Cisco UCS Manager System Monitoring Guide Using the CLI, Release 4.3

First Published: 2023-05-31

Last Modified: 2024-02-24

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

CONTENTS

Preface

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 *this font*.
TUI elements	In a Text-based User Interface, text the system displays appears in `this font`.
System output	Terminal sessions and information that the system displays appear in `this font`.
CLI commands	CLI command keywords appear in **this font**. Variables in a CLI command appear in *this font*.
[ ]	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

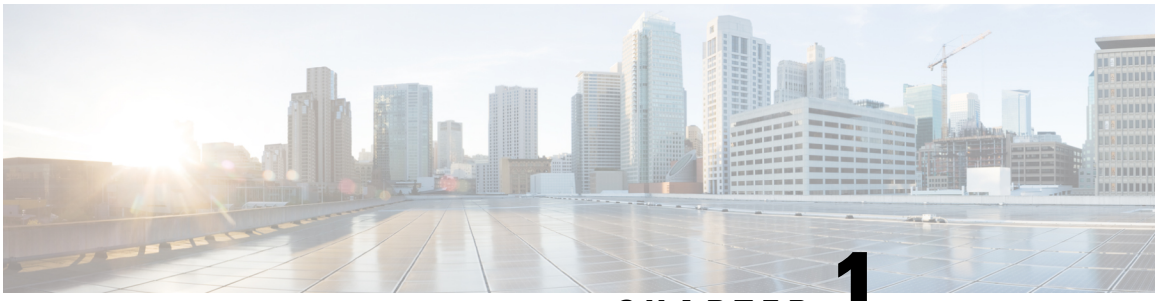
For a complete list of all B-Series documentation, see the *Cisco UCS B-Series Servers Documentation Roadmap* available at the following URL: https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/overview/guide/UCS_roadmap.html

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.

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 for This Release

New and Changed Information for This Release

New in Release 4.3(2c)

Cisco UCS X410c M7 Compute Node

The Cisco UCS X410c M7 Compute Node is the first 4-socket 4th Gen Intel® Xeon® Scalable Processors computing device to integrate into the Cisco UCS X-Series Modular System. Up to four compute nodes or two compute nodes and two GPU nodes can reside in the 7-rack-unit (7RU) Cisco UCS X9508 Server Chassis, offering high performance and efficiency gains for a wide range of mission-critical enterprise applications, memory-intensive applications and bare-metal and virtualized workloads.

The Cisco UCS X410c M7 Compute Node provides these main features:

- CPU: Four 4th Gen Intel Xeon Scalable Processors with up to 60 cores per processor
- Memory: Up to 16TB of main memory with 64x 256 GB DDR5-4800 Memory DIMMs
- Storage: Up to six hot-pluggable solid-state drives (SSDs), or non-volatile memory express (NVMe) 2.5-inch drives with a choice of enterprise-class RAID or passthrough controllers, up to two M.2 SATA drives with optional hardware RAID
- mLOM virtual interface cards:
  - Cisco UCS VIC 15420 occupies the server's modular LAN on motherboard (mLOM) slot, enabling up to 50 Gbps of unified fabric connectivity to each of the chassis’s intelligent fabric modules (IFMs) for 100 Gbps connectivity per server.
  - Cisco UCS VIC 15231 occupies the server's modular LAN on motherboard (mLOM) slot, enabling up to 100 Gbps of unified fabric connectivity to each of the chassis’s intelligent fabric modules (IFMs) for 100 Gbps connectivity per server.
  - Cisco UCS VIC 15230 (with secure boot feature) occupies the server's modular LAN on motherboard (mLOM) slot, enabling up to 100 Gbps of unified fabric connectivity to each of the chassis’s intelligent fabric modules (IFMs) for 100 Gbps connectivity per server.
- Optional mezzanine card:

- Cisco UCS 5th Gen VIC 15422 can occupy the server's mezzanine slot at the bottom rear of the chassis. This card's I/O connectors link to Cisco UCS X-Fabric technology. An included bridge card extends this VIC's 2x 50 Gbps of network connections through IFM connectors, bringing the total bandwidth to 100 Gbps per fabric (for a total of 200 Gbps per server).
- Cisco UCS PCI Mezz card for Cisco UCS X-Fabric can occupy the server's mezzanine slot at the bottom rear of the chassis. This card's I/O connectors link to Cisco UCS X-Fabric modules and enable connectivity to the Cisco UCS X440p PCIe Node.
- All VIC mezzanine cards also provide I/O connections from the X410c M7 compute node to the X440p PCIe node.

- Security: The server supports an optional trusted platform module (TPM). Additional features include a secure boot FPGA and ACT2 anti-counterfeit provisions.

Cisco UCS VIC Cards

Following Cisco UCS VIC Cards are supported from release 4.3(2c) onwards:

- Cisco UCS VIC 15427—The Cisco UCS VIC 15427 is a Quad Port CNA MLOM, 4 x 10/25/50G with Secure Boot for Cisco UCS C-Series M6 and M7 servers.
- Cisco UCS VIC 15230—The Cisco UCS VIC 15230 is a MLOM with Secure Boot for Cisco UCS X210c M6, X210c M7, and X410c M7 servers.
- Cisco UCS VIC 15237 MLOM—The Cisco UCS VIC 15237 MLOM is a MLOM, 2x40/100/200G with Secure Boot for Cisco UCS C-Series M6 and M7 servers.

New in Release 4.3(2b)

Cisco UCS X210c M7 Compute Node

The Cisco UCS X210c M7 Compute Node is the second generation of compute node to integrate into the Cisco UCS X-Series Modular System. It delivers performance, flexibility, and optimization for deployments in data centers, in the cloud, and at remote sites. This enterprise-class server offers market-leading performance, versatility, and density without compromise for workloads. Up to eight compute nodes can reside in the 7-rack-unit (7RU) Cisco UCSX-9508 Chassis, offering one of the highest densities of compute, I/O, and storage per rack unit in the industry.

The Cisco UCS X210c M7 Compute Node provides these main features:

- CPU: Up to 2x 4th Gen Intel® Xeon® Scalable Processors with up to 60 cores per processor and up to 2.625 MB Level 3 cache per core and up to 112.5 MB per CPU.
- Memory: Up to 8TB of main memory with 32x 256 GB DDR5-4800 DIMMs.
- Storage: Up to six hot-pluggable, solid-state drives (SSDs), or non-volatile memory express (NVMe) 2.5-inch drives with a choice of enterprise-class redundant array of independent disks (RAIDs) or passthrough controllers, up to two M.2 SATA and M.2 NVMe drives with optional hardware RAID.
- Optional front mezzanine GPU module: The Cisco UCS front mezzanine GPU module is a passive PCIe Gen 4.0 front mezzanine option with support for up to two U.2 NVMe drives and two HHHL GPUs.
- mLOM virtual interface cards:

- Cisco UCS Virtual Interface Card (VIC) 15420 occupies the server's modular LAN on motherboard (mLOM) slot, enabling up to 50 Gbps of unified fabric connectivity to each of the chassis intelligent fabric modules (IFMs) for 100 Gbps connectivity per server.
- ◦ Cisco UCS Virtual Interface Card (VIC) 15231 occupies the server's modular LAN on motherboard (mLOM) slot, enabling up to 100 Gbps of unified fabric connectivity to each of the chassis intelligent fabric modules (IFMs) for 100 Gbps connectivity per server.

- Optional mezzanine card:
  - Cisco UCS 5th Gen Virtual Interface Card (VIC) 15422 can occupy the server's mezzanine slot at the bottom rear of the chassis. This card's I/O connectors link to Cisco UCS X-Fabric technology. An included bridge card extends this VIC's 2x 50 Gbps of network connections through IFM connectors, bringing the total bandwidth to 100 Gbps per fabric (for a total of 200 Gbps per server).
  - Cisco UCS PCI Mezz card for X-Fabric can occupy the server's mezzanine slot at the bottom rear of the chassis. This card's I/O connectors link to Cisco UCS X-Fabric modules and enable connectivity to the Cisco UCS X440p PCIe Node.
  - All VIC mezzanine cards also provide I/O connections from the X210c M7 compute node to the X440p PCIe Node.

- Security: The server supports an optional trusted platform module (TPM). Additional features include a secure boot FPGA and ACT2 anti-counterfeit provisions.

Cisco UCS X210c M6 Compute Node

The Cisco UCS X210c M6 Compute Node is the first computing device to integrate into the Cisco UCS X-Series Modular System. Up to eight compute nodes can reside in the 7-Rack-Unit (7RU) Cisco UCS X9508 Chassis, offering one of the highest densities of compute, I/O, and storage per rack unit in the industry.

The Cisco UCS X210c M6 Compute Node provides these main features:

- CPU: Up to 2x 3rd Gen Intel® Xeon® Scalable Processors with up to 40 cores per processor and 1.5 MB Level 3 cache per core
- Memory: Up to 32x 256 GB DDR4-3200 DIMMs for up to 8 TB of main memory. Configuring up to 16x 512-GB Intel Optane™ persistent memory DIMMs can yield up to 12 TB of memory.
- Storage: Up to 6 hot-pluggable, solid-state drives (SSDs), or non-volatile memory express (NVMe) 2.5-inch drives with a choice of enterprise-class redundant array of independent disks (RAIDs) or pass-through controllers with four lanes each of PCIe Gen 4 connectivity and up to 2 M.2 SATA drives for flexible boot and local storage capabilities
- Optional front mezzanine GPU module: The Cisco UCS Front Mezzanine GPU module is a passive PCIe Gen 4 front mezzanine option with support for up to two U.2 NVMe drives and two GPUs.
- mLOM virtual interface cards:
  - Cisco UCS Virtual Interface Card (VIC) 14425 occupies the server's modular LAN on motherboard (mLOM) slot, enabling up to 50 Gbps of unified fabric connectivity to each of the chassis intelligent fabric modules (IFMs) for 100 Gbps connectivity per server.
  - Cisco UCS VIC 15231 occupies the server's modular LAN on motherboard (mLOM) slot, enabling up to 100 Gbps of unified fabric connectivity to each of the chassis intelligent fabric modules (IFMs) for 100 Gbps connectivity per server.

- Cisco UCS VIC 15420 occupies the server's modular LAN on motherboard (mLOM) slot, enabling up to 100 Gbps of unified fabric connectivity to each of the chassis intelligent fabric modules (IFMs) for 100 Gbps connectivity per server.

- Optional mezzanine card:
  - Cisco UCS VIC 14825 can occupy the server's mezzanine slot at the bottom rear of the chassis. This card's I/O connectors link to Cisco UCS X-Fabric technology. An included bridge card extends this VIC's 2x 50 Gbps of network connections through IFM connectors, bringing the total bandwidth to 100 Gbps per fabric (for a total of 200 Gbps per server).
  - Cisco UCS VIC 15422 X-Series mezz (UCSX-ME-V5Q50G) 4x25G can occupy the server's mezzanine slot at the bottom rear of the chassis. This card's I/O connectors link to Cisco UCS X-Fabric technology. An included bridge card extends this VIC's 2x 50 Gbps of network connections through IFM connectors, bringing the total bandwidth to 100 Gbps per fabric (for a total of 200 Gbps per server).
  - Cisco UCS PCI Mezz card for X-Fabric can occupy the server's mezzanine slot at the bottom rear of the chassis. This card's I/O connectors link to Cisco UCS X-Fabric modules and enable connectivity to the X440p PCIe Node.

- Security: The server supports an optional trusted platform module (TPM). Additional features include a secure boot FPGA and ACT2 anti-counterfeit provisions

Cisco UCS C240 M7 Server

The Cisco UCS C240 M7 Server is well-suited for a wide range of storage and I/O-intensive applications such as big data analytics, databases, collaboration, virtualization, consolidation, and high-performance computing in its two-socket, 2RU form factor. It incorporates the 4th Gen Intel® Xeon® Scalable Processors with up to 60 cores per socket.

In addition to the advanced features, the server is also equipped with the PCIe Gen 5.0 for high-speed I/O, a DDR5 memory bus, and expanded storage capabilities (up to 32 DDR5 DIMMs for up to 8 TB of capacity using 128 GB DIMMs (16 DIMMs per socket)) and delivers sigficant performance and efficiency gains that will improve your application performance.

You can deploy the Cisco UCS C-Series Rack Servers as standalone servers or as part of the Cisco Unified Computing System managed by Cisco Intersight or Intersight Managed Mode.

Cisco UCS C220 M7 Server

The Cisco UCS C220 M7 Server is a versatile general-purpose infrastructure and application server. This high-density, 1RU, 2-socket rack server delivers industry-leading performance and efficiency for a wide range of workloads, including virtualization, collaboration, and bare-metal applications. It incorporates the 4th Gen Intel® Xeon® Scalable Processors, with up to 52 cores per socket. With advanced features such as Intel Advanced Matrix Extensions (AMX), Data Streaming Accelerator (DSA), In-Memory Analytics Accelerator (IAA), and QuickAssist Technology (QAT), the server offers significant performance improvements.

In addition to the advanced features, the server is also equipped with the PCIe Gen 5.0 for high-speed I/O, a DDR5 memory bus, and expanded storage capabilities (up to 32 DDR5 DIMMs for up to 4 TB of capacity using 128 GB DIMMs (16 DIMMs per socket)).

You can deploy the Cisco UCS C-Series rack servers as standalone servers or as part of the Cisco Unified Computing System™ with the Cisco Intersight Infrastructure Service cloud-based management platform.

These computing innovations help reduce Total Cost of Ownership (TCO) and increase their business agility. These improvements deliver significant performance and efficiency gains that improve your application performance.

Intelligent Fabric Module (IFM for Cisco UCS X-Series Servers)

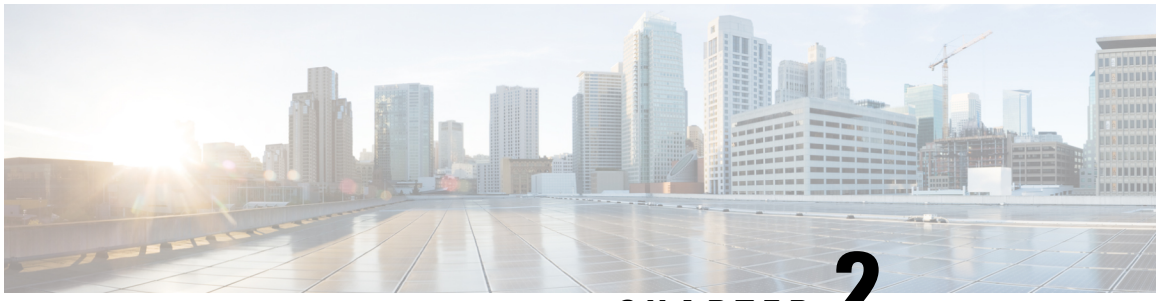
Beginning with release 4.3(2a), Cisco UCS Manager supports Cisco UCS X9508 server chassis with Cisco UCS X-Series servers. Cisco UCS X-Series servers support Intelligent Fabric Modules (IFM), which function similarly to the Input/Output Module (IOM) in Cisco UCS B-Series servers. This guide uses the term IOM to refer both IOM and IFM.

New Cisco UCS VIC Cards

Following new Cisco VIC cards are supported in this release:

- Cisco UCS VIC 15425
- Cisco UCS VIC 15235
- Cisco UCS VIC 15420
- Cisco UCS VIC 15231
- Cisco UCS VIC 14425
- Cisco UCS VIC 15422
- Cisco UCS VIC 14825

New and Changed Information for This Release



CHAPTER 2

System Monitoring Overview

System Monitoring Overview

This guide describes how to configure and use system monitoring to manage a Cisco UCS Manager environment.

Cisco UCS Manager can detect system faults: critical, major, minor, and warnings. We recommend that:

- You monitor all faults of either critical or major severity status, as immediate action is not required for minor faults and warnings.
- You monitor faults that are not of type Finite State Machine (FSM), as FSM faults will transition over time and resolve.

This guide covers the following information:

- System Log
  - System logs including faults, failures, and alarm thresholds (Syslog)
  - The three types of Syslogs: Fault, Event, and Audit logs
  - The Global Fault Policy and settings that control Syslogs
- System Event Log
  - System hardware events for servers and chassis components and their internal components (System Event Log [SEL] logs)
  - The SEL policy that controls SEL logs
- Simple Network Management Protocol
  - SNMP for monitoring devices from a central network management station and the host and user settings
  - Fault suppression policies for SNMP traps, Call Home notifications, and specific devices

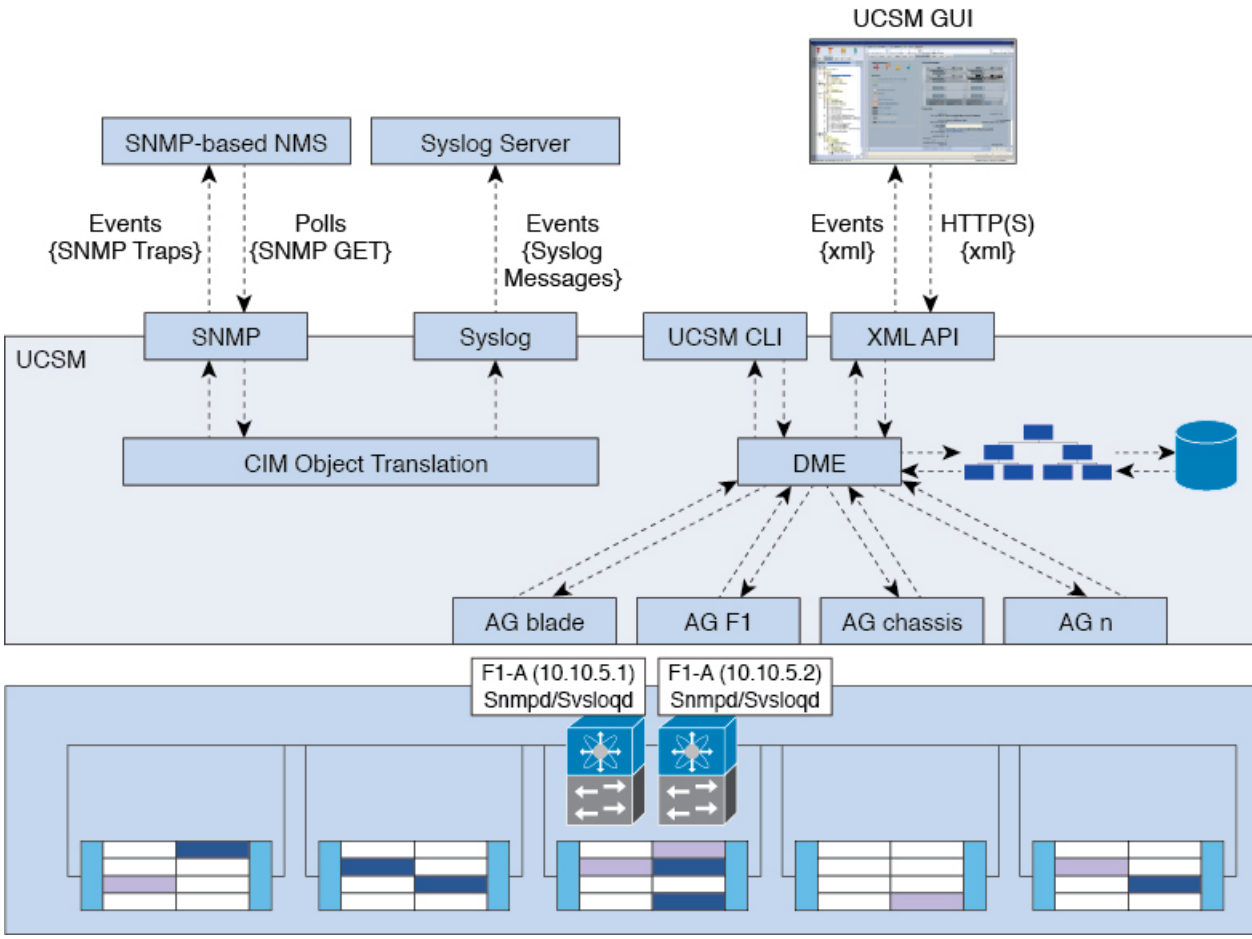
- Core File Exporter and logs, such as Syslog, Audit Log, and the System Event Log
- Statistics Collection and Threshold Policies for adapters, chassis, host, ports, and servers
- Call Home and Smart Call Home Cisco embedded device support
- Hardware monitoring using the Cisco UCS Manager user interface
- Traffic Monitoring sessions for analysis by a network analyzer
- Cisco Netflow Monitor for IP network traffic accounting, usage-based network billing, network planning, security, Denial of Service monitoring capabilities, and network monitoring

The Cisco UCS Manager Core and Fault Generation

The Cisco UCS Manager core is made up of three elements, which are the Data Management Engine, Application Gateway, and user accessible northbound interface. The northbound interface comprises of SNMP, Syslog, XML API, and UCSM CLI.

You can monitor the Cisco UCS Manager servers through XML API, SNMP, and Syslog. Both SNMP and Syslog are interfaces used only used for monitoring as they are read-only, so no configuration changes are allowed from these interfaces. Alternatively, the XML API is a monitoring interface that is read-write, which allows you to monitor Cisco UCS Manager, and change the configuration if needed.

Figure 1: Cisco UCS Manager Core and Monitoring Interfaces



Data Management Engine (DME)

The DME is the center of the Cisco UCS Manager system, which maintains:

- The Cisco UCS XML database which houses the inventory database of all physical elements (blade and rack mount servers, chassis, modules, and fabric interconnects).
- The logical configuration data for profiles, policies, pools, vNIC, and vHBA templates.
- The various networking-related configuration details like VLANs, VSANs, port channels, network uplinks, and server downlinks.

The DME monitors:

- The current health and state of all components of all physical and logical elements in a Cisco UCS domain.
- The transition information of all Finite State Machine (FSM) tasks occurring.

Only the current information of inventory, health, and configuration data of the managed endpoints are stored in the Cisco UCS XML database resulting in near real time. By default the DME does not store a historical log of faults that have occurred on a Cisco UCS domain. As fault conditions are raised on the endpoints, the

DME creates faults in the Cisco UCS XML database. As those faults are mitigated, the DME clears and removes the faults from the Cisco UCS XML database.

Application Gateway (AG)

Application Gateways are software agents that communicate directly with the endpoints to relay the health and state of the endpoints to the DME. AG-managed endpoints include servers, chassis, modules, fabric extenders, fabric interconnects, and NX-OS. The AGs actively monitor the server through the IPMI and SEL logs using the Cisco Integrated Management Controller (CIMC). They provide the DME with the health, state, configuration, and potential fault conditions of a device. The AGs manage configuration changes from the current state to the desired state during FSM transitions when changes are made to the Cisco UCS XML database.

The module AG and chassis AG communicate with the Chassis Management Controller (CMC) to get information about the health, state, configuration, and fault conditions observed by the CMC. The fabric interconnect NX-OS AG communicates directly with NX-OS to get information about the health, state, configuration, statistics, and fault conditions observed by NX-OS on the fabric interconnects. All AGs provide the inventory details to the DME about the endpoints during the various discovery processes. The AGs perform the state changes necessary to configure an endpoint during FSM-triggered transitions, monitor the health and state of the endpoints, and notify the DME of any faults.

Northbound Interfaces

The northbound interfaces include SNMP, Syslog, CLI, and XML API. The XML API present in the Apache webserver layer sends login, logout, query, and configuration requests using HTTP or HTTPS. SNMP and Syslog are both consumers of data from the DME.

SNMP informs and traps are translated directly from the fault information stored in the Cisco UCS XML database. SNMP GET requests are sent through the same object translation engine in reverse, where the DME receives a request from the object translation engine. The data is translated from the XML database to an SNMP response.

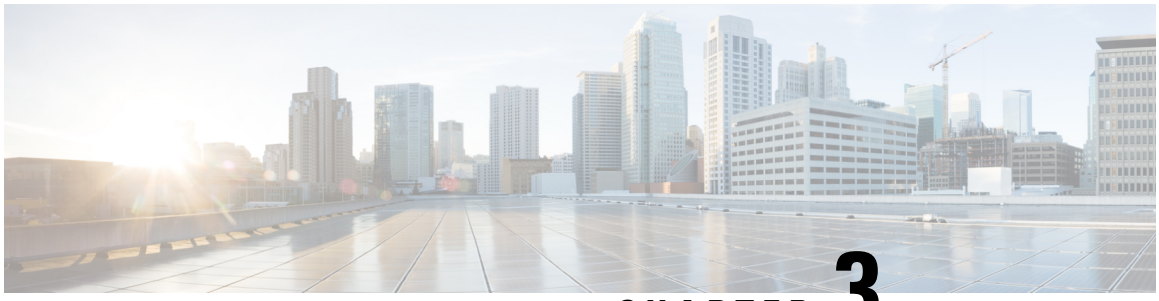
Syslog messages use the same object translation engine as SNMP, where the source of the data (faults, events, audit logs) is translated from XML into a Cisco UCS Manager-formatted Syslog message.

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.
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.

Guide	Description
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

Syslog

Syslog

Cisco UCS Manager generates system log, or syslog messages to record the following incidents that take place in the Cisco UCS Manager system:

- Routine system operations
- Failures and errors
- Critical and emergency conditions

There are three kinds of syslog entries: Fault, Event, and Audit.

Each syslog message identifies the Cisco UCS Manager process that generated the message and provides a brief description of the operation or error that occurred. The syslog is useful both in routine troubleshooting, incident handling, and management.

Cisco UCS Manager collects and logs syslog messages internally. You can send them to external syslog servers running a syslog daemon. Logging to a central syslog server helps in aggregation of logs and alerts. Some syslog messages to monitor include, DIMM problems, equipment failures, thermal problems, voltage problems, power problems, high availability (HA) cluster problems, and link failures.

Note The FSM faults, threshold faults, and unresolved policy events are not sent to syslog server. However, SNMP traps are generated for the threshold fault events.

Syslog messages contain an event code and fault code. To monitor syslog messages, you can define syslog message filters. These filters can parse the syslog messages based on the criteria you choose. You can use the following criteria to define a filter:

- By event or fault codes: Define a filter with a parsing rule to include only the specific codes that you intend to monitor. Messages that do not match these criteria are discarded.

- By severity level: Define a filter with a parsing rule to monitor syslog messages with specific severity levels. You can set syslog severity levels individually for OS functions, to facilitate logging and display of messages ranging from brief summaries to detailed information for debugging.

Cisco devices can send their log messages to a Unix-style syslog service. A syslog service simply accepts messages, then stores them in files or prints them according to a simple configuration file. This form of logging is the best available for Cisco devices because it can provide protected long-term storage of logs.

Enabling Syslog Messages to Store In a Local File

Procedure

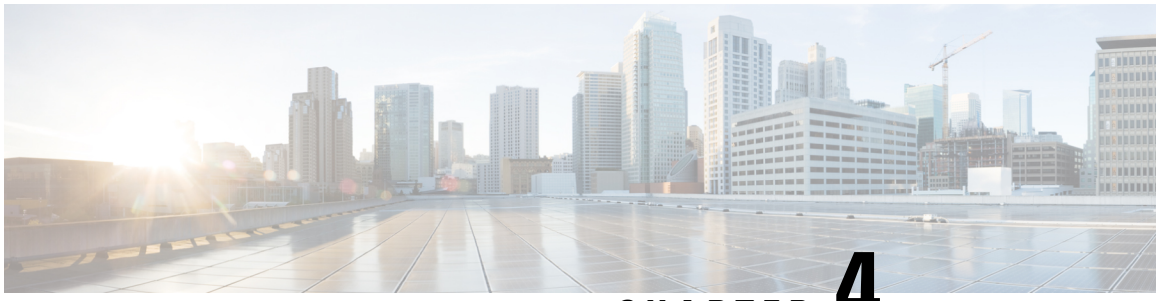
	Command or Action	Purpose
Step 1	UCS-A# **scope monitoring**	Enters monitoring mode.
Step 2	UCS-A /monitoring # {**enable** | **disable**} **syslog console**	Enables or disables the sending of syslogs to the console.
Step 3	(Optional) UCS-A /monitoring # **set syslog console level** {**emergencies** | **alerts** | **critical**}	Select the lowest message level that you want displayed. If syslogs are enabled, the system displays that level and above on the console. The level options are listed in order of decreasing urgency.The default level is Critical.
Step 4	UCS-A /monitoring # {**enable** | **disable**} **syslog monitor**	Enables or disables the monitoring of syslog information by the operating system.
Step 5	(Optional) UCS-A /monitoring # **set syslog monitor level** {**emergencies** | **alerts** | **critical** | **errors** | **warnings** | **notifications** | **information** | **debugging**}	Select the lowest message level that you want displayed. If the monitor state is enabled, the system displays that level and above. The level options are listed in order of decreasing urgency.The default level is Critical. **Note** Messages at levels below Critical are displayed on the terminal monitor only if you have entered the **terminal monitor** command.
Step 6	UCS-A /monitoring # {**enable** | **disable**} **syslog rfc-5424-compliance**	Enables or disables the writing of syslog information as per RFC 5424 format. **Note** This option is applicable only for Cisco UCS 6500 and 6400 series Fabric Interconnects.
Step 7	UCS-A /monitoring # {**enable** | **disable**} **syslog file**	Enables or disables the writing of syslog information to a syslog file.

	Command or Action	Purpose
Step 8	UCS-A /monitoring # **set syslog file name** *filename*	The name of the file in which the messages are logged. Up to 16 characters are allowed in the file name.
Step 9	(Optional) UCS-A /monitoring # **set syslog file level** {**emergencies** | **alerts** | **critical** | **errors** | **warnings** | **notifications** | **information** | **debugging**}	Select the lowest message level that you want stored to a file. If the file state is enabled, the system stores that level and above in the syslog file. The level options are listed in order of decreasing urgency.The default level is Critical.
Step 10	(Optional) UCS-A /monitoring # **set syslog file size** *filesize*	The maximum file size, in bytes, before the system begins to write over the oldest messages with the newest ones. The range is 4096 to 4194304 bytes.
Step 11	UCS-A /monitoring # {**enable** | **disable**} **syslog remote-destination** {**server-1** | **server-2** | **server-3**}	Enables or disables the sending of syslog messages to up to three external syslog servers.
Step 12	(Optional) UCS-A /monitoring # **set syslog remote-destination** {**server-1** | **server-2** | **server-3**} **level**{**emergencies** | **alerts** | **critical** | **errors** | **warnings** | **notifications** | **information** | **debugging**}	Select the lowest message level that you want stored to the external log. If the remote-destination is enabled, the system sends that level and above to the external server. The level options are listed in order of decreasing urgency.The default level is Critical.
Step 13	UCS-A /monitoring # **set syslog remote-destination** {**server-1** | **server-2** | **server-3**} **hostname** *hostname*	The hostname or IP address of the specified remote syslog server. Up to 256 characters are allowed in the hostname.
Step 14	(Optional) UCS-A /monitoring # **set syslog remote-destination** {**server-1** | **server-2** | **server-3**} **facility** {**local0** | **local1** | **local2** | **local3** | **local4** | **local5** | **local6** | **local7**}	The facility level contained in the syslog messages sent to the specified remote syslog server.
Step 15	UCS-A /monitoring # {**enable** | **disable**} **syslog source** {**audits** | **events** | **faults**}	This can be one of the following: • **audits**—Enables or disables the logging of all audit log events. • **events**—Enables or disables the logging of all system events. • **faults**—Enables or disables the logging of all system faults.
Step 16	UCS-A /monitoring # **commit-buffer**	Commits the transaction.

Example

This example shows how to enable the storage of syslog messages in a local file and commits the transaction:

```
UCS-A# scope monitoring
UCS-A /monitoring # disable syslog console
UCS-A /monitoring* # disable syslog monitor
UCS-A /monitoring* # enable syslog file
UCS-A /monitoring* # set syslog file name SysMsgsUCSA
UCS-A /monitoring* # set syslog file level notifications
UCS-A /monitoring* # set syslog file size 4194304
UCS-A /monitoring* # disable syslog remote-destination server-1
UCS-A /monitoring* # disable syslog remote-destination server-2
UCS-A /monitoring* # disable syslog remote-destination server-3
UCS-A /monitoring* # commit-buffer
UCS-A /monitoring #
```



CHAPTER 4

System Event Log

System Event Log

The System Event Log (SEL) resides on the CIMC in NVRAM. The SEL is used for troubleshooting system health. It records most server-related events, such as instances of over or under voltage, temperature events, fan events, and BIOS events. The types of events supported by SEL include BIOS events, memory unit events, processor events, and motherboard events.

The SEL logs are stored in the CIMC NVRAM, through a SEL log policy. It is best practice to periodically download and clear the SEL logs. The SEL file is approximately 40KB in size, and no further events can be recorded once it is full. It must be cleared before additional events can be recorded.

You can use the SEL policy to back up the SEL to a remote server, and optionally to clear the SEL after a backup operation occurs. Backup operations can be triggered based on specific actions, or they can be set to occur at regular intervals. You can also manually back up or clear the SEL.

The backup file is automatically generated. The filename format is sel-*SystemName-ChassisID-ServerID-ServerSerialNumber-Timestamp*.

For example, sel-UCS-A-ch01-serv01-QCI12522939-20091121160736.

Viewing the System Event Log for a Server

Viewing the System Event Log for an Individual Server

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **show sel** *chassis-id / blade-id*	Displays the system event log for the specified server.

Example

The following example displays the system event log for blade 3 in chassis 1.

```
UCS-A# show sel 1/3
     1 | 01/01/1970 01:23:27 | System Event 0x83 | Timestamp clock synch | SEL timestamp
clock updated, event is f
irst of pair | Asserted
     2 | 01/01/1970 01:23:28 | Drive slot(Bay) SAS0_LINK_STATUS | Transition to Degraded |
 Asserted
     3 | 01/01/1970 01:23:28 | Drive slot(Bay) SAS0_LINK_STATUS | Transition to On Line |
Deasserted
     4 | 01/01/1970 01:23:28 | Platform alert LED_SAS0_FAULT | LED is blinking fast |
Asserted
     5 | 01/01/1970 01:23:28 | Platform alert LED_SAS0_FAULT | LED is on | Deasserted
     6 | 01/01/1970 01:23:28 | Platform alert LED_FPID | LED is on | Asserted
     7 | 01/01/1970 01:23:28 | Platform alert LED_FPID | LED is off | Deasserted
     8 | 01/01/1970 01:23:29 | Entity presence MAIN_POWER | Device Absent | Asserted
     9 | 01/01/1970 01:23:29 | Entity presence MAIN_POWER | Device Present | Deasserted
     a | 01/01/1970 01:23:29 | Platform alert LED_SAS0_FAULT | LED is on | Asserted
    b | 01/01/1970 01:23:29 | Platform alert LED_SAS0_FAULT | LED color is green | Asserted

     c | 01/01/1970 01:23:29 | Platform alert LED_SAS0_FAULT | LED is blinking fast |
Deasserted
    d | 01/01/1970 01:23:29 | Platform alert LED_SAS0_FAULT | LED color is amber | Deasserted

     e | 01/01/1970 00:00:22 | Drive slot(Bay) SAS0_LINK_STATUS | Transition to Degraded |
 Asserted
     f | 01/01/1970 00:00:22 | Entity presence MEZZ_PRS | Device Present | Asserted
    10 | 01/01/1970 00:00:22 | Entity presence HDD1_PRS | Device Absent | Asserted
```

Viewing the System Event Log for All of the Servers in a Chassis

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *chassis-id / blade-id*	Enters chassis server mode for the specified server.
Step 2	UCS-A /chassis/server # **show sel**	Displays the system event log.

Example

The following example displays the system event log from chassis server mode for blade 3 in chassis 1.

```
UCS-A# scope server 1/3
UCS-A /chassis/server # show sel
     1 | 01/01/1970 01:23:27 | System Event 0x83 | Timestamp clock synch | SEL timestamp
clock updated, event is f
irst of pair | Asserted
     2 | 01/01/1970 01:23:28 | Drive slot(Bay) SAS0_LINK_STATUS | Transition to Degraded |
 Asserted
     3 | 01/01/1970 01:23:28 | Drive slot(Bay) SAS0_LINK_STATUS | Transition to On Line |
Deasserted
     4 | 01/01/1970 01:23:28 | Platform alert LED_SAS0_FAULT | LED is blinking fast |
Asserted
     5 | 01/01/1970 01:23:28 | Platform alert LED_SAS0_FAULT | LED is on | Deasserted
     6 | 01/01/1970 01:23:28 | Platform alert LED_FPID | LED is on | Asserted
     7 | 01/01/1970 01:23:28 | Platform alert LED_FPID | LED is off | Deasserted
     8 | 01/01/1970 01:23:29 | Entity presence MAIN_POWER | Device Absent | Asserted
     9 | 01/01/1970 01:23:29 | Entity presence MAIN_POWER | Device Present | Deasserted
     a | 01/01/1970 01:23:29 | Platform alert LED_SAS0_FAULT | LED is on | Asserted
    b | 01/01/1970 01:23:29 | Platform alert LED_SAS0_FAULT | LED color is green | Asserted

     c | 01/01/1970 01:23:29 | Platform alert LED_SAS0_FAULT | LED is blinking fast |
Deasserted
    d | 01/01/1970 01:23:29 | Platform alert LED_SAS0_FAULT | LED color is amber | Deasserted

     e | 01/01/1970 00:00:22 | Drive slot(Bay) SAS0_LINK_STATUS | Transition to Degraded |
 Asserted
     f | 01/01/1970 00:00:22 | Entity presence MEZZ_PRS | Device Present | Asserted
    10 | 01/01/1970 00:00:22 | Entity presence HDD1_PRS | Device Absent | Asserted
```

Configuring the SEL Policy

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters organization mode for the specified organization. To enter the root organization mode, type **/** as the *org-name* .
Step 2	UCS-A /org # **scope ep-log-policy sel**	Enters organization endpoint log policy mode and scopes the SEL policy.
Step 3	(Optional) UCS-A /org/ep-log-policy # **set description** *description*	Provides a description for the policy. **Note** If your description includes spaces, special characters, or punctuation, begin and end your description with quotation marks. The quotation marks will not appear in the description field of any **show** command output.

	Command or Action	Purpose
Step 4	UCS-A /org/ep-log-policy # **set backup action** [**log-full**] [**on-change-of-association**] [**on-clear**] [**timer**] [**none**]	Specifies an action or actions that will trigger a backup operation.
Step 5	UCS-A /org/ep-log-policy # **set backup clear-on-backup** {**no** | **yes**}	Specifies whether to clear the system event log after a backup operation occurs.
Step 6	UCS-A /org/ep-log-policy # **set backup destination** *URL*	Specifies the protocol, user, password, remote hostname, and remote path for the backup operation. Depending on the protocol used, specify the URL using one of the following syntaxes: • **ftp://** *username***@***hostname* / *path* • **scp://** *username* **@** *hostname* / *path* • **sftp://** *username* **@** *hostname* / *path* • **tftp://** *hostname* **:** *port-num* / *path* **Note** You can also specify the backup destination by using the **set backup hostname** , **set backup password** , **set backup protocol** , **set backup remote-path** , **set backup user** commands, or by using the **set backup destination** command. Use either method to specify the backup destination.
Step 7	UCS-A /org/ep-log-policy # **set backup format** {*ascii* | *binary*}	Specifies the format for the backup file.
Step 8	UCS-A /org/ep-log-policy # **set backup hostname** {*hostname* | *ip-addr*}	Specifies the hostname or IP address of the remote server.
Step 9	UCS-A /org/ep-log-policy # **set backup interval** {**1-hour** | **2-hours** | **4-hours** | **8-hours** | **24-hours** | **never**}	Specifies the time interval for the automatic backup operation. Specifying the **never** keyword means that automatic backups will not be made.
Step 10	UCS-A /org/ep-log-policy # **set backup password** *password*	Specifies the password for the username. This step does not apply if the TFTP protocol is used.
Step 11	UCS-A /org/ep-log-policy # **set backup protocol** {**ftp** | **scp** | **sftp** | **tftp**}	Specifies the protocol to use when communicating with the remote server.
Step 12	UCS-A /org/ep-log-policy # **set backup remote-path** *path*	Specifies the path on the remote server where the backup file is to be saved.

	Command or Action	Purpose
Step 13	UCS-A /org/ep-log-policy # **set backup user** *username*	Specifies the username the system should use to log in to the remote server. This step does not apply if the TFTP protocol is used.
Step 14	UCS-A /org/ep-log-policy # **commit-buffer**	Commits the transaction.

Example

The following example configures the SEL policy to back up the system event log (in ASCII format) every 24 hours or when the log is full, clears the system event log after a backup operation occurs, and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # scope ep-log-policy sel
UCS-A /org/ep-log-policy # set backup destination scp://user@192.168.1.10/logs
Password:
UCS-A /org/ep-log-policy* # set backup action log-full
UCS-A /org/ep-log-policy* # set backup clear-on-backup yes
UCS-A /org/ep-log-policy* # set backup format ascii
UCS-A /org/ep-log-policy* # set backup interval 24-hours
UCS-A /org/ep-log-policy* # commit-buffer
UCS-A /org/ep-log-policy #
```

Backing Up the System Event Log for a Server

Backing Up the System Event Log for an Individual Server

Before you begin

Configure the system event log policy. The manual backup operation uses the remote destination configured in the system event log policy.

Procedure

	Command or Action	Purpose
Step 1	UCS-A /chassis/server # **backup sel** *chassis-id* / *blade-id*	Backs up the system event log.
Step 2	UCS-A# **commit-buffer**	Commits the transaction.

Example

The following example backs up the system event log for blade 3 in chassis 1 and commits the transaction.

```
UCS-A# backup sel 1/3
UCS-A* # commit-buffer
UCS-A#
```

Backing Up the System Event Log for All of the Servers in a Chassis

Before you begin

Configure the system event log policy. The manual backup operation uses the remote destination configured in the system event log policy.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *chassis-id* / *blade-id*	Enters chassis server mode for the specified server.
Step 2	UCS-A /chassis/server # **backup sel**	Backs up the system event log.
Step 3	UCS-A /chassis/server # **commit-buffer**	Commits the transaction.

Example

The following example backs up the system event log from chassis server mode for blade 3 in chassis 1 and commits the transaction.

```
UCS-A# scope server 1/3
UCS-A /chassis/server # backup sel
UCS-A /chassis/server* # commit-buffer
UCS-A /chassis/server #
```

Clearing the System Event Log for a Server

Clearing the System Event Log for an Individual Server

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **clear sel** *chassis-id* / *blade-id*	Clears the system event log.
Step 2	UCS-A# **commit-buffer**	Commits the transaction.

Example

The following example clears the system event log for blade 3 in chassis 1 and commits the transaction:

```
UCS-A# clear sel 1/3
UCS-A* # commit-buffer
UCS-A#
```

Clearing the System Event Log for All of the Servers in a Chassis

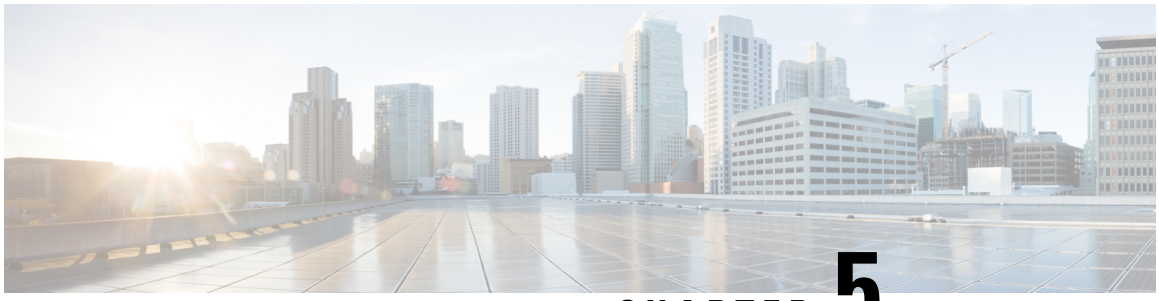
Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *chassis-id* / *blade-id*	Enters chassis server mode for the specified server.
Step 2	UCS-A /chassis/server # **clear sel**	Clears the system event log.
Step 3	UCS-A /chassis/server # **commit-buffer**	Commits the transaction.

Example

The following example clears the system event log from chassis server mode for blade 3 in chassis 1 and commits the transaction:

```
UCS-A# scope server 1/3
UCS-A /chassis/server # clear sel
UCS-A /chassis/server* # commit-buffer
UCS-A /chassis/server #
```



CHAPTER 5

Audit Logs

Audit Logs

Audit Logs record system events that occurred, where they occurred, and which users initiated them.

Viewing Audit Logs

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **show audit-logs**	Displays the audit logs. **Note** Use the *id* option to view a specific audit-log. Use the detail option to view more detailed information in the audit log output.

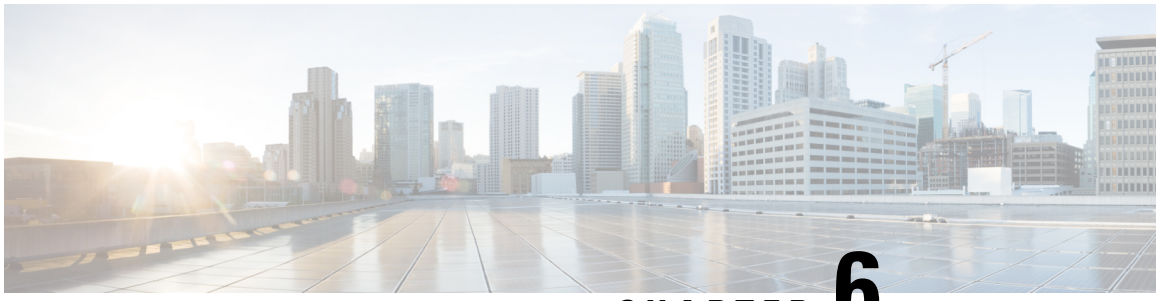
Example

The following example displays the audit logs:

```
UCS-A# scope security
UCS-A /security # show audit-logs

Audit trail logs:
    Creation Time        User       ID       Action           Description
    -------------------- ---------- -------- ---------------- -----------
    2015-12-24T12:34:02.980
                         internal    6572175 Creation         Web A: local user admin logged
 i
```

```
    2015-12-22T11:26:33.547
                         admin       6512814 Creation         Server port A/1/21 created
    2015-12-22T11:26:33.547
                         admin       6512816 Deletion         Server Port Channel A/1025
delet
    2015-12-22T11:26:33.536
                         admin       6512791 Modification     Acknowledged chassis 1.
    2015-12-22T11:25:44.755
                         admin       6512767 Modification     chassis discovery policy
modifie
    2015-12-22T11:25:01.447
                         admin       6512763 Deletion         Server Member Port A/1/23
remove
    2015-12-22T11:04:22.031
                         admin       6511644 Deletion         Server port A/1/21 deleted
    2015-12-22T11:04:22.030
                         admin       6511638 Creation         Server Port Channel A/1025
creat
    2015-12-22T11:04:22.030
 UCS-A /security #
```



CHAPTER 6

Log File Exporter

Log File Exporter

Cisco UCS Manager generates log files for each executable. The log files can be up to 20 MB in size, and up to five backups can be stored on the server. The log file exporter allows you to export the log files to a remote server before they are deleted. The log file names contain the following information:

- The name of the process
- Timestamp
- The name and ID of the fabric interconnect

Note If you do not enable log exporting, the oldest log files are deleted whenever the maximum backup file limit is reached.

Guidelines and Limitations

- We recommend that you use tftp or password-less scp or sftp for log export. When standard scp or sftp is used, the user password is stored in the configuration file in encrypted format.
- On a HA setup, the log files from each side are exported separately. If one side fails to export logs, the other side does not compensate.

Exporting Log Files to a Remote Server

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope monitoring**	Enters monitoring mode.

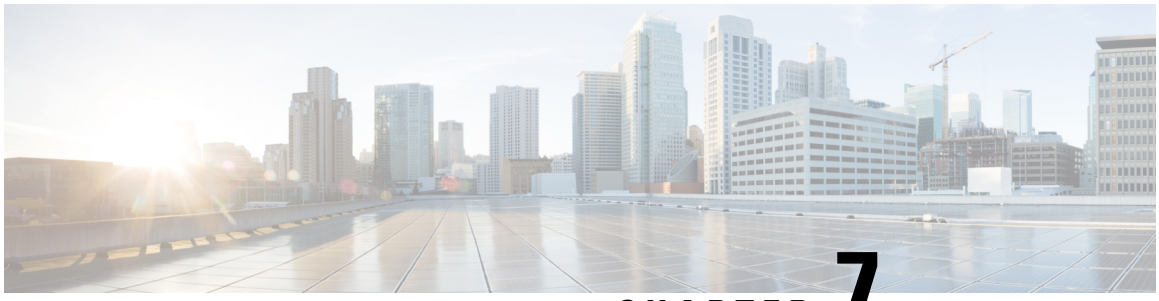
	Command or Action	Purpose
Step 2	UCS-A /monitoring # **scope sysdebug**	Enters monitoring system debug mode.
Step 3	UCS-A /monitoring/sysdebug # **scope log-export-policy**	Enters log file export mode.
Step 4	UCS-A /monitoring/sysdebug/log-export-policy # **set admin-state** {**disabled** | **enabled**}	Whether log file exporting is enabled.
Step 5	(Optional) UCS-A /monitoring/sysdebug/log-export-policy # **set desc** *description*	Provides a description for the log export policy
Step 6	UCS-A /monitoring/sysdebug/log-export-policy # **set hostname** *hostname*	Specifies the hostname of the remote server.
Step 7	UCS-A /monitoring/sysdebug/log-export-policy # **set passwd**	After you press Enter, you are prompted to enter the password. Specifies the password for the remote server username. This step does not apply if the TFTP protocol is used.
Step 8	UCS-A /monitoring/sysdebug/log-export-policy # **set passwordless-ssh** {**no** | **yes**}	Enables SSH login without a password.
Step 9	UCS-A /monitoring/sysdebug/log-export-policy # **set proto** {**scp** | **ftp** | **sftp** | **tftp**}	Specifies the protocol to use when communicating with the remote server.
Step 10	UCS-A /monitoring/sysdebug/log-export-policy # **set path** *path*	Specifies the path on the remote server where the log file is to be saved.
Step 11	UCS-A /monitoring/sysdebug/log-export-policy # **set user** *username*	Specifies the username the system should use to log in to the remote server. This step does not apply if the TFTP protocol is used.
Step 12	UCS-A /monitoring/sysdebug/log-export-policy # **commit-buffer**	Commits the transaction.

Example

The following example shows how to enable the log file exporter, specify the remote server hostname, set the protocol to scp, enable passwordless login, and commit the transaction.

```
UCS-A# scope monitoring
UCS-A /monitoring # scope sysdebug
UCS-A /monitoring/sysdebug # scope log-export-policy
UCS-A /monitoring/sysdebug/log-export-policy # set admin-state enable
UCS-A /monitoring/sysdebug/log-export-policy* # set hostname 10.10.1.1
```

```
UCS-A /monitoring/sysdebug/log-export-policy* # set path /
UCS-A /monitoring/sysdebug/log-export-policy* # set user testuser
UCS-A /monitoring/sysdebug/log-export-policy* # set proto scp
UCS-A /monitoring/sysdebug/log-export-policy* # set passwd
password:
UCS-A /monitoring/sysdebug/log-export-policy* # set passwordless-ssh yes
UCS-A /monitoring/sysdebug/log-export-policy* # commit-buffer
UCS-A /monitoring/sysdebug/log-export-policy #
```



CHAPTER 7

Core File Exporter

Core File Exporter

Critical failures in the Cisco UCS components, such as a fabric interconnect or an I/O module, can cause the system to create a core dump file. Cisco UCS Manager uses the Core File Exporter to immediately export the core dump files to a specified location on the network through TFTP. This functionality allows you to export the tar file with the contents of the core dump file. The Core File Exporter provides system monitoring and automatic export of core dump files that need to be included in TAC cases.

Configuring the Core File Exporter

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope monitoring**	Enters monitoring mode.
Step 2	UCS-A /monitoring # **scope sysdebug**	Enters monitoring system debug mode.
Step 3	UCS-A /monitoring/sysdebug # **enable core-export-target**	Enables the core file exporter. When the core file exporter is enabled and an error causes the server to perform a core dump, the system exports the core file via TFTP to the specified remote server.
Step 4	UCS-A /monitoring/sysdebug # **set core-export-target path** *path*	Specifies the path to use when exporting the core file to the remote server.
Step 5	UCS-A /monitoring/sysdebug # **set core-export-target port** *port-num*	Specifies the port number to use when exporting the core file via TFTP. The range of valid values is 1 to 65,535.

	Command or Action	Purpose
Step 6	UCS-A /monitoring/sysdebug # **set core-export-target server-description** *description*	Provides a description for the remote server used to store the core file.
Step 7	UCS-A /monitoring/sysdebug # **set core-export-target server-name** *hostname*	Specifies the hostname of the remote server to connect with via TFTP.
Step 8	UCS-A /monitoring/sysdebug # **commit-buffer**	Commits the transaction.

Example

The following example enables the core file exporter, specifies the path and port to use when sending the core file, specifies the remote server hostname, provides a description for the remote server, and commits the transaction.

```
UCS-A# scope monitoring
UCS-A /monitoring # scope sysdebug
UCS-A /monitoring/sysdebug # enable core-export-target
UCS-A /monitoring/sysdebug* # set core-export-target path /root/CoreFiles/core
UCS-A /monitoring/sysdebug* # set core-export-target port 45000
UCS-A /monitoring/sysdebug* # set core-export-target server-description CoreFile102.168.10.10
UCS-A /monitoring/sysdebug* # set core-export-target server-name 192.168.10.10
UCS-A /monitoring/sysdebug* # commit-buffer
UCS-A /monitoring/sysdebug #
```

Disabling the Core File Exporter

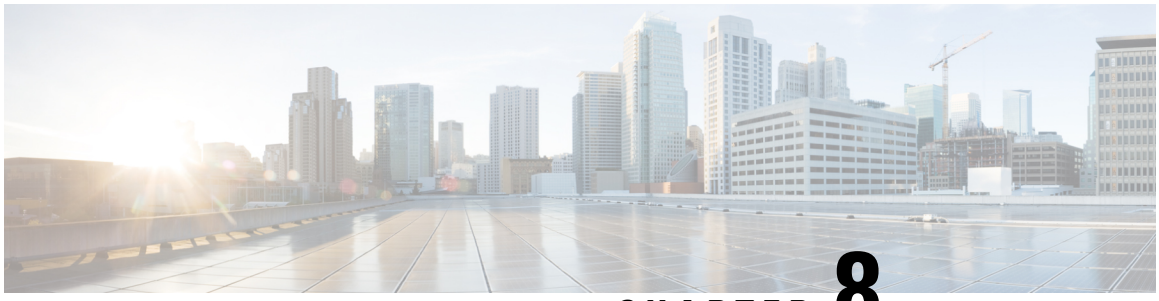
Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope monitoring**	Enters monitoring mode.
Step 2	UCS-A /monitoring # **scope sysdebug**	Enters monitoring system debug mode.
Step 3	UCS-A /monitoring/sysdebug # **disable core-export-target**	Disables the core file exporter. When the core file exporter is disabled core files are not automatically exported.
Step 4	UCS-A /monitoring/sysdebug # **commit-buffer**	Commits the transaction.

Example

The following example disables the core file exporter and commits the transaction.

```
UCS-A# scope monitoring
UCS-A /monitoring # scope sysdebug
UCS-A /monitoring/sysdebug # disable core-export-target
UCS-A /monitoring/sysdebug* # commit-buffer
UCS-A /monitoring/sysdebug #
```



CHAPTER 8

Fault Collection and Suppression

Global Fault Policy

The global fault policy controls the lifecycle of a fault in a Cisco UCS domain, including when faults are cleared, the flapping interval (the length of time between the fault being raised and the condition being cleared), and the retention interval (the length of time a fault is retained in the system).

A fault in Cisco UCS has the following lifecycle:

1. A condition occurs in the system and Cisco UCS Manager raises a fault. This is the active state.
2. When the fault is alleviated, it enters a flapping or soaking interval that is designed to prevent flapping. Flapping occurs when a fault is raised and cleared several times in rapid succession. During the flapping interval, the fault retains its severity for the length of time specified in the global fault policy.
3. If the condition reoccurs during the flapping interval, the fault returns to the active state. If the condition does not reoccur during the flapping interval, the fault is cleared.
4. The cleared fault enters the retention interval. This interval ensures that the fault reaches the attention of an administrator even if the condition that caused the fault has been alleviated and the fault has not been deleted prematurely. The retention interval retains the cleared fault for the length of time specified in the global fault policy.
5. If the condition reoccurs during the retention interval, the fault returns to the active state. If the condition does not reoccur, the fault is deleted.

Configuring the Fault Collection Policy

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope monitoring**	Enters monitoring mode.
Step 2	UCS-A /monitoring # **scope fault policy**	Enters monitoring fault policy mode.

	Command or Action	Purpose
Step 3	UCS-A /monitoring/fault-policy # **set clear-action** {**delete** | **retain**}	Specifies whether to retain or delete all cleared messages. If the **retain** option is specified, then the length of time that the messages are retained is determined by the **set retention-interval** command.
Step 4	UCS-A /monitoring/fault-policy # **set flap-interval** *seconds*	Specifies the time interval (in seconds) the system waits before changing a fault state. Flapping occurs when a fault is raised and cleared several times in rapid succession. To prevent this, the system does not allow a fault to change state until the flapping interval has elapsed after the last state change. If the fault is raised again during the flapping interval, it returns to the active state, otherwise, the fault is cleared.
Step 5	UCS-A /monitoring/fault-policy # **set retention-interval** {*days hours minutes seconds* | **forever**}	Specifies the time interval the system retains all cleared fault messages before deleting them. The system can retain cleared fault messages forever, or for the specified number of days, hours, minutes, and seconds.
Step 6	UCS-A /monitoring/fault-policy # **commit-buffer**	Commits the transaction.

Example

This example configures the fault collection policy to retain cleared fault messages for 30 days, sets the flapping interval to 10 seconds, and commits the transaction.

```
UCS-A# scope monitoring
UCS-A /monitoring # scope fault policy
UCS-A /monitoring/fault-policy # set clear-action retain
UCS-A /monitoring/fault-policy* # set flap-interval 10
UCS-A /monitoring/fault-policy* # set retention-interval 30 0 0 0
UCS-A /monitoring/fault-policy* # commit-buffer
UCS-A /monitoring/fault-policy #
```

Fault Suppression

Fault suppression allows you to suppress SNMP trap and Call Home notifications during a planned maintenance time. You can create a fault suppression task to prevent notifications from being sent whenever a transient fault is raised or cleared.

Faults remain suppressed until the time duration has expired, or the fault suppression tasks have been manually stopped by you. After the fault suppression has ended, Cisco UCS Manager will send notifications for any outstanding suppressed faults that have not been cleared.

You can configure fault suppression using the following methods.

Fixed Time Intervals or Schedules

You can use the following to specify the maintenance window during which you want to suppress faults:

- Fixed time intervals allow you to create a start time and a duration when fault suppression is active. Fixed time intervals cannot be reused.
- Schedules are used for one time occurrences or recurring time periods. They can be saved and reused.

Suppression Policies

These policies define which causes and types of faults you want to suppress. Only one policy can be assigned to a task. The following policies are defined by Cisco UCS Manager:

- **default-chassis-all-maint**—Suppresses faults for the chassis and all components installed into the chassis, including all servers, power supplies, fan modules, and IOMs.

  This policy applies only to chassis.

- **default-chassis-phys-maint**—Suppresses faults for the chassis, all fan modules, and power supplies installed into the chassis.

  This policy applies only to chassis.

- **default-fex-all-maint**—Suppresses faults for the FEX, all power supplies, fan modules, and IOMs in the FEX.

  This policy applies only to FEXes.

- **default-fex-phys-maint**—Suppresses faults for the FEX, all fan modules and power supplies in the FEX.

  This policy applies only to FEXes.

- **default-server-maint**—Suppresses faults for servers.

  This policy applies to chassis, organizations, and service profiles.

Note When applied to a chassis, only servers are affected.

Note Cisco UCS Manager does not suppress SNMP MIB-2 faults generated by NX-OS network operating system designed to support high performance, high reliability server access switches used in the data center. These SNMP MIB-2 faults have no association with this fault suppression policy.

- **default-iom-maint**—Suppresses faults for IOMs in a chassis or FEX.

  This policy applies only to chassis, FEXes, and IOMs.

Suppression Tasks

You can use these tasks to connect the schedule or fixed time interval and the suppression policy to a component.

Note After you create a suppression task, you can edit the fixed time interval or schedule of the task in both the Cisco UCS Manager GUI and Cisco UCS Manager CLI. However, you can only change between using a fixed time interval and using a schedule in the Cisco UCS Manager CLI.

Configuring Fault Suppression for a Chassis

Configuring Fault Suppression Tasks for a Chassis Using a Fixed Time Interval

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope chassis** *chassis-num*	Enters chassis mode for the specified chassis.
Step 2	UCS-A/chassis # **create fault-suppress-task** *name*	Creates a fault-suppress-task on the chassis, and enters fault-suppress-task mode. 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 3	UCS-A/chassis/fault-suppress-task # **set fault-suppress-policy** *policy-name*	Specifies the fault suppression policy that you want to apply. This can be one of the following: • **default-chassis-all-maint**—Suppresses faults for the chassis and all components installed into the chassis, including all servers, power supplies, fan modules, and IOMs. • **default-chassis-phys-maint**—Suppresses faults for the chassis, all fan modules, and power supplies installed into the chassis. • **default-server-maint**—Suppresses faults for servers. **Note** When applied to a chassis, only servers are affected. • **default-iom-maint**—Suppresses faults for IOMs in a chassis or FEX.
Step 4	UCS-A/chassis/fault-suppress-task # **create local-schedule**	Creates a local schedule and enters local-schedule mode.

	Command or Action	Purpose
Step 5	UCS-A/chassis/fault-suppress-task/local-schedule # **create occurrence single-one-time**	Creates a one-time occurrence, and enters single-one-time mode.
Step 6	UCS-A/chassis/fault-suppress-task/local-schedule/single-one-time # **set date** *month day-of-month year hour minute seconds*	Specifies the date and time that this occurrence should run.
Step 7	UCS-A/chassis/fault-suppress-task/local-schedule/single-one-time # **set max-duration** {**none** | *num-of-days num-of-hours num-of-minutes num-of-seconds*}	Specifies the maximum length of time that this task can run. To run the task until it is manually stopped, enter none or omit this step.
Step 8	UCS-A/chassis/fault-suppress-task/local-schedule/single-one-time # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create a fault suppression task called task2 for the chassis, apply the default-chassis-all-maint policy to the task, set the start date to January 1, 2013 at 11:00, and commit the transaction:

```
UCS-A# scope chassis 1
UCS-A/chassis # create fault-suppress-task task2
UCS-A/chassis/fault-suppress-task* # set fault-suppress-policy default-chassis-all-maint
UCS-A/chassis/fault-suppress-task* # create local-schedule
UCS-A/chassis/fault-suppress-task/local-schedule* # create occurrence single-one-time
UCS-A/chassis/fault-suppress-task/local-schedule* # set date jan 1 2013 11 00 00
UCS-A/chassis/fault-suppress-task/local-schedule* # commit-buffer
```

Configuring Fault Suppression Tasks for a Chassis Using a Schedule

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope chassis** *chassis-num*	Enters chassis mode for the specified chassis.
Step 2	UCS-A/chassis # **create fault-suppress-task** *name*	Creates a fault-suppress-task on the chassis, and enters the fault-suppress-task mode. 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 3	UCS-A/chassis/fault-suppress-task # **set schedule** *name*	Specifies the schedule that you want to use.

	Command or Action	Purpose
		Note The schedule must exist before you can use it in a fault suppression task. For more information about creating schedules, see Creating a Schedule, on page 56.
Step 4	UCS-A/chassis/fault-suppress-task # **set fault-suppress-policy** *policy-name*	Selects the fault suppression policy you want to apply. This can be one of the following: • **default-chassis-all-maint**—Suppresses faults for the chassis and all components installed into the chassis, including all servers, power supplies, fan modules, and IOMs. • **default-chassis-phys-maint**—Suppresses faults for the chassis, all fan modules, and power supplies installed into the chassis. • **default-server-maint**—Suppresses faults for servers. **Note** When applied to a chassis, only servers are affected. • **default-iom-maint**—Suppresses faults for IOMs in a chassis or FEX.
Step 5	UCS-A/chassis/fault-suppress-task # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create a fault suppression task called task1 for the chassis, apply the scheduler called weekly_maint and the default-chassis-all-maint policy to the task, and commit the transaction:

```
UCS-A# scope chassis 2
UCS-A/chassis # create fault-suppress-task task1
UCS-A/chassis/fault-suppress-task* # set schedule weekly_maint
UCS-A/chassis/fault-suppress-task* # set fault-suppress-policy default-chassis-all-maint
UCS-A/chassis/fault-suppress-task* # commit-buffer
```

Modifying Fault Suppression Tasks for a Chassis

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope chassis** *chassis-num*	Enters chassis mode for the specified chassis.
Step 2	UCS-A/chassis # **scope fault-suppress-task** *name*	Enters fault-suppress-task mode.
Step 3	UCS-A/chassis/fault-suppress-task # **set fault-suppress-policy** *policy-name*	Modifies the fault suppression policy. This can be one of the following: • **default-chassis-all-maint**—Suppresses faults for the chassis and all components installed into the chassis, including all servers, power supplies, fan modules, and IOMs. • **default-chassis-phys-maint**—Suppresses faults for the chassis, all fan modules, and power supplies installed into the chassis. • **default-server-maint**—Suppresses faults for servers. • **default-iom-maint**—Suppresses faults for IOMs in a chassis or FEX. **Note** To apply a different schedule to the fault suppression task, go to Step 4. To change the fixed time interval of the fault suppression task, go to Step 5.
Step 4	UCS-A/chassis/fault-suppress-task # **set schedule** *name*	Applies the schedule you want to use. **Note** If you change from a fixed time interval to a schedule, the fixed time interval is deleted when you commit. If you change from a schedule to a fixed time interval, the reference to the schedule is cleared when you commit.
Step 5	UCS-A/chassis/fault-suppress-task # **scope local-schedule**	Enters local-schedule mode.
Step 6	UCS-A/chassis/fault-suppress-task/local-schedule # **scope occurrence single-one-time**	Enters single-one-time mode.

	Command or Action	Purpose
Step 7	UCS-A/chassis/fault-suppress-task/local-schedule/single-one-time # **set date** *month day-of-month year hour minute seconds*	Specifies the date and time that this occurrence should run.
Step 8	UCS-A/chassis/fault-suppress-task/local-schedule/single-one-time # **set max-duration** {**none** | *num-of-days num-of-hours num-of-minutes num-of-seconds*}	Specifies the maximum length of time that this task can run. To run the task until it is manually stopped, enter none or omit this step.
Step 9	UCS-A/chassis/fault-suppress-task/local-schedule/single-one-time # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to change the date and the fault suppression policy of the fault suppression task called task2:

```
UCS-A# scope chassis 1
UCS-A/chassis # scope fault-suppress-task task2
UCS-A/chassis/fault-suppress-task # set fault-suppress-policy default-server-maint
UCS-A/chassis/fault-suppress-task* # scope local-schedule
UCS-A/chassis/fault-suppress-task/local-schedule* # scope occurrence single-one-time
UCS-A/chassis/fault-suppress-task/local-schedule/single-one-time* # set date dec 31 2013
11 00 00
UCS-A/chassis/fault-suppress-task/local-schedule/single-one-time* # commit-buffer
```

The following example shows how to apply a different schedule to the fault suppression task called task1:

```
UCS-A# scope chassis 1
UCS-A/chassis # scope fault-suppress-task task1
UCS-A/chassis/fault-suppress-task # set schedule monthly-maint
UCS-A/chassis/fault-suppress-task* # commit-buffer
```

Viewing Suppressed Faults and Fault Suppression Tasks for a Chassis

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope chassis** *chassis-num*	Enters chassis mode for the specified chassis.
Step 2	UCS-A/chassis # **show fault suppressed**	Displays the suppressed faults for the chassis. **Note** Only faults owned by the selected component are displayed.
Step 3	UCS-A/chassis # **scope fault-suppress-task** *name*	Enters fault-suppress-task mode.
Step 4	UCS-A/chassis/fault-suppress-task # **show detail expand**	Displays the schedule or fixed time interval for the task.

Example

The following example shows how to display the suppressed faults for a chassis:

```
UCS-A# scope chassis 1
UCS-A/chassis # show fault suppressed
Fault Suppress Task:

Name               Status             Global Schedule Suppress Policy Name
------------------ ------------------ --------------- --------------------
task1              Active             test_schedule1  Default Chassis Phys Maint

UCS-A/chassis #
```

The following example shows how to display the fault suppression task called task1:

```
UCS-A# scope chassis 1
UCS-A/chassis # scope fault-suppress-task task1
UCS-A/chassis/fault-suppress-task # show detail expand
Fault Suppress Task:
    Name: task1
    Status: Active
    Global Schedule: test_schedule1
    Suppress Policy Name: Default Chassis Phys Maint

UCS-A/chassis/fault-suppress-task #
```

Deleting Fault Suppression Tasks for a Chassis

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope chassis** *chassis-num*	Enters chassis mode for the specified chassis.
Step 2	UCS-A/chassis # **delete fault-suppress-task** *name*	Deletes the specified fault suppression task.
Step 3	UCS-A/chassis # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to delete the fault suppression task called task1:

```
UCS-A# scope chassis 1
UCS-A/chassis # delete fault-suppress-task task1
UCS-A/chassis* # commit-buffer
```

Configuring Fault Suppression for an I/O Module

Configuring Fault Suppression Tasks for an IOM Using a Fixed Time Interval

The **default-iom-maint** suppression policy is selected by default.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope** [**chassis** *chassis-num* | **fex** *fex-num*]	Enters chassis mode for the specified chassis or FEX.
Step 2	UCS-A /chassis|fex # **scope iom** *iom-id*	Enters chassis I/O module mode for the selected I/O module.
Step 3	UCS-A/chassis|fex/iom # **create fault-suppress-task** *name*	Creates a fault-suppress-task on the IOM, and enters the fault-suppress-task mode. 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 4	UCS-A/chassis|fex/iom/fault-suppress-task # **create local-schedule**	Creates a local schedule and enters local-schedule mode.
Step 5	UCS-A/chassis|fex/iom/fault-suppress-task/local-schedule # **create occurrence single-one-time**	Creates a one-time occurrence, and enters single-one-time mode.
Step 6	UCS-A/chassis|fex/iom/fault-suppress-task/local-schedule/single-one-time # **set date** *month day-of-month year hour minute seconds*	Specifies the date and time that this occurrence should run.
Step 7	UCS-A/chassis|fex/iom/fault-suppress-task/local-schedule/single-one-time # **set max-duration** {**none** | *num-of-days num-of-hours num-of-minutes num-of-seconds*}	Specifies the maximum length of time that this task can run. To run the task until it is manually stopped, enter none or omit this step.
Step 8	UCS-A/chassis|fex/iom/fault-suppress-task/local-schedule/single-one-time # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create a fault suppression task called task2 for the IOM on a chassis, set the start date to January 1, 2013 at 11:00, and commit the transaction:

```
UCS-A# scope chassis 1
UCS-A/chassis # scope iom a
UCS-A/chassis/iom # create fault-suppress-task task2
UCS-A/chassis/iom/fault-suppress-task* # create local-schedule
UCS-A/chassis/iom/fault-suppress-task/local-schedule* # create occurrence single-one-time
UCS-A/chassis/iom/fault-suppress-task/local-schedule/single-one-time* # set date jan 1 2013
```

```
 11 00 00
UCS-A/chassis/iom/fault-suppress-task/local-schedule/single-one-time* # commit-buffer
```

The following example shows how to create a fault suppression task called task2 for the IOM on a FEX, set the start date to January 1, 2013 at 11:00, and commit the transaction:

```
UCS-A# scope fex 1
UCS-A/fex # scope iom a
UCS-A/fex/iom # create fault-suppress-task task2
UCS-A/fex/iom/fault-suppress-task* # create local-schedule
UCS-A/fex/iom/fault-suppress-task/local-schedule* # create occurrence single-one-time
UCS-A/fex/iom/fault-suppress-task/local-schedule/single-one-time* # set date jan 1 2013 11
 00 00
UCS-A/fex/iom/fault-suppress-task/local-schedule/single-one-time* # commit-buffer
```

Configuring Fault Suppression Tasks for an IOM Using a Schedule

The **default-iom-maint** suppression policy is selected by default.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope** [**chassis** *chassis-num* | **fex** *fex-num*]	Enters chassis mode for the specified chassis or FEX.
Step 2	UCS-A /chassis|fex # **scope iom** *iom-id*	Enters chassis I/O module mode for the selected I/O module.
Step 3	UCS-A/chassis|fex/iom # **create fault-suppress-task** *name*	Creates a fault-suppress-task on the IOM, and enters the fault-suppress-task mode. 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 4	UCS-A/chassis|fex/iom/fault-suppress-task # **set schedule** *name*	Specifies the schedule that you want to use. **Note** The schedule must exist before you can use it in a fault suppression task. For more information about creating schedules, see Creating a Schedule, on page 56.
Step 5	UCS-A/chassis|fex/iom/fault-suppress-task # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create a fault suppression task called task1 for the IOM on a chassis, apply the scheduler called weekly_maint to the task, and commit the transaction:

```
UCS-A# scope chassis 1
UCS-A/chassis # scope iom a
UCS-A/chassis/iom # create fault-suppress-task task1
UCS-A/chassis/iom/fault-suppress-task* # set schedule weekly_maint
UCS-A/chassis/iom/fault-suppress-task* # commit-buffer
```

The following example shows how to create a fault suppression task called task1 for the IOM on a FEX, apply the scheduler called weekly_maint to the task, and commit the transaction:

```
UCS-A# scope fex 1
UCS-A/fex # scope iom a
UCS-A/fex/iom # create fault-suppress-task task1
UCS-A/fex/iom/fault-suppress-task* # set schedule weekly_maint
UCS-A/fex/iom/fault-suppress-task* # commit-buffer
```

Modifying Fault Suppression Tasks for an IOM

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope** [**chassis** *chassis-num* | **fex** *fex-num*]	Enters chassis mode for the specified chassis or FEX.
Step 2	UCS-A /chassis|fex # **scope iom** *iom-id*	Enters chassis I/O module mode for the selected I/O module.
Step 3	UCS-A/chassis|fex/iom # **scope fault-suppress-task** *name*	Enters fault-suppress-task mode. **Note** To apply a different schedule to the fault suppression task, go to Step 4. To change the fixed time interval of the fault suppression task, go to Step 5.
Step 4	UCS-A/chassis|fex/iom/fault-suppress-task # **set schedule** *name*	Applies a different schedule. **Note** If you change from a fixed time interval to a schedule, the fixed time interval is deleted when you commit. If you change from a schedule to a fixed time interval, the reference to the schedule is cleared when you commit.
Step 5	UCS-A/chassis|fex/iom/fault-suppress-task # **scope local-schedule**	Enters local-schedule mode.

	Command or Action	Purpose
Step 6	UCS-A/chassis|fex/iom/fault-suppress-task/local-schedule # **scope occurrence single-one-time**	Enters single-one-time mode.
Step 7	UCS-A/chassis|fex/iom/fault-suppress-task/local-schedule/single-one-time # **set date** *month day-of-month year hour minute seconds*	Specifies the date and time that this occurrence should run.
Step 8	UCS-A/chassis|fex/iom/fault-suppress-task/local-schedule/single-one-time # **set max-duration** {**none** | *num-of-days num-of-hours num-of-minutes num-of-seconds*}	Specifies the maximum length of time that this task can run. To run the task until it is manually stopped, enter none or omit this step.
Step 9	UCS-A/chassis|fex/iom/fault-suppress-task/local-schedule/single-one-time # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to change the date and the fault suppression policy of the fault suppression task called task2 for an IOM on a chassis:

```
UCS-A# scope chassis 1
UCS-A/chassis # scope iom a
UCS-A/chassis/iom # scope fault-suppress-task task2
UCS-A/chassis/iom/fault-suppress-task # scope local-schedule
UCS-A/chassis/iom/fault-suppress-task/local-schedule # scope occurrence single-one-time
UCS-A/chassis/iom/fault-suppress-task/local-schedule/single-one-time # set date dec 31 2013
 11 00 00
UCS-A/chassis/iom/fault-suppress-task/local-schedule/single-one-time* # commit-buffer
```

The following example shows how to apply a different schedule to the fault suppression task called task1 for an IOM on a FEX:

```
UCS-A# scope fex 3
UCS-A/fex # scope iom a
UCS-A/fex/iom # scope fault-suppress-task task1
UCS-A/fex/iom/fault-suppress-task # set schedule monthly-maint
UCS-A/fex/iom/fault-suppress-task* # commit-buffer
```

Viewing Suppressed Faults and Fault Suppression Tasks for an IOM

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope** [**chassis** *chassis-num* | **fex** *fex-num*]	Enters chassis mode for the specified chassis or FEX.
Step 2	UCS-A /chassis|fex # **scope iom** *iom-id*	Enters chassis I/O module mode for the selected I/O module.
Step 3	UCS-A/chassis|fex/iom # **show fault suppressed**	Displays the suppressed faults for the IOM. **Note** Only faults owned by the selected component are displayed.

	Command or Action	Purpose
Step 4	UCS-A/chassis|fex/iom # **scope fault-suppress-task** *name*	Enters fault-suppress-task mode.
Step 5	UCS-A/chassis|fex/iom/fault-suppress-task # **show detail expand**	Displays the schedule or fixed time interval for the task.

Example

The following example shows how to display the suppressed faults for an IOM on a chassis:

```
UCS-A# scope chassis 1
UCS-A/chassis # scope iom a
UCS-A/chassis/iom # show fault suppressed
Fault Suppress Task:

Name               Status             Global Schedule Suppress Policy Name
------------------ ------------------ --------------- --------------------
task1              Active             test_schedule1  Default Iom Maint

UCS-A/chassis/iom #
```

The following example shows how to display the fault suppression task called task1 for an IOM on a chassis:

```
UCS-A# scope chassis 1
UCS-A/chassis # scope iom a
UCS-A/chassis/iom # scope fault-suppress-task task1
UCS-A/chassis/iom/fault-suppress-task # show detail expand
Fault Suppress Task:
    Name: task1
    Status: Active
    Global Schedule: test_schedule1
    Suppress Policy Name: Default Iom Maint

UCS-A/chassis/iom/fault-suppress-task #
```

The following example shows how to display the fault suppression task called task1 for an IOM on a FEX:

```
UCS-A# scope fex 3
UCS-A/fex # scope iom a
UCS-A/fex/iom # scope fault-suppress-task task1
UCS-A/fex/iom/fault-suppress-task # show detail expand
Fault Suppress Task:
    Name: task1
    Status: Active
    Global Schedule: test_schedule1
    Suppress Policy Name: Default Iom Maint

UCS-A/chassis/iom/fault-suppress-task #
```

Deleting Fault Suppression Tasks for an IOM

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope** [**chassis** *chassis-num* | **fex** *fex-num*]	Enters chassis mode for the specified chassis or FEX.
Step 2	UCS-A /chassis|fex # **scope iom** *iom-id*	Enters chassis I/O module mode for the selected I/O module.
Step 3	UCS-A/chassis|fex/iom # **delete fault-suppress-task** *name*	Deletes the specified fault suppression task.
Step 4	UCS-A/chassis|fex/iom # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to delete the fault suppression task called task1 for an IOM on a chassis:

```
UCS-A# scope chassis 1
UCS-A/chassis # scope iom a
UCS-A/chassis/iom # delete fault-suppress-task task1
UCS-A/chassis/iom* # commit-buffer
```

The following example shows how to delete the fault suppression task called task1 for an IOM on a FEX:

```
UCS-A# scope fex 3
UCS-A/fex # scope iom a
UCS-A/fex/iom # delete fault-suppress-task task1
UCS-A/fex/iom* # commit-buffer
```

Configuring Fault Suppression for a FEX

Configuring Fault Suppression Tasks for a FEX Using a Fixed Time Interval

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope fex** *fex-num*	Enters fex mode for the specified FEX.
Step 2	UCS-A/fex # **create fault-suppress-task** *name*	Creates a fault-suppress-task on the fex, and enters the fault-suppress-task mode. This name can be between 1 and 16 alphanumeric characters. You cannot use spaces or any special characters other than - (hyphen),

	Command or Action	Purpose
		_ (underscore), : (colon), and . (period), and you cannot change this name after the object is saved.
Step 3	UCS-A/fex/fault-suppress-task # **set fault-suppress-policy** *policy-name*	Specifies the fault suppression policy you want to apply. This can be one of the following: • **default-fex-all-maint**—Suppresses faults for the FEX, all power supplies, fan modules, and IOMs in the FEX. • **default-fex-phys-maint**—Suppresses faults for the FEX, all fan modules and power supplies in the FEX. • **default-iom-maint**—Suppresses faults for IOMs in a chassis or FEX.
Step 4	UCS-A/fex/fault-suppress-task # **create local-schedule**	Creates a local schedule and enters local-schedule mode.
Step 5	UCS-A/fex/fault-suppress-task/local-schedule # **create occurrence single-one-time**	Creates a one-time occurrence, and enters single-one-time mode.
Step 6	UCS-A/fex/fault-suppress-task/local-schedule/single-one-time # **set date** *month day-of-month year hour minute seconds*	Specifies the date and time that this occurrence should run.
Step 7	UCS-A/fex/fault-suppress-task/local-schedule/single-one-time # **set max-duration** {**none** | *num-of-days num-of-hours num-of-minutes num-of-seconds*}	Specifies the maximum length of time that this task can run. To run the task until it is manually stopped, enter none or omit this step.
Step 8	UCS-A/fex/fault-suppress-task/local-schedule/single-one-time # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create a fault suppression task called task2 for the FEX, apply the default-fex-all-maint policy to the task, set the start date to January 1, 2013 at 11:00, and commit the transaction:

```
UCS-A# scope fex 1
UCS-A/fex # create fault-suppress-task task2
UCS-A/fex/fault-suppress-task* # set fault-suppress-policy default-fex-all-maint
UCS-A/fex/fault-suppress-task* # create local-schedule
UCS-A/fex/fault-suppress-task/local-schedule* # create occurrence single-one-time
UCS-A/fex/fault-suppress-task/local-schedule/single-one-time* # set date jan 1 2013 11 00
00
UCS-A/fex/fault-suppress-task/local-schedule/single-one-time* # commit-buffer
```

Configuring Fault Suppression Tasks for a FEX Using a Schedule

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope fex** *fex-num*	Enters fex mode for the specified FEX.
Step 2	UCS-A/fex # **create fault-suppress-task** *name*	Creates a fault-suppress-task on the fex, and enters the fault-suppress-task mode. 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 3	UCS-A/fex/fault-suppress-task # **set schedule** *name*	Specifies the schedule that you want to use. **Note** The schedule must exist before you can use it in a fault suppression task. For more information about creating schedules, see Creating a Schedule, on page 56.
Step 4	UCS-A/fex/fault-suppress-task # **set fault-suppress-policy** *policy-name*	Specifies the fault suppression policy that you want to apply. This can be one of the following: • **default-fex-all-maint**—Suppresses faults for the FEX, all power supplies, fan modules, and IOMs in the FEX. • **default-fex-phys-maint**—Suppresses faults for the FEX, all fan modules and power supplies in the FEX. • **default-iom-maint**—Suppresses faults for IOMs in a chassis or FEX.
Step 5	UCS-A/fex/fault-suppress-task # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create a fault suppression task called task1 for the FEX, apply the scheduler called weekly_maint and the default-fex-all-maint policy to the task, and commit the transaction:

```
UCS-A# scope fex 1
UCS-A/fex # create fault-suppress-task task1
UCS-A/fex/fault-suppress-task* # set schedule weekly_maint
```

```
UCS-A/fex/fault-suppress-task* # set fault-suppress-policy default-fex-all-maint
UCS-A/fex/fault-suppress-task* # commit-buffer
```

Modifying Fault Suppression Tasks for a FEX

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope fex** *fex-num*	Enters fex mode for the specified FEX.
Step 2	UCS-A/fex # **scope fault-suppress-task** *name*	Enters fault-suppress-task mode.
Step 3	UCS-A/fex/fault-suppress-task # **set fault-suppress-policy** *policy-name*	Modifies the fault suppression policy. This can be one of the following: • **default-fex-all-maint**—Suppresses faults for the FEX, all power supplies, fan modules, and IOMs in the FEX. • **default-fex-phys-maint**—Suppresses faults for the FEX, all fan modules and power supplies in the FEX. • **default-iom-maint**—Suppresses faults for IOMs in a chassis or FEX. **Note** To apply a different schedule to the fault suppression task, go to Step 4. To change the fixed time interval of the fault suppression task, go to Step 5.
Step 4	UCS-A/fex/fault-suppress-task # **set schedule** *name*	Applies a different schedule. **Note** If you change from a fixed time interval to a schedule, the fixed time interval is deleted when you commit. If you change from a schedule to a fixed time interval, the reference to the schedule is cleared when you commit.
Step 5	UCS-A/fex/fault-suppress-task # **scope local-schedule**	Enters local-schedule mode.
Step 6	UCS-A/fex/fault-suppress-task/local-schedule # **scope occurrence single-one-time**	Enters single-one-time mode.
Step 7	UCS-A/fex/fault-suppress-task/local-schedule/single-one-time # **set date** *month day-of-month year hour minute seconds*	Specifies the date and time that this occurrence should run.

	Command or Action	Purpose
Step 8	UCS-A/fex/fault-suppress-task/local-schedule/single-one-time # **set max-duration** {**none** | *num-of-days num-of-hours num-of-minutes num-of-seconds*}	Specifies the maximum length of time that this task can run. To run the task until it is manually stopped, enter none or omit this step.
Step 9	UCS-A/fex/fault-suppress-task/local-schedule/single-one-time # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to change the date and the fault suppression policy of the fault suppression task called task2:

```
UCS-A# scope fex 1
UCS-A/fex # scope fault-suppress-task task2
UCS-A/fex/fault-suppress-task # set fault-suppress-policy default-iom-maint
UCS-A/fex/fault-suppress-task* # scope local-schedule
UCS-A/fex/fault-suppress-task/local-schedule* # scope occurrence single-one-time
UCS-A/fex/fault-suppress-task/local-schedule/single-one-time* # set date dec 31 2013 11 00
 00
UCS-A/fex/fault-suppress-task/local-schedule/single-one-time* # commit-buffer
```

The following example shows how to apply a different schedule to the fault suppression task called task1:

```
UCS-A# scope fex 1
UCS-A/fex # scope fault-suppress-task task1
UCS-A/fex/fault-suppress-task # set schedule monthly-maint
UCS-A/fex/fault-suppress-task* # commit-buffer
```

Viewing Suppressed Faults and Fault Suppression Tasks for a FEX

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope fex** *fex-num*	Enters fex mode for the specified FEX.
Step 2	UCS-A/fex # **show fault suppressed**	Displays the suppressed faults for the FEX. **Note** Only faults owned by the selected component are displayed.
Step 3	UCS-A/fex # **scope fault-suppress-task** *name*	Enters fault-suppress-task mode.
Step 4	UCS-A/fex/fault-suppress-task # **show detail expand**	Displays the schedule or fixed time interval for the task.

Example

The following example shows how to display the suppressed faults for a FEX:

```
UCS-A# scope fex 1
UCS-A/fex # show fault suppressed
Fault Suppress Task:

Name               Status             Global Schedule Suppress Policy Name
------------------ ------------------ --------------- --------------------
task1              Active             test_schedule1  Default FEX Phys Maint

UCS-A/fex #
```

The following example shows how to display the fault suppression task called task1:

```
UCS-A# scope fex 1
UCS-A/fex # scope fault-suppress-task task1
UCS-A/fex/fault-suppress-task # show detail expand
Fault Suppress Task:
    Name: task1
    Status: Active
    Global Schedule: test_schedule1
    Suppress Policy Name: Default FEX Phys Maint

UCS-A/fex/fault-suppress-task #
```

Deleting Fault Suppression Tasks for a FEX

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope fex** *fex-num*	Enters fex mode for the specified FEX.
Step 2	UCS-A/fex # **delete fault-suppress-task** *name*	Deletes the specified fault suppression task.
Step 3	UCS-A/fex # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to delete the fault suppression task called task1:

```
UCS-A# scope fex 1
UCS-A/fex # delete fault-suppress-task task1
UCS-A/fex* # commit-buffer
```

Configuring Fault Suppression for a Server

Configuring Fault Suppression Tasks for a Server Using a Fixed Time Interval

The **default-server-maint** suppression policy is selected by default.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** [*chassis-num*/*server-num* | *dynamic-uuid*]	Enters server mode for the specified server.
Step 2	UCS-A/server # **create fault-suppress-task** *name*	Creates a fault-suppress-task on the server, and enters the fault-suppress-task mode. 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 3	UCS-A/server/fault-suppress-task # **create local-schedule**	Creates a local schedule and enters local-schedule mode.
Step 4	UCS-A/server/fault-suppress-task/local-schedule # **create occurrence single-one-time**	Creates a one-time occurrence, and enters single-one-time mode.
Step 5	UCS-A/server/fault-suppress-task/local-schedule/single-one-time # **set date** *month day-of-month year hour minute seconds*	Specifies the date and time that this occurrence should run.
Step 6	UCS-A/server/fault-suppress-task/local-schedule/single-one-time # **set max-duration** {**none** | *num-of-days num-of-hours num-of-minutes num-of-seconds*}	Specifies the maximum length of time that this task can run. To run the task until it is manually stopped, enter none or omit this step.
Step 7	UCS-A/server/fault-suppress-task/local-schedule/single-one-time # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create a fault suppression task called task2 for the server, set the start date to January 1, 2013 at 11:00, and commit the transaction:

```
UCS-A# scope server 1/1
UCS-A/server # create fault-suppress-task task2
UCS-A/server/fault-suppress-task* # create local-schedule
UCS-A/server/fault-suppress-task/local-schedule* # create occurrence single-one-time
UCS-A/server/fault-suppress-task/local-schedule/single-one-time* # set date jan 1 2013 11
00 00
UCS-A/server/fault-suppress-task/local-schedule/single-one-time* # commit-buffer
```

Configuring Fault Suppression Tasks for a Server using a Schedule

The **default-server-maint** suppression policy is selected by default.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** [*chassis-num*/*server-num* | *dynamic-uuid*]	Enters server mode for the specified server.
Step 2	UCS-A/server # **create fault-suppress-task** *name*	Creates a fault-suppress-task on the server, and enters the fault-suppress-task mode. 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 3	UCS-A/server/fault-suppress-task # **set schedule** *name*	Specifies the schedule that you want to use. **Note** The schedule must exist before you can use it in a fault suppression task. For more information about creating schedules, see Creating a Schedule, on page 56.
Step 4	UCS-A/server/fault-suppress-task # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to creates a fault suppression task called task1 for the server, apply the scheduler called weekly_maint to the task, and commit the transaction:

```
UCS-A# scope server 1/1
UCS-A/server # create fault-suppress-task task1
UCS-A/server/fault-suppress-task* # set schedule weekly_maint
UCS-A/server/fault-suppress-task* # commit-buffer
```

Modifying Fault Suppression Tasks for a Server

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** [*chassis-num*/*server-num* | *dynamic-uuid*]	Enters server mode for the specified server.
Step 2	UCS-A/server # **scope fault-suppress-task** *name*	Enters fault-suppress-task mode.

	Command or Action	Purpose
		Note To apply a different schedule to the fault suppression task, go to Step 3. To change the fixed time interval of the fault suppression task, go to Step 4.
Step 3	UCS-A/server/fault-suppress-task # **set schedule** *name*	Applies a different schedule. **Note** If you change from a fixed time interval to a schedule, the fixed time interval is deleted when you commit. If you change from a schedule to a fixed time interval, the reference to the schedule is cleared when you commit.
Step 4	UCS-A/server/fault-suppress-task # **scope local-schedule**	Enters local-schedule mode.
Step 5	UCS-A/server/fault-suppress-task/local-schedule # **scope occurrence single-one-time**	Enters single-one-time mode.
Step 6	UCS-A/server/fault-suppress-task/local-schedule/single-one-time # **set date** *month day-of-month year hour minute seconds*	Specifies the date and time that this occurrence should run.
Step 7	UCS-A/server/fault-suppress-task/local-schedule/single-one-time # **set max-duration** {**none** | *num-of-days num-of-hours num-of-minutes num-of-seconds*}	Specifies the maximum length of time that this task can run. To run the task until it is manually stopped, enter none or omit this step.
Step 8	UCS-A/server/fault-suppress-task/local-schedule/single-one-time # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to change the date and the fault suppression policy of the fault suppression task called task2:

```
UCS-A# scope server 1/1
UCS-A/server # scope fault-suppress-task task2
UCS-A/server/fault-suppress-task # scope local-schedule
UCS-A/server/fault-suppress-task/local-schedule # scope occurrence single-one-time
UCS-A/server/fault-suppress-task/local-schedule/single-one-time # set date dec 31 2013 11
00 00
UCS-A/server/fault-suppress-task/local-schedule/single-one-time* # commit-buffer
```

The following example shows how to apply a different schedule to the fault suppression task called task1:

```
UCS-A# scope server 1/1
UCS-A/server # scope fault-suppress-task task1
```

```
UCS-A/server/fault-suppress-task # set schedule monthly-maint
UCS-A/server/fault-suppress-task* # commit-buffer
```

Creating a Schedule

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system**	Enters system mode.
Step 2	UCS-A /system # **create scheduler** *sched-name*	Creates a scheduler and enters scheduler mode.
Step 3	UCS-A /system/scheduler # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example creates a scheduler called maintenancesched and commits the transaction:

```
UCS-A# scope system
UCS-A /system # create scheduler maintenancesched
UCS-A /system/scheduler* # commit-buffer
UCS-A /system/scheduler #
```

What to do next

Create a one time occurrence or recurring occurrence for the schedule.

Viewing Suppressed Faults and Fault Suppression Tasks for a Server

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** [*chassis-num*/*server-num* | *dynamic-uuid*]	Enters server mode for the specified server.
Step 2	UCS-A/server # **show fault suppressed**	Displays the suppressed faults for the server. **Note** Only faults owned by the selected component are displayed.
Step 3	UCS-A/server # **scope fault-suppress-task** *name*	Enters fault-suppress-task mode.
Step 4	UCS-A/server/fault-suppress-task # **show detail expand**	Displays the schedule or fixed time interval for the task.

Example

The following example shows how to display the suppressed faults for a server:

```
UCS-A# scope server 1/1
UCS-A/server # show fault suppressed
Fault Suppress Task:

Name               Status             Global Schedule Suppress Policy Name
------------------ ------------------ --------------- --------------------
task1              Active             test_schedule1  Default Server Maint

UCS-A/server #
```

The following example shows how to display the fault suppression task called task1:

```
UCS-A# scope server 1/1
UCS-A/server # scope fault-suppress-task task1
UCS-A/server/fault-suppress-task # show detail expand
Fault Suppress Task:
    Name: task1
    Status: Active
    Global Schedule: test_schedule1
    Suppress Policy Name: Default Server Maint

UCS-A/server/fault-suppress-task #
```

Deleting Fault Suppression Tasks for a Server

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** [*chassis-num*/*server-num* | *dynamic-uuid*]	Enters server mode for the specified server.
Step 2	UCS-A/server # **delete fault-suppress-task** *name*	Deletes the specified fault suppression task.
Step 3	UCS-A/server # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to delete the fault suppression task called task1:

```
UCS-A# scope server 1/1
UCS-A/server # delete fault-suppress-task task1
UCS-A/server* # commit-buffer
```

Configuring Fault Suppression for a Service Profile

Configuring Fault Suppression Tasks for a Service Profile Using a Fixed Time Interval

The **default-server-maint** suppression policy is selected by default.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter / as the *org-name*.
Step 2	UCS-A /org # **scope service-profile** *profile-name*	Enters service profile organization mode for the service profile.
Step 3	UCS-A /org/service-profile # **create fault-suppress-task** *name*	Creates a fault-suppress-task on the chassis, and enters the fault-suppress-task mode. 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 4	UCS-A/org/service-profile/fault-suppress-task # **create local-schedule**	Creates a local schedule and enters local-schedule mode.
Step 5	UCS-A/org/service-profile/fault-suppress-task/local-schedule # **create occurrence single-one-time**	Creates a one-time occurrence, and enters single-one-time mode.
Step 6	UCS-A/org/service-profile/fault-suppress-task/local-schedule/single-one-time # **set date** *month day-of-month year hour minute seconds*	Specifies the date and time that this occurrence should run.
Step 7	UCS-A/org/service-profile/fault-suppress-task/local-schedule/single-one-time # **set max-duration** {**none** | *num-of-days num-of-hours num-of-minutes num-of-seconds*}	Specifies the maximum length of time that this task can run. To run the task until it is manually stopped, enter none or omit this step.
Step 8	UCS-A/org/service-profile/fault-suppress-task/local-schedule/single-one-time # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create a fault suppression task called task2 under the accounting service profile, set the start date to January 1, 2013 at 11:00, and commit the transaction:

```
UCS-A# scope org /
UCS-A/org # scope service-profile accounting
UCS-A/org/service-profile # create fault-suppress-task task2
UCS-A/org/service-profile/fault-suppress-task* # create local-schedule
UCS-A/org/service-profile/fault-suppress-task/local-schedule* # create occurrence
```

```
single-one-time
UCS-A/org/service-profile/fault-suppress-task/local-schedule/single-one-time* # set date
jan 1 2013 11 00 00
UCS-A/org/service-profile/fault-suppress-task/local-schedule/single-one-time* # commit-buffer
```

Configuring Fault Suppression Tasks for a Service Profile Using a Schedule

The **default-server-maint** suppression policy is selected by default.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter / as the *org-name*.
Step 2	UCS-A /org # **scope service-profile** *profile-name*	Enters service profile organization mode for the service profile.
Step 3	UCS-A /org/service-profile # **create fault-suppress-task** *name*	Creates a fault-suppress-task on the chassis, and enters the fault-suppress-task mode. 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 4	UCS-A/org/service-profile/fault-suppress-task # **set schedule** *name*	Specifies the schedule that you want to use. **Note** The schedule must exist before you can use it in a fault suppression task. For more information about creating schedules, see Creating a Schedule, on page 56.
Step 5	UCS-A/org/service-profile/fault-suppress-task # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create a fault suppression task called task1 under the accounting service profile, apply the scheduler called weekly_maint to the task, and commit the transaction:

```
UCS-A# scope org /
UCS-A/org # scope service-profile accounting
UCS-A/org/service-profile # create fault-suppress-task task1
UCS-A/org/service-profile/fault-suppress-task* # set schedule weekly_maint
UCS-A/org/service-profile/fault-suppress-task* # commit-buffer
```

Modifying Fault Suppression Tasks for a Service Profile

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter **/** as the *org-name*.
Step 2	UCS-A /org # **scope service-profile** *profile-name*	Enters service profile organization mode for the service profile.
Step 3	UCS-A/org/service-profile # **scope fault-suppress-task** *name*	Enters fault-suppress-task mode. **Note** To apply a different schedule to the fault suppression task, go to Step 4. To change the fixed time interval of the fault suppression task, go to Step 5.
Step 4	UCS-A/org/service-profile/fault-suppress-task # **set schedule** *name*	Applies a different schedule. **Note** If you change from a fixed time interval to a schedule, the fixed time interval is deleted when you commit. If you change from a schedule to a fixed time interval, the reference to the schedule is cleared when you commit.
Step 5	UCS-A/org/service-profile/fault-suppress-task # **scope local-schedule**	Enters local-schedule mode.
Step 6	UCS-A/org/service-profile/fault-suppress-task/local-schedule # **scope occurrence single-one-time**	Enters single-one-time mode.
Step 7	UCS-A/org/service-profile/fault-suppress-task/local-schedule/single-one-time # **set date** *month day-of-month year hour minute seconds*	Specifies the date and time that this occurrence should run.
Step 8	UCS-A/org/service-profile/fault-suppress-task/local-schedule/single-one-time # **set max-duration** {**none** | *num-of-days num-of-hours num-of-minutes num-of-seconds*}	Specifies the maximum length of time that this task can run. To run the task until it is manually stopped, enter none or omit this step.
Step 9	UCS-A/org/service-profile/fault-suppress-task/local-schedule/single-one-time # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to change the date and the fault suppression policy of the fault suppression task called task2:

```
UCS-A# scope org /
UCS-A/org # scope service-profile accounting
UCS-A/org/service-profile # scope fault-suppress-task task2
UCS-A/org/service-profile/fault-suppress-task # scope local-schedule
UCS-A/org/service-profile/fault-suppress-task/local-schedule # scope occurrence
single-one-time
UCS-A/org/service-profile/fault-suppress-task/local-schedule/single-one-time # set date dec
 31 2013 11 00 00
UCS-A/org/service-profile/fault-suppress-task/local-schedule/single-one-time* # commit-buffer
```

The following example shows how to apply a different schedule to the fault suppression task called task1:

```
UCS-A# scope org /
UCS-A/org # scope service-profile accounting
UCS-A/org/service-profile # scope fault-suppress-task task1
UCS-A/org/service-profile/fault-suppress-task # set schedule monthly-maint
UCS-A/org/service-profile/fault-suppress-task* # commit-buffer
```

Viewing Suppressed Faults and Fault Suppression Tasks for a Service Profile

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter **/** as the *org-name*.
Step 2	UCS-A /org # **scope service-profile** *profile-name*	Enters service profile organization mode for the service profile.
Step 3	UCS-A/org/service-profile # **show fault suppressed**	Displays the suppressed faults for the server. **Note** Only faults owned by the selected component are displayed.
Step 4	UCS-A/org/service-profile # **scope fault-suppress-task** *name*	Enters fault-suppress-task mode.
Step 5	UCS-A/org/service-profile/fault-suppress-task # **show detail expand**	Displays the schedule or fixed time interval for the task.

Example

The following example shows how to display the suppressed faults for a service profile:

```
UCS-A# scope org /
UCS-A/org # scope service-profile accounting
```

```
UCS-A/org/service-profile # show fault suppressed
UCS-A/org/service-profile #
Fault Suppress Task:

Name               Status             Global Schedule Suppress Policy Name
------------------ ------------------ --------------- --------------------
task1              Active             test_schedule1  Default Server Maint

UCS-A/org/service-profile #
```

The following example shows how to display the fault suppression task called task1:

```
UCS-A# scope org /
UCS-A/org # scope service-profile accounting
UCS-A/org/service-profile # scope fault-suppress-task task1
UCS-A/org/service-profile/fault-suppress-task # show detail expand
Fault Suppress Task:
    Name: task1
    Status: Active
    Global Schedule: test_schedule1
    Suppress Policy Name: Default Server Maint

UCS-A/org/service-profile/fault-suppress-task #
```

Deleting Fault Suppression Tasks for a Service Profile

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter / as the *org-name*.
Step 2	UCS-A /org # **scope service-profile** *profile-name*	Enters service profile organization mode for the service profile.
Step 3	UCS-A/org/service-profile # **delete fault-suppress-task** *name*	Deletes the specified fault suppression task.
Step 4	UCS-A/org/service-profile # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to delete the fault suppression task called task1:

```
UCS-A# scope org /
UCS-A/org # scope service-profile accounting
UCS-A/org/service-profile # delete fault-suppress-task task1
UCS-A/org/service-profile* # commit-buffer
```

Configuring Fault Suppression for an Organization

Configuring Fault Suppression Tasks for an Organization Using a Fixed Time Interval

The **default-server-maint** suppression policy is selected by default.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter / as the *org-name*.
Step 2	UCS-A/org # **create fault-suppress-task** *name*	Creates a fault-suppress-task for the organization, and enters fault-suppress-task mode. 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 3	UCS-A/org/fault-suppress-task # **create local-schedule**	Creates a local schedule and enters local-schedule mode.
Step 4	UCS-A/org/fault-suppress-task/local-schedule # **create occurrence single-one-time**	Creates a one-time occurrence, and enters single-one-time mode.
Step 5	UCS-A/org/fault-suppress-task/local-schedule/single-one-time # **set date** *month day-of-month year hour minute seconds*	Specifies the date and time that this occurrence should run.
Step 6	UCS-A/org/fault-suppress-task/local-schedule/single-one-time # **set max-duration** {**none** | *num-of-days num-of-hours num-of-minutes num-of-seconds*}	Specifies the maximum length of time that this task can run. To run the task until it is manually stopped, enter none or omit this step.
Step 7	UCS-A/org/fault-suppress-task/local-schedule/single-one-time # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create a fault suppression task called task2 under the Root organization, set the start date to January 1, 2013 at 11:00, and commit the transaction:

```
UCS-A# scope org /
UCS-A/org # create fault-suppress-task task2
UCS-A/org/fault-suppress-task* # create local-schedule
UCS-A/org/fault-suppress-task/local-schedule* # create occurrence single-one-time
UCS-A/org/fault-suppress-task/local-schedule/single-one-time* # set date jan 1 2013 11 00
00
UCS-A/org/fault-suppress-task/local-schedule/single-one-time* # commit-buffer
```

Configuring Fault Suppression Tasks for an Organization Using a Schedule

The **default-server-maint** suppression policy is selected by default.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter / as the *org-name*.
Step 2	UCS-A/org # **create fault-suppress-task** *name*	Creates a fault-suppress-task for the organization, and enters the fault-suppress-task mode. 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 3	UCS-A/org/fault-suppress-task # **set schedule** *name*	Specifies the schedule that you want to use. **Note** The schedule must exist before you can use it in a fault suppression task. For more information about creating schedules, see Creating a Schedule, on page 56.
Step 4	UCS-A/org/fault-suppress-task # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create a fault suppression task called task1 under the Root organization, apply the scheduler called weekly_maint to the task, and commit the transaction:

```
UCS-A# scope org /
UCS-A/org # create fault-suppress-task task1
UCS-A/org/fault-suppress-task* # set schedule weekly_maint
UCS-A/org/fault-suppress-task* # commit-buffer
```

Modifying Fault Suppression Tasks for an Organization

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter **/** as the *org-name*.
Step 2	UCS-A/org # **scope fault-suppress-task** *name*	Enters fault-suppress-task mode. **Note** To apply a different schedule to the fault suppression task, go to Step 3. To change the fixed time interval of the fault suppression task, go to Step 4.
Step 3	UCS-A/org/fault-suppress-task # **set schedule** *name*	Applies a different schedule. **Note** If you change from a fixed time interval to a schedule, the fixed time interval is deleted when you commit. If you change from a schedule to a fixed time interval, the reference to the schedule is cleared when you commit.
Step 4	UCS-A/org/fault-suppress-task # **scope local-schedule**	Enters local-schedule mode.
Step 5	UCS-A/org/fault-suppress-task/local-schedule # **scope occurrence single-one-time**	Enters single-one-time mode.
Step 6	UCS-A/org/fault-suppress-task/local-schedule/single-one-time # **set date** *month day-of-month year hour minute seconds*	Specifies the date and time that this occurrence should run.
Step 7	UCS-A/org/fault-suppress-task/local-schedule/single-one-time # **set max-duration** {**none** | *num-of-days num-of-hours num-of-minutes num-of-seconds*}	Specifies the maximum length of time that this task can run. To run the task until it is manually stopped, enter none or omit this step.
Step 8	UCS-A/org/fault-suppress-task/local-schedule/single-one-time # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to change the date and the fault suppression policy of the fault suppression task called task2:

```
UCS-A# scope org /
UCS-A/org # scope fault-suppress-task task2
```

```
UCS-A/org/fault-suppress-task* # scope local-schedule
UCS-A/org/fault-suppress-task/local-schedule # scope occurrence single-one-time
UCS-A/org/fault-suppress-task/local-schedule/single-one-time # set date dec 31 2013 11 00
00
UCS-A/org/fault-suppress-task/local-schedule/single-one-time* # commit-buffer
```

The following example shows how to apply a different schedule to the fault suppression task called task1:

```
UCS-A# scope org
UCS-A/org # scope fault-suppress-task task1
UCS-A/org/fault-suppress-task # set schedule monthly-maint
UCS-A/org/fault-suppress-task* # commit-buffer
```

Viewing Suppressed Faults and Fault Suppression Tasks for an Organization

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter **/** as the *org-name*.
Step 2	UCS-A/org # **show fault suppressed**	Displays the suppressed faults for the organization **Note** Only faults owned by the selected component are displayed.
Step 3	UCS-A/org # **scope fault-suppress-task** *name*	Enters fault-suppress-task mode.
Step 4	UCS-A/org/fault-suppress-task # **show detail expand**	Displays the schedule or fixed time interval for the task.

Example

The following example shows how to display the suppressed faults for an organization:

```
UCS-A# scope org Finance
UCS-A/org # show fault suppressed
UCS-A/org #
Fault Suppress Task:

Name                Status              Global Schedule Suppress Policy Name
------------------- ------------------- --------------- --------------------
task1               Active              test_schedule1  Default Server Maint

UCS-A/org #
```

The following example shows how to display the fault suppression task called task1:

```
UCS-A# scope org Finance
UCS-A/org # scope fault-suppress-task task1
UCS-A/org/fault-suppress-task # show detail expand
```

```
Fault Suppress Task:
    Name: task1
    Status: Active
    Global Schedule: test_schedule1
    Suppress Policy Name: Default Server Maint

UCS-A/org/fault-suppress-task #
```

Deleting Fault Suppression Tasks for an Organization

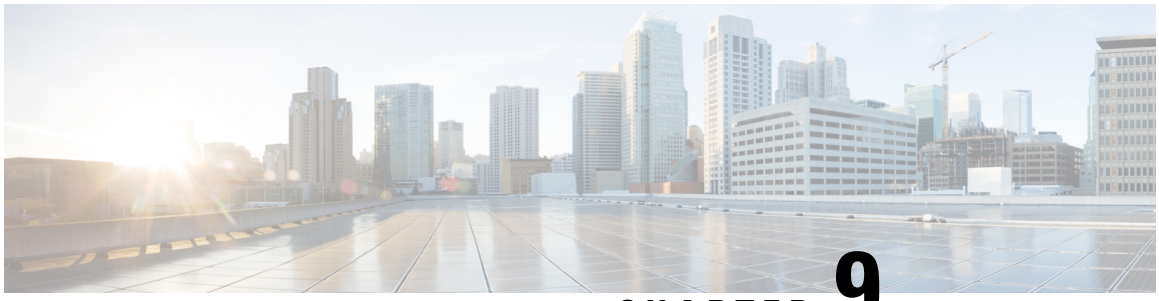
Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter / as the *org-name*.
Step 2	UCS-A/org # **delete fault-suppress-task** *name*	Deletes the specified fault suppression task.
Step 3	UCS-A/org # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to delete the fault suppression task called task1:

```
UCS-A# scope org /
UCS-A/org # delete fault-suppress-task task1
UCS-A/org* # commit-buffer
```



CHAPTER 9

SNMP Configuration

SNMP Overview

The Simple Network Management Protocol (SNMP) is an application-layer protocol that provides a message format for communication between SNMP managers and agents. SNMP provides a standardized framework and a common language for monitoring and managing devices in a network.

SNMP Functional Overview

The SNMP framework consists of three parts:

- An SNMP manager—The system used to control and monitor the activities of network devices using SNMP.
- An SNMP agent—The software component within Cisco UCS, the managed device that maintains the data for Cisco UCS, and reports the data as needed to the SNMP manager. Cisco UCS includes the agent and a collection of MIBs. To enable the SNMP agent and create the relationship between the manager and agent, enable and configure SNMP in Cisco UCS Manager.
- A managed information base (MIB)—The collection of managed objects on the SNMP agent. Cisco UCS release 1.4(1) and higher supports a larger number of MIBs than earlier releases.

Cisco UCS supports SNMPv1, SNMPv2c and SNMPv3. Both SNMPv1 and SNMPv2c use a community-based form of security. SNMP is defined in the following:

- RFC 3410 (http://tools.ietf.org/html/rfc3410)
- RFC 3411 (http://tools.ietf.org/html/rfc3411)

- RFC 3412 (http://tools.ietf.org/html/rfc3412)
- RFC 3413 (http://tools.ietf.org/html/rfc3413)
- RFC 3414 (http://tools.ietf.org/html/rfc3414)
- RFC 3415 (http://tools.ietf.org/html/rfc3415)
- RFC 3416 (http://tools.ietf.org/html/rfc3416)
- RFC 3417 (http://tools.ietf.org/html/rfc3417)
- RFC 3418 (http://tools.ietf.org/html/rfc3418)
- RFC 3584 (http://tools.ietf.org/html/rfc3584)

SNMP Notifications

A key feature of SNMP is the ability to generate notifications from an SNMP agent. These notifications do not require that requests be sent from the SNMP manager. Notifications can indicate improper user authentication, restarts, the closing of a connection, loss of connection to a neighbor router, or other significant events.

Cisco UCS Manager generates SNMP notifications as either traps or informs. Traps are less reliable than informs because the SNMP manager does not send any acknowledgment when it receives a trap, and Cisco UCS Manager cannot determine if the trap was received. An SNMP manager that receives an inform request acknowledges the message with an SNMP response Protocol Data Unit (PDU). If the Cisco UCS Manager does not receive the PDU, it can send the inform request again.

SNMP Security Levels and Privileges

SNMPv1, SNMPv2c, and SNMPv3 each represent a different security model. The security model combines with the selected security level to determine the security mechanism applied when the SNMP message is processed.

The security level determines the privileges required to view the message associated with an SNMP trap. The privilege level determines whether the message requires protection from disclosure or whether the message is authenticated. The supported security level depends on which security model is implemented. SNMP security levels support one or more of the following privileges:

- noAuthNoPriv—No authentication or encryption
- authNoPriv—Authentication but no encryption
- authPriv—Authentication and encryption

SNMPv3 provides for both security models and security levels. A security model is an authentication strategy that is set up for a user and the role in which the user resides. A security level is the permitted level of security within a security model. A combination of a security model and a security level determines which security mechanism is employed when handling an SNMP packet.

Supported Combinations of SNMP Security Models and Levels

The following table identifies the combinations of security models and levels.

Table 1: SNMP Security Models and Levels

Model	Level	Authentication	Encryption	What Happens
v1	noAuthNoPriv	Community string	No	Uses a community string match for authentication.
v2c	noAuthNoPriv	Community string	No	Uses a community string match for authentication.
v3	noAuthNoPriv	Username	No	Uses a username match for authentication.
v3	authNoPriv	HMAC-MD5 or HMAC-SHA	No	Provides authentication based on the Hash-Based Message Authentication Code (HMAC) Message Digest 5 (MD5) algorithm or the HMAC Secure Hash Algorithm (SHA).
v3	authPriv	HMAC-MD5 or HMAC-SHA	DES	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms. Provides Data Encryption Standard (DES) 56-bit encryption in addition to authentication based on the Cipher Block Chaining (CBC) DES (DES-56) standard.

SNMPv3 Security Features

SNMPv3 provides secure access to devices through a combination of authenticating and encrypting frames over the network. SNMPv3 authorizes only configured users to perform management operations and encrypts SNMP messages. The SNMPv3 User-Based Security Model (USM) refers to SNMP message-level security and offers the following services:

- Message integrity—Ensures that messages are not altered or destroyed in an unauthorized manner, and that data sequences are not altered beyond what can occur non-maliciously.
- Message origin authentication—Ensures that the identity of a message originator is verifiable.
- Message confidentiality and encryption—Ensures that information is not made available or disclosed to unauthorized individuals, entities, or processes.

SNMP Support

Cisco UCS provides the following support for SNMP:

Support for MIBs

Cisco UCS supports read-only access to MIBs.

For information about the specific MIBs available for Cisco UCS and where you can obtain them, see the http://www.cisco.com/en/US/docs/unified_computing/ucs/sw/mib/b-series/b_UCS_MIBRef.html for B-series servers, and http://www.cisco.com/en/US/docs/unified_computing/ucs/sw/mib/c-series/b_UCS_Standalone_C-Series_MIBRef.html C-series servers.

Authentication Protocols for SNMPv3 Users

Cisco UCS supports the following authentication protocols for SNMPv3 users:

- HMAC-MD5-96 (MD5)
- HMAC-SHA-96 (SHA)

AES Privacy Protocol for SNMPv3 Users

Cisco UCS uses Advanced Encryption Standard (AES) as one of the privacy protocols for SNMPv3 message encryption and conforms with RFC 3826.

The privacy password, or priv option, offers a choice of DES or 128-bit AES encryption for SNMP security encryption. If you enable AES-128 configuration and include a privacy password for an SNMPv3 user, Cisco UCS Manager uses the privacy password to generate a 128-bit AES key. The AES privacy password can have a minimum of eight characters. If the passphrases are specified in clear text, you can specify a maximum of 64 characters.

Configuring SNMP

Enabling SNMP and Configuring SNMP Properties

SNMP messages from a Cisco UCS domain display the fabric interconnect name rather than the system name.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope monitoring**	Enters monitoring mode.
Step 2	UCS-A /monitoring # **enable snmp**	Enables SNMP.
Step 3	UCS-A /monitoring # **set snmp community**	Enters snmp community mode.
Step 4	UCS-A /monitoring # **Enter a snmp community:** *community-name*	Specifies SNMP community. Use the community name as a password. The community name can be any alphanumeric string up to 32 characters.
Step 5	UCS-A /monitoring # **set snmp syscontact** *system-contact-name*	Specifies the system contact person responsible for the SNMP. The system contact name can be any alphanumeric string up to 255 characters, such as an email address or name and telephone number.

	Command or Action	Purpose
Step 6	UCS-A /monitoring # **set snmp syslocation** *system-location-name*	Specifies the location of the host on which the SNMP agent (server) runs. The system location name can be any alphanumeric string up to 512 characters.
Step 7	UCS-A /monitoring # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example enables SNMP, configures an SNMP community named SnmpCommSystem2, configures a system contact named contactperson, configures a contact location named systemlocation, and commits the transaction:

```
UCS-A# scope monitoring
UCS-A /monitoring # enable snmp
UCS-A /monitoring* # set snmp community
UCS-A /monitoring* # Enter a snmp community: SnmpCommSystem2
UCS-A /monitoring* # set snmp syscontact contactperson1
UCS-A /monitoring* # set snmp syslocation systemlocation
UCS-A /monitoring* # commit-buffer
UCS-A /monitoring #
```

What to do next

Create SNMP traps and users.

Creating an SNMP Trap

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope monitoring**	Enters monitoring mode.
Step 2	UCS-A /monitoring # **enable snmp**	Enables SNMP.
Step 3	UCS-A /monitoring # **create snmp-trap** {*hostname* | *ip-addr* | *ip6-addr*}	Creates an SNMP trap host with the specified host name, IPv4 address, or IPv6 address. The host name can be a fully qualified domain name of an IPv4 address.
Step 4	UCS-A /monitoring/snmp-trap # **set community** *community-name*	Specifies the SNMP community name to be used for the SNMP trap.
Step 5	UCS-A /monitoring/snmp-trap # **set port** *port-num*	Specifies the port to be used for the SNMP trap.
Step 6	UCS-A /monitoring/snmp-trap # **set version** {**v1** | **v2c** | **v3**}	Specifies the SNMP version and model used for the trap.

	Command or Action	Purpose
Step 7	(Optional) UCS-A /monitoring/snmp-trap # **set notificationtype** {**traps** | **informs**}	The type of trap to send. If you select v2c or v3 for the version, this can be: • **traps**—SNMP trap notifications • **informs**—SNMP inform notifications
Step 8	(Optional) UCS-A /monitoring/snmp-trap # **set v3 privilege** {**auth** | **noauth** | **priv**}	If you select v3 for the version, the privilege associated with the trap can be • **auth**—Authentication but no encryption • **noauth**—No authentication or encryption • **priv**—Authentication and encryption
Step 9	UCS-A /monitoring/snmp-trap # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example enables SNMP, creates an SNMP trap using an IPv4 address, specifies that the trap will use the SnmpCommSystem2 community on port 2, sets the version to v3, sets the notification type to traps, sets the v3 privilege to priv, and commits the transaction:

```
UCS-A# scope monitoring
UCS-A /monitoring # enable snmp
UCS-A /monitoring* # create snmp-trap 100.10.111.112
UCS-A /monitoring/snmp-trap* # set community SnmpCommSystem2
UCS-A /monitoring/snmp-trap* # set port 2
UCS-A /monitoring/snmp-trap* # set version v3
UCS-A /monitoring/snmp-trap* # set notificationtype traps
UCS-A /monitoring/snmp-trap* # set v3 privilege priv
UCS-A /monitoring/snmp-trap* # commit-buffer
UCS-A /monitoring/snmp-trap #
```

The following example enables SNMP, creates an SNMP trap using an IPv6 address, specifies that the trap will use the SnmpCommSystem3 community on port 2, sets the version to v3, sets the notification type to traps, sets the v3 privilege to priv, and commits the transaction:

```
UCS-A# scope monitoring
UCS-A /monitoring # enable snmp
UCS-A /monitoring* # create snmp-trap  2001::1
UCS-A /monitoring/snmp-trap* # set community SnmpCommSystem3
UCS-A /monitoring/snmp-trap* # set port 2
UCS-A /monitoring/snmp-trap* # set version v3
UCS-A /monitoring/snmp-trap* # set notificationtype traps
UCS-A /monitoring/snmp-trap* # set v3 privilege priv
UCS-A /monitoring/snmp-trap* # commit-buffer
UCS-A /monitoring/snmp-trap #
```

Deleting an SNMP Trap

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope monitoring**	Enters monitoring mode.
Step 2	UCS-A /monitoring # **delete snmp-trap** {*hostname* | *ip-addr*}	Deletes the specified SNMP trap host with the specified hostname or IP address.
Step 3	UCS-A /monitoring # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes the SNMP trap at IP address 192.168.100.112 and commits the transaction:

```
UCS-A# scope monitoring
UCS-A /monitoring # delete snmp-trap 192.168.100.112
UCS-A /monitoring* # commit-buffer
UCS-A /monitoring #
```

Generating Test SNMP Traps

You can generate a test SNMP trap without making any software or physical configuration change to the system.

Procedure

	Command or Action	Purpose
Step 1	connect nxos	Connects to the NX-OS operating system software.
Step 2	(nxos)# **test pfm snmp test-trap** ?	Returns the list of test trap options.
Step 3	(nxos)# **test pfm snmp test-trap** {fan | powersupply | temp_sensor}	Generates a test SNMP trap. • fan - Generate a test SNMP Trap for fan • powersupply -Generate a test SNMP Trap for Power Supply. • temp_sensor - Generate a test SNMP Trap for Temperature.

What to do next

While you run the NX-OS command, you can open another SSH session to the fabric interconnect and verify that SNMP packets are sent out from the fabric interconnect's management interface.

For complete packet:

```
(nxos)# ethanalyzer local interface mgmt capture-filter "udp port 162" limit-captured-frames
 0 detail
```

To capture just packet headers

```
(nxos)# ethanalyzer local interface mgmt capture-filter "udp port 162" limit-captured-frames
 0
```

Creating an SNMPv3 User

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope monitoring**	Enters monitoring mode.
Step 2	UCS-A /monitoring # **enable snmp**	Enables SNMP.
Step 3	UCS-A /monitoring # **create snmp-user** *user-name*	Creates the specified SNMPv3 user. An SNMP username cannot be the same as a local username. Choose an SNMP username that does not match a local username.
Step 4	UCS-A /monitoring/snmp-user # **set aes-128** {**no** | **yes**}	Enables or disables the use of AES-128 encryption.
Step 5	UCS-A /monitoring/snmp-user # **set auth** {**md5** | **sha**}	Specifies the use of MD5 or DHA authentication.
Step 6	UCS-A /monitoring/snmp-user # **set password**	Specifies the user password. After you enter the **set password** command, you are prompted to enter and confirm the password. **Note** • The *Password Strength Check* option is supported only for locally authenticated users and is not supported for SNMPv3 users. • For more information on the password guidelines, see the *Guidelines for Cisco UCS Passwords* section in Cisco UCS Manager Administration Management Guide.
Step 7	UCS-A /monitoring/snmp-user # **set priv-password**	Specifies the user privacy password. After you enter the **set priv-password** command, you are prompted to enter and confirm the privacy password.

	Command or Action	Purpose
Step 8	UCS-A /monitoring/snmp-user # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example enables SNMP, creates an SNMPv3 user named snmp-user14, disables AES-128 encryption, specifies the use of MD5 authentication, sets the password and privacy password, and commits the transaction:

```
UCS-A# scope monitoring
UCS-A /monitoring # enable snmp
UCS-A /monitoring* # create snmp-user snmp-user14
UCS-A /monitoring/snmp-user* # set aes-128 no
UCS-A /monitoring/snmp-user* # set auth md5
UCS-A /monitoring/snmp-user* # set password
Enter a password:
Confirm the password:
UCS-A /monitoring/snmp-user* # set priv-password
Enter a password:
Confirm the password:
UCS-A /monitoring/snmp-user* # commit-buffer
UCS-A /monitoring/snmp-user #
```

Deleting an SNMPv3 User

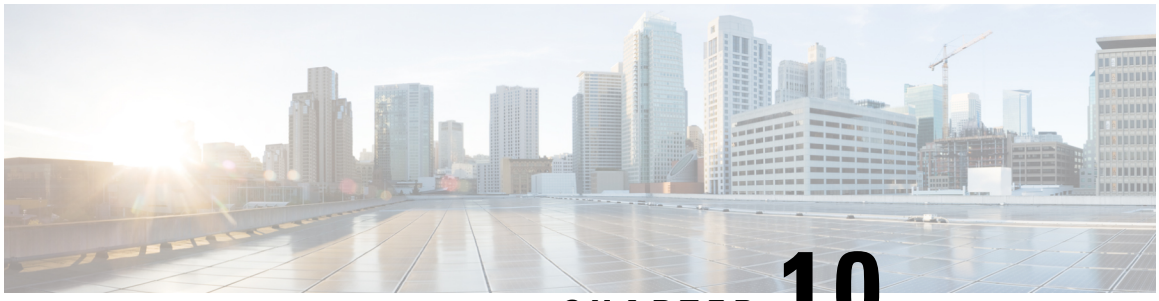
Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope monitoring**	Enters monitoring mode.
Step 2	UCS-A /monitoring # **delete snmp-user** *user-name*	Deletes the specified SNMPv3 user.
Step 3	UCS-A /monitoring # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes the SNMPv3 user named snmp-user14 and commits the transaction:

```
UCS-A# scope monitoring
UCS-A /monitoring # delete snmp-user snmp-user14
UCS-A /monitoring* # commit-buffer
UCS-A /monitoring #
```



CHAPTER 10

Statistics Collection Policy Configuration

Statistics Collection Policy

A statistics collection policy defines how frequently statistics are collected (collection interval) and how frequently the statistics are reported (reporting interval). Reporting intervals are longer than collection intervals so that multiple statistical data points can be collected during the reporting interval. This provides Cisco UCS Manager with sufficient data to calculate and report minimum, maximum, and average values.

For NIC statistics, Cisco UCS Manager displays the average, minimum, and maximum of the change since the last collection of statistics. If the values are 0, there has been no change since the last collection.

Statistics can be collected and reported for the following five functional areas of the Cisco UCS system:

- Adapter — Statistics related to the adapters
- Chassis — Statistics related to the chassis
- Host — This policy is a placeholder for future support
- Port — Statistics related to the ports, including server ports, uplink Ethernet ports, and uplink Fibre Channel ports
- Server — Statistics related to servers

Note Cisco UCS Manager has one default statistics collection policy for each of the five functional areas. You cannot create additional statistics collection policies and you cannot delete the existing default policies. You can only modify the default policies.

The values that are displayed for delta counter in Cisco UCS Manager are calculated as the difference between the last two samples in a collection interval. In addition, Cisco UCS Manager displays the average, minimum, and maximum delta values of the samples in the collection interval.

Configuring a Statistics Collection Policy

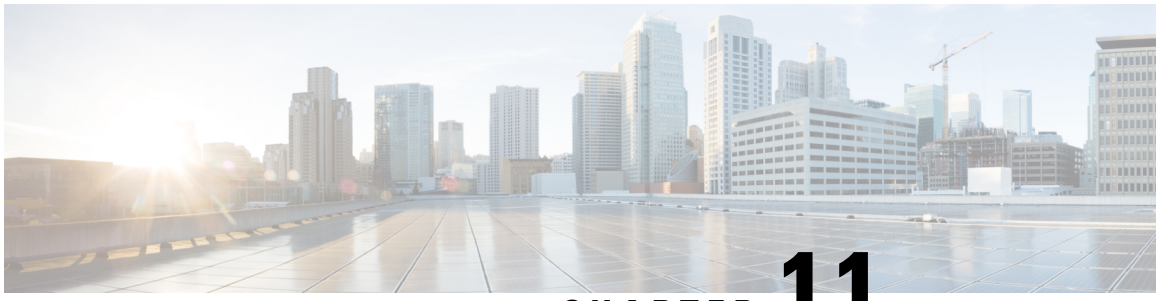
Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope monitoring**	Enters monitoring mode.
Step 2	UCS-A/monitoring # **scope stats-collection-policy** {**adapter** | **chassis** | **host** | **port** | **server**}	Enters statistics collection policy mode for the specified policy type.
Step 3	UCS-A /monitoring/stats-collection-policy # **set collection-interval** {**1minute** | **2minutes** | **30seconds** | **5minutes**}	Specifies the interval at which statistics are collected from the system.
Step 4	UCS-A /monitoring/stats-collection-policy # **set reporting-interval** {**15minutes** | **30minutes** | **60minutes**}	Specifies the interval at which collected statistics are reported.
Step 5	UCS-A /monitoring/stats-collection-policy # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example creates a statistics collection policy for ports, sets the collection interval to one minute, the reporting interval to 30 minutes, and commits the transaction:

```
UCS-A# scope monitoring
UCS-A /monitoring # scope stats-collection-policy port
UCS-A /monitoring/stats-collection-policy* # set collection-interval 1minute
UCS-A /monitoring/stats-collection-policy* # set reporting-interval 30minutes
UCS-A /monitoring/stats-collection-policy* # commit-buffer
UCS-A /monitoring/stats-collection-policy #
```



CHAPTER 11

Call Home and Smart Call Home Configuration

Call Home in UCS Overview

Call Home provides an email-based notification for critical system policies. A range of message formats are available for compatibility with pager services or XML-based automated parsing applications. You can use this feature to page a network support engineer, email a Network Operations Center, or use Cisco Smart Call Home services to generate a case with the Technical Assistance Center.

The Call Home feature can deliver alert messages containing information about diagnostics and environmental faults and events.

The Call Home feature can deliver alerts to multiple recipients, referred to as Call Home destination profiles. Each profile includes configurable message formats and content categories. A predefined destination profile is provided for sending alerts to the Cisco TAC, but you also can define your own destination profiles.

When you configure Call Home to send messages, Cisco UCS Manager executes the appropriate CLI **show** command and attaches the command output to the message.

Cisco UCS delivers Call Home messages in the following formats:

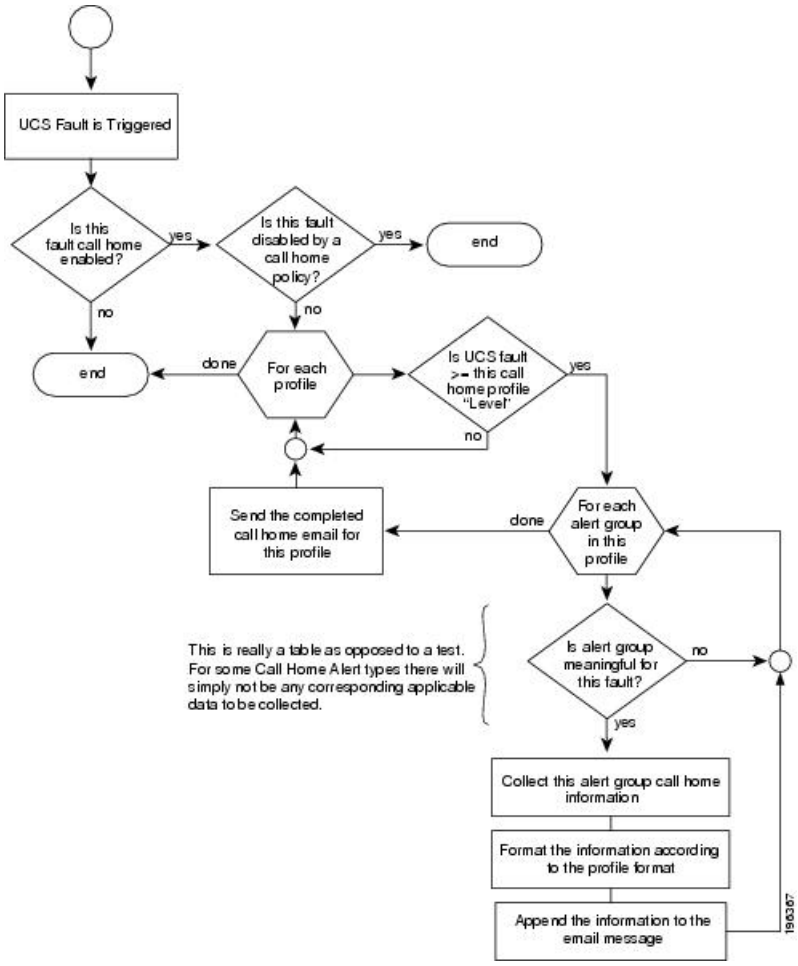
- Short text format which provides a one or two line description of the fault that is suitable for pagers or printed reports.

- Full text format which provides fully formatted message with detailed information that is suitable for human reading.
- XML machine-readable format that uses Extensible Markup Language (XML) and Adaptive Messaging Language (AML) XML Schema Definition (XSD). The AML XSD is published on the Cisco.com website. The XML format enables communication with the Cisco Systems Technical Assistance Center.

For information about the faults that can trigger Call Home email alerts, see the *Cisco UCS Faults and Error Messages Reference*.

The following figure shows the flow of events after a Cisco UCS fault is triggered in a system with Call Home configured:

Figure 2: Flow of Events after a Fault is Triggered



SMTP Authentication

Beginning with release 4.2(3b), UCS Manager supports secured authentication for the transport email with the SMTP server.

You can toggle **SMTP Authentication** between

- **Off**—SMTP Authentication is not used for this Cisco UCS domain.

- **On**—SMTP Authentication is used for this Cisco UCS domain.

Note SMTP server should be capable of supporting STARTTLS, SSL based SMTP communication.

You should also install the server root CA certificate on the SMTP-Client (switch) for succesfull connection between SSL to SMTP-AUTH server.

Call Home Considerations and Guidelines

How you configure Call Home depends on how you intend to use the feature. The information you need to consider before you configure Call Home includes the following:

Destination Profile

You must configure at least one destination profile. The destination profile or profiles that you use depends upon whether the receiving entity is a pager, email, or automated service such as Cisco Smart Call Home.

If the destination profile uses email message delivery, you must specify a Simple Mail Transfer Protocol (SMTP) server when you configure Call Home.

Contact Information

The contact email, phone, and street address information should be configured so that the receiver can determine the origin of messages received from the Cisco UCS domain.

Cisco Smart Call Home sends the registration email to this email address after you send a system inventory to begin the registration process.

If an email address includes special characters, such as # (hash), spaces, or & (ampersand), the email server might not be able to deliver email messages to that address. Cisco recommends that you use email addresses which comply with RFC2821 and RFC2822 and include only 7bit ASCII characters.

IP Connectivity to Email Server or HTTP Server

The fabric interconnect must have IP connectivity to an email server or the destination HTTP server. In a cluster configuration, both fabric interconnects must have IP connectivity. This connectivity ensures that the current, active fabric interconnect can send Call Home email messages. The source of these email messages is always the IP address of a fabric interconnect. The virtual IP address assigned to Cisco UCS Manager in a cluster configuration is never the source of the email.

Note Ensure that you add each fabric interconnect IP in the SMTP server. Call Home email messages cannot be delivered if the fabric interconnect IPs are not configured in the SMTP server.

Smart Call Home

If Cisco Smart Call Home is used, the following are required:

- An active service contract must cover the device being configured.

- The customer ID associated with the Smart Call Home configuration in Cisco UCS must be the CCO (Cisco.com) account name associated with a support contract that includes Smart Call Home.

SMTP Authentication

Beginning with release 4.2(3b), UCS Manager supports secured authentication for the transport email with the SMTP server.

You can toggle **SMTP Authentication** between

- **Off**—SMTP Authentication is not used for this Cisco UCS domain.
- **On**—SMTP Authentication is used for this Cisco UCS domain.

Note SMTP server should be capable of supporting STARTTLS, SSL based SMTP communication.

You should also install the server root CA certificate on the SMTP-Client (switch) for succesfull connection between SSL to SMTP-AUTH server.

Cisco UCS Faults and Call Home Severity Levels

Because Call Home is present across several Cisco product lines, Call Home has its own standardized severity levels. The following table describes how the underlying Cisco UCS fault levels map to the Call Home severity levels. You need to understand this mapping when you configure the Level setting for Call Home profiles.

Table 2: Mapping of Faults and Call Home Severity Levels

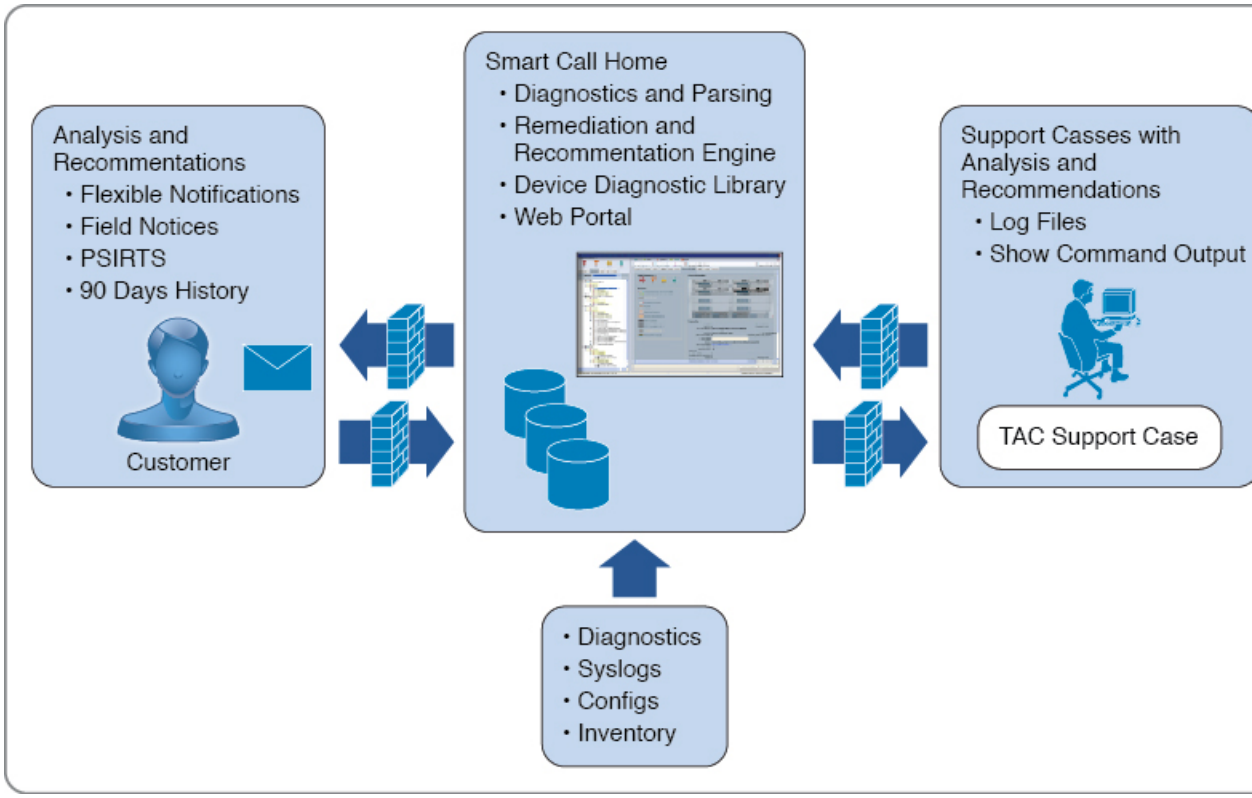
Call Home Severity	Cisco UCS Fault	Call Home Meaning
(9) Catastrophic	N/A	Network-wide catastrophic failure.
(8) Disaster	N/A	Significant network impact.
(7) Fatal	N/A	System is unusable.
(6) Critical	Critical	Critical conditions, immediate attention needed.
(5) Major	Major	Major conditions.
(4) Minor	Minor	Minor conditions.
(3) Warning	Warning	Warning conditions.
(2) Notification	Info	Basic notifications and informational messages. Possibly independently insignificant.
(1) Normal	Clear	Normal event, signifying a return to normal state.

Call Home Severity	Cisco UCS Fault	Call Home Meaning
(0) debug	N/A	Debugging messages.

Cisco Smart Call Home

Cisco Smart Call Home is a web application which leverages the Call Home feature of Cisco UCS. Smart Call Home offers proactive diagnostics and real-time email alerts of critical system events, which results in higher network availability and increased operational efficiency. Smart Call Home is a secure connected service offered by Cisco Unified Computing Support Service and Cisco Unified Computing Mission Critical Support Service for Cisco UCS.

Figure 3: Cisco Smart Call Home Features



Note Using Smart Call Home requires the following:

- A Cisco.com ID associated with a corresponding Cisco Unified Computing Support Service or Cisco Unified Computing Mission Critical Support Service contract for your company.
- Cisco Unified Computing Support Service or Cisco Unified Computing Mission Critical Support Service for the device to be registered.
- Beginning with release 4.2(3b), UCS Manager supports secured authentication for the transport email with the SMTP server. You require SMTP server, which is capable of supporting STARTTLS, SSL based SMTP communication.

You can configure and register Cisco UCS Manager to send Smart Call Home email alerts to either the Smart Call Home System or the secure Transport Gateway. Email alerts sent to the secure Transport Gateway are forwarded to the Smart Call Home System using HTTPS.

Note For security reasons, we recommend using the Transport Gateway option. The Transport Gateway can be downloaded from Cisco.com.

To configure Smart Call Home, do the following:

- Enable the Smart Call Home feature.
- Configure the contact information.
- Configure the email information.
- Configure the SMTP server information.
- Configure the default CiscoTAC-1 profile.

Note In order to apply Callhome sendtestAlert functionality at least one of the email destination should be set for profiles other than CiscoTAC-1.

- Send a Smart Call Home inventory message to start the registration process.
- Ensure that the Cisco.com ID you plan to use as the Call Home Customer ID for the Cisco UCS domain has the contract numbers from the registration added to its entitlements. You can update the ID in the **Account Properties**under **Additional Access** in the Profile Manager on Cisco.com.

SMTP Authentication

Beginning with release 4.2(3b), UCS Manager supports secured authentication for the transport email with the SMTP server.

You can toggle **SMTP Authentication** between

- **Off**—SMTP Authentication is not used for this Cisco UCS domain.

- **On**—SMTP Authentication is used for this Cisco UCS domain.

Note SMTP server should be capable of supporting STARTTLS, SSL based SMTP communication.

You should also install the server root CA certificate on the SMTP-Client (switch) for succesfull connection between SSL to SMTP-AUTH server.

Anonymous Reporting

After you upgrade to the latest release of Cisco UCS Manager, by default, you are prompted with a dialog box to enable anonymous reporting.

To enable anonymous reporting, you need to enter details about the SMTP server and the data file that is stored on the fabric switch. This report is generated every seven days and is compared with the previous version of the same report. When Cisco UCS Manager identifies changes in the report, the report is sent as an e-mail.

Configuring Call Home

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope monitoring**	Enters monitoring mode.
Step 2	UCS-A /monitoring # **scope callhome**	Enters monitoring call home mode.
Step 3	UCS-A /monitoring/callhome # **enable**	Enables Call Home.
Step 4	UCS-A /monitoring/callhome # **set contact** *name*	Specifies the name of the main Call Home contact person.
Step 5	UCS-A /monitoring/callhome # **set email** *email-addr*	Specifies the email address of the main Call Home contact person. **Note** If an email address includes special characters, such as # (hash), spaces, or & (ampersand), the email server might not be able to deliver email messages to that address. Cisco recommends that you use email addresses which comply with RFC2821 and RFC2822 and include only 7bit ASCII characters.

	Command or Action	Purpose
Step 6	UCS-A /monitoring/callhome # **set phone-contact** *phone-num*	Specifies the phone number of the main Call Home contact person. The phone number must be in international format, starting with a + (plus sign) and a country code. **Note** On Cisco UCS 6454, UCS 64108, and UCS 6536 Fabric Interconnects, ensure to limit the phone number within 17 characters. Cisco UCS Manager system may raise a fault when the phone number limit exceeds 17 characters.
Step 7	UCS-A /monitoring/callhome # **set street-address** *street-addr*	Specifies the street address of the main Call Home contact person. Enter up to 255 ASCII characters.
Step 8	UCS-A /monitoring/callhome # **set customer-id** *id-num*	Specifies the CCO identification number that includes the contract numbers for the support contract in its entitlements. The number can be up to 255 alphanumeric characters in free format.
Step 9	UCS-A /monitoring/callhome # **set contract-id** *id-num*	Specifies the contract identification number from the service agreement. The number can be up to 255 alphanumeric characters in free format.
Step 10	UCS-A /monitoring/callhome # **set site-id** *id-num*	Specifies the site identification number from the service agreement. The number can be up to 255 alphanumeric characters in free format.
Step 11	UCS-A /monitoring/callhome # **set from-email** *email-addr*	Specifies the email address to use for the **From** field in Call Home messages.
Step 12	UCS-A /monitoring/callhome # **set reply-to-email** *email-addr*	Specifies the email address to use for the **Reply To** field in Call Home messages.
Step 13	UCS-A /monitoring/callhome # **set hostname** {*hostname* | *ip-addr* | *ip6-addr*}	Specifies the hostname, IPv4 or IPv6 address of the SMTP server that Call Home uses to send email messages.
Step 14	UCS-A /monitoring/callhome # **set port** *port-num*	Specifies the SMTP server port that Call Home uses to send email messages. Valid port numbers are 1 to 65535.
Step 15	UCS-A /monitoring/callhome # **set throttling** {**off** | **on**}	Enables or disables Call Home throttling. When enabled, throttling prevents too many Call Home email messages from being sent

	Command or Action	Purpose
		for the same event. By default, throttling is enabled.
Step 16	UCS-A /monitoring/callhome # **set urgency** {**alerts** | **critical** | **debugging** | **emergencies** | **errors** | **information** | **notifications** | **warnings**}	Specifies the urgency level for Call Home email messages. In the context of a large UCS deployment with several pairs of fabric interconnects, the urgency level potentially allows you to attach significance to Call Home messages from one particular Cisco UCS domain versus another. In the context of a small UCS deployment involving only two fabric interconnects, the urgency level holds little meaning.
Step 17	UCS-A /monitoring/callhome # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example configures Call Home with and IPv4 hostname and commits the transaction:

```
UCS-A# scope monitoring
UCS-A /monitoring* # scope callhome
UCS-A /monitoring/callhome* # enable
UCS-A /monitoring/callhome* # set contact "Steve Jones"
UCS-A /monitoring/callhome* # set email admin@MyCompany.com
UCS-A /monitoring/callhome* # set phone-contact +1-001-408-555-1234
UCS-A /monitoring/callhome* # set street-address "123 N. Main Street, Anytown, CA, 99885"
UCS-A /monitoring/callhome* # set customer-id 1234567
UCS-A /monitoring/callhome* # set contract-id 99887766
UCS-A /monitoring/callhome* # set site-id 5432112
UCS-A /monitoring/callhome* # set from-email person@MyCompany.com
UCS-A /monitoring/callhome* # set reply-to-email person@MyCompany.com
UCS-A /monitoring/callhome* # set hostname 192.168.100.12
UCS-A /monitoring/callhome* # set port 25
UCS-A /monitoring/callhome* # set throttling on
UCS-A /monitoring/callhome* # set urgency information
UCS-A /monitoring/callhome* # commit-buffer
UCS-A /monitoring/callhome #
```

The following example configures Call Home with and IPv6 hostname and commits the transaction:

```
UCS-A# scope monitoring
UCS-A /monitoring* # scope callhome
UCS-A /monitoring/callhome* # enable
UCS-A /monitoring/callhome* # set contact "Steve Jones"
UCS-A /monitoring/callhome* # set email admin@MyCompany.com
UCS-A /monitoring/callhome* # set phone-contact +1-001-408-555-1234
UCS-A /monitoring/callhome* # set street-address "123 N. Main Street, Anytown, CA, 99885"
UCS-A /monitoring/callhome* # set customer-id 1234567
UCS-A /monitoring/callhome* # set contract-id 99887766
UCS-A /monitoring/callhome* # set site-id 5432112
UCS-A /monitoring/callhome* # set from-email person@MyCompany.com
UCS-A /monitoring/callhome* # set reply-to-email person@MyCompany.com
UCS-A /monitoring/callhome* # set hostname 2001::25
UCS-A /monitoring/callhome* # set port 25
```

```
UCS-A /monitoring/callhome* # set throttling on
UCS-A /monitoring/callhome* # set urgency information
UCS-A /monitoring/callhome* # commit-buffer
UCS-A /monitoring/callhome #
```

Enabling Call Home

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope monitoring**	Enters monitoring mode.
Step 2	UCS-A /monitoring # **scope callhome**	Enters monitoring call home mode.
Step 3	UCS-A /monitoring/callhome # **enable**	Enables Call Home.
Step 4	UCS-A /monitoring/callhome # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example enables Call Home and commits the transaction:

```
UCS-A# scope monitoring
UCS-A /monitoring # scope callhome
UCS-A /monitoring/callhome # enable
UCS-A /monitoring/callhome* # commit-buffer
UCS-A /monitoring/callhome #
```

Disabling Call Home

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope monitoring**	Enters monitoring mode.
Step 2	UCS-A /monitoring # **scope callhome**	Enters monitoring call home mode.
Step 3	UCS-A /monitoring/callhome # **disable**	Enables Call Home.
Step 4	UCS-A /monitoring/callhome # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example disables Call Home and commits the transaction:

```
UCS-A# scope monitoring
UCS-A /monitoring # scope callhome
UCS-A /monitoring/callhome # disable
UCS-A /monitoring/callhome* # commit-buffer
UCS-A /monitoring/callhome #
```

Configuring System Inventory Messages

Configuring System Inventory Messages

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope monitoring**	Enters monitoring mode.
Step 2	UCS-A /monitoring # **scope callhome**	Enters monitoring call home mode.
Step 3	UCS-A /monitoring/callhome # **scope inventory**	Enters monitoring call home inventory mode.
Step 4	UCS-A /monitoring/callhome/inventory # **set send-periodically** {**off** | **on**}	Enables or disables the sending of inventory messages. When the **on** keyword is specified, inventory messages are automatically sent to the Call Home database.
Step 5	UCS-A /monitoring/callhome/inventory # **set interval-days** *interval-num*	Specifies the time interval (in days) at which inventory messages will be sent.
Step 6	UCS-A /monitoring/callhome/inventory # **set timeofday-hour** *hour*	Specifies the hour (using 24-hour format) that inventory messages are sent.
Step 7	UCS-A /monitoring/callhome/inventory # **set timeofday-minute** *minute*	Specifies the number of minutes after the hour that inventory messages are sent.
Step 8	UCS-A /monitoring/callhome/inventory # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example configures Call Home system inventory messages and commits the transaction:

```
UCS-A# scope monitoring
UCS-A /monitoring* # scope callhome
UCS-A /monitoring/callhome* # scope inventory
UCS-A /monitoring/callhome/inventory* # set send-periodically on
UCS-A /monitoring/callhome/inventory* # set interval-days 15
UCS-A /monitoring/callhome/inventory* # set timeofday-hour 21
UCS-A /monitoring/callhome/inventory* # set timeofday-minute 30
UCS-A /monitoring/callhome/inventory* # commit-buffer
UCS-A /monitoring/callhome/inventory #
```

Sending a System Inventory Message

Use this procedure if you need to manually send a system inventory message outside of the scheduled messages.

Note The system inventory message is sent only to those recipients defined in CiscoTAC-1 profile.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope monitoring**	Enters monitoring mode.
Step 2	UCS-A /monitoring # **scope callhome**	Enters monitoring call home mode.
Step 3	UCS-A /monitoring/callhome # **scope inventory**	Enters monitoring call home inventory mode.
Step 4	UCS-A /monitoring/callhome/inventory # **send**	Sends the system inventory message to the Call Home database.

Example

The following example sends the system inventory message to the Call Home database:

```
UCS-A# scope monitoring
UCS-A /monitoring # scope callhome
UCS-A /monitoring/callhome # scope inventory
UCS-A /monitoring/callhome/inventory* # send
```

Configuring Call Home Profiles

Call Home Profiles

Call Home profiles determine which alerts are sent to designated recipients. You can configure the profiles to send email alerts for events and faults at a desired severity level and for specific alert groups that represent categories of alerts. You can also use these profiles to specify the format of the alert for a specific set of recipients and alert groups.

Alert groups and Call Home profiles enable you to filter the alerts and ensure that a specific profile only receives certain categories of alerts. For example, a data center may have a hardware team that handles issues with fans and power supplies. This hardware team does not care about server POST failures or licensing issues. To ensure that the hardware team only receives relevant alerts, create a Call Home profile for the hardware team and check only the "environmental" alert group.

By default, you must configure the Cisco TAC-1 profile. You can also create additional profiles to send email alerts to one or more alert groups, when events occur at the level that you specify and provide the recipients with the appropriate amount of information about those alerts.

For example, you may want to configure two profiles for faults with a major severity:

- A profile that sends an alert to the Supervisor alert group in the short text format. Members of this group receive a one- or two-line description of the fault that they can use to track the issue.
- A profile that sends an alert to the CiscoTAC alert group in the XML format. Members of this group receive a detailed message in the machine-readable format preferred by the Cisco Systems Technical Assistance Center.

Call Home Alert Groups

An alert group is a predefined subset of Call Home alerts. Alert groups allow you to select the set of Call Home alerts that you want to send to a predefined or custom Call Home profile. Cisco UCS Manager sends Call Home alerts to e-mail destinations in a destination profile only under the following conditions:

- If the Call Home alert belongs to one of the alert groups associated with that destination profile.
- If the alert has a Call Home message severity at or above the message severity set in the destination profile.

Each alert that Cisco UCS Manager generates fits into a category represented by an alert group. The following table describes those alert groups:

Alert Group	Description
Cisco TAC	All critical alerts from the other alert groups destined for Smart Call Home.
Diagnostic	Events generated by diagnostics, such as the POST completion on a server.
Environmental	Events related to power, fan, and environment-sensing elements such as temperature alarms. **Note** A Call Home alert is not generated when fans or PSUs are manually removed from the chassis. This is by design.

Configuring a Call Home Profile

By default, you must configure the Cisco TAC-1 profile, However, you can also create additional profiles to send email alerts to one or more specified groups when events occur at the level that you specify.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope monitoring**	Enters monitoring mode.
Step 2	UCS-A /monitoring # **scope callhome**	Enters monitoring call home mode.
Step 3	UCS-A /monitoring/callhome # **create profile** *profile-name*	Enters monitoring call home profile mode.
Step 4	UCS-A /monitoring/callhome/profile # **set level** {**critical** | **debug** | **disaster** | **fatal** |	Specifies the event level for the profile. Each profile can have its own unique event level.

	Command or Action	Purpose
	major | **minor** | **normal** | **notification** | **warning**}	Cisco UCS faults that are greater than or equal to the event level will trigger this profile.
Step 5	UCS-A /monitoring/callhome/profile # **set alertgroups** *group-name* • **ciscotac** • **diagnostic** • **environmental** • **inventory** • **license** • **lifecycle** • **linecard** • **supervisor** • **syslogport** • **system** • **test**	Specifies one or more groups that are alerted based on the profile. The *group-name* argument can be one or more of the following keywords entered on the same command line:
Step 6	(Optional) UCS-A /monitoring/callhome/profile # **add alertgroups** *group-names*	Adds one or more groups to the existing list of groups that are alerted based on the Call Home profile. **Note** You must use the **add alertgroups** command to add more alert groups to the existing alert group list. Using the **set alertgroups** command will replace any pre-existing alert groups with a new group list.
Step 7	UCS-A /monitoring/callhome/profile # **set format** {**shorttxt** | **xml**}	Specifies the formatting method to use for the e-mail messages.
Step 8	UCS-A /monitoring/callhome/profile # **set maxsize** *id-num*	Specifies the maximum size (in characters) of the email message.
Step 9	UCS-A /monitoring/callhome/profile # **create destination** *email-addr*	Specifies the email address to which Call Home alerts should be sent. This email address receives Callhome Alerts/Faults. Use multiple **create destination** commands in monitoring call home profile mode to specify multiple email recipients. Use the **delete destination** command in monitoring call home profile mode to delete a specified email recipient.
Step 10	UCS-A /monitoring/callhome/profile/destination # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example configures a Call Home profile and commits the transaction:

```
UCS-A# scope monitoring
UCS-A /monitoring* # scope callhome
UCS-A /monitoring/callhome* # create profile TestProfile
UCS-A /monitoring/callhome/profile* # set level normal
UCS-A /monitoring/callhome/profile* # set alertgroups test diagnostic
UCS-A /monitoring/callhome/profile* # set format xml
UCS-A /monitoring/callhome/profile* # set maxsize 100000
UCS-A /monitoring/callhome/profile* # create destination admin@MyCompany.com
UCS-A /monitoring/callhome/profile/destination* # commit-buffer
UCS-A /monitoring/callhome/profile/destination #
```

Deleting a Call Home Profile

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope monitoring**	Enters monitoring mode.
Step 2	UCS-A /monitoring # **scope callhome**	Enters monitoring call home mode.
Step 3	UCS-A /monitoring/callhome # **delete profile** *profile-name*	Deletes the specified profile.
Step 4	UCS-A /monitoring/callhome # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes the Call Home profile named TestProfile and commits the transaction:

```
UCS-A# scope monitoring
UCS-A /monitoring # scope callhome
UCS-A /monitoring/callhome # delete profile TestProfile
UCS-A /monitoring/callhome* # commit-buffer
UCS-A /monitoring/callhome #
```

Sending a Test Call Home Alert

Before you begin

Configure Call Home and a Call Home Profile.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope monitoring**	Enters monitoring mode.
Step 2	UCS-A /monitoring # **scope callhome**	Enters monitoring call home mode.
Step 3	UCS-A /monitoring/callhome # **send-test-alert** {[**alert-group** {**diagnostic** | **environmental**}] [**alert-level** {**critical** | **debug** | **fatal** | **major** | **minor** | **normal** | **notify** | **warning**}] [**alert-message-type** {**conf** | **diag** | **env** | **inventory** | **syslog** | **test**}] [**alert-message-subtype** {**delta** | **full** | **goldmajor** | **goldminor** | **goldnormal** | **major** | **minor** | **nosubtype** | **test**}] [**alert-description** *description*]}	Sends a test Call Home alert. The test Call Home alert must specify all **alert-*** parameters or Cisco UCS Manager cannot generate the test message. The **alert-*** parameters include the following: • **alert-description**—Alert description • **alert-group**—Alert group • **alert-level**—Event severity level • **alert-message-type**—Message type • **alert-message-subtype**—Message subtype When a test Call Home alert is sent, Call Home responds as it would to any other alert and delivers it to the configured destination email addresses.

Example

The following example sends a test Call Home alert to the configured destination email address of the environmental alert group:

```
UCS-A# scope monitoring
UCS-A /monitoring # scope callhome
UCS-A /monitoring/callhome # send-test-alert alert-group diagnostic
alert-level critical alert-message-type test alert-message-subtype major
alert-description "This is a test alert"
```

Configuring Call Home Policies

Call Home Policies

Call Home policies determine whether or not Call Home alerts are sent for a specific type of fault or system event. By default, Call Home is enabled to send alerts for certain types of faults and system events.

Note You can configure Cisco UCS Manager not to process the default faults and system events.

To disable alerts for a type of fault or event, you must first create a Call Home policy for that type and then disable the policy.

Configuring a Call Home Policy

Tip By default, email alerts are sent for all critical system events. However, you can optionally configure Call Home policies to enable or disable sending email alerts for other critical system events.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope monitoring**	Enters monitoring mode.
Step 2	UCS-A /monitoring # **scope callhome**	Enters monitoring call home mode.
Step 3	UCS-A /monitoring/callhome # **create policy** {**equipment-inoperable** | **fru-problem** | **identity-unestablishable** | **thermal-problem** | **voltage-problem**}	Creates the specified policy and enters monitoring call home policy mode.
Step 4	UCS-A /monitoring/callhome/policy # {**disabled** | **enabled**}	Disables or enables the sending of email alerts for the specified policy.
Step 5	UCS-A /monitoring/callhome/policy # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example creates a Call Home policy that disables the sending of email alerts for system events pertaining to voltage problems and commits the transaction:

```
UCS-A# scope monitoring
UCS-A /monitoring* # scope callhome
UCS-A /monitoring/callhome* # create policy voltage-problem
UCS-A /monitoring/callhome/policy* # disabled
UCS-A /monitoring/callhome/policy* # commit-buffer
UCS-A /monitoring/callhome/policy #
```

Disabling a Call Home Policy

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope monitoring**	Enters monitoring mode.
Step 2	UCS-A /monitoring # **scope callhome**	Enters monitoring call home mode.

	Command or Action	Purpose
Step 3	UCS-A /monitoring/callhome # **scope policy** {**equipment-inoperable** | **fru-problem** | **identity-unestablishable** | **thermal-problem** | **voltage-problem**}	Enters monitoring call home policy mode for the specified policy.
Step 4	UCS-A /monitoring/callhome/policy # **disable**	Disables the specified policy.
Step 5	UCS-A /monitoring/callhome/policy # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example disables the Call Home policy named voltage-problem and commits the transaction:

```
UCS-A# scope monitoring
UCS-A /monitoring # scope callhome
UCS-A /monitoring/callhome # scope policy voltage-problem
UCS-A /monitoring/callhome/policy # disable
UCS-A /monitoring/callhome/policy* # commit-buffer
UCS-A /monitoring/callhome/policy #
```

Enabling a Call Home Policy

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope monitoring**	Enters monitoring mode.
Step 2	UCS-A /monitoring # **scope callhome**	Enters monitoring call home mode.
Step 3	UCS-A /monitoring/callhome # **scope policy** {**equipment-inoperable** | **fru-problem** | **identity-unestablishable** | **thermal-problem** | **voltage-problem**}	Enters monitoring call home policy mode for the specified policy.
Step 4	UCS-A /monitoring/callhome/policy # **enable**	Enables the specified policy.
Step 5	UCS-A /monitoring/callhome/policy # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example enables the Call Home policy named voltage-problem and commits the transaction:

```
UCS-A# scope monitoring
UCS-A /monitoring # scope callhome
UCS-A /monitoring/callhome # scope policy voltage-problem
```

```
UCS-A /monitoring/callhome/policy # enable
UCS-A /monitoring/callhome/policy* # commit-buffer
UCS-A /monitoring/callhome/policy #
```

Deleting a Call Home Policy

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope monitoring**	Enters monitoring mode.
Step 2	UCS-A /monitoring # **scope callhome**	Enters monitoring call home mode.
Step 3	UCS-A /monitoring/callhome # **delete policy** {**equipment-inoperable** | **fru-problem** | **identity-unestablishable** | **thermal-problem** | **voltage-problem**}	Deletes the specified policy
Step 4	UCS-A /monitoring/callhome # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes the Call Home policy named voltage-problem and commits the transaction:

```
UCS-A# scope monitoring
UCS-A /monitoring # scope callhome
UCS-A /monitoring/callhome # delete policy voltage-problems
UCS-A /monitoring/callhome* # commit-buffer
UCS-A /monitoring/callhome #
```

Configuring Anonymous Reporting

Enabling Anonymous Reporting

Procedure

	Command or Action	Purpose
Step 1	UCS-A # **scope monitoring**	Enters monitoring mode.
Step 2	UCS-A/monitoring # **scope callhome**	Enters monitoring call home mode.
Step 3	(Optional) UCS-A/monitoring/callhome # **show anonymous-reporting**	Displays if anonymous reporting is enabled or disabled.

	Command or Action	Purpose
Step 4	UCS-A/monitoring/callhome # **enable anonymous-reporting**	Enables anonymous reporting on Smart Call Home.
Step 5	UCS-A/monitoring/callhome # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to enable anonymous reporting on the Call Home server:

```
UCS-A # scope monitoring
UCS-A/monitoring #scope callhome
UCS-A/mnitoring/callhome # show anonymous-reporting
Anonymous Reporting:
    Admin State
    -----------
    Off
UCS-A/monitoring/callhome* # enable anonymous-reporting
UCS-A/monitoring/callhome # commit-buffer
UCS-A/monitoring/callhome # show anonymous-reporting
Anonymous Reporting:
    Admin State
    -----------
    On
```

Disabling Anonymous Reporting

Procedure

	Command or Action	Purpose
Step 1	UCS-A # **scope monitoring**	Enters monitoring mode.
Step 2	UCS-A/monitoring # **scope callhome**	Enters monitoring call home mode.
Step 3	(Optional) UCS-A/monitoring/callhome # **show anonymous-reporting**	Displays if anonymous reporting is enabled or disabled.
Step 4	UCS-A/monitoring/callhome # **disable anonymous-reporting**	Disables anonymous reporting on the Smart Call Home server.
Step 5	UCS-A/monitoring/callhome # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to disable anonymous reporting on the Call Home server:

```
UCS-A # scope monitoring
UCS-A/monitoring # scope callhome
```

```
UCS-A/mnitoring/callhome # show anonymous-reporting
Anonymous Reporting:
    Admin State
    -----------
    On
UCS-A/monitoring/callhome* # disable anonymous-reporting
UCS-A/monitoring/callhome # commit-buffer
UCS-A/monitoring/callhome # show anonymous-reporting
Anonymous Reporting:
    Admin State
    -----------
    Off
```

Viewing Anonymous Reports

Procedure

	Command or Action	Purpose
Step 1	UCS-A # **scope monitoring**	Enters monitoring mode.
Step 2	UCS-A/monitoring # **scope callhome**	Enters monitoring call home mode.
Step 3	UCS-A/monitoring/callhome # **scope anonymous-reporting**	Enters anonymous reporting mode.
Step 4	UCS-A/monitoring/callhome/anonymous-reporting # **show detail**	Displays the SMTP server address and server port.
Step 5	UCS-A/monitoring/callhome/anonymous-reporting # **show inventory**	Displays the anonymous reporting information.
Step 6	UCS-A/monitoring/callhome/anonymous-reporting # **show content**	Displays the anonymous report sample information.

Example

The following example shows how to display anonymous reports from the Call Home server:

```
UCS-A # scope monitoring
UCS-A/monitoring # scope callhome
UCS-A/monitoring/callhome # scope anonymous-reporting
UCS-A/monitoring/callhome/anonymous-reporting # show detail
UCS-A/monitoring/callhome/anonymous-reporting # show inventory
UCS-A/monitoring/callhome/anonymous-reporting # show content
<anonymousData>
<discreteData
smartCallHomeContract="false"
ethernetMode="EndHost"
fcMode="EndHost"
disjointL2Used="false"
fabricFailoverUsed="false"
numVnicAdaptTempl="3"
numServiceProfiles="7"
updatingSPtemplUsed="false"
initialSPtemplUsed="true"
```

```
lanConnPolicyUsed="true"
sanConnPolicyUsed="false"
updatingAdaptTemplUsed="false"
initialAdaptTemplUsed="true"
numMsoftVMnets="10"
numOfVMs="3"
discreteFEX="false"
ucsCentralConnected="false"/>
<bladeUnit
chassisId="1"
slotId="4"
…..
```

Configuring Smart Call Home

Configuring Smart Call Home

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope monitoring**	Enters monitoring mode.
Step 2	UCS-A /monitoring # **scope callhome**	Enters monitoring call home mode.
Step 3	UCS-A /monitoring/callhome # **enable**	Enables Call Home.
Step 4	UCS-A /monitoring/callhome # **set contact** *name*	Cisco Smart Call Home sends the registration email to this email address.
Step 5	UCS-A /monitoring/callhome # **set email** *email-addr*	Specifies the email address of the main Call Home contact person. Cisco Smart Call Home sends the registration email to this email address.
Step 6	UCS-A /monitoring/callhome # **set phone-contact** *phone-num*	Specifies the phone number of the main Call Home contact person. The phone number must be in international format, starting with a + (plus sign) and a country code.
Step 7	UCS-A /monitoring/callhome # **set street-address** *street-addr*	Specifies the street address of the main Call Home contact person.
Step 8	UCS-A /monitoring/callhome # **set customer-id** *id-num*	Specifies the CCO identification number that includes the contract numbers for the support contract in its entitlements. The number can be up to 255 alphanumeric characters in free format.
Step 9	UCS-A /monitoring/callhome # **set contract-id** *id-num*	Specifies the contract identification number from the service agreement. The number can

	Command or Action	Purpose
		be up to 255 alphanumeric characters in free format.
Step 10	UCS-A /monitoring/callhome # **set site-id** *id-num*	Specifies the site identification number from the service agreement. The number can be up to 255 alphanumeric characters in free format.
Step 11	UCS-A /monitoring/callhome # **set from-email** *email-addr*	Specifies the email address to use for the From field in Call Home messages.
Step 12	UCS-A /monitoring/callhome # **set reply-to-email** *email-addr*	Specifies the email address to use for the Reply To field in Call Home messages.
Step 13	UCS-A /monitoring/callhome # **set hostname** {*hostname* | *ip-addr*}	Specifies the hostname or IP address of the SMTP server that Call Home uses to send email messages.
Step 14	UCS-A /monitoring/callhome # **set port** *port-num*	Specifies the SMTP server port that Call Home uses to send email messages. Valid port numbers are 1 to 65535.
Step 15	UCS-A /monitoring/callhome # **set throttling** {**off** | **on**}	Enables or disables Call Home throttling. When enabled, throttling prevents too many Call Home email messages from being sent for the same event. By default, throttling is enabled.
Step 16	UCS-A /monitoring/callhome # **set urgency** {**alerts** | **critical** | **debugging** | **emergencies** | **errors** | **information** | **notifications** | **warnings**}	Specifies the urgency level for Call Home email messages.
Step 17	UCS-A /monitoring/callhome # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example configures Call Home and commits the transaction:

```
UCS-A# scope monitoring
UCS-A /monitoring* # scope callhome
UCS-A /monitoring/callhome* # enable
UCS-A /monitoring/callhome* # set contact "Steve Jones"
UCS-A /monitoring/callhome* # set email admin@MyCompany.com
UCS-A /monitoring/callhome* # set phone-contact +1-001-408-555-1234
UCS-A /monitoring/callhome* # set street-address "123 N. Main Street, Anytown, CA, 99885"
UCS-A /monitoring/callhome* # set customer-id 1234567
UCS-A /monitoring/callhome* # set contract-id 99887766
UCS-A /monitoring/callhome* # set site-id 5432112
UCS-A /monitoring/callhome* # set from-email person@MyCompany.com
UCS-A /monitoring/callhome* # set reply-to-email person@MyCompany.com
UCS-A /monitoring/callhome* # set hostname 192.168.100.12
UCS-A /monitoring/callhome* # set port 25
UCS-A /monitoring/callhome* # set throttling on
```

```
UCS-A /monitoring/callhome* # set urgency information
UCS-A /monitoring/callhome* # commit-buffer
UCS-A /monitoring/callhome #
```

What to do next

Continue to "Configuring the Default Cisco TAC-1 Profile, on page 104" to configure a Call Home profile for use with Smart Call Home.

Configuring the Default Cisco TAC-1 Profile

The following are the default settings for the CiscoTAC-1 profile:

Note In order to apply Callhome sendtestAlert functionality at least one of the Email Destination should be set for profiles other than CiscoTAC-1.

- Level is normal
- Only the CiscoTAC alert group is selected
- Format is xml
- Maximum message size is 5000000

Before you begin

Complete the "Configuring Smart Call Home, on page 102" section.

Procedure

	Command or Action	Purpose
Step 1	UCS-A /monitoring/callhome # **scope profile CiscoTac-1**	Enters monitoring call home profile mode for the default Cisco TAC-1 profile.
Step 2	UCS-A /monitoring/callhome/profile # **set level normal**	Specifies the **normal** event level for the profile.
Step 3	UCS-A /monitoring/callhome/profile # **set alertgroups ciscotac**	Specifies the **ciscotac** alert group for the profile.
Step 4	UCS-A /monitoring/callhome/profile # **set format xml**	Specifies the e-mail message format to **xml** .
Step 5	UCS-A /monitoring/callhome/profile # **set maxsize 5000000**	Specifies the maximum size of **5000000** for email messages.
Step 6	UCS-A /monitoring/callhome/profile # **create destination callhome@cisco.com**	Specifies the email recipient to **callhome@cisco.com** .
Step 7	UCS-A /monitoring/callhome/profile/destination # **exit**	Exits to monitoring call home profile mode.

	Command or Action	Purpose
Step 8	UCS-A /monitoring/callhome/profile # **exit**	Exits to monitoring call home mode.

Example

The following example configures the default Cisco TAC-1 profile for use with Smart Call Home:

```
UCS-A /monitoring/callhome* # scope profile CiscoTac-1
UCS-A /monitoring/callhome/profile* # set level normal
UCS-A /monitoring/callhome/profile* # set alertgroups ciscotac
UCS-A /monitoring/callhome/profile* # set format xml
UCS-A /monitoring/callhome/profile* # set maxsize 5000000
UCS-A /monitoring/callhome/profile* # create destination callhome@cisco.com
UCS-A /monitoring/callhome/profile/destination* # exit
UCS-A /monitoring/callhome/profile* # exit
UCS-A /monitoring/callhome* #
```

What to do next

Continue to "Configuring a System Inventory Message for Smart Call Home, on page 105" to configure system inventory messages for use with Smart Call Home.

Configuring a System Inventory Message for Smart Call Home

Before you begin

Complete the "Configuring the Default Cisco TAC-1 Profile, on page 104" section.

Procedure

	Command or Action	Purpose
Step 1	UCS-A /monitoring/callhome # **scope inventory**	Enters monitoring call home inventory mode.
Step 2	UCS-A /monitoring/callhome/inventory # **set send-periodically** {**off** | **on**}	Enables or disables the sending of inventory messages. When the **on** keyword is specified, inventory messages are automatically sent to the Call Home database.
Step 3	UCS-A /monitoring/callhome/inventory # **set interval-days** *interval-num*	Specifies the the time interval (in days) at which inventory messages will be sent.
Step 4	UCS-A /monitoring/callhome/inventory # **set timeofday-hour** *hour*	Specifies the hour (using 24-hour format) that inventory messages are sent.
Step 5	UCS-A /monitoring/callhome/inventory # **set timeofday-minute** *minute*	Specifies the number of minutes after the hour that inventory messages are sent.
Step 6	UCS-A /monitoring/callhome/inventory # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example configures Call Home system inventory messages and commits the transaction:

```
UCS-A /monitoring/callhome* # scope inventory
UCS-A /monitoring/callhome/inventory* # set send-periodically on
UCS-A /monitoring/callhome/inventory* # set interval-days 15
UCS-A /monitoring/callhome/inventory* # set timeofday-hour 21
UCS-A /monitoring/callhome/inventory* # set timeofday-minute 30
UCS-A /monitoring/callhome/inventory* # commit-buffer
UCS-A /monitoring/callhome/inventory #
```

What to do next

Continue to "Registering Smart Call Home, on page 106" to send an inventory message that starts the Smart Call Home registration process.

Registering Smart Call Home

Before you begin

Complete the "Configuring a System Inventory Message for Smart Call Home, on page 105" section.

Procedure

	Command or Action	Purpose
Step 1	UCS-A /monitoring/callhome/inventory # **send**	Sends the system inventory message to the Smart Call Home database. When Cisco receives the system inventory, a Smart Call Home registration email is sent to the email address that you configured as the email address for the main Smart Call Home contact.

Example

The following example sends the system inventory message to the Smart Call Home database:

```
UCS-A /monitoring/callhome/inventory # send
```

What to do next

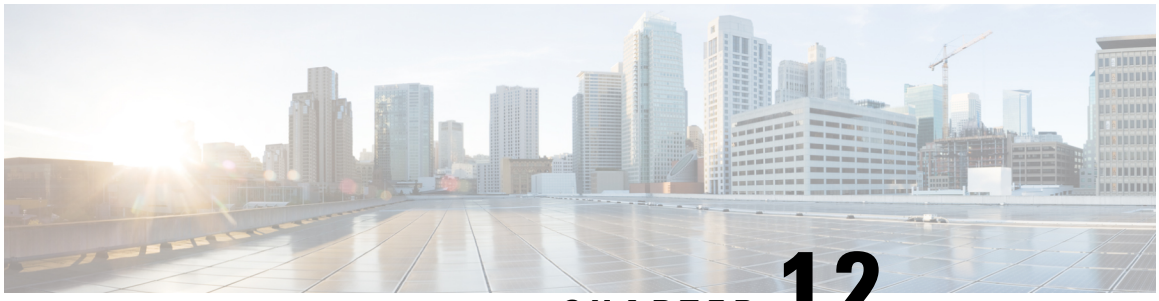
When you receive the registration email from Cisco, do the following to complete registration for Smart Call Home:

1. Click the link in the email.

   The link opens the Cisco Smart Call Home portal in your web browser.

2. Log into the Cisco Smart Call Home portal.
3. Follow the steps provided by Cisco Smart Call Home.

   After you agree to the terms and conditions, the Cisco Smart Call Home registration for the Cisco UCS domain is complete.



CHAPTER 12

Database Health Monitoring

Cisco UCS Manager Database Health Monitoring

Cisco UCS Manager uses a SQLite database stored on the Fabric Interconnects to persist configuration and inventory. Data corruption on both the Flash and NVRAM storage devices can cause failures and loss of customer configuration data. Cisco UCS Manager provides several proactive health check and recovery mechanisms to improve the integrity of the Cisco UCS Manager database. These mechanisms enable active monitoring of the database health.

- **Periodic Health Check**— A periodic check of database integrity ensures that any corruption is caught and recovered proactively. See Triggering Health Check, on page 110, and Changing Health Check Interval, on page 110.
- **Periodic Backup**— A periodic internal full state backup of the system ensures a smoother route to recovery in the case of any unrecoverable errors. See Changing Internal Backup Interval, on page 109.

Changing Internal Backup Interval

You can change the interval at which the internal backup is done. To disable the backup the value can be set to 0.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system**	Enters the system.
Step 2	UCS-A /system# **set mgmt-db-check-policy internal-backup-interval** *days*	Specifies the time interval (in days) at which the integrity backup is done.
Step 3	UCS-A /system* # **commit-buffer**	Commits the transaction.

Example

This example changes the time interval at which the check runs to two days, and commits the transaction.

```
UCS-A# scope system
UCS-A /system # set mgmt-db-check-policy health-check-interval 2
UCS-A /system* # commit-buffer
UCS-A /system #
```

Triggering Health Check

Use the following commands to trigger an immediate full database integrity check.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system**	Enters the system.
Step 2	UCS-A /system # **start-db-check**	Triggers health check.
Step 3	UCS-A /system # **commit-buffer**	Commits the transaction.

Changing Health Check Interval

You can change the interval at which the integrity check runs. To disable the periodic check entirely set the value for to 0.

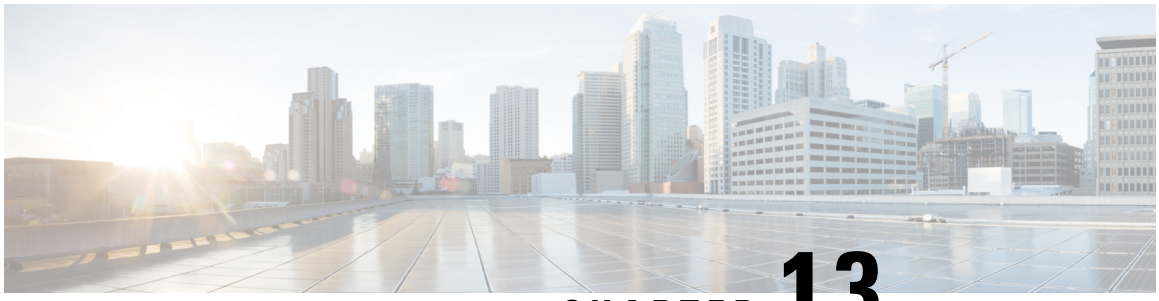
Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system**	Enters the system.
Step 2	UCS-A /system# **set mgmt-db-check-policy health-check-interval** *hours*	Specifies the time interval (in hours) at which the integrity check runs.
Step 3	UCS-A /system* # **commit-buffer**	Commits the transaction.

Example

This example changes the time interval at which the check runs to two hours , and commits the transaction.

```
UCS-A# scope system
UCS-A /system # set mgmt-db-check-policy health-check-interval 2
UCS-A /system* # commit-buffer
UCS-A /system #
```



CHAPTER 13

Hardware Monitoring

System Monitoring CLI Command Cheat Sheet

The following table provides a brief summary of Cisco UCS Manager CLI commands you use to monitor managed objects in the system.

Managed Object	Monitoring Command	Description
Hardware		
Chassis	**show chassis** [**adaptor** | **cmc** | **decommissioned** | **detail** | **environment** | **fabric** | **fi-iom** | **firmware** | **fsm** | **inventory** | **psu** | **version**]	Displays chassis information.
Fabric Interconnect	**show fabric-interconnect** [**a** | **b**] [**detail** | **environment** | **firmware** | **fsm** | **inventory** | **mac-aging** | **mode** | **version**]	Displays Fabric Interconnect information.
FEX	**show fex** [**detail** | **firmware** | **fsm** | **inventory** | **version**]	Displays Fabric Extender information
IOM	**show iom** [**firmware** | **health** | **version**]	Displays Fabric Input/Output Module information.

Managed Object	Monitoring Command	Description
Server	**show server** [**actual-boot-order** **adapter** | **assoc** | **bios** | **boot-order** | **cpu** | **decommissioned** | **environment** | **firmware** | **health** | **identity** | **inventory** | **memory** | **status** | **storage** | **version**]	Displays server information .
System	**show system** [**detail** | **firmware** | **version**]	Displays system information.
System	**scope monitoring** [**show**] [**baseline-faults** | **callhome** | **event** | **fault** | **fault-suppress-policy** | **fsm** | **mgmt-if-mon-policy** | **new-faults** | **snmp** | **snmp-trap** | **snmp-user** | **stats-collection-policy** | **stats-threshold-policy** | **syslog**]	Displays information about commands in Monitoring mode.
Logs		
Event	**show event** [*event-id* | **detail**]	Displays the Event log.
Fault	**show fault** [*fault-id* | **cause** | **detail** | **severity** | **suppressed**]	Displays the Fault log.
SEL	**show sel** [*chassis-id*/*blade-id* | *rack-id*]	Displays the System Event Log for the chassis, blade, or rack-mount server.
Syslog	**scope monitoring** [**show**] [**syslog**]	Displays the Syslog.

Managing the Chassis

Turning On the Locator LED for a Chassis

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope chassis** *chassis-num*	Enters chassis mode for the specified chassis.
Step 2	UCS-A /chassis # **enable locator-led**	Turns on the chassis locator LED.
Step 3	UCS-A /chassis # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example turns on the locator LED for chassis 2 and commits the transaction:

```
UCS-A# scope chassis 2
UCS-A /chassis # enable locator-led
UCS-A /chassis* # commit-buffer
UCS-A /chassis #
```

Turning Off the Locator LED for a Chassis

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope chassis** *chassis-num*	Enters chassis mode for the specified chassis.
Step 2	UCS-A /chassis # **disable locator-led**	Turns off the chassis locator LED.
Step 3	UCS-A /chassis # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example turns off the locator LED for chassis 2 and commits the transaction:

```
UCS-A# scope chassis 2
UCS-A /chassis # disable locator-led
UCS-A /chassis* # commit-buffer
UCS-A /chassis #
```

Managing Blade Servers

Turning On the Locator LED for a Blade Server

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *chassis-num* / *server-num*	Enters chassis server mode for the specified chassis.
Step 2	UCS-A /chassis/server # **enable locator-led** [**multi-master** | **multi-slave**]	Turns on the blade server locator LED.
Step 3	UCS-A /chassis/server # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example turns on the locator LED for blade server 4 in chassis 2 and commits the transaction:

```
UCS-A# scope server 2/4
UCS-A /chassis/server # enable  locator-led
UCS-A /chassis/server* # commit-buffer
UCS-A /chassis/server #
```

Turning Off the Locator LED for a Blade Server

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *chassis-num* / *server-num*	Enters chassis mode for the specified chassis.
Step 2	UCS-A /chassis/server # **disable locator-led** [**multi-master** | **multi-slave**]	Turns off the blade server locator LED.
Step 3	UCS-A /chassis/server # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example turns off the locator LED for blade server 4 in chassis 2 and commits the transaction:

```
UCS-A# scope chassis 2/4
UCS-A /chassis/server # disable locator-led
UCS-A /chassis/server* # commit-buffer
UCS-A /chassis/server #
```

Managing Rack-Mount servers

Turning On the Locator LED for a Rack-Mount Server

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *server-num*	Enters server mode for the specified rack-mount server.

	Command or Action	Purpose
Step 2	UCS-A /server # **enable locator-led**	Turns on the rack-mount server locator LED.
Step 3	UCS-A /server # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example turns on the locator LED for rack-mount server 2 and commits the transaction:

```
UCS-A# scope server 2
UCS-A /server # enable  locator-led
UCS-A /server* # commit-buffer
UCS-A /server #
```

Turning Off the Locator LED for a Rack-Mount Server

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *server-num*	Enters server mode for the specified rack-mount server.
Step 2	UCS-A /server # **disable locator-led**	Turns off the rack-mount server locator LED.
Step 3	UCS-A /server # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example turns off the locator LED for rack-mount server 2 and commits the transaction:

```
UCS-A# scope server 2
UCS-A /server # disable  locator-led
UCS-A /server* # commit-buffer
UCS-A /server #
```

Showing the Status for a Rack-Mount Server

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **show server status**	Shows the status for all servers in the Cisco UCS domain.

Example

The following example shows the status for all servers in the Cisco UCS domain. The servers numbered 1 and 2 do not have a slot listed in the table because they are rack-mount servers.

```
Server Slot  Status    Availability  Overall Status  Discovery
-------------------------------------------------------------------
1/1          Equipped  Unavailable   Ok              Complete
1/2          Equipped  Unavailable   Ok              Complete
1/3          Equipped  Unavailable   Ok              Complete
1/4          Empty     Unavailable   Ok              Complete
1/5          Equipped  Unavailable   Ok              Complete
1/6          Equipped  Unavailable   Ok              Complete
1/7          Empty     Unavailable   Ok              Complete
1/8          Empty     Unavailable   Ok              Complete
1            Equipped  Unavailable   Ok              Complete
2            Equipped  Unavailable   Ok              Complete
```

Monitoring Fan Modules

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope chassis** *chassis-num*	Enters chassis mode for the specified chassis.
Step 2	UCS-A /chassis # **show environment fan**	Displays the environment status for all fans within the chassis. This includes the following information: • Overall status • Operability • Power state • Thermal status • Threshold status • Voltage status
Step 3	UCS-A /chassis # **scope fan-module** *tray-num module-num*	Enters fan module chassis mode for the specified fan module. **Note** Each chassis contains one tray, so the tray number in this command is always 1.
Step 4	UCS-A /chassis/fan-module # **show** [**detail** | **expand**]	Displays the environment status for the specified fan module.

Example

The following example displays information about the fan modules in chassis 1:

```
UCS-A# scope chassis 1
UCS-A /chassis # show environment fan
Chassis 1:
    Overall Status: Power Problem
    Operability: Operable
    Power State: Redundancy Failed
    Thermal Status: Upper Non Recoverable

    Tray 1 Module 1:
        Threshold Status: OK
        Overall Status: Operable
        Operability: Operable
        Power State: On
        Thermal Status: OK
        Voltage Status: N/A

        Fan Module Stats:
             Ambient Temp (C): 25.000000

        Fan 1:
            Threshold Status: OK
            Overall Status: Operable
            Operability: Operable
            Power State: On
            Thermal Status: OK
            Voltage Status: N/A

        Fan 2:
            Threshold Status: OK
            Overall Status: Operable
            Operability: Operable
            Power State: On
            Thermal Status: OK
            Voltage Status: N/A

    Tray 1 Module 2:
        Threshold Status: OK
        Overall Status: Operable
        Operability: Operable
        Power State: On
        Thermal Status: OK
        Voltage Status: N/A

        Fan Module Stats:
             Ambient Temp (C): 24.000000

        Fan 1:
            Threshold Status: OK
            Overall Status: Operable
            Operability: Operable
            Power State: On
            Thermal Status: OK
            Voltage Status: N/A

        Fan 2:
            Threshold Status: OK
            Overall Status: Operable
            Operability: Operable
            Power State: On
```

```
            Thermal Status: OK
            Voltage Status: N/A
```

The following example displays information about fan module 2 in chassis 1:

```
UCS-A# scope chassis 1
UCS-A /chassis # scope fan-module 1 2
UCS-A /chassis/fan-module # show detail
Fan Module:
    Tray: 1
    Module: 2
    Overall Status: Operable
    Operability: Operable
    Threshold Status: OK
    Power State: On
    Presence: Equipped
    Thermal Status: OK
    Product Name: Fan Module for UCS 5108 Blade Server Chassis
    PID: N20-FAN5
    VID: V01
    Vendor: Cisco Systems Inc
    Serial (SN): NWG14350B6N
    HW Revision: 0
    Mfg Date: 1997-04-01T08:41:00.000
```

Monitoring Management Interfaces

Management Interfaces Monitoring Policy

The management interfaces monitoring policy defines how the mgmt0 Ethernet interface on the fabric interconnect is monitored. If Cisco UCS Manager detects a management interface failure, a failure report is generated. If the configured number of failure reports is reached, the system assumes that the management interface is unavailable and generates a fault. By default, the management interfaces monitoring policy is enabled.

When the management interface of a fabric interconnect which is currently the managing instance fails, Cisco UCS Manager first confirms if the status of the subordinate fabric interconnect is up. In addition, if there are no current failure reports logged against the fabric interconnect, Cisco UCS Manager modifies the managing instance for the endpoints.

If the affected fabric interconnect is currently the primary in a high availability setup, a failover of the management plane is triggered. This failover does not affect the data plane. You can set the following properties related to monitoring the management interface:

- The type of mechanism used to monitor the management interface.
- The interval at which the status of the management interface is monitored.
- The maximum number of monitoring attempts that can fail before the system assumes that the management is unavailable and generates a fault message.

Important When the management interface fails on a fabric interconnect, the managing instance may not change if one of the following occurs:

- A path to the endpoint through the subordinate fabric interconnect does not exist.
- The management interface for the subordinate fabric interconnect has failed.
- The path to the endpoint through the subordinate fabric interconnect has failed.

Configuring the Management Interfaces Monitoring Policy

Procedure

Step 1 Enter monitoring mode.

UCS-A# **scope monitoring**

Step 2 Enable or disable the management interfaces monitoring policy.

UCS-A /monitoring # **set mgmt-if-mon-policy admin-state** {**enabled** | **disabled**}

Step 3 Specify the number of seconds that the system should wait between data recordings.

UCS-A /monitoring # **set mgmt-if-mon-policy poll-interval**

Enter an integer between 90 and 300.

Step 4 Specify the maximum number of monitoring attempts that can fail before the system assumes that the management interface is unavailable and generates a fault message.

UCS-A /monitoring # **set mgmt-if-mon-policy max-fail-reports** *num-mon-attempts*

Enter an integer between 2 and 5.

Step 5 Specify the monitoring mechanism that you want the system to use.

UCS-A /monitoring # **set mgmt-if-mon-policy monitor-mechanism** {**mii-status** | **ping-arp-targets** | **ping-gateway**

- **mii-status** —The system monitors the availability of the Media Independent Interface (MII).
- **ping-arp-targets** —The system pings designated targets using the Address Resolution Protocol (ARP).
- **ping-gateway** —The system pings the default gateway address specified for this Cisco UCS domain in the management interface.

Step 6 If you selected **mii-status** as your monitoring mechanism, configure the following properties:

a) Specify the number of seconds that the system should wait before requesting another response from the MII if a previous attempt fails.

UCS-A /monitoring # **set mgmt-if-mon-policy mii-retry-interval** *num-seconds*

Enter an integer between 3 and 10.

b) Specify the number of times that the system polls the MII until the system assumes that the interface is unavailable.

UCS-A /monitoring # **set mgmt-if-mon-policy mii-retry-count** *num-retries*

Enter an integer between 1 and 3.

Step 7 If you selected **ping-arp-targets** as your monitoring mechanism, configure the following properties:

a) Specify the first IPv4 or IPv6 address the system pings.

UCS-A /monitoring # **set mgmt-if-mon-policy** {*arp-target1* | *ndisc-target1*} {*ipv4-addr* | *ipv6-addr*}

Type 0.0.0.0 for an IPv4 address to remove the ARP target or :: for an IPv6 address to remove the N-disc target.

b) Specify the second IPv4 or IPv6 address the system pings.

UCS-A /monitoring # **set mgmt-if-mon-policy** {*arp-target2* | *ndisc-target2*} {*ipv4-addr* | *ipv6-addr*}

Type 0.0.0.0 for an IPv4 address to remove the ARP target or :: for an IPv6 address to remove the N-disc target.

c) Specify the third IPv4 or IPv6 address the system pings.

UCS-A /monitoring # **set mgmt-if-mon-policy** {*arp-target3* | *ndisc-target3*} {*ipv4-addr* | *ipv6-addr*}

Type 0.0.0.0 for an IPv4 address to remove the ARP target or :: for an IPv6 address to remove the N-disc target.

Note The ping IPv4 ARP or IPv6 N-disc targets must be in the same subnet or prefix, respectively, as the fabric interconnect.

d) Specify the number of ARP requests to send to the target IP addresses.

UCS-A /monitoring # **set mgmt-if-mon-policy arp-requests** *num-requests*

Enter an integer between 1 and 5.

e) Specify the number of seconds to wait for responses from the ARP targets before the system assumes that they are unavailable.

UCS-A /monitoring # **set mgmt-if-mon-policy arp-deadline** *num-seconds*

Enter a number between 5 and 15.

Step 8 If you selected **ping-gateway** as your monitoring mechanism, configure the following properties:

a) Specify the number of times the system should ping the gateway.

UCS-A /monitoring # **set mgmt-if-mon-policy ping-requests**

Enter an integer between 1 and 5.

b) Specify the number of seconds to wait for a response from the gateway until the system assumes that the address is unavailable.

UCS-A /monitoring # **set mgmt-if-mon-policy ping-deadline**

Enter an integer between 5 and 15.

Step 9 UCS-A /monitoring # **commit-buffer**

Commits the transaction to the system configuration.

Example

The following example creates a monitoring interface management policy using the Media Independent Interface (MII) monitoring mechanism and commits the transaction:

```
UCS-A# scope monitoring
UCS-A /monitoring # set mgmt-if-mon-policy admin-state enabled
UCS-A /monitoring* # set mgmt-if-mon-policy poll-interval 250
UCS-A /monitoring* # set mgmt-if-mon-policy max-fail-reports 2
UCS-A /monitoring* # set mgmt-if-mon-policy monitor-mechanism set mii-status
UCS-A /monitoring* # set mgmt-if-mon-policy mii-retry-count 3
UCS-A /monitoring* # set mgmt-if-mon-policy mii-retry-interval 7
UCS-A /monitoring* # commit-buffer
UCS-A /monitoring #
```

Local Storage Monitoring

Local storage monitoring in Cisco UCS provides status information on local storage that is physically attached to a blade or rack server. This includes RAID controllers, physical drives and drive groups, virtual drives, RAID controller batteries (Battery Backup Unit), Transportable Flash Modules (TFM), supercapacitors, FlexFlash controllers, and SD cards.

Cisco UCS Manager communicates directly with the LSI MegaRAID controllers and FlexFlash controllers using an out-of-band interface, which enables real-time updates. Some of the information that is displayed includes:

- RAID controller status and rebuild rate.
- The drive state, power state, link speed, operability, and firmware version of physical drives.
- The drive state, operability, strip size, access policies, drive cache, and health of virtual drives.
- The operability of a BBU, whether it is a supercap or battery, and information about the TFM.

  LSI storage controllers use a Transportable Flash Module (TFM) powered by a supercapacitor to provide RAID cache protection.
- Information on SD cards and FlexFlash controllers, including RAID health and RAID state, card health, and operability.
- Information on operations that are running on the storage component, such as rebuild, initialization, and relearning.

Note After a CIMC reboot or build upgrades, the status, start time, and end times of operations running on the storage component may not be displayed correctly.

- Detailed fault information for all local storage components.

Note All faults are displayed on the **Faults** tab.

Support for Local Storage Monitoring

The type of monitoring supported depends upon the Cisco UCS server.

Supported Cisco UCS Servers for Local Storage Monitoring

Through Cisco UCS Manager, you can monitor local storage components for the following servers:

- Cisco UCS X210c M7 Compute Node
- Cisco UCS X210c M6 Compute Node
- Cisco UCS B200 M6 Server
- Cisco UCS B200 M5 Server
- Cisco UCS B480 M5 Server
- Cisco UCS C240 M7 Server
- Cisco UCS C220 M7 Server
- Cisco UCS C240 M6 Server
- Cisco UCS C245 M6 Server
- Cisco UCS C220 M6 Server
- Cisco UCS C225 M6 Server
- Cisco UCS C240 M5 Server
- Cisco UCS C480 M5 Server
- Cisco UCS C220 M5 Server

Prerequisites for Local Storage Monitoring

These prerequisites must be met for local storage monitoring or legacy disk drive monitoring to provide useful status information:

- The drive must be inserted in the server drive bay.
- The server must be powered on.
- The server must have completed discovery.
- The results of the BIOS POST complete must be TRUE.

Legacy Disk Drive Monitoring

Note The following information is applicable only for B200 M1/M2 and B250 M1/M2 blade servers.

The legacy disk drive monitoring for Cisco UCS provides Cisco UCS Manager with blade-resident disk drive status for supported blade servers in a Cisco UCS domain. Disk drive monitoring provides a unidirectional fault signal from the LSI firmware to Cisco UCS Manager to provide status information.

The following server and firmware components gather, send, and aggregate information about the disk drive status in a server:

- Physical presence sensor—Determines whether the disk drive is inserted in the server drive bay.
- Physical fault sensor—Determines the operability status reported by the LSI storage controller firmware for the disk drive.
- IPMI disk drive fault and presence sensors—Sends the sensor results to Cisco UCS Manager.
- Disk drive fault LED control and associated IPMI sensors—Controls disk drive fault LED states (on/off) and relays the states to Cisco UCS Manager.

Turning On the Local Disk Locator LED

Procedure

Step 1 UCS-A# **scope server** *id*

Enters server mode for the specified server.

Step 2 UCS-A/server # **scope local-disk** *id*

Enters the RAID controller for the specified local disk.

Step 3 UCS-A /server/local-disk # **enable locator-led**

Turns on the disk locator LED.

Step 4 UCS-A/server/local-disk* # **commit-buffer**

Commits the command to the system configuration.

Example

The following example displays how to turn on the local disk Locator LED:

```
UCS-A# scope server 1
UCS-A /server/raid-controller # scope local-disk 2
USA-A /server/raid-controller/local-disk # enable locator-led
USA-A /server/raid-controller/local-disk* # commit-buffer
```

Turning Off the Local Disk Locator LED

Procedure

Step 1 UCS-A# **scope server** *id*

Enters server mode for the specified server.

Step 2 UCS-A/server # **scope local-disk** *id*

Enters the RAID controller for the specified local disk.

Step 3 UCS-A/server/local-disk # **disable locator-led**

Turns off the disk locator LED.

Step 4 UCS-A/server/raid-controller/local-disk* # **commit-buffer**

Commits the command to the system configuration.

Example

The following example displays how to disable the local disk Locator LED:

```
UCS-A# server 1
UCS-A /server # scope local-disk 2
USA-A /server/local-disk # disable locator-led
USA-A /server/local-disk* # commit-buffer
```

Viewing the Local Disk Locator LED State

Procedure

Step 1 UCS-A# **scope server** *id*

Enters server mode for the specified server.

Step 2 UCS-A/server # **scope local-disk** *id*

Enters the RAID controller for the specified local disk.

Step 3 UCS-A/server/local-disk # **show locator-led**

Shows the state of the disk locator LED.

Example

The following example shows that the state of the local disk Locator LED is on:

```
USA-A# scope server 1
USA-A /server # scope local-disk 2
USA-A /serverlocal-disk # show locator-led
Locator LED:
    Equipment        Operational State
    ---------------- -----------------
    1/SAS-1/2        On
```

Flash Life Wear Level Monitoring

Flash life wear level monitoring enables you to monitor the life span of solid state drives. You can view both the percentage of the flash life remaining, and the flash life status. Wear level monitoring is supported on the Fusion IO mezzanine card with the following Cisco UCS blade servers:

- Cisco UCS B22 M3 blade server
- Cisco UCS B200 M3 blade server
- Cisco UCS B420 M3 blade server
- Cisco UCS B200 M4 blade server
- Cisco UCS B260 M4 blade server
- Cisco UCS B460 M4 blade server

Note Wear level monitoring requires the following:

- Cisco UCS Manager must be at release 2.2(2a) or greater.
- The Fusion IO mezzanine card firmware must be at version 7.1.15 or greater.

Viewing Flash Life Status

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *chassis-id* / *server-id*	Enters chassis server mode for the specified server.
Step 2	UCS-A /chassis/server # **show raid-controller detail expand**	Displays details for the RAID controller.

Example

The following example shows how to display the flash life status for server 3:

```
UCS-A# scope server 1/3
UCS-A /chassis/server # show raid-controller detail expand
```

```
RAID Controller:
    ID: 1
    Type: FLASH
    PCI Addr: 131:00.0
    Vendor: Cisco Systems Inc
    Model: UCSC-F-FIO-1205M
    Serial: 1315D2B52
    HW Rev: FLASH
    Raid Support: No
    OOB Interface Supported: No
    Rebuild Rate: N/A
    Controller Status: Unknown

    Flash Life:
    Flash Percentage: N/A
    FLash Status: Error(244)

UCS-A /chassis/server #
```

Viewing the Status of Local Storage Components

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *chassis-id* / *server-id*	Enters chassis server mode for the specified server.
Step 2	UCS-A /chassis/server # **show inventory storage**	Displays the local and virtual storage information for the server.

Example

The following example shows how to display the local disk status for server 2:

```
UCS-A# scope server 1/2
UCS-A /chassis/server # show inventory storage
Server 1/2:
    Name:
    User Label:
    Equipped PID: UCSB-B200-M3
    Equipped VID: V01
    Equipped Serial (SN): FCH16207KXG
    Slot Status: Equipped
    Acknowledged Product Name: Cisco UCS B200 M3
    Acknowledged PID: UCSB-B200-M3
    Acknowledged VID: V01
    Acknowledged Serial (SN): FCH16207KXG
    Acknowledged Memory (MB): 98304
    Acknowledged Effective Memory (MB): 98304
    Acknowledged Cores: 12
    Acknowledged Adapters: 1
    Motherboard:
        Product Name: Cisco UCS B200 M3
        PID: UCSB-B200-M3
        VID: V01
        Vendor: Cisco Systems Inc
        Serial (SN): FCH16207KXG
```

```
HW Revision: 0

RAID Controller 1:
    Type: SAS
    Vendor: LSI Logic   Symbios Logic
    Model: LSI MegaRAID SAS 2004 ROMB
    Serial: LSIROMB-0
    HW Revision: B2
    PCI Addr: 01:00.0
    Raid Support: RAID0, RAID1
    OOB Interface Supported: Yes
    Rebuild Rate: 31
    Controller Status: Optimal

  Local Disk 1:
        Product Name: 146GB 6Gb SAS 10K RPM SFF HDD/hot plug/drive sled mounted
        PID: A03-D146GA2
        VID: V01
        Vendor: SEAGATE
        Model: ST9146803SS
        Vendor Description: Seagate Technology LLC
        Serial: 3SD31S4X
        HW Rev: 0
        Block Size: 512
        Blocks: 285155328
        Operability: Operable
        Oper Qualifier Reason: N/A
        Presence: Equipped
        Size (MB): 139236
        Drive State: Online
        Power State: Active
        Link Speed: 6 Gbps
        Device Type: HDD

    Local Disk 2:
        Product Name: 600G AL12SE SAS Hard Disk Drive
        PID: A03-D600GA2
        VID: V01
        Vendor: TOSHIBA
        Model: MBF2600RC
        Vendor Description: Toshiba Corporation
        Serial: EA00PB109T4A
        HW Rev: 0
        Block Size: 512
        Blocks: 1169920000
        Operability: Operable
        Oper Qualifier Reason: N/A
        Presence: Equipped
        Size (MB): 571250
        Drive State: Online
        Power State: Active
        Link Speed: 6 Gbps
        Device Type: HDD

    Local Disk Config Definition:
        Mode: RAID 1 Mirrored
        Description:
        Protect Configuration: No

    Virtual Drive 0:
        Type: RAID 1 Mirrored
        Block Size: 512
        Blocks: 285155328
        Operability: Operable
```

```
                Presence: Equipped
                Size (MB): 139236
                Lifecycle: Allocated
                Drive State: Optimal
                Strip Size (KB): 64
                Access Policy: Read Write
                Read Policy: Normal
                Configured Write Cache Policy: Write Through
                Actual Write Cache Policy: Write Through
                IO Policy: Direct
                Drive Cache: No Change
                Bootable: False

UCS-A /chassis/server #
```

The following example shows how to display the local disk status for server 2 with PCIe\NVMe Flash Storage:

```
UCS-A# scope server 1/2
UCS-A /chassis/server # show inventory storage
Server 1/2:
Name:
    Acknowledged Serial (SN): FCH1901V0FK
    Acknowledged Product Name: Cisco UCS C240 M4S2
    Acknowledged PID: UCSC-C240-M4S2
    Acknowledged VID: 0
    Acknowledged Memory (MB): 16384
    Acknowledged Effective Memory (MB): 16384
    Acknowledged Cores: 24
    Acknowledged Adapters: 4
    Motherboard:
        Product Name: Cisco UCS C240 M4S2
        PID: UCSC-C240-M4S2
        VID: V01
        Vendor: Cisco Systems Inc
        Serial (SN): FCH1901V0FK
        HW Revision: 0

        Raid Controller 1:
            Type: NVMe
            Vendor: HGST
            Model: HUSPR3280ADP301
            Serial: STM0001A74F2
            HW Revision:
            PCI Addr: 42:00.0
            Raid Support: No
            OOB Interface Supported: Yes
            Rebuild Rate: 0
            Controller Status: Optimal


            Local Disk 2:
                Product Name: Cisco UCS 800GB 2.5 in NVMe based PCIeSSD
                PID: UCS-SDHPCIE800GB
                VID:
                Vendor: HGST
                Model: HUSPR3280ADP301
                Vendor Description:
                Serial: 14310CF8E975
                HW Rev: 0
                Block Size: 512
                Blocks: 285155328
                Operability: NA
                Oper Qualifier Reason: N/A
```

```
                Presence: Equipped
                Size: 94413
                Drive State: NA
                Power State: NA
                Link Speed: NA
                Device Type: SSD
                Thermal: N/A
UCS-A /chassis/server #
```

The following example shows how to display the local disk status for Cisco UCS (P3600) 2.5 inches 800 GB NVMe based PCIe SSD:

```
RAID Controller:
    ID: 1
    Type: NVME
    PCI Addr: 69:00.0
    Vendor: Intel
    Model: SSDPE2ME800G4K
    Serial: CVMD6083003D800GGN
    HW Rev:
    Raid Support: No
    OOB Interface Supported: Yes
    Mode: NVME
    Rebuild Rate: 0
    Controller Status: Optimal
    Config State: Not Applied
    Pinned Cache Status: Disabled
    Sub OEM ID: 0
    Supported Strip Sizes: Not Applicable
    Default Strip Size: Unknown
   PCI Slot: FrontPCIe5
    Product Variant: default
    Product Name: Cisco UCS (P3600) 2.5 inches 800 GB NVMe based PCIe SSD
    PID: UCS-PCI25-8003
    VID:
    Part Number:
    Storage Controller Admin State: Unspecified
    Vendor Id: 0x8086
    Subvendor Id: 0x1137
    Device Id: 0x953
    Subdevice Id: 0x15b
    Current Task:

    Local Disk:
        ID: 5
        Block Size: 512
        Physical Block Size: Unknown
        Blocks: 1562822656
        Size: 763097
        Technology:
        Operability: N/A
        Oper Qualifier Reason: N/A
        Presence: Equipped
        Connection Protocol: NVME
        Product Variant: default
        Product Name: Cisco UCS (P3600) 2.5 inches 800 GB NVMe based PCIe SSD
        PID: UCS-PCI25-8003
        VID:
        Vendor: Intel
        Model: SSDPE2ME800G4K
        Vendor Description:
        Serial: CVMD6083003D800GGN
        HW Rev: 0
        Drive State: Unknown
```

```
        Power State: Unknown
        Link Speed: Unknown
        Enclosure Association Type: Unknown
        Device Version: N/A
        Device Type: SSD
        Thermal: N/A
        Admin State Type: N/A
        Admin Virtual Drive ID: Unspecified
        Current Task:
```

The following example shows how to display the status for Cisco UCS (P3600) HHHL 2000 GB NVMe based PCIe SSD:

```
RAID Controller:
    ID: 3
    Type: NVME
    PCI Addr: 01:00.0
    Vendor: Intel
    Model: SSDPEDME020T401
    Serial: CVMD543200AQ2P0EGN
    HW Rev:
    Raid Support: No
    OOB Interface Supported: Yes
    Mode: NVME
    Rebuild Rate: 0
    Controller Status: Optimal
    Config State: Not Applied
    Pinned Cache Status: Disabled
    Sub OEM ID: 0
    Supported Strip Sizes: Not Applicable
    Default Strip Size: Unknown
    PCI Slot: 2
    Product Variant: default
    Product Name: Cisco UCS (P3600) HHHL 2000 GB NVMe based PCIe SSD
    PID: UCSC-F-I20003
    VID:
    Part Number:
    Storage Controller Admin State: Unspecified
    Vendor Id: 0x8086
    Subvendor Id: 0x1137
    Device Id: 0x953
    Subdevice Id: 0x1ac
    Current Task:

    Embedded Storage:
        Size: 2000000
        Block Size: 512
        Number Of Blocks: 3906250000
```

Viewing the Status of a Disk Drive

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope chassis** *chassis-num*	Enters chassis mode for the specified chassis.
Step 2	UCS-A /chassis # **scope server** *server-num*	Enters server chassis mode.

	Command or Action	Purpose
Step 3	UCS-A /chassis/server # **scope raid-controller** *raid-contr-id* {**sas** | **sata**}	Enters RAID controller server chassis mode.
Step 4	UCS-A /chassis/server/raid-controller # **show local-disk** [*local-disk-id* | **detail** | **expand**]	

Example

The following example shows the status of a disk drive:

```
UCS-A# scope chassis 1
UCS-A /chassis # scope server 6
UCS-A /chassis/server # scope raid-controller 1 sas
UCS-A /chassis/server/raid-controller # show local-disk 1

Local Disk:
    ID: 1
    Block Size: 512
    Blocks: 60545024
    Size (MB): 29563
    Operability: Operable
    Presence: Equipped
```

Viewing RAID Controller Operations

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *chassis-id* / *server-id*	Enters chassis server mode for the specified server.
Step 2	UCS-A /chassis/server # **show raid-controller operation**	Displays the long running operations for the RAID controller.

Example

The following example shows how to display the RAID controller operations for server 3:

```
UCS-A# scope server 1/3
UCS-A /chassis/server # show raid-controller operation

    Name: Rebuild
    Affected Object: sys/chassis-1/blade-3/board/storage-SAS-1/disk-1
    State: In Progress
    Progress: 4
    Start Time: 2013-11-05T12:02:10.000
    End Time: N/A

UCS-A /chassis/server #
```

Viewing RAID Controller Stats

The following procedure shows how to display controller stats for a server with PCIe\NVMe Flash Storage:

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *chassis-id* / *server-id*	Enters chassis server mode for the specified server.
Step 2	UCS-A /chassis/server # **scope raid-controller** *raid-contr-id* **{flash | sas | sata | sd | unknown}**	Enters RAID controller server chassis mode.
Step 3	UCS-A /chassis/server/raid-controller # **show stats**	Displays the raid controller stats.

Example

The following example shows how to display the RAID controller stats:

```
UCS-A# scope server 1/3
UCS-A /chassis/server # scope raid-controller
UCS-A /chassis/server/raid-controller # show stats

Nvme Stats:
    Time Collected: 2016-06-22T12:37:55.043
    Monitored Object: sys/rack-unit-6/board/storage-NVME-1/nvme-stats
    Suspect: Yes
    Temperature (C): 27.000000
    Life Used Percentage: 0
    Thresholded: 0

UCS-A /chassis/server/raid-controller #
```

Monitoring RAID Battery Status

This procedure applies only to Cisco UCS servers that support RAID configuration and TFM. If the Battery Backup Unit (BBU) has failed or is predicted to fail, you should replace the unit as soon as possible.

Procedure

	Command or Action	Purpose
Step 1	UCS-A # **scope chassis** *chassis-num*	Enters chassis mode for the specified chassis.
Step 2	UCS-A /chassis #**scope server** *server-num*	Enters server chassis mode.
Step 3	UCS-A /chassis/server # **scope raid-controller** *raid-contr-id* {**flash** | **sas** | **sata** | **sd** | **unknown**}	Enters RAID controller server chassis mode.
Step 4	UCS-A /chassis/server/raid-controller # **show raid-battery expand**	Displays the RAID battery status.

Example

This example shows how to view information on the BBU of a server:

```
UCS-A # scope chassis 1
UCS-A /chassis #scope server 3
UCS-A /chassis/server #scope raid-controller 1 sas
UCS-A /chassis/server/raid-controller # show raid-battery expand
RAID Battery:
    Battery Type: Supercap
    Presence: Equipped
    Operability: Operable
    Oper Qualifier Reason:
    Vendor: LSI
    Model: SuperCaP
    Serial: 0
    Capacity Percentage: Full
    Battery Temperature (C): 54.000000

    Transportable Flash Module:
        Presence: Equipped
        Vendor: Cisco Systems Inc
        Model: UCSB-RAID-1GBFM
        Serial: FCH164279W6
```

Graphics Card Monitoring

Graphics Card Server Support

With Cisco UCS Manager, you can view the properties for certain graphics cards and controllers. Graphics cards are supported on the following servers:

- Cisco UCS X410c M7 Compute Node
- Cisco UCS X210c M7 Compute Node
- Cisco UCS X210c M6 Compute Node
- Cisco UCS C240 M7 Server
- Cisco UCS C220 M7 Server
- Cisco UCS C240 M6 Server
- Cisco UCS C220 M6 Server
- Cisco UCS C245 M6 Server
- Cisco UCS C225 M6 Server
- Cisco UCS C240 M5 Server
- Cisco UCS C220 M5 Server
- Cisco UCS B200 M6 Server
- Cisco UCS B480 M6 Server

- Cisco UCS B200 M5 Server
- Cisco UCS B480 M5 Server

Note Certain NVIDIA Graphics Processing Units (GPU) do not support Error Correcting Code (ECC) and vGPU together. Cisco recommends that you refer to the release notes published by NVIDIA for the respective GPU to know whether it supports ECC and vGPU together.

Viewing Graphics Card Properties

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *blade-id*	Enters server mode for the specified server.
Step 2	UCS-A /server # **show graphics-card detail**	Displays information about the graphics card.

Example

The following example shows how to display the graphics card properties on server 1:

```
UCS-A# scope server 1
UCS-A /server # show graphics-card detail

ID: 1
Slot Id: 2
Magma Expander Slot Id:
Is Supported: Yes
Vendor: Cisco Systems Inc
Model: UCSB-GPU-M6
Serial: FHH1924002B
Mode: Graphics
PID: UCSB-GPU-M6
Firmware Version: 84.04.89.00.01|2754.0200.01.02
Vendor Id: 0x10de
Subvendor Id: 0x10de
Device Id: 0x13f3
Subdevice Id: 0x1143

UCS-A /server #
```

Viewing Graphics Controller Properties

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *blade-id*	Enters server mode for the specified server.

	Command or Action	Purpose
Step 2	UCS-A /server # **scope graphics-card** *card-id*	Enters graphics card mode for the specified graphics card.
Step 3	UCS-A /server/graphics-card # **show graphics-controller detail**	Displays information about the graphics controllers.

Example

The following example shows how to display the graphics controller properties for graphics card 1 on server 1:

```
UCS-A# scope server 1
UCS-A /server # scope graphics-card 1
UCS-A /server/graphics-card # show graphics-controller detail
Graphics Controller:
    ID: 1
    Pci Address: 07:00.0

    ID: 2
    Pci Address: 08:00.0
UCS-A /server/graphics-card #
```

PCI Switch Monitoring

PCI Switch Server Support

With Cisco UCS Manager, you can view the properties for PCI switches. PCI switches are supported on the following servers:

- Cisco UCS C480 M5 ML Server

Viewing PCI Switch Properties

PCI Switch properties are visible only for servers which support PCI switch.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *server-num*	Enters server mode for the specified server.
Step 2	UCS-A /server # **show pci-switch**	Displays information about the PCI switches.
Step 3	UCS-A /server # **scope pci-switch** *pci-switch-number*	Enters the PCI switch mode for the specified PCI switch.
Step 4	UCS-A /server # **show detail**	

Example

The following example shows how to display the PCI switch properties:

```
UCS-A# scope server 1
UCS-A /server # show pci-switch
Pci Switch:
ID Pci Switch name Firmware Version
--- --------------- ----------------
1 PCI-Switch-1 xxxx
2 PCI-Switch-2 xxxxxxx
3 PCI-Switch-3 xxx
4 PCI-Switch-4 xxxxx
UCS-A /server # scope pci-switch 1
UCS-A /server/pci-switch #show detail

Pci Switch:
ID: 1
Pci Switch name: PCI-Switch-1
No of Adapters: 3
Switch Status: Good
Switch Temperature (C): 45.000000
Switch Product Revision: 0XxB
Firmware Version: xxxx
Vendor Id: xxx
Subvendor Id: xxx
Device Id: xxxx
Subdevice Id: xxxx
Switch Vendor: xxxxx
Pci Address: xx:00.0
UCS-A /server/pci-switch #
```

Managing Transportable Flash Module and Supercapacitor

LSI storage controllers use a Transportable Flash Module (TFM) powered by a supercapacitor to provide RAID cache protection. With Cisco UCS Manager, you can monitor these components to determine the status of the battery backup unit (BBU). The BBU operability status can be one of the following:

- **Operable**—The BBU is functioning successfully.
- **Inoperable**—The TFM or BBU is missing, or the BBU has failed and needs to be replaced.
- **Degraded**—The BBU is predicted to fail.

TFM and supercap functionality is supported beginning with Cisco UCS Manager Release 2.1(2).

TFM and Supercap Guidelines and Limitations

Supported Cisco UCS Servers for TFM and Supercap

The following Cisco UCS servers support TFM and supercap:

- Cisco UCS X410c M7 Compute Node
- Cisco UCS C240 M7 Server

- Cisco UCS C220 M7 Server
- Cisco UCS X210c M7 Compute Node
- Cisco UCS X210c M6 Compute Node
- Cisco UCS B200 M6 Server
- Cisco UCS B200 M5 Server
- Cisco UCS B480 M5 Server
- Cisco UCS C220 M5 Server
- Cisco UCS C240 M5 Server
- Cisco UCS C480 M5 Server

TPM Monitoring

Trusted Platform Module (TPM) is included on all Cisco UCS M3 blade and rack-mount servers. Operating systems can use TPM to enable encryption. For example, Microsoft's BitLocker Drive Encryption uses the TPM on Cisco UCS servers to store encryption keys.

Cisco UCS Manager enables monitoring of TPM, including whether TPM is present, enabled, or activated.

Viewing TPM Properties

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *chassis-id / server-id*	Enters chassis server mode for the specified server.
Step 2	UCS-A /chassis/server # **scope tpm** *tpm-id*	Enters TPM mode for the specified TPM ID.
Step 3	UCS-A /chassis/server/tpm # **show**	Displays the TPM properties.
Step 4	UCS-A /chassis/server/tpm # **show detail**	Displays detailed TPM properties.

Example

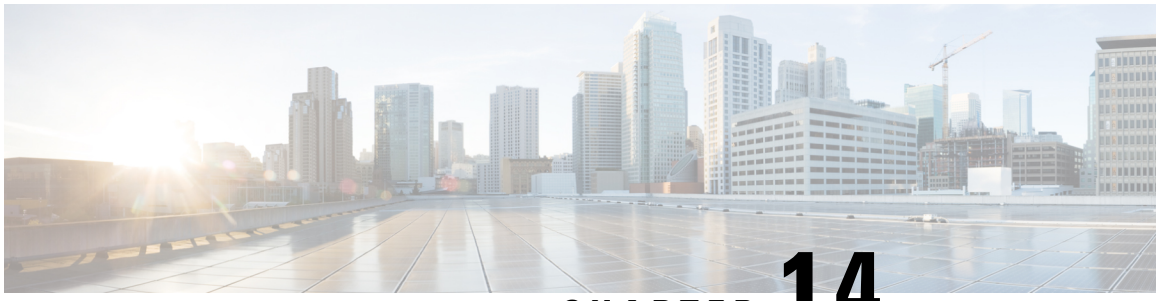
The following example shows how to display the TPM properties for blade 3 in chassis 1:

```
UCS-A# scope server 1/3
UCS-A /chassis/server # scope tpm 1
UCS-A /chassis/server/tpm # show

Trusted Platform Module:
    Presence: Equipped
    Enabled Status: Enabled
    Active Status: Activated
    Ownership: Unowned
```

```
UCS-A /chassis/server/tpm # show detail

Trusted Platform Module:
    Enabled Status: Enabled
    Active Status: Activated
    Ownership: Unowned
    Tpm Revision: 1
    Model: UCSX-TPM1-001
    Vendor: Cisco Systems Inc
    Serial: FCH16167DBJ
UCS-A /chassis/server/tpm #
```



CHAPTER 14

Netflow Monitoring

NetFlow Monitoring

NetFlow is a standard network protocol for collecting IP traffic data. NetFlow enables you to define a flow in terms of unidirectional IP packets that share certain characteristics. All packets that match the flow definition are collected and exported to one or more external NetFlow Collectors, where they can be further aggregated, analyzed, and used for application-specific processing.

Cisco UCS Manager uses NetFlow-capable adapters (Cisco UCS Cisco UCS VIC 1300 series, Cisco UCS VIC 1400 series, Cisco UCS VIC 14000 series, and Cisco UCS VIC 15000 series) to communicate with the routers and switches that collect and export flow information.

Starting from 4.3(2b) release, NetFlow monitoring is supported on Cisco UCS 6400 and 6500 series Fabric Interconnects.

Network Flows

A flow is a set of unidirectional IP packets that have common properties such as, the source or destination of the traffic, routing information, and protocol used. Flows are collected when they match the definitions in the flow record definition.

Flow Record Definitions

A flow record definition contains information about the properties used to define the flow, which can include both characteristic properties or measured properties. Characteristic properties, also called flow keys, are the

properties that define the flow. Cisco UCS Manager supports IPv4, IPv6, and Layer 2 keys. Measured characteristics, also called flow values or non-keys, measurable values such as the number of bytes contained in all packets of the flow, or the total number of packets.

A flow record definition is a specific combination of flow keys and flow values. The two types of flow record definitions are:

- **System-defined**—Default flow record definitions supplied by Cisco UCS Manager.
- **User-defined**—Flow record definitions that you can create yourself.

Flow Exporters, Flow Exporter Profiles, and Flow Collectors

Flow exporters transfer the flows to the flow connector based on the information in a flow exporter profile. The flow exporter profile contains the networking properties used to export NetFlow packets. The networking properties include a VLAN, the source IP address, and the subnet mask for each fabric interconnect.

Note In the Cisco UCS Manager GUI, the networking properties are defined in an exporter interface that is included in the profile. In the Cisco UCS Manager CLI, the properties are defined in the profile.

Flow collectors receive the flows from the flow exporter. Each flow collector contains an IP address, port, external gateway IP, and VLAN that defines where the flows are sent.

Flow Monitors and Flow Monitor Sessions

A flow monitor consists of a flow definition, one or two flow exporters, and a timeout policy. You can use a flow monitor to specify which flow information you want to gather, and where you want to collect it from. Each flow monitor operates in either the egress or ingress direction.

A flow monitor session contains up to four flow monitors: two flow monitors in the ingress direction and two flow monitors in the egress direction. A flow monitor session can also be associated with a vNIC.

NetFlow Limitations

The following limitations apply to NetFlow monitoring:

- NetFlow monitoring is supported on Cisco UCS VIC 1300, 1400, 14000, and 15000 series adapters. On Cisco UCS VIC 1200 series adapters, NetFlow is not recommended with FCoE traffic.
- For Cisco UCS 6400 series and 6500 series Fabric Interconnects:
  - Netflow monitoring includes both host receive and transmit directions. However, for Cisco UCS 6400 series and Cisco UCS 6536 Fabric Interconnects, NetFlow monitoring session applied to the Host Receive Direction Monitor will enable both transmit and receive monitoring, while NetFlow monitoring session applied to the Host Transmit Direction Monitor is a NO-OP.
  - Vethernet interface netflow monitor will always have **NFM_RECORD_L2_SRC_VLAN** enabled.
  - **Active Timeout** and **Inactive Timeout** values in **Flow Timeout Policy** cannot be modified.
- You can have up to 64 flow record definitions, flow exporters, and flow monitors.

- NetFlow is not supported in vNIC template objects.
- PVLANs and local VLANs are not supported for service VLANs.
- All VLANs must be public and must be common to both fabric interconnects.
- VLANs must be defined as an exporter interface before they can be used with a flow collector.
- You cannot use NetFlow with usNIC, Virtual Machine Queue, Virtual Machine Multiple Queues, RoCE, SRIOV, Geneve, or Linux ARFS enabled vNIC.
- Enabling NetFlow Monitoring does not allow you to downgrade Cisco UCS Manager software. To downgrade, disable Netflow Monitoring feature.

Enabling or Disabling NetFlow Monitoring

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope eth-flow-mon**	Enters the ethernet flow monitor mode.
Step 2	UCS-A /eth-flow-mon # *enable|disable*	Enables the Netflow feature and deploys any existing configuration present in Cisco UCS Manager onto NX-OS. Or, disables the Netflow feature and removes any configuration from the NX-OS. Even when you disable NetFlow monitoring, Cisco UCS Manager retains the Netflow configuration and deploys the same configuration when you enable Netflow monitoring. **Note** Disabling Netflow removes all Netflow related configuration from backend. All the flow sessions, which are in use are removed.
Step 3	UCS-A /eth-flow-mon # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to disable NetFlow monitoring:

```
UCS-A# scope eth-flow-mon
UCS-A /eth-flow-mon # disable
Warning: Disabling Netflow will Remove all Netflow related configuration from backend.
All the flow session which is in use will get cleaned up.
UCS-A /eth-flow-mon* # commit-buffer
UCS-A /eth-flow-mon #
```

Configuring a Flow Record Definition

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope eth-flow-mon**	Enters the ethernet flow monitor mode.
Step 2	UCS-A /eth-flow-mon # **enter flow-record** *flow-record-name*	Enters flow record mode for the specified flow record.
Step 3	UCS-A /eth-flow-mon/flow-record # **set keytype** {**ipv4keys** | **ipv6keys** | **l2keys**}	Specifies the key type.
Step 4	UCS-A /eth-flow-mon/flow-record # **set ipv4keys** {**dest-port** | **ip-protocol** | **ip-tos** | **ipv4-dest-address** | **ipv4-src-address** | **src-port**}	Specifies the attributes for the key type that you selected in Step 3. **Note** Use this command only if you chose **ipv4keys** in step 3.
Step 5	UCS-A /eth-flow-mon/flow-record # **set ipv6keys** {**dest-port** | **ip-protocol** | **ipv6-dest-address** | **ipv6-src-address** | **src-port**}	Specifies the attributes for the key type that you selected in Step 3. **Note** Use this command only if you chose **ipv6keys** in Step 3.
Step 6	UCS-A /eth-flow-mon/flow-record # **set l2keys** {**dest-mac-address** | **ethertype** | **src-mac-address**}	Specifies the attributes for the key type that you chose in Step 3. **Note** Use this command only if you selected **l2keys** in step 3.
Step 7	UCS-A /eth-flow-mon/flow-record # **set nonkeys** {**counter-bytes-long** | **counter-packets-long** | **sys-uptime-first** | **sys-uptime-last**}	Specifies the nonkey attributes.
Step 8	UCS-A /eth-flow-mon/flow-record # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create a flow record definition with Layer 2 keys and commit the transaction:

```
UCS-A# scope eth-flow-mon
UCS-A /eth-flow-mon # enter flow-record r1
UCS-A /eth-flow-mon/flow-record* # set keytype l2keys
UCS-A /eth-flow-mon/flow-record* #set l2keys dest-mac-address src-mac-address
UCS-A /eth-flow-mon/flow-record* # set nonkeys sys-uptime counter-bytes counter-packets
UCS-A /eth-flow-mon/flow-record* # commit-buffer
UCS-A /eth-flow-mon/flow-record #
```

Configuring an Exporter Profile

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope eth-flow-mon**	Enters the ethernet flow monitor mode.
Step 2	UCS-A /eth-flow-mon # **scope flow-profile** *profile-name*	Enters the flow profile mode for the specified profile.
Step 3	UCS-A /eth-flow-mon/flow-profile # **show config**	Displays the flow profile configuration.
Step 4	UCS-A /eth-flow-mon/flow-profile # **enter vlan** *vlan-name*	Specifies the VLAN associated with the exporter profile. PVLANs and local VLAN are not supported. All VLAN must be public and must be common to both fabric interconnects.
Step 5	UCS-A /eth-flow-mon/flow-profile/vlan # **enter fabric** {**a** | **b**}	Enters flow profile mode for the specified fabric.
Step 6	UCS-A /eth-flow-mon/flow-profile/vlan/fabric/ # **set addr** *ip-addr* **subnet** *ip-addr*	Specifies the source IP and subnet mask for the exporter profile on the fabric. **Important** Make sure the IP address you specify is unique within the Cisco UCS domain. IP address conflicts can occur if you specify an IP address that is already being used by Cisco UCS Manager.
Step 7	UCS-A /eth-flow-mon/flow-profile/vlan/fabric/ # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to configure the default exporter profile, set the source IP and subnet mask for the exporter interface on each fabric, and commit the transaction:

```
UCS-A# scope eth-flow-mon
UCS-A /eth-flow-mon # scope flow-profile default
UCS-A /eth-flow-mon/flow-profile # enter vlan 100
UCS-A /eth-flow-mon/flow-profile/vlan* # enter fabric a
UCS-A /eth-flow-mon/flow-profile/vlan/fabric* # set addr 10.10.10.10 subnet 255.255.255.0
UCS-A /eth-flow-mon/flow-profile/vlan/fabric* # up
UCS-A /eth-flow-mon/flow-profile/vlan* # enter fabric b
UCS-A /eth-flow-mon/flow-profile/vlan/fabric* # set addr 10.10.10.11 subnet 255.255.255.0
UCS-A /eth-flow-mon/flow-profile/vlan/fabric* # commit-buffer
UCS-A /eth-flow-mon/flow-profile/vlan/fabric #
```

Configuring a Netflow Collector

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope eth-flow-mon**	Enters the ethernet flow monitor mode.
Step 2	UCS-A /eth-flow-mon # **enter flow-collector** *flow-collector-name*	Enters the flow collector mode for the specified flow collector.
Step 3	UCS-A /eth-flow-mon/flow-collector # **set dest-port** *port_number*	Specifies the destination port for the flow collector.
Step 4	UCS-A /eth-flow-mon/flow-collector # **set vlan** *vlan_id*	Specifies the VLAN ID for the flow collector.
Step 5	UCS-A /eth-flow-mon/flow-collector # **enter ip-if**	Enters IPv4 configuration mode.
Step 6	UCS-A /eth-flow-mon/flow-collector/ip-if # **set addr** *ip-address*	Specifies the exporter IP address.
Step 7	UCS-A /eth-flow-mon/flow-collector/ip-if # **set exporter-gw** *gw-address*	Specifies the exporter gateway address.
Step 8	UCS-A /eth-flow-mon/flow-collector/ip-if # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to configure a NetFlow collector, set the exporter IP and gateway address, and commit the transaction:

```
UCS-A# scope eth-flow-mon
UCS-A /eth-flow-mon # enter flow-collector c1
UCS-A /eth-flow-mon/flow-collector* # set dest-port 9999
UCS-A /eth-flow-mon/flow-collector* # set vlan vlan100
UCS-A /eth-flow-mon/flow-collector* # enter ip-if
UCS-A /eth-flow-mon/flow-collector/ip-if* # set addr 20.20.20.20
UCS-A /eth-flow-mon/flow-collector/ip-if* # set exporter-gw 10.10.10.1
UCS-A /eth-flow-mon/flow-collector/ip-if* # commit-buffer
UCS-A /eth-flow-mon/flow-collector/ip-if #
```

Configuring a Flow Exporter

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope eth-flow-mon**	Enters the ethernet flow monitor mode.

	Command or Action	Purpose
Step 2	UCS-A /eth-flow-mon # **enter flow-exporter** *flow-exporter-name*	Enters the flow exporter mode for the specified flow exporter.
Step 3	UCS-A /eth-flow-mon/flow-exporter # **set dscp** *dscp_number*	Specifies the differentiated services code point.
Step 4	UCS-A /eth-flow-mon/flow-exporter # **set flow-collector** *flow-collector_name*	Specifies the flow collector.
Step 5	UCS-A /eth-flow-mon/flow-exporter # **set exporter-stats-timeout** *timeout_number*	Specifies the timeout period for resending NetFlow flow exporter data.
Step 6	UCS-A /eth-flow-mon/flow-exporter # **set interface-table-timeout** *timeout_number*	Specifies the time period for resending the NetFlow flow exporter interface table.
Step 7	UCS-A /eth-flow-mon/flow-exporter # **set template-data-timeout** *timeout_number*	Specifies the timeout period for resending NetFlow template data.
Step 8	UCS-A /eth-flow-mon/flow-exporter # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to configure a flow exporter, set the timeout values, and commit the transaction:

```
UCS-A# scope eth-flow-mon
UCS-A /eth-flow-mon # enter flow-exporter ex1
UCS-A /eth-flow-mon/flow-exporter* # set dscp 6
UCS-A /eth-flow-mon/flow-exporter* # set flow-collector c1
UCS-A /eth-flow-mon/flow-exporter* # set exporter-stats-timeout 600
UCS-A /eth-flow-mon/flow-exporter* # set interface-table-timeout 600
UCS-A /eth-flow-mon/flow-exporter* # set template-data-timeout 600
UCS-A /eth-flow-mon/flow-exporter* # commit-buffer
UCS-A /eth-flow-mon/flow-exporter #
```

Configuring a Flow Monitor

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope eth-flow-mon**	Enters the ethernet flow monitor mode.
Step 2	UCS-A /eth-flow-mon # **enter flow-monitor** *flow-monitor-name*	Enters the flow monitor mode for the specified flow monitor.
Step 3	UCS-A /eth-flow-mon/flow-monitor # **set flow-record** *flow-record-name*	Specifies the flow record.

	Command or Action	Purpose
Step 4	UCS-A /eth-flow-mon/flow-monitor # **create flow-exporter** *flow-exporter-name*	Specifies the first flow exporter.
Step 5	UCS-A /eth-flow-mon/flow-monitor # **create flow-exporter** *flow-exporter-name*	Specifies the second flow exporter.
Step 6	UCS-A /eth-flow-mon/flow-monitor # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create a flow monitor and commit the transaction:

```
UCS-A# scope eth-flow-mon
UCS-A /eth-flow-mon # enter flow-monitor m1
UCS-A /eth-flow-mon/flow-monitor* # set flow-record r1
UCS-A /eth-flow-mon/flow-monitor* # create flow-exporter ex1
UCS-A /eth-flow-mon/flow-monitor* # create flow-exporter ex2
UCS-A /eth-flow-mon/flow-monitor* # commit-buffer
UCS-A /eth-flow-mon/flow-monitor #
```

Configuring a Flow Monitor Session

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope eth-flow-mon**	Enters the ethernet flow monitor mode.
Step 2	UCS-A /eth-flow-mon # **enter flow-mon-session** *flow-monitor-session-name*	Enters the flow monitor session mode for the specified flow monitor session.
Step 3	UCS-A /eth-flow-mon/flow-mon-session # **create flow-monitor** *flow-monitor-1*	Specifies the first flow monitor.
Step 4	UCS-A /eth-flow-mon/flow-mon-session # **create flow-monitor** *flow-monitor-2*	Specifies the second flow monitor.
Step 5	UCS-A /eth-flow-mon/flow-mon-session # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create a flow monitor session with two flow monitors:

```
UCS-A# scope eth-flow-mon
UCS-A /eth-flow-mon # enter flow-mon-session s1
UCS-A /eth-flow-mon/flow-mon-session* # create flow-monitor m1
UCS-A /eth-flow-mon/flow-mon-session* # create flow-monitor m2
UCS-A /eth-flow-mon/flow-mon-session* # commit-buffer
UCS-A /eth-flow-mon/flow-mon-session #
```

Configuring a NetFlow Cache Active and Inactive Timeout

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope eth-flow-mon**	Enters the ethernet flow monitor mode.
Step 2	UCS-A /eth-flow-mon # **scope flow-timeout** *timeout-name*	Enters the flow timeout mode for the specified flow timeout.
Step 3	UCS-A /eth-flow-mon/flow-timeout # **set cache-timeout-active** *timeout-value*	Specifies the active timeout value. This value can be between 60 and 4092 seconds. The default value is 120 seconds.
Step 4	UCS-A /eth-flow-mon/flow-timeout # **set cache-timeout-inactive** *timeout-value*	Specifies the inactive timeout value. This value can be between 15 and 4092 seconds. The default value is 15 seconds.
Step 5	UCS-A /eth-flow-mon/flow-timeout # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to change the NetFlow timeout values and commit the transaction:

```
UCS-A# scope eth-flow-mon
UCS-A /eth-flow-mon # scope flow-timeout default
UCS-A /eth-flow-mon/flow-timeout # set cache-timeout-active 1800
UCS-A /eth-flow-mon/flow-timeout* # set cache-timeout-inactive 20
UCS-A /eth-flow-mon/flow-timeout* # commit-buffer
UCS-A /eth-flow-mon/flow-timeout #
```

Associating a Flow Monitor Session to a vNIC

Procedure

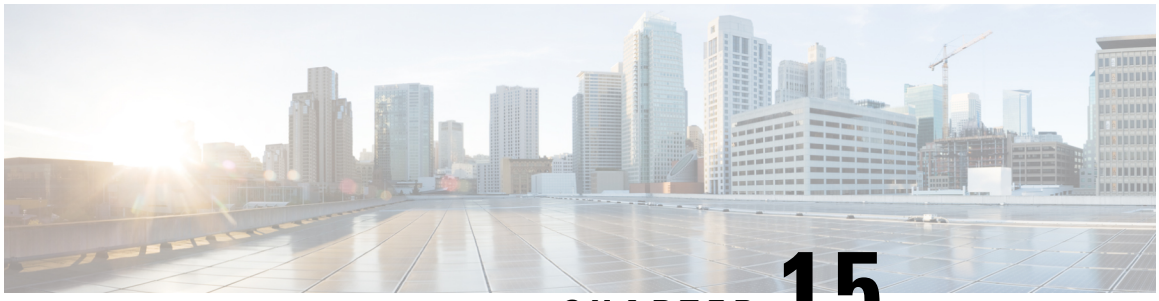
	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter / as the *org-name*.
Step 2	UCS-A /org # **scope service-profile** *profile-name*	Enters the organization service profile mode for the specified service profile.
Step 3	UCS-A /org/service-profile # **scope vnic** *vnic-name*	Enters the organization service profile mode for the specified vNIC.

	Command or Action	Purpose
Step 4	UCS-A /org/service-profile/vnic # **enter flow-mon-src** *flow-monitor-session-name*	Associates the flow monitor session to the 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 associate the flow monitor session s1 to the vNIC eth5:

```
UCS-A# scope org /
UCS-A /org # scope service-profile sp1
UCS-A /org/service-profile # scope vnic eth5
UCS-A /org/service-profile/vnic # enter flow-mon-src s1
UCS-A /org/service-profile/vnic # commit-buffer
```



CHAPTER 15

Traffic Monitoring

Traffic Monitoring

Traffic monitoring copies traffic from one or more source ports and sends the copied traffic to a dedicated destination port for analysis by a network analyzer. This feature is also known as Switched Port Analyzer (SPAN).

Types of Traffic Monitoring Sessions

There are two types of monitoring sessions:

- Ethernet
- Fibre Channel

The type of destination port determines what kind of monitoring session you need. For an Ethernet traffic monitoring session, the destination port must be an unconfigured physical port. For a Fibre Channel traffic monitoring session, the destination port must be a Fibre Channel uplink port except when you are using Cisco UCS 6536 Fabric Interconnect, Cisco UCS 6454 Fabric Interconnect, Cisco UCS 6400 Series Fabric Interconnect and 6300 Series Fabric Interconnects.

Note For Cisco UCS 6332, 6332-16UP, 64108, 6454, and 6536 Fabric Interconnects, you cannot choose Fibre Channel destination ports. The destination port must be an unconfigured physical Ethernet port.

Traffic Monitoring Across Ethernet

An Ethernet traffic monitoring session can monitor any of the following traffic source and destination ports:

Source Ports	Destination Ports
• Uplink Ethernet port • Ethernet port channel • VLAN • Service profile vNIC • Service profile vHBA • FCoE port • Port channels • Unified uplink port • VSAN	Unconfigured Ethernet Port

Note All traffic sources must be located within the same switch as the destination port. A port configured as a destination port cannot also be configured as a source port. A member port of a port channel cannot be configured individually as a source. If the port channel is configured as a source, all member ports are source ports.

A server port can be a source, only if it is a non-virtualized rack server adapter-facing port.

Traffic Monitoring for Cisco UCS 6500, 6400 Series Fabric Interconnects

- Cisco UCS 6500, 6400 Series Fabric Interconnects do not support a Fibre Channel port as a destination port. Therefore, an Ethernet port is the only option for configuring any traffic monitoring session on this Fabric Interconnect.
- Cisco UCS 6500, 6400 Series Fabric Interconnects support monitoring traffic in the transmit direction for more than two sources per Fabric Interconnect.
- You can monitor or use SPAN on port channels sources for traffic in the transmit and receive directions.
- You can configure a port as a destination port for only one monitor session.
- You can monitoring Port-Channel as a source in the transmit direction.
- You cannot monitor vEth as a source in the transmit direction.

Traffic Monitoring for Cisco UCS 6300 Fabric Interconnects

- Cisco UCS 6300 Fabric Interconnect supports port-based mirroring.
- Cisco UCS 6300 Fabric Interconnects support VLAN SPAN only in the receive direction.
- Ethernet SPAN is port based on the Cisco UCS 6300 Fabric Interconnect.

Traffic Monitoring Across Fibre Channel

You can monitor Fibre Channel traffic using either a Fibre Channel traffic analyzer or an Ethernet traffic analyzer. When Fibre Channel traffic is monitored with an Ethernet traffic monitoring session, at an Ethernet destination port, the destination traffic is FCoE. The Cisco UCS 6300 Fabric Interconnect supports FC SPAN only on the ingress side.

A Fibre Channel traffic monitoring session can monitor any of the following traffic source and destination ports:

Source Ports	Destination Ports
• FC Port • FC Port Channel • Uplink Fibre Channel port • SAN port channel • VSAN • Service profile vHBA • Fibre Channel storage port	• Fibre Channel uplink port • Unconfigured Ethernet Port (Cisco UCS 6536, 64108, 6454, 6332, and 6332-16UP Fabric Interconnects)

Guidelines and Recommendations for Traffic Monitoring

When configuring or activating traffic monitoring, consider the following guidelines:

Traffic Monitoring Sessions

A traffic monitoring session is disabled by default when created. To begin monitoring traffic, first activate the session. A traffic monitoring session must be unique on any fabric interconnect within the Cisco UCS pod. Create each monitoring session with a unique name and unique VLAN source. To monitor traffic from a server, add all vNICs from the service profile corresponding to the server.

Note No more than 32 VLANs can be added to a SPAN monitoring session.

Maximum Number of Supported Active Traffic Monitoring Sessions Per Fabric-Interconnect

You can create and store up to 16 traffic monitoring sessions, but only four can be active at the same time. For each Cisco UCS 6536, 6400 Series Fabric Interconnect and 6300 Fabric Interconnect, you can only monitor up to four traffic directions. The receive and transmit directions each count as one monitoring session, while the bi-direction monitoring session is counted as 2. For example:

- Four active sessions—If each session is configured to monitor traffic in only one direction.
- Two active sessions—If each session is configured to monitor traffic bidirectionally.
- Three active sessions—If one session is unidirectional and the second session is bidirectional.

Note Traffic monitoring can impose a significant load on your system resources. To minimize the load, select sources that carry as little unwanted traffic as possible and disable traffic monitoring when it is not needed.

vNIC

Because a traffic monitoring destination is a single physical port, a traffic monitoring session can monitor only a single fabric. To monitor uninterrupted vNIC traffic across a fabric failover, create two sessions, one per fabric and connect two analyzers. Add the vNIC as the traffic source using the exact same name for both sessions. If you change the port profile of a virtual machine, any associated vNICs being used as source ports are removed from monitoring, and you must reconfigure the monitoring session. If a traffic monitoring session was configured on a dynamic vNIC under a release earlier than Cisco UCS Manager Release 2.0, you must reconfigure the traffic monitoring session after upgrading. Cisco UCS 6500, 6400 Series Fabric Interconnects do not support traffic monitoring traffic from a vNIC in the transmit direction.

vHBA

A vHBA can be a source for either an Ethernet or Fibre Channel monitoring session, but it cannot be a source for both simultaneously. When a VHBA is set as the SPAN source, the SPAN destination only receives VN-Tagged frames. It does not receive direct FC frames. Cisco UCS 6500, 6400 Series Fabric Interconnects do not support traffic monitoring traffic from a vHBA in the transmit direction.

Creating an Ethernet Traffic Monitoring Session

Note This procedure describes creating an Ethernet traffic monitoring session. To create a Fibre Channel traffic monitoring session, the following changes are required:

- Enter the **scope fc-traffic-mon** command instead of the **scope eth-traffic-mon** command in Step 1.
- Enter the **create fc-mon-session** command instead of the **create eth-mon-session** command in Step 3.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope eth-traffic-mon**	Enters Ethernet traffic monitoring command mode.
Step 2	UCS-A /eth-traffic-mon # **scope fabric** {**a** | **b**}	Enters traffic monitoring command mode for the specified fabric.
Step 3	UCS-A /eth-traffic-mon/fabric # **create eth-mon-session** *session-name*	Creates a traffic monitoring session with the specified name.
Step 4	UCS-A /eth-traffic-mon/fabric/eth-mon-session # **create dest-interface** *slot-num port-num*	Configures the interface at the specified slot and port number to be the destination for the

	Command or Action	Purpose
		traffic monitoring session. Enters the command mode for the interface.
Step 5	UCS-A /eth-traffic-mon/fabric/eth-mon-session/dest-interface # **set speed***admin-speed*	Sets the data transfer rate of the port channel to be monitored. This can be: • 1gbps—1 Gbps • 10gbps—10 Gbps • 20gbps—20 Gbps • 40gbps—40 Gbps
Step 6	UCS-A /eth-traffic-mon/fabric/eth-mon-session/dest-interface # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example creates an Ethernet traffic monitoring session to copy and forward traffic to the destination port at slot 2, port 12, sets the admin speed to 20 Gbps, and commits the transaction:

```
UCS-A# scope eth-traffic-mon
UCS-A /eth-traffic-mon # scope fabric a
UCS-A /eth-traffic-mon/fabric # create eth-mon-session EthMonitor33
UCS-A /eth-traffic-mon/fabric/eth-mon-session* # create dest-interface 2 12
UCS-A /eth-traffic-mon/fabric/eth-mon-session/dest-interface* # set speed 20gbps
UCS-A /eth-traffic-mon/fabric/eth-mon-session/dest-interface* # commit-buffer
UCS-A /eth-traffic-mon/fabric/eth-mon-session/dest-interface #
```

What to do next

- Add traffic sources to the traffic monitoring session.
- Activate the traffic monitoring session.

Creating a Fibre Channel Traffic Monitoring Session

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope fc-traffic-mon**	Enters Fibre Channel traffic monitoring command mode.
Step 2	UCS-A /fc-traffic-mon # **scope fabric** {**a** | **b**}	Enters Fibre Channel traffic monitoring command mode for the specified fabric.

	Command or Action	Purpose
Step 3	UCS-A /fc-traffic-mon/fabric # **create fc-mon-session** *session-name*	Creates a Fibre Channel traffic monitoring session with the specified name.
Step 4	UCS-A /fc-traffic-mon/fabric/fc-mon-session # **create dest-interface** *slot-num port-num*	Creates and enters the command mode of the destination slot and port for the Fibre Channel traffic monitoring session.
Step 5	UCS-A /fc-traffic-mon/fabric/fc-mon-session/dest-interface # **set speed***admin-speed*	Sets the data transfer rate of the port channel to be monitored. This can be: • 1gbps—1 Gbps • 2gbps—2 Gbps • 4gbps—4 Gbps • 8gbps—8 Gbps • auto—Cisco UCS determines the data transfer rate.
Step 6	UCS-A /fc-traffic-mon/fabric/fc-mon-session/dest-interface # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example creates a Fibre channel traffic monitoring session to copy and forward traffic to the destination port at slot 1, port 10, sets the admin speed to 8 Gbps, and commits the transaction:

```
UCS-A# scope fc-traffic-mon
UCS-A /fc-traffic-mon # scope fabric a
UCS-A /fc-traffic-mon/fabric # create fc-mon-session FCMonitor
UCS-A /fc-traffic-mon/fabric/fc-mon-session* # create dest-interface 1 10
UCS-A /fc-traffic-mon/fabric/fc-mon-session/dest-interface* # set speed 8gbps
UCS-A /fc-traffic-mon/fabric/fc-mon-session/dest-interface* # commit-buffer
UCS-A /fc-traffic-mon/fabric/fc-mon-session/dest-interface #
```

What to do next

- Add traffic sources to the traffic monitoring session.
- Activate the traffic monitoring session.

Adding Traffic Sources to a Monitoring Session

Adding an Uplink Source Port to a Monitoring Session

Note This procedure describes adding an Ethernet uplink port as a source for a traffic monitoring session. To add a Fibre Channel uplink port as a source, enter the **scope fc-uplink** command instead of the **scope eth-uplink** command in Step 1.

Before you begin

A traffic monitoring session must be created.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope eth-uplink**	Enters Ethernet uplink command mode.
Step 2	UCS-A /eth-uplink # **scope fabric** {**a** | **b**}	Enters uplink fabric mode for the specified fabric.
Step 3	UCS-A /eth-uplink/fabric # **scope interface** *slot-num port-num*	Enters the interface command mode for the specified uplink port.
Step 4	UCS-A /eth-uplink/fabric/interface # **create mon-src** *session-name*	Adds the uplink port as a source to the specified monitoring session.
Step 5	(Optional) UCS-A /eth-uplink/fabric/interface/mon-src # **set direction** {**both** | **receive** | **transmit**}	Specifies the traffic direction to be monitored. **Note** If you do not select any direction, the default direction is Rx.
Step 6	UCS-A /eth-uplink/fabric/interface/mon-src # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example adds the ingress traffic on Ethernet uplink port 3 on slot 2 of fabric A as a source for a monitoring session and commits the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # scope fabric a
UCS-A /eth-uplink/fabric # scope interface 2 3
UCS-A /eth-uplink/fabric/interface # create mon-src Monitor23
UCS-A /eth-uplink/fabric/interface/mon-src* # set direction receive
UCS-A /eth-uplink/fabric/interface/mon-src* # commit-buffer
UCS-A /eth-uplink/fabric/interface/mon-src #
```

What to do next

You can add additional sources to the traffic monitoring session.

Adding a vNIC or vHBA Source to a Monitoring Session

Note This procedure describes adding a vNIC as a source for a traffic monitoring session. To add a vHBA as a source, enter the **scope vhba** command instead of the **scope vnic** command in Step 2.

Before you begin

A traffic monitoring session must be created.

Procedure

	Command or Action	Purpose
Step 1	Switch-A# **scope system**	Enters system mode.
Step 2	Switch-A /system # **scope vm-mgmt**	Enters VM management mode.
Step 3	(Optional) Switch-A /system/vm-mgmt # **show virtual-machine**	Displays the running virtual machines.
Step 4	Switch-A /system/vm-mgmt # **scope virtual-machine** *uuid*	Enters command mode for the virtual machine that contains the dynamic vNIC.
Step 5	(Optional) Switch-A /system/vm-mgmt/virtual-machine # **show expand**	Displays the virtual machine details, including the vNIC MAC address.
Step 6	Switch-A /system/vm-mgmt/virtual-machine # **scope vnic** *mac-address*	Enters the command mode for the vNIC at the specified MAC address.
Step 7	Switch-A /system/vm-mgmt/virtual-machine/vnic # **create mon-src** *session-name*	Adds the vNIC as a source to the specified monitoring session.
Step 8	(Optional) Switch-A /system/vm-mgmt/virtual-machine/vnic/mon-src # **set direction** {**both** | **receive** | **transmit**}	Specifies the traffic direction to be monitored.
Step 9	Switch-A /system/vm-mgmt/virtual-machine/vnic/mon-src # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example adds the ingress traffic on a dynamic vNIC as a source for a monitoring session and commits the transaction:

```
Switch-A# scope system
Switch-A /system # scope vm-mgmt
Switch-A /system/vm-mgmt # show virtual-machine
Virtual Machine:
    UUID: 42327c42-e00c-886f-e3f7-e615906f51e9
    Service Profile: org-root/ls-dsw-bld1-esx
    Server: sys/chassis-1/blade-1
    Status: Online
.
.
.
Switch-A /system/vm-mgmt # scope virtual-machine 42327c42-e00c-886f-e3f7-e615906f51e9
Switch-A /system/vm-mgmt/virtual-machine # show expand
Virtual Machine:
    UUID: 42327c42-e00c-886f-e3f7-e615906f51e9
    Service Profile: org-root/ls-dsw-bld1-esx
    Server: sys/chassis-1/blade-1
    Status: Online

    vNIC:
        Name:
        Status: Online
        MAC Address: 00:50:56:B2:00:00

        VIF:
            Vif Id: 32772
            Status: Online
            Phys Fabric ID: B
            Virtual Fabric:
Switch-A /system/vm-mgmt/virtual-machine # scope vnic 00:50:56:B2:00:00
Switch-A /system/vm-mgmt/virtual-machine/vnic # create mon-src Monitor23
Switch-A /system/vm-mgmt/virtual-machine/vnic/mon-src* # set direction receive
Switch-A /system/vm-mgmt/virtual-machine/vnic/mon-src* # commit-buffer

Switch-A /system/vm-mgmt/virtual-machine/vnic/mon-src #
```

What to do next

You can add additional sources to the traffic monitoring session.

Adding a VLAN or VSAN Source to a Monitoring Session

Note This procedure describes adding a VLAN as a source for a traffic monitoring session. To add a VSAN as a source, the following changes are required:

- Enter the **scope fc-uplink** command instead of the **scope eth-uplink** command in Step 1.
- Enter the **create vsan** command instead of the **create vlan** command in Step 3.

Before you begin

A traffic monitoring session must be created.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope eth-uplink**	Enters Ethernet uplink command mode.
Step 2	UCS-A /eth-uplink # **scope fabric** {**a** | **b**}	Enters uplink fabric mode for the specified fabric. **Note** This step is required when adding a local VLAN as a source. To add a global VLAN as a source, omit this step.
Step 3	UCS-A /eth-uplink/fabric # **create vlan** *vlan-name* *vlan-id*	Creates a named VLAN, specifies the VLAN name and VLAN ID, and enters uplink VLAN mode.
Step 4	UCS-A /eth-uplink/fabric/vlan # **create mon-src** *session-name*	Adds the VLAN as a source to the specified monitoring session.
Step 5	UCS-A /eth-uplink/fabric/vlan/mon-src # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example adds a local VLAN as a source for an Ethernet monitoring session and commits the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # scope fabric a
UCS-A /eth-uplink/fabric # create vlan vlan23 23
UCS-A /eth-uplink/fabric/vlan # create mon-src Monitor23
UCS-A /eth-uplink/fabric/vlan/mon-src* # commit-buffer
UCS-A /eth-uplink/fabric/vlan/mon-src #
```

What to do next

You can add additional sources to the traffic monitoring session.

Adding a Storage Port Source to a Monitoring Session

Note This procedure describes adding a Fibre Channel storage port as a source for a Fibre Channel traffic monitoring session. To add an FCoE storage port as a source for an Ethernet traffic monitoring session, enter the **create interface fcoe** command instead of the **create interface fc** command in Step 3.

Before you begin

A traffic monitoring session must be created.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope fc-storage**	Enters Fibre Channel storage port command mode.
Step 2	UCS-A /fc-storage # **scope fabric** {**a** | **b**}	Enters Fibre Channel storage port fabric mode for the specified fabric.
Step 3	UCS-A /fc-storage/fabric # **create interface fc** *slot-num port-num*	Creates a Fibre Channel storage port interface and enters the interface command mode.
Step 4	UCS-A /fc-storage/fabric/fc # **create mon-src** *session-name*	Adds the storage port as a source to the specified monitoring session.
Step 5	UCS-A /fc-storage/fabric/fc/mon-src # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example adds a Fibre Channel storage port on port 3 of slot 2 as a source for a Fibre Channel monitoring session 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 3
UCS-A /fc-storage/fabric/fc* # create mon-src Monitor23
UCS-A /fc-storage/fabric/fc/mon-src* # commit-buffer
UCS-A /fc-storage/fabric/fc/mon-src #
```

What to do next

You can add additional sources to the traffic monitoring session.

Activating a Traffic Monitoring Session

Note This procedure describes activating an Ethernet traffic monitoring session. To activate a Fibre Channel traffic monitoring session, the following changes are required:

- Enter the **scope fc-traffic-mon** command instead of the **scope eth-traffic-mon** command in Step 1.
- Enter the **scope fc-mon-session** command instead of the **scope eth-mon-session** command in Step 3.

Before you begin

Configure a traffic monitoring session.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope eth-traffic-mon**	Enters Ethernet traffic monitoring command mode.
Step 2	UCS-A /eth-traffic-mon # **scope fabric** {**a** | **b**}	Enters traffic monitoring command mode for the specified fabric.
Step 3	UCS-A /eth-traffic-mon/fabric # **scope eth-mon-session** *session-name*	Enters the command mode of the traffic monitoring session with the specified name.
Step 4	UCS-A /eth-traffic-mon/fabric/eth-mon-session # **disable** | **enable**	Disables or enables the traffic monitoring session.
Step 5	UCS-A /eth-traffic-mon/fabric/eth-mon-session # **commit-buffer**	Commits the transaction to the system configuration.

When activated, the traffic monitoring session begins forwarding traffic to the destination when a traffic source is configured.

Example

The following example activates an Ethernet traffic monitoring session and commits the transaction:

```
UCS-A# scope eth-traffic-mon
UCS-A /eth-traffic-mon # scope fabric a
UCS-A /eth-traffic-mon/fabric # scope eth-mon-session Monitor33
UCS-A /eth-traffic-mon/fabric/eth-mon-session # enable
UCS-A /eth-traffic-mon/fabric/eth-mon-session* # commit-buffer
UCS-A /eth-traffic-mon/fabric/eth-mon-session # show

Ether Traffic Monitoring Session:
    Name       Admin State       Oper State   Oper State Reason
    ---------- ----------------- ------------ -----------------
    Monitor33  Enabled           Up           Active

UCS-A /eth-traffic-mon/fabric/eth-mon-session #
```

Deleting a Traffic Monitoring Session

Note This procedure describes deleting an Ethernet traffic monitoring session. To delete a Fibre Channel traffic monitoring session, the following changes are required:

- Enter the **scope fc-traffic-mon** command instead of the **scope eth-traffic-mon** command in Step 1.
- Enter the **delete fc-mon-session** command instead of the **delete eth-mon-session** command in Step 3.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope eth-traffic-mon**	Enters Ethernet traffic monitoring command mode.
Step 2	UCS-A /eth-traffic-mon # **scope fabric** {**a** | **b**}	Enters traffic monitoring command mode for the specified fabric.
Step 3	UCS-A /eth-traffic-mon/fabric # **delete eth-mon-session** *session-name*	Deletes the traffic monitoring session with the specified name.
Step 4	UCS-A /eth-traffic-mon/fabric # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes an Ethernet traffic monitoring session and commits the transaction:

```
UCS-A# scope eth-traffic-mon
UCS-A /eth-traffic-mon # scope fabric a
UCS-A /eth-traffic-mon/fabric # delete eth-mon-session Monitor33
UCS-A /eth-traffic-mon/fabric* # commit-buffer
UCS-A /eth-traffic-mon/fabric #
```

SPAN Restrictions for the Cisco UCS Mini

Consider the following guidelines and restrictions when configuring the SPAN feature on Cisco UCS Mini

- FC port as SPAN destination is not supported.
- VSAN as SPAN source is not supported.
- FC uplink ports as SPAN source is not supported.