Cisco UCS Manager Infrastructure Management Using the CLI, Release 3.1

First Published: 2016-01-20

Last Modified: 2017-04-27

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`.

Text Type	Indication
CLI commands	CLI command keywords appear in **this font**. Variables in a CLI command appear in *this font*.
[ ]	Elements in square brackets are optional.
{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: http://www.cisco.com/go/unifiedcomputing/b-series-doc.

For a complete list of all C-Series documentation, see the *Cisco UCS C-Series Servers Documentation Roadmap* available at the following URL: http://www.cisco.com/go/unifiedcomputing/c-series-doc.

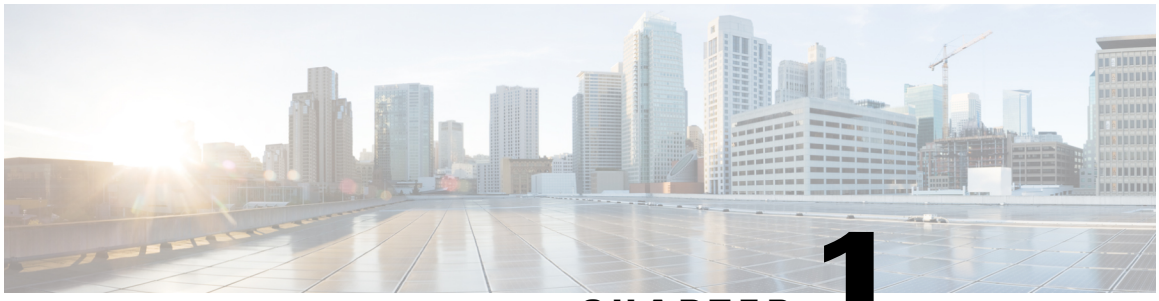
For information on supported firmware versions and supported UCS Manager versions for the rack servers that are integrated with the UCS Manager for management, refer to Release Bundle Contents for Cisco UCS Software.

Other Documentation Resources

Follow Cisco UCS Docs on Twitter to receive document update notifications.

Documentation Feedback

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



CHAPTER 1

New and Changed Information

New and Changed Information for This Release

The following table provides an overview of the significant changes to this guide for this current release. The table does not provide an exhaustive list of all changes made to this guide or of all new features in this release.

Table 1: New Features and Changed Behavior in Cisco UCS Manager, Release 3.1(3)

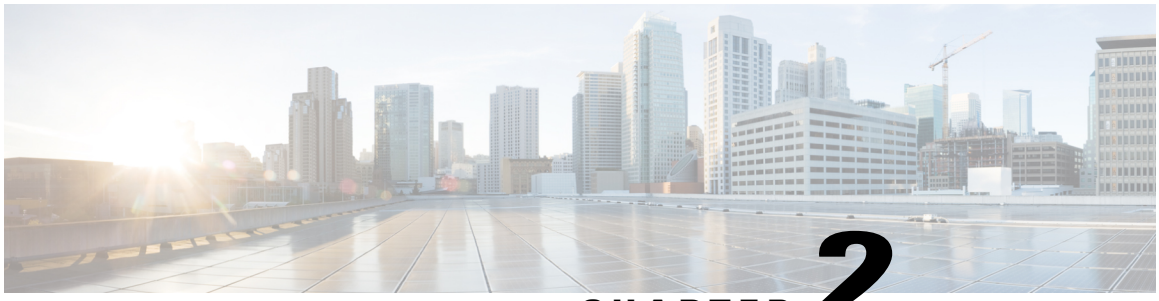
Feature	Description	Where Documented
Cisco UCS C3260/C3X60 re-branding.	Beginning with Cisco UCS Manager Release 3.1(3), Cisco UCS C3260/C3X60 is renamed to Cisco UCS S3260. You may still see certain components in the system labeled as C3260/C3X60. For this release, the terms S3260 and C3260/C3X60 are used interchangeably. Both, S3260 and C3260/C3X60, refer to the same hardware component.	Chassis Management in Cisco UCS Manager CLI , on page 29
Smart SSD	Cisco UCS Manager supports monitoring SSD health. This feature is called Smart SSD feature.	Smart SSD, on page 80

Feature	Description	Where Documented
Power Transition Log	The Power Transition Log was added which logs the last five server power transitions, the power transition source timestamp of the latest power transition, and the count of the last consecutive server power transitions from the same source.	Viewing the Power Transition Log, on page 96

Table 2: New Features and Changed Behavior in Cisco UCS Manager, Release 3.1(2)

Feature	Description	Where Documented
Server Factory Reset	Factory reset of servers.	Resetting a Rack-Mount Server to Factory Default Settings, on page 87 Resetting a Blade Server to Factory Default Settings, on page 71 Resetting a Cisco UCS S3260 Server Node to Factory Default Settings, on page 101
Enable 'hardware multicast hw-hash' on server port-channels	Multicast Hardware Hash—In a portchannel, by default, ingress multicasttraffic on any port in the fabricinterconnect (FI) selects a particularlink between the IOM and the fabricinterconnect to egress the traffic. Toreduce potential issues with thebandwidth, and to provide effectiveload balancing of the ingress multicasttraffic, hardware hashing is used formulticast traffic. When multicasthardware hashing is enabled, all linksbetween the IOM and the fabricinterconnect in a port channel can beused for multicast traffic.	Configuring the Chassis/FEX Discovery Policy, on page 19

Feature	Description	Where Documented
UCSM HA should allowreplacement (re-election) of HAquorum chassis - HA Version	HA Version Holder Replacement—Insome situations, the shared storagedevices that are selected as highavailability (HA) version holdersbecome unreachable for an extendedperiod of time. You can now specifynew preferred HA version holderscorresponding to the devices that arefunctioning correctly. When you triggera reelection of version holders, thesenew preferred HA devices are selectedfirst.	HA Version Holder Replacement, on page 24



CHAPTER 2

Overview

Cisco UCS Manager User CLI Documentation

Cisco UCS Manager offers you a new 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.
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.

Guide	Description
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.

Infrastructure Management Guide Overview

This guide provides an overview of the physical and virtual infrastructure used in Cisco Unified Computing System (UCS) and managed through Cisco UCS Manager. It also provides detailed information about managing these infrastructure components. The following table summarizes the overall organization of the guide.

Topic	Description
Overview	Conceptual overview of Cisco UCS architecture including Cisco Fabric Interconnects, I/O Module, Chassis, Servers, and Virtualization in Cisco UCS.
Equipment Policies	Equipment policies such as Chassis/FEX discovery policy, Chassis connectivity policy, and Rack Server discovery policy.
Chassis Management	Overview of the chassis supported and procedures to manage them.
I/O Module Management	Overview of I/O Modules and procedures to manage them.
Power Management in Cisco UCS	Overview of UCS Power Management policies, Global Power policies, and Power Capping.
Blade Server Management	Overview of Blade Servers and procedures to manage them.
Rack-Mount Server Management	Overview of Rack-Mount Servers and procedures to manage them.
S3260 Server Node Management	Overview of S3260 Server Node and procedures to manage them.

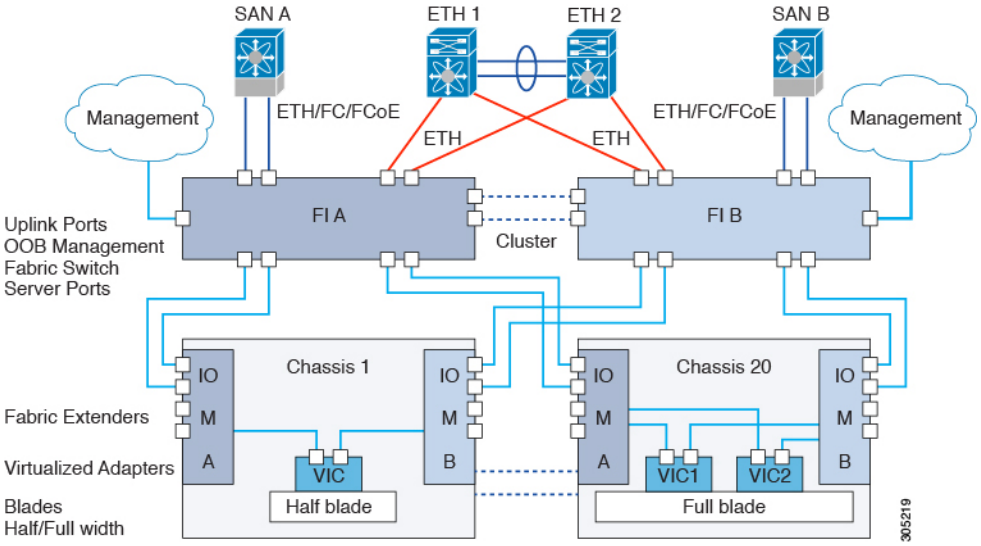
Topic	Description
Virtual Interface Management	Overview of Virtualization in Cisco UCS, Virtual Interfaces and procedures to manage them.
Server Troubleshooting	Common troubleshooting scenarios for Servers.

Cisco Unified Computing System Overview

Cisco UCS has a unique architecture that integrates compute, data network access, and storage network access into a common set of components under a single-pane-of-glass management interface.

Cisco UCS fuses access layer networking and servers. This high-performance, next-generation server system provides a data center with a high degree of workload agility and scalability. The hardware and software components support Cisco's unified fabric, which runs multiple types of data center traffic over a single converged network adapter.

Figure 1: Cisco Unified Computing System Architecture



Architectural Simplification

The simplified architecture of Cisco UCS reduces the number of required devices and centralizes switching resources. By eliminating switching inside a chassis, network access-layer fragmentation is significantly reduced. Cisco UCS implements Cisco unified fabric within racks and groups of racks, supporting Ethernet and Fibre Channel protocols over 10 Gigabit Cisco Data Center Ethernet and Fibre Channel over Ethernet (FCoE) links. This radical simplification reduces the number of switches, cables, adapters, and management points by up to two-thirds. All devices in a Cisco UCS domain remain under a single management domain, which remains highly available through the use of redundant components.

High Availability

The management and data plane of Cisco UCS is designed for high availability and redundant access layer fabric interconnects. In addition, Cisco UCS supports existing high availability and disaster recovery solutions for the data center, such as data replication and application-level clustering technologies.

Scalability

A single Cisco UCS domain supports multiple chassis and their servers, all of which are administered through one Cisco UCS Manager. For more detailed information about the scalability, speak to your Cisco representative.

Flexibility

A Cisco UCS domain allows you to quickly align computing resources in the data center with rapidly changing business requirements. This built-in flexibility is determined by whether you choose to fully implement the stateless computing feature. Pools of servers and other system resources can be applied as necessary to respond to workload fluctuations, support new applications, scale existing software and business services, and accommodate both scheduled and unscheduled downtime. Server identity can be abstracted into a mobile service profile that can be moved from server to server with minimal downtime and no need for additional network configuration.

With this level of flexibility, you can quickly and easily scale server capacity without having to change the server identity or reconfigure the server, LAN, or SAN. During a maintenance window, you can quickly do the following:

- Deploy new servers to meet unexpected workload demand and rebalance resources and traffic.
- Shut down an application, such as a database management system, on one server and then boot it up again on another server with increased I/O capacity and memory resources.

Optimized for Server Virtualization

Cisco UCS has been optimized to implement VM-FEX technology. This technology provides improved support for server virtualization, including better policy-based configuration and security, conformance with a company's operational model, and accommodation for VMware's VMotion.

Cisco UCS Building Blocks and Connectivity

Figure 2: Cisco UCS Building Blocks and Connectivity

As shown in the figure above, the primary components included within Cisco UCS are as follows:

- **Cisco UCS Manager—**Cisco UCS Manager is the centralized management interface for Cisco UCS. For more information on Cisco UCS Manager, see *Introduction to Cisco UCS manager* in *Cisco UCS Manager Getting Started Guide*
- **Cisco UCS Fabric Interconnects—**The Cisco UCS Fabric Interconnect is the core component of Cisco UCS deployments, providing both network connectivity and management capabilities for the Cisco UCS system. The Cisco UCS Fabric Interconnects run the Cisco UCS Manager control software and consist of the following components:
  - Cisco UCS 6200 series Fabric Interconnects, Cisco UCS 6332 series Fabric Interconnects, and Cisco UCS Mini
  - Transreceivers for network and storage connectivity
  - Expansion modules for the various Fabric Interconnects
  - Cisco UCS Manager software

  For more information on Cisco UCS Fabric Interconnects, see Cisco UCS Fabric Infrastructure Portfolio, on page 10.
- **Cisco UCS I/O Modules and Cisco UCS Fabric Extender—**IOM modules are also known as Cisco FEXs or simply FEX modules. These modules serve as line cards to the FIs in the same way that Nexus series switches can have remote line cards. IOM modules also provide interface connections to blade servers. They multiplex data from blade servers and provide this data to FIs and do the same in the reverse direction. In production environments, IOM modules are always used in pairs to provide redundancy and failover.

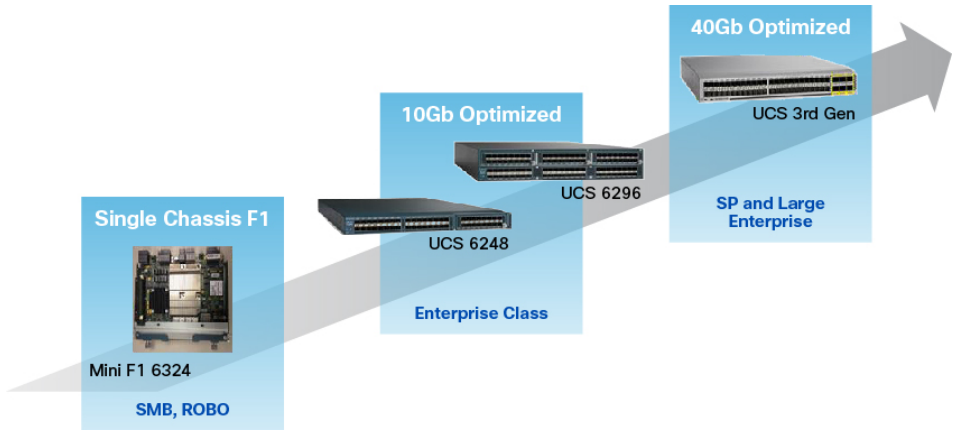
Important The 40G backplane setting is not applicable for 22xx IOMs.

- **Cisco UCS Blade Server Chassis—**The Cisco UCS 5100 Series Blade Server Chassis is a crucial building block of Cisco UCS, delivering a scalable and flexible architecture for current and future data center needs, while helping reduce total cost of ownership.
- **Cisco UCS Blade and Rack Servers—**Cisco UCS Blade servers are at the heart of the UCS solution. They come in various system resource configurations in terms of CPU, memory, and hard disk capacity. All blade servers are based on Intel Xeon processors. There is no AMD option available. The Cisco UCS rack-mount servers are standalone servers that can be installed and controlled individually. Cisco provides Fabric Extenders (FEXs) for the rack-mount servers. FEXs can be used to connect and manage rack-mount servers from FIs. Rack-mount servers can also be directly attached to the fabric interconnect. Small and Medium Businesses (SMBs) can choose from different blade configurations as per business needs
- **Cisco UCS I/O Adapters—**Cisco UCS B-Series Blade Servers are designed to support up to two network adapters. This design can reduce the number of adapters, cables, and access-layer switches by as much as half because it eliminates the need for multiple parallel infrastructure for both LAN and SAN at the server, chassis, and rack levels.

Cisco UCS Fabric Infrastructure Portfolio

The Cisco UCS fabric interconnects are top-of-rack devices and provide unified access to the Cisco UCS domain. The following illustration shows the evolution of the Cisco UCS fabric interconnects product family. The Cisco UCS Infrastructure hardware is now in its third generation.

Note The Cisco UCS 6100 Series Fabric Interconnects and Cisco UCS 2104 I/O Modules have reached end of life.



Cisco UCS I/O Modules and Cisco UCS Fabric Extenders

Cisco UCS 2200 and 2300 Series Fabric Extenders, also known as I/O modules, bring the unified fabric into the blade server enclosure, providing multiple 10 Gigabit Ethernet connections between servers and the fabric interconnect, simplifying diagnostics, cabling, and management.

The Cisco UCS 2200 Series extends the I/O fabric between the Cisco UCS 6200 Series fabric interconnects and the Cisco UCS 5100 Series blade server chassis. The Cisco UCS 2300 Series extends the I/O fabric between the Cisco UCS 6300 Series fabric interconnects and the Cisco UCS 5100 Series blade server chassis. These fabric extenders enable a lossless and deterministic Fibre Channel over Ethernet (FCoE) fabric to connect all servers and chassis together. Because the fabric extender is similar to a distributed line card, it does not perform any switching, and is managed as an extension of the fabric interconnects.

The Cisco UCS 2200 Series manages the fabric interconnect and the chassis environment, including the power supply, fans, and blade servers. Therefore, separate chassis management modules are not required. These fabric extenders fit into the back of the Cisco UCS 5100 Series chassis. Each Cisco UCS 5100 Series chassis can support up to two fabric extenders, allowing increased capacity and redundancy.

Cisco UCS 2232PP and Cisco UCS 2232TM-E extend the fabric between the Cisco UCS 6200 Series fabric interconnects and rack mount servers. Cisco UCS 2348UPQ extends the fabric between the Cisco UCS 6300 Series fabric interconnects and rack mount servers.

I/O Module Management, on page 37 provides more details about managing I/O Modules.

Cisco UCS Chassis

Cisco UCS Manager Release 3.1(1) provides support for Cisco UCS 5108 Blade Server Chassis

Chassis Management, on page 29 provides details on managing the chassis through Cisco UCS Manager.

Cisco UCS 5108 Blade Server Chassis

The Cisco UCS 5108 Blade Server Chassis, is six rack units (6RU) high, can mount in an industry-standard 19-inch rack, and uses standard front-to-back cooling. A chassis can accommodate up to eight half-width, or four full-width Cisco UCS B-Series Blade Servers form factors within the same chassis. By incorporating unified fabric and fabric-extender technology, the Cisco Unified Computing System enables the chassis to:

- Have fewer physical components
- Require no independent management
- Be more energy efficient than a traditional blade-server chassis

The Cisco UCS 5108 Blade Server Chassis is supported with all generations of fabric interconnects.

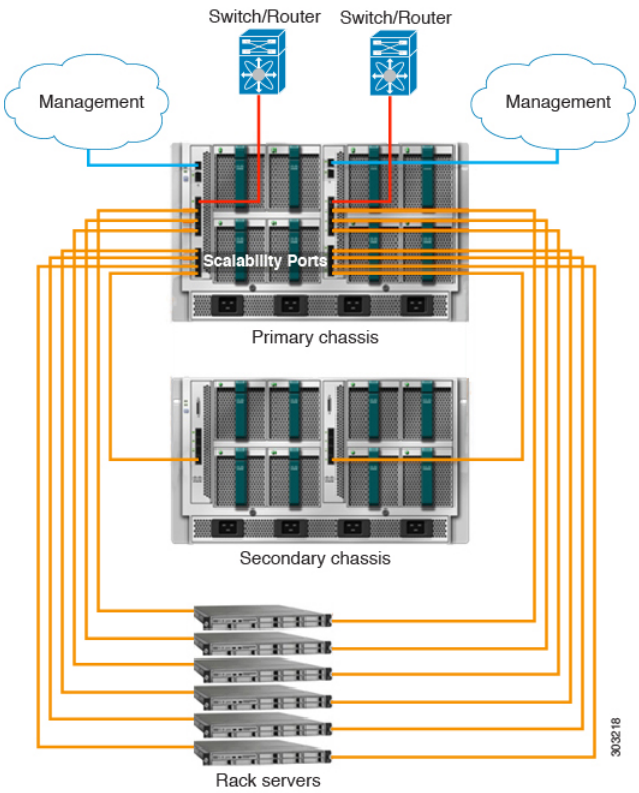
Cisco UCS Mini Infrastructure

The Cisco UCS Mini solution extends the Cisco UCS architecture into environments with requirements for smaller domains, including branch and remote offices, point-of-sale locations, and smaller IT environments. Cisco UCS Mini has three main infrastructure components:

- Cisco UCS 6324 fabric interconnect

- Cisco UCS blade server chassis
- Cisco UCS blade or rack mount servers

Figure 3: Cisco UCS Mini



In the Cisco UCS Mini solution, the Cisco UCS 6324 fabric interconnect is collapsed into the IO Module form factor, and is inserted into the IOM slot of the blade server chassis. The Cisco UCS 6324 fabric interconnect has 24 10G ports available on it. Sixteen of these ports are server facing, two 10G ports are dedicated to each of the eight half width blade slots. The remaining eight ports are divided into groups of four 1/10G Enhanced Small Form-Factor Pluggable (SFP+) ports and one 40G Quad Small Form-factor Pluggable (QSFP) port, which is called the 'scalability port'.

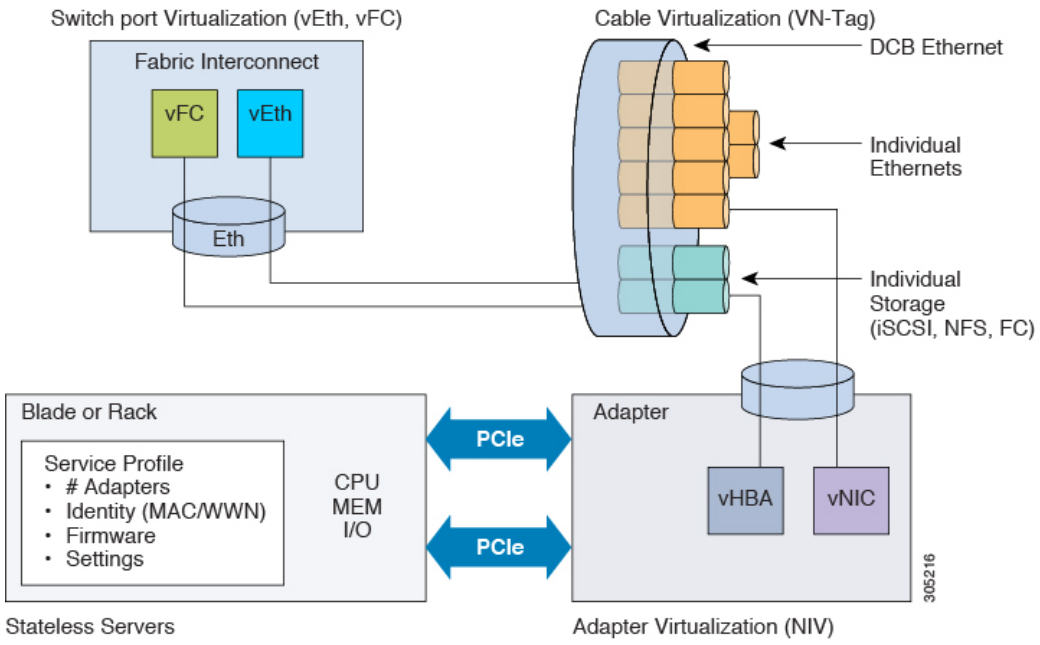
Cisco UCS Manager Release 3.1(1) introduces support for a second UCS 5108 chassis to an existing single-chassis Cisco UCS 6324 fabric interconnect setup. This extended chassis enables you to configure an additional 8 servers. Unlike the primary chassis, the extended chassis supports IOMs. Currently, it supports UCS-IOM-2204XP and UCS-IOM-2208XP IOMs. The extended chassis can only be connected through the scalability port on the FI-IOM.

Important Currently, Cisco UCS Manager supports only one extended chassis for UCS Mini.

Cisco UCS Infrastructure Virtualization

Cisco UCS is a single integrated system with switches, cables, adapters, and servers all tied together and managed by unified management software. One capability that enables this unification is the ability to virtualize every component of the system at every level. Switch port, cables, adapter, and servers can all be virtualized. Because of the virtualization capabilities at every component of the system, you have the unique capability to provide rapid provisioning of any service on any server on any blade through a system that is wired once. The following image illustrates these virtualization capabilities.

Figure 4: Virtualization Capabilities of Cisco UCS



Switch Port Virtualization

The physical interfaces provide physical connectivity for what are actually logical virtual interfaces on the fabric interconnects-virtual Fibre Channel interfaces (vFC) and virtual Ethernet interfaces (vEth). The logical connectivity to a server is provided through these virtual interfaces.

Cable Virtualization

The physical cables that connect to physical switch ports provide the infrastructure for logical and virtual cables. These virtual cables connect to virtual adapters on any given server in the system.

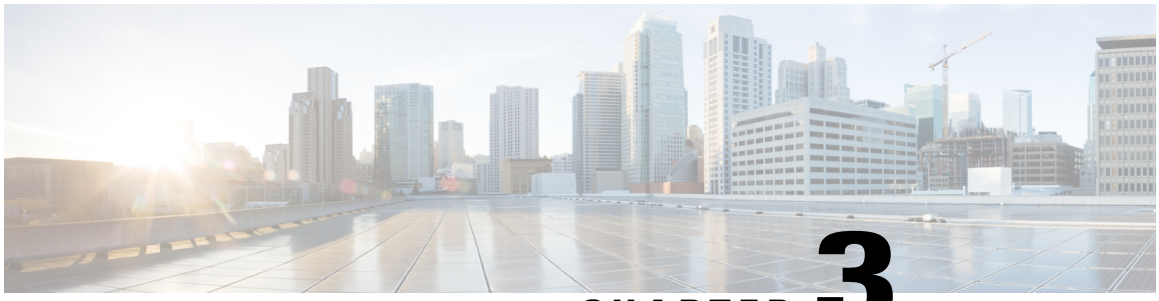
Adapter Virtualization

On the server, you have physical adapters, which provide physical infrastructure for virtual adapters. A virtual network interface card (vNIC) or virtual host bus adapter (vHBA) logically connects a host to a virtual interface on the fabric interconnect and allows the host to send and receive traffic through that interface. Each virtual interface in the fabric interconnect corresponds to a vNIC.

An adapter that is installed on the server appears to the server as multiple adapters through standard PCIe virtualization. When the server scans the PCIe bus, the virtual adapters provisioned appear to be physically plugged into the PCIe bus.

Server Virtualization

Server virtualization provides you with the ability of stateless servers. As part of the physical infrastructure, you have physical servers. However, the configuration of a server is derived from the service profile to which it is associated. All service profiles are centrally managed and stored in a database on the fabric interconnect. A service profile defines all of the settings of the server, for example, the number of adapters, virtual adapters, the identity of these adapters, the firmware of the adapters, and the firmware of the server. It contains all the settings of the server that you would typically configure on a physical machine. Because the service profile is abstracted from the physical infrastructure, you can apply it to any physical server and the physical server will be configured according to the configuration defined in the service profile. *Cisco UCS Manager Server Management Guide* provides detailed information about managing service profiles.



CHAPTER 3

Equipment Policies

Chassis/FEX Discovery Policy

The chassis/FEX discovery policy determines how the system reacts when you add a new chassis or FEX. Cisco UCS Manager uses the settings in the chassis/FEX discovery policy to determine the minimum threshold for the number of links between the chassis or FEX and the fabric interconnect and whether to group links from the IOM to the fabric interconnect in a fabric port channel.

In a Cisco UCS Mini setup, chassis discovery policy is supported only on the extended chassis.

Chassis Links

If you have a Cisco UCS domain with some of the chassis' wired with one link, some with two links, some with four links, and some with eight links, Cisco recommends configuring the chassis/FEX discovery policy for the minimum number links in the domain so that Cisco UCS Manager can discover all chassis.

Tip To establish the highest available chassis connectivity in a Cisco UCS domain where Fabric Interconnect is connected to different types of IO Modules supporting different max number of uplinks, select platform max value. Setting the platform max ensures that Cisco UCS Manager discovers the chassis including the connections and servers only when the maximum supported IOM uplinks are connected per IO Module.

After the initial discovery, re-acknowledge the chassis' that are wired for a greater number of links and Cisco UCS Manager configures the chassis to use all available links.

Cisco UCS Manager cannot discover any chassis that is wired for fewer links than are configured in the chassis/FEX discovery policy. For example, if the chassis/FEX discovery policy is configured for four links, Cisco UCS Manager cannot discover any chassis that is wired for one link or two links. Re-acknowledgement of the chassis resolves this issue.

The following table provides an overview of how the chassis/FEX discovery policy works in a multi-chassis Cisco UCS domain:

Table 3: Chassis/FEX Discovery Policy and Chassis Links

Number of Links Wired for the Chassis	1-Link Discovery Policy	2-Link Discovery Policy	4-Link Discovery Policy	8-Link Discovery Policy	Platform-Max Discovery Policy
1 link between IOM and fabric interconnects	Chassis is discovered by Cisco UCS Manager and added to the Cisco UCS domain as a chassis wired with 1 link.	Chassis connections and servers cannot be discovered by Cisco UCS Manager and are not added to the Cisco UCS domain.	Chassis connections and servers cannot be discovered by Cisco UCS Manager and are not added to the Cisco UCS domain.	Chassis connections and servers cannot be discovered by Cisco UCS Manager and are not added to the Cisco UCS domain.	Chassis connections and servers cannot be discovered by Cisco UCS Manager and are not added to the Cisco UCS domain.
2 links between IOM and fabric interconnects	Chassis is discovered by Cisco UCS Manager and added to the Cisco UCS domain as a chassis wired with 1 link. After initial discovery, reacknowledge the chassis and Cisco UCS Manager recognizes and uses the additional links.	Chassis is discovered by Cisco UCS Manager and added to the Cisco UCS domain as a chassis wired with 2 link.	Chassis connections and servers cannot be discovered by Cisco UCS Manager and are not added to the Cisco UCS domain.	Chassis connections and servers cannot be discovered by Cisco UCS Manager and are not added to the Cisco UCS domain.	Chassis connections and servers cannot be discovered by Cisco UCS Manager and are not added to the Cisco UCS domain.

Number of Links Wired for the Chassis	1-Link Discovery Policy	2-Link Discovery Policy	4-Link Discovery Policy	8-Link Discovery Policy	Platform-Max Discovery Policy
4 links between IOM and fabric interconnects	Chassis is discovered by Cisco UCS Manager and added to the Cisco UCS domain as a chassis wired with 1 link. After initial discovery, reacknowledge the chassis and Cisco UCS Manager recognizes and uses the additional links.	Chassis is discovered by Cisco UCS Manager and added to the Cisco UCS domain as a chassis wired with 2 links. After initial discovery, reacknowledge the chassis and Cisco UCS Manager recognizes and uses the additional links.	Chassis is discovered by Cisco UCS Manager and added to the Cisco UCS domain as a chassis wired with 4 link.	Chassis connections and servers cannot be discovered by Cisco UCS Manager and are not added to the Cisco UCS domain.	If the IOM has 4 links, the chassis is discovered by Cisco UCS Manager and added to the Cisco UCS domain as a chassis wired with 4 links. If the IOM has 8 links, the chassis is not fully discovered by Cisco UCS Manager.
8 links between IOM and fabric interconnects	Chassis is discovered by Cisco UCS Manager and added to the Cisco UCS domain as a chassis wired with 1 link. After initial discovery, reacknowledge the chassis and Cisco UCS Manager recognizes and uses the additional links.	Chassis is discovered by Cisco UCS Manager and added to the Cisco UCS domain as a chassis wired with 2 links. After initial discovery, reacknowledge the chassis and Cisco UCS Manager recognizes and uses the additional links.	Chassis is discovered by Cisco UCS Manager and added to the Cisco UCS domain as a chassis wired with 4 links. After initial discovery, reacknowledge the chassis and Cisco UCS Manager recognizes and uses the additional links.	Chassis is discovered by Cisco UCS Manager and added to the Cisco UCS domain as a chassis wired with 8 links.	Chassis is discovered by Cisco UCS Manager and added to the Cisco UCS domain as a chassis wired with 8 links.

Link Grouping

For hardware configurations that support fabric port channels, link grouping determines whether all of the links from the IOM to the fabric interconnect are grouped in to a fabric port channel during chassis discovery.

If the link grouping preference is set to **Port Channel**, all of the links from the IOM to the fabric interconnect are grouped in a fabric port channel. If set to **None**, links from the IOM are pinned to the fabric interconnect.

After a fabric port channel is created through Cisco UCS Manager, you can add or remove links by changing the link group preference and re-acknowledging the chassis, or by enabling or disabling the chassis from the port channel.

Note The link grouping preference only takes effect if both sides of the links between an IOM or FEX and the fabric interconnect support fabric port channels. If one side of the links does not support fabric port channels, this preference is ignored and the links are not grouped in a port channel.

Multicast Hardware Hash

In a port channel, by default, ingress multicast traffic on any port in the fabric interconnect (FI) selects a particular link between the IOM and the fabric interconnect to egress the traffic. To reduce potential issues with the bandwidth, and to provide effective load balancing of the ingress multicast traffic, hardware hashing is used for multicast traffic. When multicast hardware hashing is enabled, all links between the IOM and the fabric interconnect in a port channel can be used for multicast traffic.

Pinning

Pinning in Cisco UCS is only relevant to uplink ports. If you configure **Link Grouping Preference** as **None** during chassis discovery, the IOM forwards traffic from a specific server to the fabric interconnect through its uplink ports by using static route pinning.

The following table showcases how pinning is done between an IOM and the fabric interconnect based on the number of active fabric links between the IOM and the fabric interconnect.

Table 4: Pinning on an IOM

Number of Active Fabric Links	Server slot pinned to fabric link
1-Link	All the HIF ports are pinned to the active link
2-Link	1,3,5,7 to link-1 2,4,6,8 to link-2
4-Link	1,5 to link-1 2,6 to link-2 3,7 to link-3 4,8 to link-4

Number of Active Fabric Links	Server slot pinned to fabric link
8-Link (Applies only to 2208XP)	1 to link-1 2 to link-2 3 to link-3 4 to link-4 5 to link-5 6 to link-6 7 to link-7 8 to link-8

Only 1,2,4 and 8 links are supported. 3,5,6, and 7 links are not valid configurations.

Port-Channeling

While pinning traffic from a specific server to an uplink port provides you with greater control over the unified fabric and ensures optimal utilization of uplink port bandwidth, it could also mean excessive traffic over certain circuits. This issue can be overcome by using port channeling. Port channeling groups all links between the IOM and the fabric interconnect into one port channel. The port channel uses a load balancing algorithm to decide the link over which to send traffic. This results in optimal traffic management.

Cisco UCS supports port-channeling only through the Link Aggregation Control Protocol (LACP). For hardware configurations that support fabric port channels, link grouping determines whether all of the links from the IOM to the fabric interconnect are grouped into a fabric port channel during chassis discovery. If the **Link Grouping Preference** is set to **Port Channel**, all of the links from the IOM to the fabric interconnect are grouped in a fabric port channel. If this parameter is set to **None**, links from the IOM to the fabric interconnect are not grouped in a fabric port channel.

Once a fabric port channel is created, links can be added or removed by changing the link group preference and reacknowledging the chassis, or by enabling or disabling the chassis from the port channel.

Configuring the Chassis/FEX Discovery Policy

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org /**	Enters the root organization mode. **Note** The chassis/FEX discovery policy can be accessed only from the root organization.
Step 2	UCS-A /org # **scope chassis-disc-policy**	Enters organization chassis/FEX discovery policy mode.

	Command or Action	Purpose
Step 3	UCS-A /org/chassis-disc-policy # **set action** {**1-link** | **2-link** | **4-link** | **8-link** | **platform-max**}	Specifies the minimum threshold for the number of links between the chassis or FEX and the fabric interconnect.
Step 4	UCS-A /org/chassis-disc-policy # **set descr** *description*	(Optional) Provides a description for the chassis/FEX discovery policy. **Note** If your description includes spaces, special characters, or punctuation, you must begin and end your description with quotation marks. The quotation marks will not appear in the description field of any **show** command output.
Step 5	UCS-A /org/chassis-disc-policy # **set link-aggregation-pref** {**none** | **port-channel**}	Specifies whether the links from the IOMs or FEXes to the fabric interconnects are grouped in a port channel. **Note** The link grouping preference only takes effect if both sides of the links between an IOM or FEX and the fabric interconnect support fabric port channels. If one side of the links does not support fabric port channels, this preference is ignored and the links are not grouped in a port channel.
Step 6	UCS-A /org/chassis-disc-policy # **set multicast-hw-hash** {**disabled** | **enabled**}	Specifies whether the all the links between the IOM and the fabric interconnect in a port channel can be used for multicast traffic. • **disabled—**Only one link between the IOM and the fabric interconnect is used for multicast traffic • **enabled—**All links between the IOM and the fabric interconnect can be used for multicast traffic
Step 7	UCS-A /org/chassis-disc-policy # **set qualifier** *qualifier*	(Optional) Uses the specified server pool policy qualifications to associate this policy with a server pool.
Step 8	UCS-A /org/chassis-disc-policy # **commit-buffer**	Commits the transaction to the system configuration.

The following example scopes to the default chassis/FEX discovery policy, sets it to discover chassis with four links to a fabric interconnect, provides a description for the policy, specifies the server pool policy qualifications that will be used to qualify the chassis, and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # scope chassis-disc-policy
UCS-A /org/chassis-disc-policy* # set action 4-link
UCS-A /org/chassis-disc-policy* # set descr "This is an example chassis/FEX discovery
policy."
UCS-A /org/chassis-disc-policy* # set qualifier ExampleQual
UCS-A /org/chassis-disc-policy* # commit-buffer
UCS-A /org/chassis-disc-policy #
```

The following example scopes to the default chassis/FEX discovery policy, sets it to discover chassis with eight links to a fabric interconnect, provides a description for the policy, sets the link grouping preference to port channel, specifies the server pool policy qualifications that will be used to qualify the chassis, and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # scope chassis-disc-policy
UCS-A /org/chassis-disc-policy* # set action 8-link
UCS-A /org/chassis-disc-policy* # set descr "This is an example chassis/FEX discovery
policy."
UCS-A /org/chassis-disc-policy* # set link-aggregation-pref port-channel
UCS-A /org/chassis-disc-policy* # set qualifier ExampleQual
UCS-A /org/chassis-disc-policy* # commit-buffer
UCS-A /org/chassis-disc-policy #
```

The following example scopes to the default chassis/FEX discovery policy, sets it to discover chassis with four links to a fabric interconnect, provides a description for the policy, sets the link grouping preference to port channel, enables multicast hardware hashing, specifies the server pool policy qualifications that will be used to qualify the chassis, and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # scope chassis-disc-policy
UCS-A /org/chassis-disc-policy* # set action 4-link
UCS-A /org/chassis-disc-policy* # set descr "This is an example chassis/FEX discovery
policy."
UCS-A /org/chassis-disc-policy* # set link-aggregation-pref port-channel
UCS-A /org/chassis-disc-policy* # set multicast-hw-hash enabled
UCS-A /org/chassis-disc-policy* # set qualifier ExampleQual
UCS-A /org/chassis-disc-policy* # commit-buffer
UCS-A /org/chassis-disc-policy #
```

What to Do Next

To customize fabric port channel connectivity for a specific chassis, configure the chassis connectivity policy.

Chassis Connectivity Policy

The chassis connectivity policy determines the whether a specific chassis is included in a fabric port channel after chassis discovery. This policy is helpful for users who want to configure one or more chassis differently from what is specified in the global chassis discovery policy. The chassis connectivity policy also allows for different connectivity modes per fabric interconnect, further expanding the level of control offered with regards to chassis connectivity.

By default, the chassis connectivity policy is set to global. This means that connectivity control is configured when the chassis is newly discovered, using the settings configured in the chassis discovery policy. Once the chassis is discovered, the chassis connectivity policy controls whether the connectivity control is set to none or port channel.

Important The 40G backplane setting is not applicable for 22xx IOMs.

The chassis connectivity policy is created by Cisco UCS Manager only when the hardware configuration supports fabric port channels.

In a Cisco UCS Mini setup, the creation of a chassis connectivity policy is supported only on the extended chassis.

Configuring a Chassis Connectivity Policy

Changing the connectivity mode for a chassis might result in decreased VIF namespace.

Caution Changing the connectivity mode for a chassis results in chassis re-acknowledgement. Traffic might be disrupted during this time.

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 chassis-conn-policy** *chassis-num* [**a** | **b**}	Enters chassis connection policy organization mode for the specified chassis and fabric.
Step 3	UCS-A /org/chassis-conn-policy # **set link-aggregation-pref** {**global** | **none** | **port-channel**}	Specifies whether the links from the IOMs or FEXes to the fabric interconnects are grouped in a port channel. • **None—**No links are grouped in a port channel • **Port Channel—**All links from an IOM to a fabric interconnect are grouped in a port channel. • **Global—**The chassis inherits this configuration from the chassis discovery policy. This is the default value.
Step 4	UCS-A /org/chassis-conn-policy # **commit-buffer**	Commits the transaction to the system configuration.

The following example shows how to change the fabric port channel connectivity for two chassis. Chassis 6, fabric A is changed to port channel and chassis 12, fabric B is changed to discrete links:

```
UCS-A# scope org /
UCS-A /org # scope chassis-conn-policy 6 a
UCS-A /org/chassis-conn-policy # set link-aggregation-pref port-channel
UCS-A /org/chassis-conn-policy* # up
UCS-A /org* # scope chassis-conn-policy 12 b
UCS-A /org/chassis-conn-policy* # set link-aggregation-pref none
UCS-A /org/chassis-conn-policy* # commit-buffer
UCS-A /org/chassis-conn-policy #
```

Rack Server Discovery Policy

The rack server discovery policy determines how the system reacts when you add a new rack-mount server. Cisco UCS Manager uses the settings in the rack server discovery policy to determine whether any data on the hard disks are scrubbed and whether server discovery occurs immediately or needs to wait for explicit user acknowledgement.

Cisco UCS Manager cannot discover any rack-mount server that has not been correctly cabled and connected to the fabric interconnects. For information about how to integrate a supported Cisco UCS rack-mount server with Cisco UCS Manager, see the appropriate rack-mount server integration guide.

Configuring the Rack Server Discovery Policy

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org /**	Enters the root organization mode. **Note** The rack server discovery policy can be accessed only from the root organization.
Step 2	UCS-A /org # **scope rackserver-disc-policy**	Enters organization rack server discovery policy mode.
Step 3	UCS-A /org/rackserver-disc-policy # **set action** {**immediate** | **user-acknowledged**}	Specifies the way the system reacts when you add a new rack server.
Step 4	UCS-A /org/rackserver-disc-policy # **set descr** *description*	(Optional) Provides a description for the rack server discovery policy. **Note** If your description includes spaces, special characters, or punctuation, you must begin and end your description with quotation marks. The quotation marks will not appear in the description field of any **show** command output.
Step 5	UCS-A /org/rackserver-disc-policy # **set scrub-policy** *scrub-pol-name*	Specifies the scrub policy that should run on a newly discovered rack server.
Step 6	UCS-A /org/rackserver-disc-policy # **commit-buffer**	Commits the transaction to the system configuration.

The following example scopes to the default rack server discovery policy, sets it to immediately discover new rack servers, provides a description for the policy, specifies a scrub policy called scrubpol1, and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # scope rackserver-disc-policy
UCS-A /org/rackserver-disc-policy* # set action immediate
UCS-A /org/rackserver-disc-policy* # set descr "This is an example rackserver discovery
policy."
UCS-A /org/rackserver-disc-policy* # set scrub-policy scrubpol1
UCS-A /org/rackserver-disc-policy* # commit-buffer
UCS-A /org/rackserver-disc-policy #
```

Aging Time for the MAC Address Table

To efficiently switch packets between ports, the fabric interconnect maintains a MAC address table. It dynamically builds the MAC address table by using the MAC source address from the packets received and the associated port on which the packets were learned. The fabric interconnect uses an aging mechanism, defined by a configurable aging timer, to determine how long an entry remains in the MAC address table. If an address remains inactive for a specified number of seconds, it is removed from the MAC address table.

You can configure the amount of time (age) that a MAC address entry (MAC address and associated port) remains in the MAC address table.

Configuring the Aging Time for the MAC Address Table

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope eth-uplink**	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # **set mac-aging** {*dd hh mm ss* | **mode-default** | **never**}	Specifies the aging time for the MAC address table. Use the **mode-default** keyword to set the aging time to a default value dependent on the configured Ethernet switching mode. Use the **never** keyword to never remove MAC addresses from the table regardless of how long they have been idle.
Step 3	UCS-A /eth-uplink # **commit-buffer**	Commits the transaction to the system configuration.

The following example sets the aging time for the MAC address table to one day and 12 hours and commits the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # set mac-aging 01 12 00 00
UCS-A /eth-uplink* # commit-buffer
UCS-A /eth-uplink #
```

HA Version Holder Replacement

In releases earlier than Cisco UCS Manager Release 3.1(2), version holders are selected on a first come first serve basis. As chassis and rack servers are discovered, they can become version holders if they meet the requirements, and if the number of version holders has not reached the maximum permitted number. After a device is marked as a version holder, it persists as a version holder until it is decommissioned or removed. For example, if the connection status between the device and one or both fabric interconnects goes down, the device will not be removed as version holder.

In some situations, the shared storage devices that are selected as high availability (HA) version holders become unreachable for an extended period of time. Cisco UCS Manager Release 3.1(2) introduces the ability

to specify new preferred HA version holders corresponding to the devices that are functioning correctly. When you trigger a reelection of version holders, these new preferred HA devices are selected first.

Guidelines for Preferred HA Version Holder Replacement

Consider the following guidelines when replacing HA version holders:

- Both fabric interconnects must be up for device reelection to be triggered.
- Cisco UCS Mini does not support preferred HA version holder replacement.
- A preferred version holder can be any device that is currently supported for shared storage.
- You can specify up to five preferred version holder devices. However, only three devices will be selected for active HA access.
- When you trigger shared storage device reelection, it removes all currently active devices and selects a new set of active devices. This set of devices may include previously active devices. Devices that are specified as preferred version holders are selected first as active devices.
- You can trigger reelection of shared storage devices at any time. However, the device will be selected as a version holder only in the following scenarios:
  - When the connection path is both fabric interconnect A and B for UCS B Series blade chassis
  - When the connection status is both fabric interconnect A and B for UCS C Series racks
- For a device to be selected as a version holder, the following requirements must be met:
  - There must be less than three devices selected for active HA access.
  - Chassis removal must not be in progress.
  - A chassis that has been removed from the system must not be used as a version holder.
  - The connection path must be both fabric interconnect A and B.
- Replacement of HA version holders can be done only through Cisco UCS Manager CLI.

Creating a Preferred Version Holder

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system**	Enters system mode.
Step 2	UCS-A /system # **create preferred-ha-device** *device-serial*	Creates the specified preferred HA device.
Step 3	UCS-A /system/ preferred-ha-device # **commit-buffer**	Commits the transaction to the system configuration.

	Command or Action	Purpose
Step 4	UCS-A /system/ preferred-ha-device* # **exit**	Enters system mode.
Step 5	UCS-A /system # **show preferred-ha-devices**	Displays the list of preferred HA version holders and whether they are active or not.

This example shows how to create a preferred version holder:

```
UCS-A# scope system
UCS-A /system # create preferred-ha-device FCH1606V02F
UCS-A /system/ preferred-ha-device* # commit-buffer
UCS-A /system/ preferred-ha-device # exit
UCS-A /system # show preferred-ha-devices

Preferred Version Holder:
    Chassis Serial Active
    -------------- ------
    FCH1606V02F    Yes
    FOX1636H6R3    Yes
    FOX1636H6R4    No
```

What to Do Next

Trigger a reelection of version holders.

Deleting a Preferred Version Holder

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system**	Enters system mode.
Step 2	UCS-A /system # **delete preferred-ha-device** *device-serial*	Deletes the specified preferred HA device.
Step 3	UCS-A /system/ preferred-ha-device* # **commit-buffer**	Commits the transaction to the system configuration.
Step 4	UCS-A /system/ preferred-ha-device # **exit**	Enters system mode.
Step 5	UCS-A /system # **show preferred-ha-devices**	Displays the list of preferred HA version holders and whether they are active or not.

This example shows how to delete a preferred version holder:

```
UCS-A# scope system
UCS-A /system # delete preferred-ha-device FCH1606V02F
UCS-A /system/ preferred-ha-device* # commit-buffer
UCS-A /system/ preferred-ha-device # exit
UCS-A /system # show preferred-ha-devices
```

```
Preferred Version Holder:
    Chassis Serial Active
    ------------- ------
    FOX1636H6R3    Yes
    FOX1636H6R4    No
```

Triggering the Reelection of Version Holders

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system**	Enters system mode.
Step 2	UCS-A /system # **re-elect-ha-devices**	Triggers reelection of version holders for HA devices.

This example shows how to trigger the reelection of version holders:

```
UCS-A# scope system
UCS-A /system # re-elect-ha-devices
```

Displaying Operational Version Holders

You can use this command to display all operational version holders, including preferred version holders.

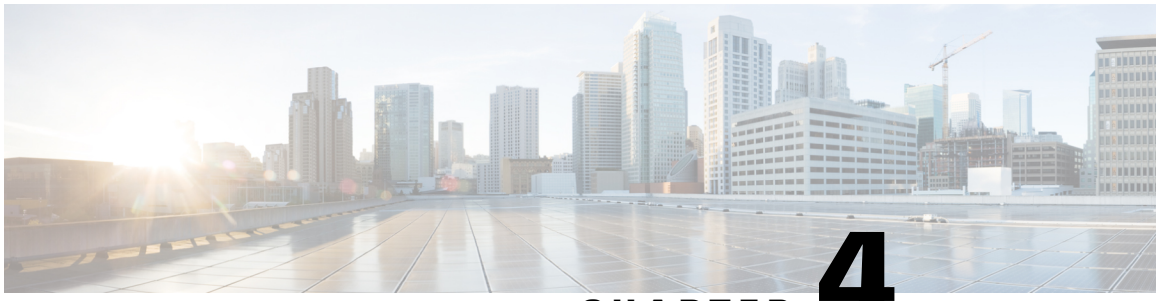
Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system**	Enters system mode.
Step 2	UCS-A /system # **show operational-ha-devices**	Displays the list of all currently operational HA version holders.

This example shows how to display all currently operational version holders:

```
UCS-A# scope system
UCS-A /system # show operational-ha-devices

Current Version Holder:
    Serial
    ------
    FOX1636H6R5
```



CHAPTER 4

Chassis Management

Chassis Management in Cisco UCS Manager CLI

You can manage and monitor all chassis in a Cisco UCS domain through Cisco UCS Manager CLI.

The Cisco UCS S3260 Chassis

Cisco UCS Manager Release 3.1(2) introduces support for the Cisco UCS S3260 chassis on Cisco UCS 6300 Series, and 6200 Series fabric interconnect setups.

The Cisco UCS S3260 chassis is a 4U chassis that is designed to operate in a standalone environment and also as part of the Cisco Unified Computing System. It has the following main components:

- Four 1050 Watt AC modular power supplies (2 + 2 shared and redundant mode of operation)
- Two System IO Controller (SIOC) slots
- Two storage server slots out of which one can be used for storage expansion

Note The second server slot in the chassis can be utilized by an HDD expansion tray module for an additional four 3.5” drives.

- 56 3.5” drive bays with an optional 4 x 3.5” HDD expansion tray module instead of the second server
- Up to 360 TB storage capacity by using 6 TB HDDs
- Serial Attached SCSI (SAS) expanders that can be configured to assign the 3.5” drives to individual server modules
- The two servers in the chassis can be replaced by a single, dual-height server with an IO expander

Cisco UCS 5108 Blade Server Chassis

The Cisco UCS 5100 Series Blade Server Chassis is logically part of the fabric interconnects, thus creating a single, coherent management domain and decreasing management complexity. In the management domain, server management is handled by the fabric interconnect, while I/O and network management is extended to every chassis and blade server. Basing the I/O infrastructure on a unified fabric allows the Cisco Unified Computing System to have a simple and streamlined chassis yet offer a comprehensive set of I/O options. This results in the chassis having only five basic components:

- The physical chassis with passive midplane and active environmental monitoring circuitry
- Four power-supply bays with power entry in the rear, and redundant-capable, hot-swappable power supply units accessible from the front panel
- Eight hot-swappable fan trays, each with two fans
- Two fabric extender slots accessible from the back panel
- Eight blade server slots accessible from the front panel

The blade server chassis has flexible partitioning with removable dividers to handle two blade server form factors:

- Half-width blade servers have access to power and two 10GBASE-KR connections, one to each fabric extender slot.
- Full-width blade servers connect to power and two connections to each fabric extender.

Extended Chassis for UCS Mini

Cisco UCS Manager Release 3.1(1) introduces support for an extended UCS 5108 chassis to an existing single-chassis Cisco UCS 6324 fabric interconnect setup. This extended chassis enables you to configure an additional 8 servers. Unlike the primary chassis, the extended chassis supports IOMs. Currently, it supports UCS-IOM-2204XP and UCS-IOM-2208XP IOMs. The extended chassis can only be connected through the scalability port on the FI-IOM.

Important Currently, Cisco UCS Manager supports only one extended chassis for UCS Mini.

To use a extended chassis, do the following:

- Connect the second Cisco UCS 5108 chassis to the existing single-chassis Cisco UCS 6324 Series fabric interconnect configuration through the scalability port.
- Configure the chassis discovery policy.
- Configure the server ports and wait for the second chassis to be discovered.

Guidelines for Removing and Decommissioning Chassis

Consider the following guidelines when deciding whether to remove or decommission a chassis using Cisco UCS Manager:

Decommissioning a Chassis

Decommissioning is performed when a chassis is physically present and connected but you want to temporarily remove it from the Cisco UCS Manager configuration. Because it is expected that a decommissioned chassis will be eventually recommissioned, a portion of the chassis' information is retained by Cisco UCS Manager for future use.

Removing a Chassis

Removing is performed when you physically remove a chassis from the system. Once the physical removal of the chassis is completed, the configuration for that chassis can be removed in Cisco UCS Manager.

Note You cannot remove a chassis from Cisco UCS Manager if it is physically present and connected.

If you need to add a removed chassis back to the configuration, it must be reconnected and then rediscovered. During rediscovery Cisco UCS Manager will assign the chassis a new ID that may be different from ID that it held before.

Acknowledging a Chassis

Perform the following procedure if you increase or decrease the number of links that connect the chassis to the fabric interconnect. Acknowledging the chassis ensures that Cisco UCS Manager is aware of the change in the number of links and that traffics flows along all available links.

After you enable or disable a port on a fabric interconnect, wait for at least 1 minute before you re-acknowledge the chassis. If you re-acknowledge the chassis too soon, the pinning of server traffic from the chassis might not get updated with the changes to the port that you enabled or disabled.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **acknowledge chassis** *chassis-num*	Acknowledges the specified chassis.
Step 2	UCS-A# **commit-buffer**	Commits the transaction to the system configuration.

The following example acknowledges chassis 2 and commits the transaction:

```
UCS-A# acknowledge chassis 2
UCS-A* # commit-buffer
UCS-A #
```

Decommissioning a Chassis

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **decommission chassis** *chassis-num*	Decommissions the specified chassis.
Step 2	UCS-A# **commit-buffer**	Commits the transaction to the system configuration.

The decommission may take several minutes to complete.

The following example decommissions chassis 2 and commits the transaction:

```
UCS-A# decommission chassis 2
UCS-A* # commit-buffer
UCS-A # show chassis

Chassis:
    Chassis    Overall Status           Admin State
    ---------- ------------------------ -----------
             1 Operable                 Acknowledged
             2 Accessibility Problem    Decommission
UCS-A #
```

Removing a Chassis

Before You Begin

Physically remove the chassis before performing the following procedure.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **remove chassis** *chassis-num*	Removes the specified chassis.
Step 2	UCS-A# **commit-buffer**	Commits the transaction to the system configuration.

The removal may take several minutes to complete.

The following example removes chassis 2 and commits the transaction:

```
UCS-A# remove chassis 2
UCS-A* # commit-buffer
UCS-A #
```

Recommissioning a Chassis

This procedure returns the chassis to the configuration and applies the chassis discovery policy to the chassis. After this procedure, you can access the chassis and any servers in it.

Note This procedure is not applicable for Cisco UCS S3260 Chassis.

Before You Begin

Collect the following information about the chassis to be recommissioned by using the **show chassis decommissioned** or **show chassis inventory** commands:

- Vendor name
- Model name
- Serial number

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **recommission chassis** *vendor-name model-name serial-num*	Recommissions the specified chassis.
Step 2	UCS-A# **commit-buffer**	Commits the transaction to the system configuration. **Note** After recommissioning a chassis and committing the transaction, if you immediately run the **show chassis** command, you may not see any change in the Admin State of the chassis. It may take a while before the state of the chassis changes after it is recommissioned.

The following example recommissions a Cisco UCS 5108 chassis and commits the transaction:

```
UCS-A# show chassis

Chassis:
    Chassis    Overall Status           Admin State
    ---------- ------------------------ -----------
             1 Accessibility Problem    Decommission

UCS-A# recommission chassis "Cisco Systems Inc" "N20-C6508" FOX1252GNNN
UCS-A* # commit-buffer
UCS-A #
```

Renumbering a Chassis

Note You cannot renumber a blade server through Cisco UCS Manager. The ID assigned to a blade server is determined by its physical slot in the chassis. To renumber a blade server, you must physically move the server to a different slot in the chassis.

Note This procedure is not applicable for Cisco UCS S3260 Chassis.

Before You Begin

If you are swapping IDs between chassis, you must first decommission both chassis, then wait for the chassis decommission FSM to complete before proceeding with the renumbering steps.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **show chassis inventory**	Displays information about your chassis.
Step 2	Verify that the chassis inventory does not include the following:	• The chassis you want to renumber • A chassis with the number you want to use If either of these chassis are listed in the chassis inventory, decommission those chassis. You must wait until the decommission FSM is complete and the chassis are not listed in the chassis inventory before continuing. This might take several minutes. To see which chassis have been decommissioned, issue the **show chassis decommissioned** command.
Step 3	UCS-A# **recommission chassis** *vendor-name model-name serial-num* [*chassis-num*]	Recommissions and renumbers the specified chassis.

	Command or Action	Purpose
Step 4	UCS-A# **commit-buffer**	Commits the transaction to the system configuration.

The following example decommissions two Cisco UCS chassis (chassis 8 and 9), switches their IDs, and commits the transaction:

```
UCS-A# show chassis inventory

    Chassis    PID       Vendor          Serial (SN) HW Revision
    ------- --------- ---------------   ----------- -------------
          1 N20-C6508 Cisco Systems Inc FOX1252GAAA 0
          2 N20-C6508 Cisco Systems Inc FOX1252GBBB 0
          3 N20-C6508 Cisco Systems Inc FOX1252GCCC 0
          4 N20-C6508 Cisco Systems Inc FOX1252GDDD 0
          5 N20-C6508 Cisco Systems Inc FOX1252GEEE 0
          6 N20-C6508 Cisco Systems Inc FOX1252GFFF 0
          7 N20-C6508 Cisco Systems Inc FOX1252GGGG 0
          8 N20-C6508 Cisco Systems Inc FOX1252GHHH 0
          9 N20-C6508 Cisco Systems Inc FOX1252GIII 0
         10 N20-C6508 Cisco Systems Inc FOX1252GJJJ 0
         11 N20-C6508 Cisco Systems Inc FOX1252GKKK 0
         12 N20-C6508 Cisco Systems Inc FOX1252GLLL 0
         13 N20-C6508 Cisco Systems Inc FOX1252GMMM 0
         14 N20-C6508 Cisco Systems Inc FOX1252GNNN 0

UCS-A# decommission chassis 8
UCS-A*# commit-buffer
UCS-A# decommission chassis 9
UCS-A*# commit-buffer
UCS-A# show chassis inventory

    Chassis    PID       Vendor          Serial (SN) HW Revision
    ------- --------- ---------------   ----------- -------------
          1 N20-C6508 Cisco Systems Inc FOX1252GAAA 0
          2 N20-C6508 Cisco Systems Inc FOX1252GBBB 0
          3 N20-C6508 Cisco Systems Inc FOX1252GCCC 0
          4 N20-C6508 Cisco Systems Inc FOX1252GDDD 0
          5 N20-C6508 Cisco Systems Inc FOX1252GEEE 0
          6 N20-C6508 Cisco Systems Inc FOX1252GFFF 0
          7 N20-C6508 Cisco Systems Inc FOX1252GGGG 0
         10 N20-C6508 Cisco Systems Inc FOX1252GJJJ 0
         11 N20-C6508 Cisco Systems Inc FOX1252GKKK 0
         12 N20-C6508 Cisco Systems Inc FOX1252GLLL 0
         13 N20-C6508 Cisco Systems Inc FOX1252GMMM 0
         14 N20-C6508 Cisco Systems Inc FOX1252GNNN 0

UCS-A# show chassis decommissioned

    Chassis    PID       Vendor          Serial (SN) HW Revision
    ------- --------- ---------------   ----------- -------------
          8 N20-C6508 Cisco Systems Inc FOX1252GHHH 0
          9 N20-C6508 Cisco Systems Inc FOX1252GIII 0

UCS-A# recommission chassis "Cisco Systems Inc" "N20-C6508" FOX1252GHHH 9
UCS-A* # commit-buffer
UCS-A# recommission chassis "Cisco Systems Inc" "N20-C6508" FOX1252GIII 8
UCS-A* # commit-buffer
UCS-A # show chassis inventory

    Chassis    PID       Vendor          Serial (SN) HW Revision
    ------- --------- ---------------   ----------- -------------
          1 N20-C6508 Cisco Systems Inc FOX1252GAAA 0
          2 N20-C6508 Cisco Systems Inc FOX1252GBBB 0
          3 N20-C6508 Cisco Systems Inc FOX1252GCCC 0
          4 N20-C6508 Cisco Systems Inc FOX1252GDDD 0
          5 N20-C6508 Cisco Systems Inc FOX1252GEEE 0
```

```
 6 N20-C6508 Cisco Systems Inc FOX1252GFFF 0
 7 N20-C6508 Cisco Systems Inc FOX1252GGGG 0
 8 N20-C6508 Cisco Systems Inc FOX1252GIII 0
 9 N20-C6508 Cisco Systems Inc FOX1252GHHH 0
10 N20-C6508 Cisco Systems Inc FOX1252GJJJ 0
11 N20-C6508 Cisco Systems Inc FOX1252GKKK 0
12 N20-C6508 Cisco Systems Inc FOX1252GLLL 0
13 N20-C6508 Cisco Systems Inc FOX1252GMMM 0
14 N20-C6508 Cisco Systems Inc FOX1252GNNN 0
```

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.

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 #
```

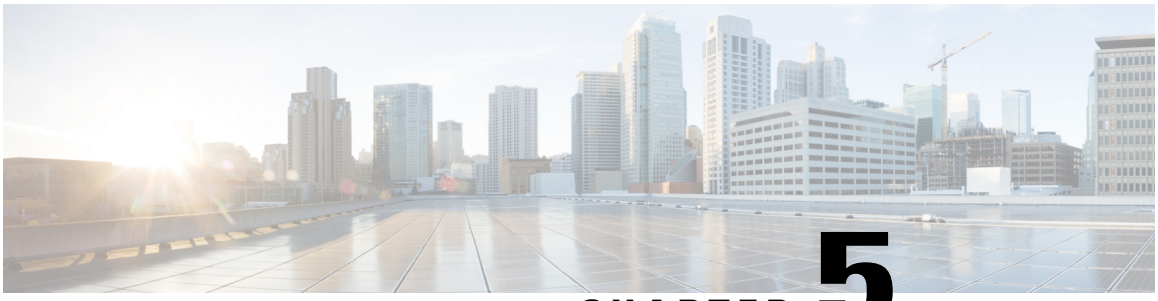
Turning Off the Locator LED for a Chassis

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.

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 #
```



CHAPTER 5

I/O Module Management

I/O Module Management in Cisco UCS Manager CLI

You can manage and monitor all I/O modules in a Cisco UCS domain through Cisco UCS Manager CLI. Cisco UCS Manager Release 3.1 introduces the Cisco UCS-IOM-2304 I/O module with 40 GbE connectivity to the Cisco UCS 6300 Series Fabric Interconnect. The *Cisco UCS Manager Getting Started Guide* provides more information about this functionality.

Acknowledging an IO Module

Cisco UCS Manager Release 2.2(4) introduces the ability to acknowledge a specific IO module in a chassis.

Note This operation rebuilds the network connectivity between the IO module and the Fabrics to which it is connected.

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 # **acknowledge iom** {**1** | **2**}	Acknowledges the specified IOM in the chassis.

	Command or Action	Purpose
Step 3	UCS-A /chassis* # **commit-buffer**	Commits the transaction to the system configuration.

The following example acknowledges IO Module 1 and commits the transaction:

```
UCS-A# scope chassis 1
UCS-A /chassis # acknowledge iom 1
UCS-A /chassis* # commit-buffer
UCS-A /chassis #
```

Resetting the I/O Module

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 iom** {**a b**}	Enters chassis IOM mode for the specified IOM.
Step 3	UCS-A /chassis/iom # **reset**	Resets the IOM.
Step 4	UCS-A /chassis/iom # **commit-buffer**	Commits the transaction to the system configuration.

The following example resets the IOM on fabric A and commits the transaction:

```
UCS-A# scope chassis 1
UCS-A /chassis # scope iom a
UCS-A /chassis/iom # reset
UCS-A /chassis/iom* # commit-buffer
UCS-A /chassis/iom #
```

Resetting an I/O Module from a Peer I/O Module

Sometimes, I/O module upgrades can result in failures or I/O modules can become unreachable from Cisco UCS Manager due to memory leaks. You can now reboot an I/O module that is unreachable through its peer I/O module.

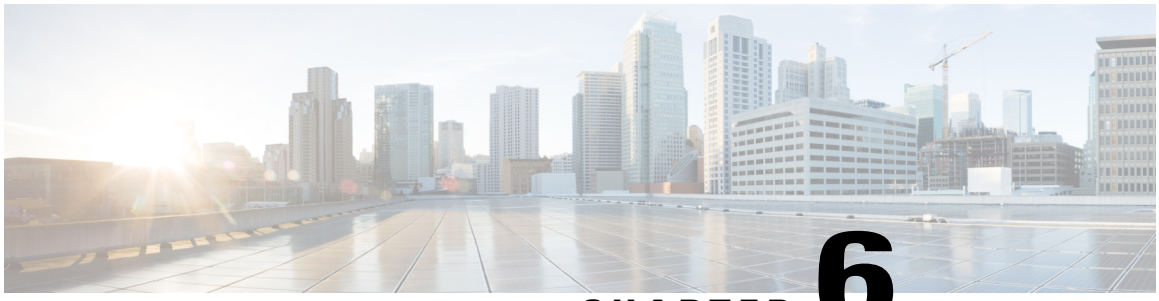
Resetting the I/O module restores the I/O module to factory default settings, deletes all cache files and temporary files, but retains the size-limited OBFL file.

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 iom** {*a b*}	Enters chassis IOM mode for the specified IOM. Specify the peer IOM of the IOM that you want to reset.
Step 3	UCS-A /chassis/iom # **reset-peer**	Resets the peer IOM of the specified IOM.
Step 4	UCS-A /chassis/iom* # **commit-buffer**	Commits the transaction to the system configuration.

This example shows how to reset IOM b from IOM a:

```
UCS-A# scope chassis 1
UCS-A /chassis # scope iom a
UCS-A /chassis/iom # reset-peer
UCS-A /chassis/iom* # commit-buffer
```



CHAPTER 6

SIOC Management

SIOC Management in Cisco UCS Manager

You can manage and monitor all System Input/Output Controllers (SIOC) in a Cisco UCS domain through Cisco UCS Manager.

SIOC Removal or Replacement

You can remove or replace an SIOC from a chassis. Removal or replacement of an SIOC is a service-affecting operation, which requires you to power down the entire chassis.

Guidelines for SIOC Removal

- To remove the active SIOC, or both SIOCs, shut down and remove power from the entire chassis. You must disconnect all power cords to completely remove power.
- Removal of SIOCs from a chassis results in the entire chassis being disconnected from Cisco UCS Manager.

SIOC Removal

Do the following to remove an SIOC from the system:

1. Shut down and remove power from the entire chassis. You must disconnect all power cords to completely remove power.
2. Disconnect the cables connecting the SIOC to the system.
3. Remove the SIOC from the system.

SIOC Replacement

Do the following to remove an SIOC from the system and replace it with another SIOC:

1. Shut down and remove power from the entire chassis. You must disconnect all power cords to completely remove power.
2. Disconnect the cables connecting the SIOC to the system.
3. Remove the SIOC from the system.
4. Connect the new SIOC to the system.
5. Connect the cables to the SIOC.
6. Connect power cords and then power on the system.
7. Acknowledge the new SIOC.

   The server connected to the replaced SIOC is rediscovered.

Note If the firmware of the replaced SIOC is not the same version as the peer SIOC, then it is recommended to update the firmware of the replaced SIOC by re-triggering chassis profile association.

Acknowledging an SIOC

Cisco UCS Manager has the ability to acknowledge a specific SIOC in a chassis. Perform the following procedure when you replace an SIOC in a chassis.

Caution This operation rebuilds the network connectivity between the SIOC and the fabric interconnects to which it is connected. The server corresponding to this SIOC becomes unreachable, and traffic is disrupted.

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 # **acknowledge sioc** {**1** | **2**}	Acknowledges the specified SIOC in the chassis.
Step 3	UCS-A /chassis* # **commit-buffer**	Commits the transaction to the system configuration.

The following example acknowledges SIOC 1 and commits the transaction:

```
UCS-A# scope chassis 3
UCS-A /chassis # acknowledge sioc 1
UCS-A /chassis* # commit-buffer
UCS-A /chassis #
```

Resetting the CMC

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 sioc** {**1** | **2**}	Enters the specified SIOC in the chassis.
Step 3	UCS-A /chassis/sioc # **scope cmc**	Enters the CMC of the selected SIOC slot.
Step 4	UCS-A /chassis/sioc/cmc # **reset**	Resets the CMC.
Step 5	UCS-A /chassis/sioc/cmc* # **commit-buffer**	Commits the transaction to the system configuration.

The following example resets the CMC on SIOC 1 and commits the transaction:

```
UCS-A# scope chassis 1
UCS-A /chassis # scope sioc 1
UCS-A /chassis/sioc # scope cmc
UCS-A /chassis/sioc/cmc # reset
UCS-A /chassis/sioc/cmc* # commit-buffer
```

CMC Secure Boot

With Chassis Management Controller (CMC) secure boot, only Cisco-signed firmware images can be installed and run on the CMC. When the CMC is updated, the image is certified before the firmware is flashed. If certification fails, the firmware is not flashed. This prevents unauthorized access to the CMC firmware.

Guidelines and Limitations for CMC Secure Boot

- CMC secure boot is supported only on the Cisco UCS S3260 chassis.
- When chassis association is in progress, enabling secure boot on one of the SIOCs will result in a failed operation.
- After CMC secure boot is enabled, it cannot be disabled.
- CMC secure boot is specific to the SIOC on which it is enabled. If you replace the SIOC on which CMC secure boot is enabled, the **Secure boot operational state** field will now display the secure boot status of the new SIOC.
- After CMC secure boot is enabled on a chassis, you cannot move the chassis back to standalone mode and downgrade the firmware to a CMC firmware image earlier than Cisco IMC Release 2.0(13).
- The **Secure boot operational state** field shows the secure boot status. This can be one of the following:

- **Disabled—**When CMC secure boot is not enabled. This is the default state.
- **Enabling—**When CMC secure boot is being enabled.
- **Enabled—**When CMC secure boot is enabled.

Enabling CMC Secure Boot

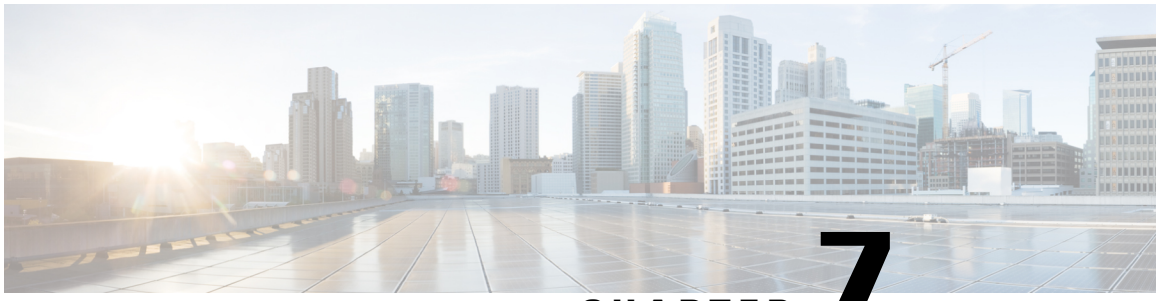
Cisco UCS Manager Release 3.1(2) introduces the ability to enable Chassis Management Controller (CMC) secure boot so that only Cisco-signed firmware images can be installed and run on the CMC.

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 sioc** {**1** | **2**}	Enters the specified SIOC in the chassis.
Step 3	UCS-A /chassis/sioc # **scope cmc**	Enters the CMC of the selected SIOC slot.
Step 4	UCS-A /chassis/sioc/cmc # **enable secure-boot**	Enables CMC secure boot. If you run this command when the secure boot state is **enabled**, Cisco UCS Manager will display an error message and the operation will fail. **Note** This is an irreversible operation. You cannot disable CMC secure boot.
Step 5	UCS-A /chassis/sioc/cmc* # **commit-buffer**	Commits the transaction to the system configuration.

The following example enables CMC secure boot on SIOC 1 and commits the transaction:

```
UCS-A# scope chassis 1
UCS-A /chassis # scope sioc 1
UCS-A /chassis/sioc # scope cmc
UCS-A /chassis/sioc/cmc # enable secure-boot
Warning: This is an irreversible operation.
Do you want to proceed? [Y/N] Y
UCS-A /chassis/sioc/cmc* # commit-buffer
```



CHAPTER 7

Power Management in Cisco UCS

Power Capping in Cisco UCS

You can control the maximum power consumption on a server through power capping, as well as manage the power allocation in the Cisco UCS Manager for the UCS B-Series Blade Servers, UCS Mini, and mixed UCS domains.

UCS Manager supports power capping on the following servers:

- UCS Mini 6324
- UCS 6300 Series Fabric Interconnects

You can use Policy Driven Chassis Group Power Cap, or Manual Blade Level Power Cap methods to allocate power that applies to all of the servers in a chassis.

Cisco UCS Manager provides the following power management policies to help you allocate power to your servers:

Power Management Policies	Description
Power Policy	Specifies the redundancy for power supplies in all chassis in a Cisco UCS domain.
Power Control Policies	Specifies the priority to calculate the initial power allocation for each blade in a chassis.
Global Power Allocation	Specifies the Policy Driven Chassis Group Power Cap or the Manual Blade Level Power Cap to apply to all servers in a chassis.
Global Power Profiling	Specifies how the power cap values of the servers are calculated. If it is enabled, the servers will be profiled during discovery through benchmarking. This policy applies when the Global Power Allocation Policy is set to Policy Driven Chassis Group Cap.

Power Policy for Cisco UCS Servers

The power policy is global and is inherited by all of the chassis' managed by the Cisco UCS Manager instance. You can add the power policy to a service profile to specify the redundancy for power supplies in all chassis' in the Cisco UCS domain. This policy is also known as the PSU policy.

For more information about power supply redundancy, see *Cisco UCS 5108 Server Chassis Hardware Installation Guide*.

Configuring the Power 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 psu-policy**	Enters PSU policy mode.
Step 3	UCS-A /org/psu-policy # **set redundancy** {**grid** | **n-plus-1** | **non-redund**}	Specifies one of the following redundancy types: • **grid —**Two power sources are turned on, or the chassis requires greater than N+1 redundancy. If one source fails (which causes a loss of power to one or two PSUs), the surviving PSUs on the other power circuit continue to provide power to the chassis. • **n-plus-1 —**The total number of PSUs to satisfy non-redundancy, plus one additional PSU for redundancy, are turned on and equally share the power load for the chassis. If any additional PSUs are installed, Cisco UCS Manager sets them to a "turned-off" state. • **non-redund —**All installed power supplies (PSUs) are turned on and the load is evenly balanced. Only smaller configurations (requiring less than 2500W) can be powered by a single PSU. For more information about power redundancy, see the *Cisco UCS 5108 Server Chassis Installation Guide*.
Step 4	UCS-A /org/psu-policy # **commit-buffer**	Commits the transaction to the system configuration.

The following example configures the power policy to use grid redundancy and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # scope psu-policy
UCS-A /org/psu-policy # set redundancy grid
UCS-A /org/psu-policy* # commit-buffer
UCS-A /org/psu-policy #
```

Power Supply for Redundancy Method

PSU Redundancy	Max Power @ 220 V (Watts)	Max Power @ 110 V (Watts)
1+1 (N+1) OR 1 (N)	2500	1300
2+1 (N+1) OR 2 (N) or 2+2 (Grid)	5000	2600

PSU Redundancy	Max Power @ 220 V (Watts)	Max Power @ 110 V (Watts)
3+1 (N+1) OR 3 (N)	5472	3900
4 (N)	5472	5200

Policy Driven Power Capping

Policy Driven Chassis Group Power Capping

When you select the Policy Driven Chassis Group Power Cap in the Global Cap Policy, Cisco UCS can maintain the over-subscription of servers without risking power failures. You can achieve over-subscription through a two-tier process. For example, at the chassis level, Cisco UCS divides the amount of power available among members of the power group, and at the blade level, the amount of power allotted to a chassis is divided among blades based on priority.

Each time a service profile is associated or disassociated, Cisco UCS Manager recalculates the power allotment for each blade server within the chassis. If necessary, power from lower-priority service profiles is redistributed to higher-priority service profiles.

UCS power groups cap power in less than one second to safely protect data center circuit breakers. A blade must stay at its cap for 20 seconds before the chassis power distribution is optimized. This is intentionally carried out over a slower timescale to prevent reacting to transient spikes in demand.

Note The system reserves enough power to boot a server in each slot, even if that slot is empty. This reserved power cannot be leveraged by servers requiring more power. Blades that fail to comply with the power cap are penalized.

Power Control Policy

Cisco UCS uses the priority set in the power control policy along with the blade type and configuration to calculate the initial power allocation for each blade within a chassis. During normal operation, the active blades within a chassis can borrow power from idle blades within the same chassis. If all blades are active and reach the power cap, service profiles with higher priority power control policies take precedence over service profiles with lower priority power control policies.

Priority is ranked on a scale of 1-10, where 1 indicates the highest priority and 10 indicates lowest priority. The default priority is 5.

For mission-critical application a special priority called no-cap is also available. Setting the priority to no-cap prevents Cisco UCS from leveraging unused power from a particular server. With this setting, the server is allocated the maximum amount of power possible for that type of server.

Note You must include the power control policy in a service profile and that service profile must be associated with a server for it to take effect.

Creating a Power Control 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 # **create power-control-policy** *power-control-pol-name*	Creates a power control policy and enters power control policy mode.
Step 3	UCS-A /org/power-control-policy # **set priority** {*priority-num* | **no-cap**}	Specifies the priority for the power control policy.
Step 4	UCS-A /org/power-control-policy # **commit-buffer**	Commits the transaction to the system configuration.

The following example creates a power control policy called powerpolicy15, sets the priority at level 2, and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # create power-control-policy powerpolicy15
UCS-A /org/power-control policy* # set priority 2
UCS-A /org/power-control policy* # commit-buffer
UCS-A /org/power-control policy #
```

What to Do Next

Include the power control policy in a service profile.

Deleting a Power Control 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 # **delete power-control-policy** *power-control-pol-name*	Deletes the specified power control policy.

	Command or Action	Purpose
Step 3	UCS-A /org # **commit-buffer**	Commits the transaction to the system configuration.

The following example deletes a power control policy called powerpolicy15 and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # delete power-control-policy powerpolicy15
UCS-A /org* # commit-buffer
UCS-A /org #
```

Power Groups in UCS Manager

A power group is a set of chassis that all draw power from the same power distribution unit (PDU). In Cisco UCS Manager, you can create power groups that include one or more chassis, then set a peak power cap in AC watts for that power grouping.

Implementing power capping at the chassis level requires the following:

- IOM, CIMC, and BIOS version 1.4 or higher
- Two Power Supply Units (PSUs)

The peak power cap is a static value that represents the maximum power available to all blade servers within a given power group. If you add or remove a blade from a power group, but do not manually modify the peak power value, the power group adjusts the peak power cap to accommodate the basic power-on requirements of all blades within that power group.

A minimum of 890 AC watts should be set for each chassis. This converts to 800 watts of DC power, which is the minimum amount of power required to power an empty chassis. To associate a half-width blade, the group cap needs to be set to 1475 AC watts. For a full-width blade, it needs to be set to 2060 AC watts.

After a chassis is added to a power group, all service profile associated with the blades in the chassis become part of that power group. Similarly, if you add a new blade to a chassis, that blade inherently becomes part of the chassis' power group.

Note Creating a power group is not the same as creating a server pool. However, you can populate a server pool with members of the same power group by creating a power qualifier and adding it to server pool policy.

When a chassis is removed or deleted, the chassis gets removed from the power group.

UCS Manager supports explicit and implicit power groups.

- **Explicit**: You can create a power group, add chassis' and racks, and assign a budget for the group.
- **Implicit**: Ensures that the chassis is always protected by limiting the power consumption within safe limits. By default, all chassis that are not part of an explicit power group are assigned to the default group and the appropriate caps are placed. New chassis that connect to UCS Manager are added to the default power group until you move them to a different power group.

The following table describes the error messages you might encounter while assigning power budget and working with power groups.

Error Message	Cause	Recommended Action

`Insufficient budget for power group POWERGROUP_NAME`
and/or
`Chassis N cannot be capped as group cap is low. Please consider raising the cap.`
and/or
`Admin committed insufficient for power group GROUP_NAME, using previous value N`
and/or
`Power cap application failed for chassis N`
	One of these messages displays if you did not meet the minimum limit when assigning the power cap for a chassis, or the power requirement increased because of the addition of blades or change of power policies.	Increase the power cap limit to the **Minimum Power Cap for Allowing Operations (W)** value displayed on the **Power Group** page for the specified power group.

`Chassis N cannot be capped as the available PSU power is not enough for the chassis and the blades. Please correct the problem by checking input power or replace the PSU`	Displays when the power budget requirement for the chassis is more than the PSU power that is available.	Check the PSU input power and redundancy policy to ensure that enough power is available for the chassis. If a PSU failed, replace the PSU.
`Power cap application failed for server N`	Displays when the server is consuming more power than allocated and cannot be capped, or the server is powered on when no power is allocated.	Do not power on un-associated servers.
`P-State lowered as consumption hit power cap for server`	Displays when the server is capped to reduce the power consumption below the allocated power.	This is an information message. If a server should not be capped, in the service profile set the value of the power control policy **Power Capping** field to **no-cap**.
`Chassis N has a mix of high-line and low-line PSU input power sources.`	This fault is raised when a chassis has a mix of high-line and low-line PSU input sources connected.	This is an unsupported configuration. All PSUs must be connected to similar power sources.

Creating a Power Group

Before You Begin

Ensure that the global power allocation policy is set to Policy Driven Chassis Group Cap.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope power-cap-mgmt**	Enters power cap management mode.
Step 2	UCS-A /power-cap-mgmt # **create power-group** *power-group-name*	Creates a power group and enters power group mode.
Step 3	UCS-A /power-cap-mgmt/power-group # **set peak** {*peak-num* | **disabled** | **uninitialized**}	Specifies the maximum peak power (in watts) available to the power group.
Step 4	UCS-A /power-cap-mgmt/power-group # **create chassis** *chassis-id*	Adds the specified chassis to the power group and enters power group chassis mode.
Step 5	UCS-A /power-cap-mgmt/power-group # **create rack** *rack-id*	Adds the specified rack to the power group.
Step 6	UCS-A /power-cap-mgmt/power-group # **create fex** *fex-id*	Adds the specified FEX to the power group.
Step 7	UCS-A /power-cap-mgmt/power-group # **create fi** *fi-id*	Adds the specified FI to the power group.
Step 8	UCS-A /power-cap-mgmt/power-group/chassis # **commit-buffer**	Commits the transaction to the system configuration.

The following example creates a power group called powergroup1, specifies the maximum peak power for the power group (10000 watts), adds chassis 1 to the group, and commits the transaction:

```
UCS-A# scope power-cap-mgmt
UCS-A /power-cap-mgmt # create power-group powergroup1
UCS-A /power-cap-mgmt/power-group* # set peak 10000
UCS-A /power-cap-mgmt/power-group* # create chassis 1
UCS-A /power-cap-mgmt/power-group/chassis* # commit-buffer
UCS-A /power-cap-mgmt/power-group/chassis #
```

Deleting a Power Group

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope power-cap-mgmt**	Enters power cap management mode.
Step 2	UCS-A /power-cap-mgmt # **delete power-group** *power-group-name*	Deletes the specified power group.

	Command or Action	Purpose
Step 3	UCS-A /power-cap-mgmt/power-group/chassis # **commit-buffer**	Commits the transaction to the system configuration.

The following example deletes a power group called powergroup1 and commits the transaction:

```
UCS-A# scope power-cap-mgmt
UCS-A /power-cap-mgmt # delete power-group powergroup1
UCS-A /power-cap-mgmt* # commit-buffer
UCS-A /power-cap-mgmt #
```

Blade Level Power Capping

Manual Blade Level Power Cap

When manual blade-level power cap is configured in the global cap policy, you can set a power cap for each blade server in a Cisco UCS domain.

The following configuration options are available:

- **Watts—**You can specify the maximum amount of power that the server can consume at one time. This maximum can be any amount between 0 watts and 1100 watts.
- **Unbounded—**No power usage limitations are imposed on the server. The server can use as much power as it requires.

If the server encounters a spike in power usage that meets or exceeds the maximum configured for the server, Cisco UCS Manager does not disconnect or shut down the server. Instead, Cisco UCS Manager reduces the power that is made available to the server. This reduction can slow down the server, including a reduction in CPU speed.

Note If you configure the manual blade-level power cap using **Equipment** > **Policies** > **Global Policies** > **Global Power Allocation Policy**, the priority set in the Power Control Policy is no longer relevant.

Setting the Blade-Level Power Cap for a Server

Before You Begin

Ensure that the global power allocation policy is set to Manual Blade Level Cap.

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 # **set power-budget committed** {**unbounded** | *watts*}	Commits the server to one of the following power usage levels: • **unbounded —**Does not impose any power usage limitations on the server. • *watts* **—**Allows you to specify the upper level for power usage by the server. If you choose this setting, enter the maximum number of watts that the server can use. The range is 0 to 10000000 watts.
Step 3	UCS-A /chassis/server # **commit-buffer**	Commits the transaction to the system configuration.
Step 4	UCS-A /chassis/server # **show power-budget**	(Optional) Displays the power usage level setting.

The following example limits the power usage for a server to unbounded and then to 1000 watts and commits the transaction:

```
UCS-A# scope server 1/7
UCS-A /chassis/server # show power-budget

Budget:
    AdminCommitted (W)
    ------------------
    139
UCS-A /chassis/server # set power-budget committed unbounded
UCS-A /chassis/server* # commit-buffer
UCS-A /chassis/server # show power-budget

Budget:
    AdminCommitted (W)
    ------------------
    Unbounded

UCS-A /chassis/server # set power-budget committed 1000
UCS-A /chassis/server* # commit-buffer
UCS-A /chassis/server # show power-budget

Budget:
    AdminCommitted (W)
    ------------------
    1000
UCS-A /chassis/server #
```

Viewing Server Statistics

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 stats**	Displays the following server statistics: • Ethernet Port Error • Ethernet Port Multicast • Ethernet Port • Virtual Interface • Motherboard Power • PC Ie Fatal Completion Error • PC Ie Fatal Protocol Error • PC Ie Fatal Receiving Error • PC Ie Fatal Error • Memory Error • DIMM Env • CPU Env

The following example shows the section on motherboard power usage statistics:

```
UCS-A# scope server 2/4
UCS-A /chassis/server # show stats

Motherboard Power Statistics:
Time Collected: 2016-07-11T20:51:24.722
Monitored Object: sys/chassis-1/blade-1/board/power-stats
Suspect: No
Consumed Power (W): 126.000000
Input Voltage (V): 11.859000
Input Current (A): 10.624842
Thresholded: 0

UCS-A /chassis/server #
```

Global Power Profiling Policy Configuration

Global Power Profiling Policy

The Global Power Profiling Policy specifies how power allocation is applied to all of the servers in a chassis. The policy applies when you set the Global Power Allocation Policy to **policy-driven-chassis-group-cap**. You can set the Global Power Profiling Policy to one of the following:

- **Disabled—**The minimum and maximum power cap values of the blades are calculated based on the static power consumption values of each of the components.
- **Enabled—**The minimum and maximum power cap values of the blades are measured as part of the server discovery. These values are similar to the actual power consumption of the blades.

Note After enabling the Global Power Profiling Policy, you must re-acknowledge the blades to obtain the minimum and maximum power cap.

Important Power profiling is not supported in Cisco UCS B460 M4 blades.

Configuring the Global Power Profile Policy

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope power-cap-mgmt**	Enters power cap management mode.
Step 2	UCS-A /power-cap-mgmt # **set profile-policy** {**no** | **yes**}	Enables or disables the global power profiling policy.
Step 3	UCS-A /power-cap-mgmt # **commit-buffer**	Commits the transaction to the system configuration.

The following example shows how to enable the global power profile policy and commit the transaction:

```
UCS-A# scope power-cap-mgmt
UCS-A /power-cap-mgmt # set profile-policy yes
UCS-A /power-cap-mgmt* # commit-buffer
UCS-A /power-cap-mgmt #
```

Global Power Allocation Policy

The Global Power Allocation Policy allows you to specify the Policy Driven Chassis Group Power Cap or Manual Blade-level Power Cap power allocation method applied to servers in a chassis.

Cisco recommends using the default Policy Driven Chassis Group Power Cap power allocation method.

Important Any change to the Manual Blade level Power Cap configuration results in the loss of any groups or configuration options set for the Policy Driven Chassis Group Power Cap.

Configuring the Global Power Allocation Policy

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope power-cap-mgmt**	Enters power cap management mode.
Step 2	UCS-A /power-cap-mgmt # **set cap-policy** {**manual-blade-level-cap** | **policy-driven-chassis-group-cap**}	Sets the global cap policy to the specified power cap management mode. By default, the global cap policy is set to policy driven chassis group cap.
Step 3	UCS-A /power-cap-mgmt # **commit-buffer**	Commits the transaction to the system configuration.

The following example sets the global cap policy to manual blade power cap and commits the transaction:

```
UCS-A# scope power-cap-mgmt
UCS-A /power-cap-mgmt # set cap-policy manual-blade-level-cap
UCS-A /power-cap-mgmt* # commit-buffer
UCS-A /power-cap-mgmt #
```

Viewing the Power Cap Values for Servers

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope power-cap-mgmt**	Enters power cap management mode.
Step 2	UCS-A /power-cap-mgmt # **show power-measured**	Displays the minimum and maximum power cap values.

The following example shows how to display the minimum and maximum power cap values:

```
UCS-A# scope power-cap-mgmt
UCS-A /power-cap-mgmt # show power-measured

Measured Power:
    Device Id (W)  Minimum power (W) Maximum power (W) OperMethod
    -------------- ----------------- ----------------- ----------
    blade   1/1    234               353               Pnuos

UCS-A /power-cap-mgmt #
```

Power Management During Power-on Operations

Boot Staggering during Power on

Cisco UCS Manager attempts to boot as many blades as possible based on the amount of available power. If the power required to boot a blade is not available, Cisco UCS Manager staggers the boot in the Finite State Machine (FSM) CheckPowerAvailability stage, and raises the following fault on the blade: Insufficient power available to power-on server x/y.

When the required power becomes available, the FSM proceeds with blade power on. After a blade powers off, the allocated power budget is reclaimed.

Note When the power budget that was allocated to the blade is reclaimed, the allocated power displays as 0 Watts.

Limitation

If you power on a blade outside of the Cisco UCS Manager, it does not allocate any power budget to the blade, and raises the following fault: Power cap application failed for server x/y.

Power Allocation during Service Profile Association

The power allocated to a blade during service profile association depends on the Power Control Policy used, and the power that is available from the power group. After the power is allocated to a server during a successful service profile association, the blade is guaranteed the minimum power cap. If the Power Control Policy priority is set to no-cap, a blade is allocated a potential maximum power cap, which might exceed the measured maximum power cap that displays.

Note If the priority of an associated blade is changed to no-cap, and is not able to allocate the maximum power cap, you might see one of the following faults:

- `PSU-insufficient`—There is not enough available power for the PSU.
- `Group-cap-insufficient`—The group cap value is not sufficient for the blade.

Power Sync Policy

Cisco UCS Manager includes a global (default) power sync policy to address power synchronization issues between the associated service profiles and the servers. You can use the power sync policy to synchronize the power state when the power state of the service profile differs from the actual power state of the server. The policy allows you to control when to synchronize the power state on the associated service profiles for the servers. The power sync policy does not affect other power-related policies.

The power synchronization policy applies to all the service profiles by default. You cannot delete the default power sync policy, but you can edit the default policy. You can create your own power sync policies and apply them to the service profiles. You can also create a power sync policy that is specific to a service profile and it always takes precedence over the default policy.

Cisco UCS Manager creates a fault on the associated service profile when the power sync policy referenced in the service profile does not exist. Cisco UCS Manager automatically clears the fault once you create a power sync policy for the specified service profile or change the reference to an existing policy in the service profile.

Power Synchronization Behavior

Cisco UCS Manager synchronizes the power state only when the actual power state of the server is OFF. The current power synchronization behavior is based on the actual power state and the preferred power state after shallow association occurs.
For example, the following events trigger shallow association:

- Fabric Interconnects(FI) and IOM disconnected.
- IOM reset
- FI power loss or reboot
- Chassis reacknowledgment
- Chassis power loss
- Service profile change

The following table describes the current power synchronization behavior:

Event	Preferred Power State	Actual Power State Before Event	Actual Power State After Event
Shallow Association	ON	OFF	ON
Shallow Association	OFF	OFF	OFF
Shallow Association	ON	ON	ON
Shallow Association	OFF	ON	ON

Displaying the Global Power Sync 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 power-sync-policy default**	Enters the global power sync policy mode.
Step 3	UCS-A /org/power/-sync-policy # **show** {**detail** | **expand** | **detail expand** }	Displays the global power sync policy information.

The following example displays the global (default) power sync policy:

```
UCS-A # scope org
UCS-A /org # scope power-sync-policy default-sync
UCS-A /org/power-sync-policy # show expand

Power Sync Policy:
    Name                Power Sync Option
    ------------------- -----------------
    default             Default Sync

UCS-A /org/power-sync-policy # show detail expand

Power Sync Policy:
    Full Name: org-root/power-sync-default
    Name: default
    Description:
    Power Sync Option: Default Sync
    Policy Owner: Local

UCS-A /org/power-sync-policy #
```

Setting Global Policy Reference for a Service Profile

To refer the global power sync policy in a service profile, use the following commands in service profile mode:

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.

	Command or Action	Purpose
Step 2	UCS-A/org # **scope service-profile** *service-profile-name*	Enters the service profile mode for the specified service profile. The name of the service profile can be a minimum of two characters and a maximum up to 32 characters.
Step 3	UCS-A /org/service-profile # **set power-sync-policy default**	Specifies the global power sync policy that can be referenced in the service profile. You can also change the policy reference from the default to other power sync policies using this command.
Step 4	UCS-A /org/service-profile* # **commit-buffer**	Commits the transaction to the system configuration.

The following example sets the reference to the global power sync policy for use in the service profile.

```
UCS-A # scope org
    UCS-A/org # scope service-profile spnew
    UCS-A/org/service-profile # set power-sync-policy default
    UCS-A/org/service-profile* # commit-buffer
```

Creating a Power Sync 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 # **create power-sync-policy** *power-sync-pol-name*	Creates a power sync policy and enters power sync policy mode. The power sync policy name can be up to 16 characters.
Step 3	UCS-A /org/power-sync-policy* # **set descr** *optionall-description*	(Optional) Specifies the description of the power-sync-policy. You can also modify the description using the descr keyword.
Step 4	UCS-A /org/power-sync-policy* # **set sync-option** { **always-sync** | **default-sync** | **initial-only-sync** }	Specifies the power synchronization option to the physical server. You can also modify the power synchronization option using the sync-option keyword. This can be one of the following: • **Default Sync—**After the initial server association, any configuration change or management connectivity changes that you perform trigger a server reassociation. This option synchronizes the desired power state to the physical server if the physical server power state is off and the desired power state is on. This is the default behavior. • **Always Sync—**When the initial server association or the server reassociation occurs, this option always synchronizes

	Command or Action	Purpose
		the desired power state to the physical server even if the physical server power state is on and the desired power state is off. • **Initial Only Sync—**This option only synchronizes the power to a server when a service profile is associated to the server for the first time or when the server is re-commissioned. When you set this option, resetting the power state from the physical server side does not affect the desired power state on the service profile.
Step 5	UCS-A /org/power-sync-policy* # **commit-buffer**	Commits the transaction to the system configuration.

The following example creates a power sync policy called newSyncPolicy, sets the default sync-option, and commits the transaction to the system configuration:

```
UCS-A # scope org
UCS-A /org # create power-sync-policy newSyncPolicy
UCS-A /org/power-sync-policy* # set decsr newSyncPolicy
UCS-A /org/power-sync-policy* # set sync-option default-sync
UCS-A /org/power-sync-policy* # commit-buffer
UCS-A /org/power-sync-policy #
```

What to Do Next

Include the power sync policy in a service profile or in a service profile template.

Deleting a Power Sync 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 # **delete power-sync-policy** *power-sync-pol-name*	Deletes the specified power sync policy.
Step 3	UCS-A /org # **commit buffer**	Commits the transaction to the system configuration.

The following example deletes the power sync policy called spnew and commits the transaction to the system:

```
UCS-A # scope org
UCS-A /org # delete power-sync-policy spnew
UCS-A /org # commit-buffer
```

Displaying All Power Sync Policies

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 # **show power-sync-policy** {**detail** | **expand** | **detail expand** }	Displays the default, local, and other power sync policies.

The following example displays power sync policies that are defined:

```
UCS-A # scope org
UCS-A /org # show power-sync-policy expand
Power Sync Policy:
    Name                 Power Sync Option
    -------------------- -----------------
    default              Default Sync
    policy-1             Default Sync

UCS-A /org # show power-sync-policy detail expand
Power Sync Policy:
    Full Name: org-root/power-sync-default
    Name: default
    Description:
    Power Sync Option: Default Sync
    Policy Owner: Local

    Full Name: org-root/power-sync-policy-1
    Name: policy-1
    Description:
    Power Sync Option: Default Sync
    Policy Owner: Local

UCS-A /org #
```

Creating a Local Policy

To create a local power sync policy that you want to use by any service profile, create a power sync definition for the power sync 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 service-profile** *service-profile-name*	Enters the service profile mode for the specified service profile. The name of the service profile can be a minimum of two characters and a maximum up to 32 characters.
Step 3	UCS-A /org/service-profile # **create power-sync-definition**	Enters the power sync definition mode. You can create a power sync policy definition that you defined for the power sync policy.
Step 4	UCS-A /org/service-profile/power-sync-definition* # **set descr** *optional-description*	(Optional) Specifies the description of the power-sync-policy. You can also change the description using the descr keyword.
Step 5	UCS-A /org/service-profile/power-sync-definition* # **set sync-option** { **always-sync** | **default-sync** | **initial-only-sync** }	Specifies the power synchronization option to the physical server. You can also change the power synchronization option using the sync-option keyword.
Step 6	UCS-A /org/service-profile/power-sync-definition* # **commit-buffer**	Commits the transaction to the system configuration.

The following example creates a local policy using the policy sync definition, sets the sync-option, and commits the transaction to the system configuration:

```
 UCS-A # scope org
 UCS-A/org # scope service-profile spnew
UCS-A/org/service-profile # create power-sync-definition
UCS-A/org/service-profile/power-sync-definition* # set decsr spnew
UCS-A/org/service-profile/power-sync-definition* # set sync-option default-sync
UCS-A/org/service-profile/power-sync-definition* #  commit-buffer
```

Showing a Local 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.

	Command or Action	Purpose
Step 2	UCS-A/org # **scope service-profile** *service-profile-name*	Enters the service profile mode for the specified service profile. The name of the service profile can be a minimum of two characters and a maximum up to 32 characters.
Step 3	UCS-A /org/service-profile # **show power-sync-policy** {**detail** | **expand** | **detail expand** }	(Optional) Displays the local policy in the power-sync-policy mode.
Step 4	UCS-A /org/service-profile # **show power-sync-definition** {**detail** | **expand** | **detail expand** }	Displays the local policy for the specified service policy in the power-sync-definition mode. **Note** If you do not have a definition for the power sync policy, you can still use the command, but you cannot see anything displayed.

The following example displays the local policy in use by the service profile spnew:

```
 UCS-A # scope org
UCS-A/org # scope service-profile spnew
UCS-A/org/service-profile # show power-sync-definition expand

Power Sync Definition:
    Name                 Power Sync Option
    -------------------- -----------------
    spnew                Always Sync

UCS-A/org/service-profile # show power-sync-definition detail expand

Power Sync Definition:
    Full Name: org-root/ls-sp2/power-sync-def
    Name: spnew
    Description: optional description
    Power Sync Option: Always Sync
    Policy Owner: Local

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

Deleting a Local 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 service-profile** *service-profile-name*	Enters the service profile mode for the specified service profile. The name of the service profile can be a minimum of two characters and a maximum up to 32 characters.

	Command or Action	Purpose
Step 3	UCS-A /org/service-profile # **delete power-sync-definition**	Enters the power sync definition mode. You can delete a power sync policy definition that you defined for the power sync policy.
Step 4	UCS-A /org/service-profile* # **commit-buffer**	Commits the transaction to the system configuration.

The following example deletes the local policy in use by the service profile.

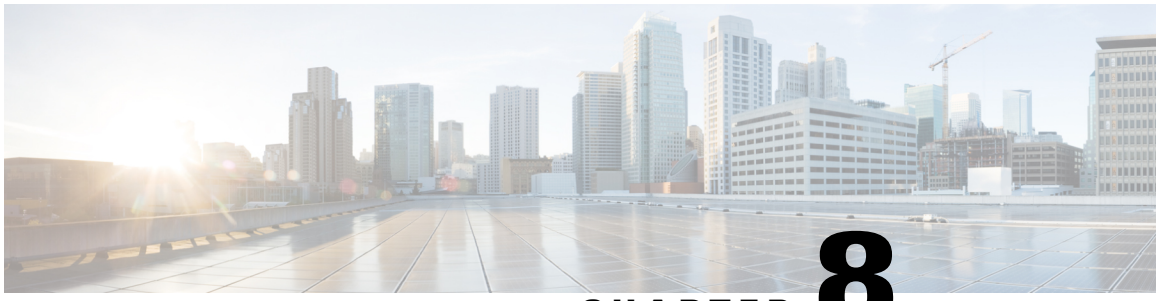
```
UCS-A # scope org
    UCS-A/org # scope service-profile spnew
    UCS-A/org/service-profile # delete power-sync-definition
    UCS-A/org/service-profile* # commit-buffer
```

Rack Server Power Management

Power capping is not supported for rack servers.

UCS Mini Power Management

You can manage power of the blade servers in 6324 Fabric Interconnect (FI), which is used for remote offices and branch sites, and for limited server deployments. UCS Manager supports Dual Line Power Supply Unit and 110V when used with the 6324 Fabric Interconnect. You can manager how you want to allocate power when using 110V power supplies, because they might not provide enough power for a fully loaded chassis. Dual power supplies is standard for both AC and DC-48V on the UCS Mini 6324.



CHAPTER 8

Blade Server Hardware Management

Blade Server Management

You can manage and monitor all blade servers in a Cisco UCS domain through Cisco UCS Manager. You can perform some blade server management tasks, such as changes to the power state, from the server and service profile.

The remaining management tasks can only be performed on the server.

The power supply units go into power save mode when a chassis has two blades or less. When a third blade is added to the chassis and is fully discovered, the power supply units return to regular mode.

If a blade server slot in a chassis is empty, Cisco UCS Manager provides information, errors, and faults for that slot. You can also re-acknowledge the slot to resolve server mismatch errors and to have Cisco UCS Manager rediscover the blade server in the slot.

Guidelines for Removing and Decommissioning Blade Servers

Consider the following guidelines when deciding whether to remove or decommission a blade server using Cisco UCS Manager:

Decommissioning a Blade Server

If you want to temporarily decommission a physically present and connected blade server, you can temporarily remove it from the configuration. A portion of the server's information is retained by Cisco UCS Manager for future use, in case the blade server is recommissioned.

Removing a Blade Server

Removing is performed when you physically remove a blade server from the Cisco UCS Manager by disconnecting it from the chassis. You cannot remove a blade server from Cisco UCS Manager if it is physically present and connected to a chassis. After the physical removal of the blade server is completed, the configuration for that blade server can be removed in Cisco UCS Manager.

During removal, active links to the blade server are disabled, all entries from databases are removed, and the server is automatically removed from any server pools that it was assigned to during discovery.

Note Only servers added to a server pool automatically during discovery are removed automatically. Servers that were manually added to a server pool must be removed manually.

To add a removed blade server back to the configuration, it must be reconnected, then rediscovered. When a server is reintroduced to Cisco UCS Manager, it is treated as a new server and is subject to the deep discovery process. For this reason, it is possible for Cisco UCS Manager to assign the server a new ID that might be different from the ID that it held before.

Recommendations for Avoiding Unexpected Server Power Changes

If a server is not associated with a service profile, you can use any available means to change the server power state, including the physical Power or Reset buttons on the server.

If a server is associated with, or assigned to, a service profile, you should only use the following methods to change the server power state:

- In Cisco UCS Manager GUI, go to the **General** tab for the server or the service profile associated with the server and select **Boot Server** or **Shutdown Server** from the **Actions** area.

- In Cisco UCS Manager CLI, scope to the server or the service profile associated with the server and use the **power up** or **power down** commands.

Important Do *not* use any of the following options on an associated server that is currently powered off:

- **Reset** in the GUI
- **cycle cycle-immediate** or **reset hard-reset-immediate** in the CLI
- The physical Power or Reset buttons on the server

If you reset, cycle, or use the physical power buttons on a server that is currently powered off, the server's actual power state might become out of sync with the desired power state setting in the service profile. If the communication between the server and Cisco UCS Manager is disrupted or if the service profile configuration changes, Cisco UCS Manager might apply the desired power state from the service profile to the server, causing an unexpected power change.

Power synchronization issues can lead to an unexpected server restart, as shown below:

Desired Power State in Service Profile	Current Server Power State	Server Power State After Communication Is Disrupted
Up	Powered Off	Powered On
Down	Powered On	Powered On **Note** Running servers are not shut down regardless of the desired power state in the service profile.

Booting a Blade Server

Before You Begin

Associate a service profile with a blade server or server pool.

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 service-profile** *profile-name*	Enters organization service profile mode for the specified service profile.

	Command or Action	Purpose
Step 3	UCS-A /org/service-profile # **power up**	Boots the blade server associated with the service profile.
Step 4	UCS-A /org/service-profile # **commit-buffer**	Commits the transaction to the system configuration.

The following example boots the blade server associated with the service profile named ServProf34 and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # scope service-profile ServProf34
UCS-A /org/service-profile* # power up
UCS-A /org/service-profile* # commit-buffer
UCS-A /org/service-profile #
```

Shutting Down a Blade Server

When you use this procedure to shut down a server with an installed operating system, Cisco UCS Manager triggers the OS into a graceful shutdown sequence.

Note When a blade server that is associated with a service profile is shut down, the VIF down alerts F0283 and F0479 are automatically suppressed.

Before You Begin

Associate a service profile with a blade server or server pool.

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 service-profile** *profile-name*	Enters organization service profile mode for the specified service profile.
Step 3	UCS-A /org/service-profile # **power down**	Shuts down the blade server associated with the service profile.
Step 4	UCS-A /org/service-profile # **commit-buffer**	Commits the transaction to the system configuration.

The following example shuts down the blade server associated with the service profile named ServProf34 and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # scope service-profile ServProf34
UCS-A /org/service-profile # power down
UCS-A /org/service-profile* # commit-buffer
UCS-A /org/service-profile #
```

Resetting a Blade Server to Factory Default Settings

You can now reset a blade server to its factory settings. By default, the factory reset operation does not affect storage drives and flexflash drives. This is to prevent any loss of data. However, you can choose to reset these devices to a known state as well.

Important Resetting storage devices will result in loss of data.

Perform the following procedure to reset the server to factory default settings.

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 /chassis/server # **reset factory-default** [**delete-flexflash-storage** | **delete-storage** [**create-initial-storage-volumes**]]	Resets server settings to factory default using the following command options: • **factory-default—**Resets the server to factory defaults without deleting storage • **delete-flexflash-storage—**Resets the server to factory defaults and deletes flexflash storage • **delete-storage—**Resets the server to factory defaults and deletes all storage • **create-initial-storage-volumes—**Resets the server to factory defaults, deletes all storage, sets all disks to their initial state **Important** Do not use the **create-initial-storage-volumes** command option if you want to use storage profiles. Creating initial volumes when you are using storage profiles may result in configuration errors.
Step 3	UCS-A /chassis/server* # **commit-buffer**	Commits any pending transactions.

The following example resets the server settings to factory default without deleting storage, and commits the transaction:

```
UCS-A# scope server 2/4
UCS-A /chassis/server # reset factory-default
UCS-A /chassis/server* # commit-buffer
```

The following example resets the server settings to factory default, deletes flexflash storage, and commits the transaction:

```
UCS-A# scope server 2/4
UCS-A /chassis/server # reset factory-default delete-flexflash-storage

UCS-A /chassis/server* # commit-buffer
```

The following example resets the server settings to factory default, deletes all storage, and commits the transaction:

```
UCS-A# scope server 2/4
UCS-A /chassis/server # reset factory-default delete-storage
UCS-A /chassis/server* # commit-buffer
```

The following example resets the server settings to factory default, deletes all storage, sets all disks to their initial state, and commits the transaction:

```
UCS-A# scope server 2/4
UCS-A /chassis/server # reset factory-default delete-storage create-initial-storage-volumes
UCS-A /chassis/server* # commit-buffer
```

Power Cycling 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 blade server.
Step 2	UCS-A /chassis/server # **cycle** {**cycle-immediate** | **cycle-wait**}	Power cycles the blade server. Use the **cycle-immediate** keyword to immediately begin power cycling the blade server; use the **cycle-wait** keyword to schedule the power cycle to begin after all pending management operations have completed.
Step 3	UCS-A# **commit-buffer**	Commits the transaction to the system configuration.

The following example immediately power cycles blade server 4 in chassis 2 and commits the transaction:

```
UCS-A#  scope server 2/4
UCS-A /chassis/server # cycle cycle-immediate
UCS-A /chassis/server* # commit-buffer
UCS-A /chassis/server #
```

Performing a Hard Reset on a Blade Server

When you reset a server, Cisco UCS Manager sends a pulse on the reset line. You can choose to gracefully shut down the operating system. If the operating system does not support a graceful shutdown, the server is power cycled. The option to have Cisco UCS Manager complete all management operations before it resets the server does not guarantee the completion of these operations before the server is reset.

Note If you are trying to boot a server from a power-down state, you should not use **Reset**.

If you continue the power-up with this process, the desired power state of the servers become out of sync with the actual power state and the servers might unexpectedly shut down at a later time. To safely reboot the selected servers from a power-down state, click **Cancel**, then select the **Boot Server** action.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *chassis-num* / *server-num*	Enters chassis server mode for the specified server.
Step 2	UCS-A /chassis/server # **reset** {**hard-reset-immediate** | **hard-reset-wait**}	Performs a hard reset of the blade server. Use the **hard-reset-immediate** keyword to immediately begin hard resetting the server; use the **hard-reset-wait** keyword to schedule the hard reset to begin after all pending management operations have completed.
Step 3	UCS-A /server # **commit-buffer**	Commits the transaction to the system configuration.

The following example performs an immediate hard reset of blade server 4 in chassis 2 and commits the transaction:

```
UCS-A#  scope server 2/4
UCS-A /chassis/server # reset hard-reset-immediate
UCS-A /chassis/server* # commit-buffer
UCS-A /chassis/server #
```

Acknowledging a Blade Server

Perform the following procedure to rediscover the server and all endpoints in the server. For example, you can use this procedure if a server is stuck in an unexpected state, such as the discovery state.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **acknowledge server** *chassis-num* / *server-num*	Acknowledges the specified blade server.

	Command or Action	Purpose
Step 2	UCS-A# **commit-buffer**	Commits the transaction to the system configuration.

The following example acknowledges server 4 in chassis 2 and commits the transaction:

```
UCS-A#  acknowledge server 2/4
UCS-A* # commit-buffer
UCS-A #
```

Removing a Blade Server from a Chassis

Before You Begin

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **remove server** *chassis-num* / *server-num*	Removes the specified blade server.
Step 2	UCS-A# **commit-buffer**	Commits the transaction to the system configuration.
Step 3	Go to the physical location of the chassis and remove the server hardware from the slot.	For instructions on how to remove the server hardware, see the *Cisco UCS Hardware Installation Guide* for your chassis.

The following example removes blade server 4 in chassis 2 and commits the transaction:

```
UCS-A#  remove server 2/4
UCS-A* # commit-buffer
UCS-A #
```

What to Do Next

If you physically re-install the blade server, you must re-acknowledge the slot for the Cisco UCS Manager to rediscover the server.

For more information, see Acknowledging a Blade Server, on page 73.

Decommissioning a Blade Server

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **decommission server** *chassis-num* / *server-num*	Decommissions the specified blade server.
Step 2	UCS-A# **commit-buffer**	Commits the transaction to the system configuration.

The following example decommissions blade server 4 in chassis 2 and commits the transaction:

```
UCS-A# decommission server 2/4
UCS-A* # commit-buffer
UCS-A #
```

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. For the Cisco UCS B460 M4 blade server, you can add the following keywords: • **multi-master—**Turns on the LED for the master node only. • **multi-slave—**Turns on the LED for the slave node only.
Step 3	UCS-A /chassis/server # **commit-buffer**	Commits the transaction to the system configuration.

The following example turns on the locator LED on 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 #
```

The following example turns on the locator LED for the master node only on blade server 7 in chassis 2 and commits the transaction:

```
UCS-A# scope chassis 2/7
UCS-A /chassis/server # enable locator-led multi-master
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. For the Cisco UCS B460 M4 blade server, you can add the following keywords: • **multi-master—**Turns off the LED for the master node only. • **multi-slave—**Turns off the LED for the slave node only.
Step 3	UCS-A /chassis/server # **commit-buffer**	Commits the transaction to the system configuration.

The following example turns off the locator LED on 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 #
```

The following example turns off the locator LED for the master node on blade server 7 in chassis 2 and commits the transaction:

```
UCS-A# scope chassis 2/7
UCS-A /chassis/server # disable locator-led multi-master
UCS-A /chassis/server* # commit-buffer
UCS-A /chassis/server #
```

Resetting the CMOS for a Blade Server

Sometimes, troubleshooting a server might require you to reset the CMOS. Resetting the CMOS is not part of the normal maintenance of a 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 # **reset-cmos**	Resets the CMOS for the blade server.
Step 3	UCS-A /chassis/server # **commit-buffer**	Commits the transaction to the system configuration.

The following example resets the CMOS for blade server 4 in chassis 2 and commits the transaction:

```
UCS-A# scope server 2/4
UCS-A /chassis/server # reset-cmos
UCS-A /chassis/server* # commit-buffer
UCS-A /chassis/server #
```

Resetting the CIMC for a Blade Server

Sometimes, with the firmware, troubleshooting a server might require you to reset the CIMC. Resetting the CIMC is not part of the normal maintenance of a server. After you reset the CIMC, the server boots with the running version of the firmware for that server.

If the CIMC is reset, the power monitoring functions of Cisco UCS become briefly unavailable until the CIMC reboots. Typically, the reset only takes 20 seconds; however, it is possible that the peak power cap can exceed during that time. To avoid exceeding the configured power cap in a low power-capped environment, consider staggering the rebooting or activation of CIMCs.

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 # **scope CIMC**	Enters chassis server CIMC mode
Step 3	UCS-A /chassis/server/CIMC # **reset**	Resets the CIMC for the blade server.
Step 4	UCS-A /chassis/server/CIMC # **commit-buffer**	Commits the transaction to the system configuration.

The following example resets the CIMC for blade server 4 in chassis 2 and commits the transaction:

```
UCS-A# scope server 2/4
UCS-A /chassis/server # scope CIMC
UCS-A /chassis/server/cimc # reset
UCS-A /chassis/server/cimc* # commit-buffer
UCS-A /chassis/server/cimc #
```

Clearing TPM for a Blade Server

You can clear TPM only on Cisco UCS M4 blade and rack-mount servers that include support for TPM.

Caution Clearing TPM is a potentially hazardous operation. The OS may stop booting. You may also see loss of data.

Before You Begin

TPM must be enabled.

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# /chassis/server # **scope tpm** *tpm-ID*	Enters org TPM mode for the specified TPM.
Step 3	UCS-A# /chassis/server/tpm # **set adminaction clear-config**	Specifies that the TPM is to be cleared.
Step 4	UCS-A# /chassis/server/tpm # **commit-buffer**	Commits the transaction to the system configuration.

The following example shows how to clear TPM for a blade server:

```
UCS-A# scope server 1/3
UCS-A# /chassis/server # scope tpm 1
UCS-A# /chassis/server/tpm # set adminaction clear-config
UCS-A#/chassis/server/tpm* # commit-buffer
```

Issuing an NMI from a Blade Server

Perform the following procedure if the system remains unresponsive and you need Cisco UCS Manager to issue a Non-Maskable Interrupt (NMI) to the BIOS or operating system from the CIMC. This action creates a core dump or stack trace, depending on the operating system installed on the 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 /chassis/server # **diagnostic-interrupt**	
Step 3	UCS-A /chassis/server* # **commit-buffer**	Commits any pending transactions.

The following example sends an NMI from server 4 in chassis 2 and commits the transaction:

```
UCS-A# scope server 2/4
UCS-A /chassis/server # diagnostic-interrupt
UCS-A /chassis/server* # commit-buffer
UCS-A /chassis/server #
```

Health LED Alarms

The blade health LED is located on the front of each Cisco UCS B-Series blade server. Cisco UCS Manager allows you to view the sensor faults that cause the blade health LED to change color from green to amber or blinking amber.

The health LED alarms display the following information:

Name	Description
Severity column	The severity of the alarm. This can be one of the following: • Critical—The blade health LED is blinking amber. • Minor—The blade health LED is amber.
Description column	A brief description of the alarm.
Sensor ID column	The ID of the sensor the triggered the alarm.
Sensor Name column	The name of the sensor that triggered the alarm.

Viewing Health LED 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 health-led expand**	Displays the health LED and sensor alarms for the selected server.

The following example shows how to display the health LED status and sensor alarms for chassis 1 server 3:

```
UCS-A# scope server 1/3
UCS-A /chassis/server # show health-led expand
Health LED:
    Severity: Normal
    Reason:
    Color: Green
    Oper State: On

UCS-A /chassis/server #
```

Smart SSD

Beginning with release 3.1(3), Cisco UCS Manager supports monitoring SSD health. This feature is called Smart SSD. It provides statistical information about the properties like wear status in days, percentage life remaining, and so on. For every property, a minimum, a maximum and an average value is recorded and displayed. The feature also allows you to provide threshold limit for the properties.

Note The Smart SSD feature is supported only for a selected range of SSDs. It is not supported for any HDDs.

The SATA range of supported SSDs are:

- Intel
- Samsung
- Micron

The SAS range of supported SSDs are:

- Toshiba
- Sandisk
- Samsung
- Micron

Note

- Power-On Hours and Power Cycle Count are not available on SAS SSDs.
- Smart SSD feature is supported only on M4 servers and later.

Viewing SSD Health Statistics

Perform this procedure to view the SSD Health statistics.

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 stats**	Displays the SSD health statistics for the specified server.

The following example displays the SSD health statistics for blade 3 in chassis 1:

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

Ssd Health Stats:
    Time Collected: 2016-12-07T19:35:15.920
    Monitored Object: sys/chassis-1/blade-3/board/storage-SAS-1/ssd-health-stats-1
    Suspect: No
    Id: 1
    Power Cycle Count: 1022
    Power On Hours: 4793
    Percentage Life Left: 92
    Wear Status In Days: 1679
    Thresholded: 0

    Time Collected: 2016-12-07T19:35:38.912
    Monitored Object: sys/chassis-1/blade-3/board/storage-SAS-1/ssd-health-stats-2
    Suspect: No
    Id: 2
    Power Cycle Count: 1017
    Power On Hours: 4270
    Percentage Life Left: 87
    Wear Status In Days: 1587
    Thresholded: 0

    Time Collected: 2016-12-07T19:35:15.920
    Monitored Object: sys/chassis-1/blade-3/board/storage-SAS-4/ssd-health-stats-1
    Suspect: No
    Id: 1
    Power Cycle Count: 1506
    Power On Hours: 5029
    Percentage Life Left: 98
    Wear Status In Days: 1788
    Thresholded: 0

    Time Collected: 2016-12-07T19:35:15.920
    Monitored Object: sys/chassis-1/blade-3/board/storage-SAS-4/ssd-health-stats-2
    Suspect: No
```

```
    Id: 2
    Power Cycle Count: 58
    Power On Hours: 4731
    Percentage Life Left: 100
    Wear Status In Days: 1825
    Thresholded: 0
UCS-A /chassis/server #
```

CHAPTER 9

Rack-Mount Server Hardware Management

Rack-Mount Server Management

You can manage and monitor all rack-mount servers that are integrated with a Cisco UCS domain through Cisco UCS Manager. All management and monitoring features are supported for rack-mount servers except power capping. Some rack-mount server management tasks, such as changes to the power state, can be performed from both the server and service profile. The remaining management tasks can only be performed on the server.

Cisco UCS Manager provides information, errors, and faults for each rack-mount server that it has discovered.

Tip For information on how to integrate a supported Cisco UCS rack-mount server with Cisco UCS Manager, see the Cisco UCS C-series server integration guide or Cisco UCS S-series server integration guide for your Cisco UCS Manager release.

Guidelines for Removing and Decommissioning Rack-Mount Servers

Consider the following guidelines when deciding whether to remove or decommission a rack-mount server using Cisco UCS Manager:

Decommissioning a Rack-Mount server

Decommissioning is performed when a rack-mount server is physically present and connected but you want to temporarily remove it from the configuration. Because it is expected that a decommissioned rack-mount server will be eventually recommissioned, a portion of the server's information is retained by Cisco UCS Manager for future use.

Removing a Rack-Mount server

Removing is performed when you physically remove the server from the system by disconnecting the rack-mount server from the fabric extender. You cannot remove a rack-mount server from Cisco UCS Manager if it is physically present and connected to the fabric extender. Once the rack-mount server is disconnected, the configuration for that rack-mount server can be removed in Cisco UCS Manager.

During removal, management interfaces are disconnected, all entries from databases are removed, and the server is automatically removed from any server pools that it was assigned to during discovery.

Note Only those servers added to a server pool automatically during discovery will be removed automatically. Servers that have been manually added to a server pool have to be removed manually.

If you need to add a removed rack-mount server back to the configuration, it must be reconnected and then rediscovered. When a server is reintroduced to Cisco UCS Manager it is treated like a new server and is subject to the deep discovery process. For this reason, it's possible that Cisco UCS Manager will assign the server a new ID that may be different from the ID that it held before.

Recommendations for Avoiding Unexpected Server Power Changes

If a server is not associated with a service profile, you can use any available means to change the server power state, including the physical Power or Reset buttons on the server.

If a server is associated with, or assigned to, a service profile, you should only use the following methods to change the server power state:

- In Cisco UCS Manager GUI, go to the **General** tab for the server or the service profile associated with the server and select **Boot Server** or **Shutdown Server** from the **Actions** area.
- In Cisco UCS Manager CLI, scope to the server or the service profile associated with the server and use the **power up** or **power down** commands.

Important Do *not* use any of the following options on an associated server that is currently powered off:

- **Reset** in the GUI
- **cycle cycle-immediate** or **reset hard-reset-immediate** in the CLI
- The physical Power or Reset buttons on the server

If you reset, cycle, or use the physical power buttons on a server that is currently powered off, the server's actual power state might become out of sync with the desired power state setting in the service profile. If the communication between the server and Cisco UCS Manager is disrupted or if the service profile configuration changes, Cisco UCS Manager might apply the desired power state from the service profile to the server, causing an unexpected power change.

Power synchronization issues can lead to an unexpected server restart, as shown below:

Desired Power State in Service Profile	Current Server Power State	Server Power State After Communication Is Disrupted
Up	Powered Off	Powered On
Down	Powered On	Powered On **Note** Running servers are not shut down regardless of the desired power state in the service profile.

Booting a Rack-Mount Server

Before You Begin

Associate a service profile with a rack-mount server.

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 service-profile** *profile-name*	Enters organization service profile mode for the specified service profile.
Step 3	UCS-A /org/service-profile # **power up**	Boots the rack-mount server associated with the service profile.
Step 4	UCS-A /org/service-profile # **commit-buffer**	Commits the transaction to the system configuration.

The following example boots the rack-mount server associated with the service profile named ServProf34 and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # scope service-profile ServProf34
UCS-A /org/service-profile # power up
UCS-A /org/service-profile* # commit-buffer
UCS-A /org/service-profile #
```

Shutting Down a Rack-Mount Server

When you use this procedure to shut down a server with an installed operating system, Cisco UCS Manager triggers the OS into a graceful shutdown sequence.

Before You Begin

Associate a service profile with a rack-mount server.

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 service-profile** *profile-name*	Enters organization service profile mode for the specified service profile.
Step 3	UCS-A /org/service-profile # **power down**	Shuts down the rack-mount server associated with the service profile.
Step 4	UCS-A /org/service-profile # **commit-buffer**	Commits the transaction to the system configuration.

The following example shuts down the rack-mount server associated with the service profile named ServProf34 and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # scope service-profile ServProf34
UCS-A /org/service-profile # power down
UCS-A /org/service-profile* # commit-buffer
UCS-A /org/service-profile #
```

Resetting a Rack-Mount Server to Factory Default Settings

You can now reset a rack-mount server to its factory settings. By default, the factory reset operation does not affect storage, including storage drives and flexflash drives. This is to prevent any loss of data. However, you can choose to reset these devices to a known state as well.

Important Resetting storage devices will result in loss of data.

Perform the following procedure if you need to reset the server to factory default settings.

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 # **reset factory-default** [**delete-flexflash-storage** | **delete-storage** [**create-initial-storage-volumes**]]	Resets server settings to factory default using the following command options: • **factory-default—**Resets the server to factory defaults without deleting storage • **delete-flexflash-storage—**Resets the server to factory defaults and deletes flexflash storage • **delete-storage—**Resets the server to factory defaults and deletes all storage • **create-initial-storage-volumes—**Resets the server to factory defaults, deletes all storage, sets all disks to their initial state **Important** Do not use the **create-initial-storage-volumes** command option if you want to use storage profiles. Creating initial volumes when you are using storage profiles may result in configuration errors.

	Command or Action	Purpose
Step 3	UCS-A /server # **commit-buffer**	Commits the transaction to the system configuration.

The following example resets the server settings to factory default without deleting storage, and commits the transaction:

```
UCS-A# scope server 2
UCS-A /server # reset factory-default
UCS-A /server* # commit-buffer
UCS-A /server #
```

The following example resets the server settings to factory default, deletes flexflash storage, and commits the transaction:

```
UCS-A# scope server 2
UCS-A /server # reset factory-default delete-flexflash-storage
UCS-A /server* # commit-buffer
```

The following example resets the server settings to factory default, deletes all storage, and commits the transaction:

```
UCS-A# scope server 2
UCS-A /server # reset factory-default delete-storage
UCS-A /server* # commit-buffer
```

The following example resets the server settings to factory default, deletes all storage, sets all disks to their initial state, and commits the transaction:

```
UCS-A# scope server 2
UCS-A /server # reset factory-default delete-storage create-initial-storage-volumes
UCS-A /server* # commit-buffer
```

Power Cycling 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 # **cycle** {**cycle-immediate** | **cycle-wait**}	Power cycles the rack-mount server. Use the **cycle-immediate** keyword to immediately begin power cycling the rack-mount server; use the **cycle-wait** keyword to schedule the power cycle to begin after all pending management operations have completed.
Step 3	UCS-A# **commit-buffer**	Commits the transaction to the system configuration.

The following example immediately power cycles rack-mount server 2 and commits the transaction:

```
UCS-A#  scope server 2
UCS-A /server # cycle cycle-immediate
```

```
UCS-A /server* # commit-buffer
UCS-A /server #
```

Performing a Hard Reset on a Rack-Mount Server

When you reset a server, Cisco UCS Manager sends a pulse on the reset line. You can choose to gracefully shut down the operating system. If the operating system does not support a graceful shutdown, the server is power cycled. The option to have Cisco UCS Manager complete all management operations before it resets the server does not guarantee the completion of these operations before the server is reset.

Note If you are trying to boot a server from a power-down state, you should not use **Reset**.

If you continue the power-up with this process, the desired power state of the servers become out of sync with the actual power state and the servers might unexpectedly shut down at a later time. To safely reboot the selected servers from a power-down state, click **Cancel**, then select the **Boot Server** action.

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 # **reset** {**hard-reset-immediate** | **hard-reset-wait**}	Performs a hard reset of the rack-mount server. Use the **hard-reset-immediate** keyword to immediately begin hard resetting the rack-mount server; use the **hard-reset-wait** keyword to schedule the hard reset to begin after all pending management operations have completed.
Step 3	UCS-A /server # **commit-buffer**	Commits the transaction to the system configuration.

The following example performs an immediate hard reset of rack-mount server 2 and commits the transaction:

```
UCS-A#  scope server 2
UCS-A /server # reset hard-reset-immediate
UCS-A /server* # commit-buffer
UCS-A /server #
```

Acknowledging a Rack-Mount Server

Perform the following procedure to rediscover the server and all endpoints in the server. For example, you can use this procedure if a server is stuck in an unexpected state, such as the discovery state.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **acknowledge server** *server-num*	Acknowledges the specified rack-mount server.
Step 2	UCS-A# **commit-buffer**	Commits the transaction to the system configuration.

The following example acknowledges rack-mount server 2 and commits the transaction:

```
UCS-A#  acknowledge server 2
UCS-A* # commit-buffer
UCS-A #
```

Decommissioning a Rack-Mount Server

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **decommission server** *server-num*	Decommissions the specified rack-mount server.
Step 2	UCS-A# **commit-buffer**	Commits the transaction to the system configuration.

The following example decommissions rack-mount server 2 and commits the transaction:

```
UCS-A# decommission server 2
UCS-A* # commit-buffer
UCS-A #
```

Renumbering a Rack-Mount Server

Before You Begin

If you are swapping IDs between servers, you must first decommission both servers, then wait for the server decommission FSM to complete before proceeding with the renumbering steps.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **show server inventory**	Displays information about your servers.
Step 2	Verify that the server inventory does not include the following:	• The rack-mount server you want to renumber • A rack-mount server with the number you want to use

	Command or Action	Purpose
		If either of these rack-mount servers are listed in the server inventory, decommission those servers. You must wait until the decommission FSM is complete and the rack-mount servers are not listed in the server inventory before continuing. This might take several minutes. To see which servers have been decommissioned, issue the **show server decommissioned** command.
Step 3	UCS-A# **recommission server** *vendor-name model-name serial-numnew-id*	Recommissions and renumbers the specified rack-mount server.
Step 4	UCS-A# **commit-buffer**	Commits the transaction to the system configuration.

The following example decommissions a rack-mount server with ID 2, changes the ID to 3, recommissions that server, and commits the transaction:

```
UCS-A# show server inventory

Server  Equipped PID Equipped VID Equipped Serial (SN) Slot Status      Ackd Memory (MB)
Ackd Cores
------- ------------ ------------ -------------------- ---------------- ----------------
----------
1/1     UCSB-B200-M3 V01          FCH1532718P          Equipped         131072
16
1/2     UCSB-B200-M3 V01          FCH153271DF          Equipped         131072
16
1/3     UCSB-B200-M3 V01          FCH153271DL          Equipped         114688
16
1/4     UCSB-B200-M3 V01                               Empty
1/5                                                    Empty
1/6                                                    Empty
1/7     N20-B6730-1  V01          JAF1432CFDH          Equipped         65536
16
1/8                                                    Empty
1       R200-1120402W V01          QCI1414A02J          N/A              49152
12
2       R210-2121605W V01          QCI1442AHFX          N/A              24576            8
4       UCSC-BSE-SFF-C200 V01      QCI1514A0J7          N/A              8192             8

UCS-A# decommission server 2
UCS-A*# commit-buffer
UCS-A# show server decommissioned

Vendor            Model         Serial (SN) Server
----------------- ----------    ----------- ------
Cisco Systems Inc R210-2121605W QCI1442AHFX 2

UCS-A# recommission chassis "Cisco Systems Inc" "R210-2121605W" QCI1442AHFX 3
UCS-A* # commit-buffer
UCS-A # show server inventory

Server  Equipped PID Equipped VID Equipped Serial (SN) Slot Status      Ackd Memory (MB)
Ackd Cores
------- ------------ ------------ -------------------- ---------------- ----------------
----------
1/1     UCSB-B200-M3 V01          FCH1532718P          Equipped         131072
16
1/2     UCSB-B200-M3 V01          FCH153271DF          Equipped         131072
16
1/3     UCSB-B200-M3 V01          FCH153271DL          Equipped         114688
```

```
16
1/4      UCSB-B200-M3 V01                              Empty
1/5                                                    Empty
1/6                                                    Empty
1/7      N20-B6730-1  V01          JAF1432CFDH          Equipped         65536
16
1/8                                                    Empty
1        R200-1120402W V01          QCI1414A02J          N/A              49152
12
3        R210-2121605W V01          QCI1442AHFX          N/A              24576            8
4        UCSC-BSE-SFF-C200 V01      QCI1514A0J7          N/A              8192             8
```

Removing a Rack-Mount Server

Before You Begin

Physically disconnect the CIMC LOM cables that connect the rack-mount server to the fabric extender before performing the following procedure. For high availability setups, remove both cables.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **remove server** *server-num*	Removes the specified rack-mount server.
Step 2	UCS-A# **commit-buffer**	Commits the transaction to the system configuration.

The following example removes rack-mount server 4 and commits the transaction:

```
UCS-A#  remove server 4
UCS-A* # commit-buffer
UCS-A #
```

What to Do Next

If you physically reconnect the rack-mount server, you must re-acknowledge it for the Cisco UCS Manager to rediscover the server.

For more information, see Acknowledging a Rack-Mount Server, on page 89.

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.
Step 2	UCS-A /server # **enable locator-led**	Turns on the rack-mount server locator LED.

	Command or Action	Purpose
Step 3	UCS-A /server # **commit-buffer**	Commits the transaction to the system configuration.

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.

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 #
```

Resetting the CMOS for a Rack-Mount Server

Sometimes, troubleshooting a server might require you to reset the CMOS. Resetting the CMOS is not part of the normal maintenance of a server.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *server-num*	Enters server mode for the rack-mount server.
Step 2	UCS-A /server # **reset-cmos**	Resets the CMOS for the rack-mount server.

	Command or Action	Purpose
Step 3	UCS-A /server # **commit-buffer**	Commits the transaction to the system configuration.

The following example resets the CMOS for rack-mount server 2 and commits the transaction:

```
UCS-A# scope server 2
UCS-A /server # reset-cmos
UCS-A /server* # commit-buffer
UCS-A /server #
```

Resetting the CIMC for a Rack-Mount Server

Sometimes, with the firmware, troubleshooting a server might require you to reset the CIMC. Resetting the CIMC is not part of the normal maintenance of a server. After you reset the CIMC, the server boots with the running version of the firmware for that 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 # **scope CIMC**	Enters server CIMC mode
Step 3	UCS-A /server/CIMC # **reset**	Resets the CIMC for the rack-mount server.
Step 4	UCS-A /server/CIMC # **commit-buffer**	Commits the transaction to the system configuration.

The following example resets the CIMC for rack-mount server 2 and commits the transaction:

```
UCS-A# scope server 2
UCS-A /server # scope CIMC
UCS-A /server/cimc # reset
UCS-A /server/cimc* # commit-buffer
UCS-A /server/cimc #
```

Clearing TPM for a Rack-Mount Server

You can clear TPM only on Cisco UCS M4 blade and rack-mount servers that include support for TPM.

Caution Clearing TPM is a potentially hazardous operation. The OS may stop booting. You may also see loss of data.

Before You Begin

TPM must be enabled.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *server-num*	Enters server mode for the rack-mount server.
Step 2	UCS-A# /server # **scope tpm** *tpm-ID*	Enters org TPM mode for the specified TPM.
Step 3	UCS-A# /server/tpm # **set adminaction clear-config**	Specifies that the TPM is to be cleared.
Step 4	UCS-A# /server/tpm # **commit-buffer**	Commits the transaction to the system configuration.

The following example shows how to clear TPM for a rack-mount server:

```
UCS-A# scope server 3
UCS-A# /server # scope tpm 1
UCS-A# /server/tpm # set adminaction clear-config
UCS-A# /server/tpm* # commit-buffer
```

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.

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
```

Issuing an NMI from a Rack-Mount Server

Perform the following procedure if the system remains unresponsive and you need Cisco UCS Manager to issue a Non-Maskable Interrupt (NMI) to the BIOS or operating system from the CIMC. This action creates a core dump or stack trace, depending on the operating system installed on the 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 /chassis/server # **diagnostic-interrupt**	
Step 3	UCS-A /chassis/server* # **commit-buffer**	Commits any pending transactions.

The following example sends an NMI from server 4 in chassis 2 and commits the transaction:

```
UCS-A# scope server 2/4
UCS-A /chassis/server # diagnostic-interrupt
UCS-A /chassis/server* # commit-buffer
UCS-A /chassis/server #
```

Viewing the Power Transition Log

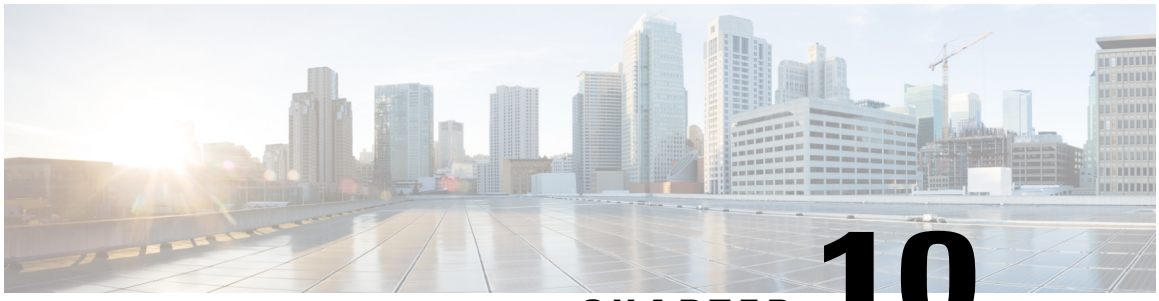
Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *server-num*	Enters server mode for the rack-mount server.
Step 2	UCS-A# /chassis/server # **show power-transition-log**	Displays the computeRebootLog instances for the specified server.

The following example shows how to view the power transition log for server 3.

```
UCS-A# scope server 3
UCS-A# /chassis/server # show power-transition-log
Last 5 server reboots (Newest first):

Pwr Change Source              Last pwr transition timestamp
-----------------------------  -----------------------------
UCSM TURNUP                    2016-10-28T09:35:04.498
HOST PWR TRANSITION            2016-10-27T17:06:56.157
UCSM TURNUP                    2016-10-27T17:06:24.734
UCSM ASSOCIATE                 2016-10-27T17:06:24.068
UCSM SERVER DISCOVER          2016-10-27T16:56:56.153
```



CHAPTER 10

S3260 Server Node Hardware Management

Cisco UCS S3260 Server Node Management

You can manage and monitor all Cisco UCS S3260 server nodes in a Cisco UCS domain through Cisco UCS Manager. You can perform some server management tasks, such as changes to the power state, from the server and service profile.

The remaining management tasks can only be performed on the server.

If a server slot in a chassis is empty, Cisco UCS Manager provides information, errors, and faults for that slot. You can also re-acknowledge the slot to resolve server mismatch errors and rediscover the server in the slot.

Booting a Server from the Service Profile

Before You Begin

Associate a service profile with a server or server pool.

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 service-profile** *profile-name*	Enters organization service profile mode for the specified service profile.
Step 3	UCS-A /org/service-profile # **power up**	Boots the server associated with the service profile.
Step 4	UCS-A /org/service-profile* # **commit-buffer**	Commits the transaction to the system configuration.

The following example boots the server associated with the service profile named ServProf34 and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # scope service-profile ServProf34
UCS-A /org/service-profile # power up
UCS-A /org/service-profile* # commit-buffer
UCS-A /org/service-profile #
```

Acknowledging a Server

Perform the following procedure to rediscover the server and all endpoints in the server. For example, you can use this procedure if a server is stuck in an unexpected state, such as the discovery state.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **acknowledge server** *chassis-num* / *server-num*	Acknowledges the specified server.
Step 2	UCS-A*# **commit-buffer**	Commits the transaction to the system configuration.

The following example acknowledges server 1 in chassis 3 and commits the transaction:

```
UCS-A#  acknowledge server 3/1
UCS-A* # commit-buffer
UCS-A #
```

Power Cycling a Server

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *chassis-num* / *server-num*	Enters chassis server mode for the specified server.
Step 2	UCS-A /chassis/server # **cycle** {**cycle-immediate** | **cycle-wait**}	Power cycles the server. Use the **cycle-immediate** keyword to immediately begin power cycling the server; use the **cycle-wait** keyword to schedule the power cycle to begin after all pending management operations have completed.
Step 3	UCS-A /chassis/server* # **commit-buffer**	Commits the transaction to the system configuration.

The following example immediately power cycles server 1 in chassis 3 and commits the transaction:

```
UCS-A#  scope server 3/1
UCS-A /chassis/server # cycle cycle-immediate
UCS-A /chassis/server* # commit-buffer
UCS-A /chassis/server #
```

Shutting Down a Server

When you use this procedure to shut down a server with an installed operating system, Cisco UCS Manager triggers the OS into a graceful shutdown sequence.

Before You Begin

Associate a service profile with a server or server pool.

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*.

	Command or Action	Purpose
Step 2	UCS-A /org # **scope service-profile** *profile-name*	Enters organization service profile mode for the specified service profile.
Step 3	UCS-A /org/service-profile # **power down**	Shuts down the server associated with the service profile.
Step 4	UCS-A /org/service-profile* # **commit-buffer**	Commits the transaction to the system configuration.

The following example shuts down the server associated with the service profile named ServProf34 and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # scope service-profile ServProf34
UCS-A /org/service-profile # power down
UCS-A /org/service-profile* # commit-buffer
UCS-A /org/service-profile #
```

Performing a Hard Reset on a Server

When you reset a server, Cisco UCS Manager sends a pulse on the reset line. You can choose to gracefully shut down the operating system. If the operating system does not support a graceful shutdown, the server is power cycled. The option to have Cisco UCS Manager complete all management operations before it resets the server does not guarantee the completion of these operations before the server is reset.

Note If you are trying to boot a server from a power-down state, you should not use **Reset**.

If you continue the power-up with this process, the desired power state of the servers become out of sync with the actual power state and the servers might unexpectedly shut down at a later time. To safely reboot the selected servers from a power-down state, click **Cancel**, then select the **Boot Server** action.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *chassis-num* / *server-num*	Enters chassis server mode for the specified server.
Step 2	UCS-A /chassis/server # **reset** {**hard-reset-immediate** | **hard-reset-wait**}	Performs a hard reset of the server. Use the: • **hard-reset-immediate** keyword to immediately begin hard resetting the server. • **hard-reset-wait** keyword to schedule the hard reset to begin after all pending management operations have completed.

	Command or Action	Purpose
Step 3	UCS-A /server* # **commit-buffer**	Commits the transaction to the system configuration.

The following example performs an immediate hard reset of server 1 in chassis 3 and commits the transaction:

```
UCS-A#  scope server 3/1
UCS-A /chassis/server # reset hard-reset-immediate
UCS-A /chassis/server* # commit-buffer
UCS-A /chassis/server #
```

Resetting a Cisco UCS S3260 Server Node to Factory Default Settings

You can now reset a Cisco UCS S3260 Server Node to its factory settings. By default, the factory reset operation does not affect storage drives. This is to prevent any loss of data. However, you can choose to reset these devices to a known state as well.

The following guidelines apply to Cisco UCS S3260 Server Nodes when using scrub policies:

- For Cisco UCS S3260 Server Nodes, you cannot delete storage by using the scrub policy.
- Cisco UCS S3260 Server Nodes do not support FlexFlash drives.
- For Cisco UCS S3260 Server Nodes, you can only reset the BIOS by using the scrub policy.

Important Resetting storage devices will result in loss of data.

Perform the following procedure to reset the server to factory default settings.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *chassis-num* / *server-num*	Enters chassis server mode for the specified server.
Step 2	UCS-A /chassis/server # **reset factory-default** [**delete-flexflash-storage** | **delete-storage** [**create-initial-storage-volumes**]]	Resets server settings to factory default using the following command options: • **factory-default—**Resets the server to factory defaults without deleting storage **Note** This operation resets the BIOS. • **delete-flexflash-storage—**Resets the server to factory defaults and deletes flexflash storage **Note** This operation is not supported on Cisco UCS S3260 Server Nodes.

	Command or Action	Purpose
		• **delete-storage—**Resets the server to factory defaults and deletes all storage • **create-initial-storage-volumes—**Resets the server to factory defaults, deletes all storage, sets all disks to their initial state
Step 3	UCS-A /chassis/server* # **commit-buffer**	Commits any pending transactions.

The following example resets the server settings to factory default without deleting storage, and commits the transaction:

```
UCS-A# scope server 3/1
UCS-A /chassis/server # reset factory-default
UCS-A /chassis/server* # commit-buffer
```

The following example resets the server settings to factory default, deletes flexflash storage, and commits the transaction:

```
UCS-A# scope server 3/1
UCS-A /chassis/server # reset factory-default delete-flexflash-storage

UCS-A /chassis/server* # commit-buffer
```

The following example resets the server settings to factory default, deletes all storage, and commits the transaction:

```
UCS-A# scope server 3/1
UCS-A /chassis/server # reset factory-default delete-storage
UCS-A /chassis/server* # commit-buffer
```

The following example resets the server settings to factory default, deletes all storage, sets all disks to their initial state, and commits the transaction:

```
UCS-A# scope server 3/1
UCS-A /chassis/server # reset factory-default delete-storage create-initial-storage-volumes
UCS-A /chassis/server* # commit-buffer
```

Removing a Server from a Chassis

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **remove server** *chassis-num* / *server-num*	Removes the specified server.

	Command or Action	Purpose
Step 2	UCS-A*# **commit-buffer**	Commits the transaction to the system configuration.
Step 3	Go to the physical location of the chassis and remove the server hardware from the slot.	For instructions on how to remove the server hardware, see the *Cisco UCS Hardware Installation Guide* for your chassis.

The following example removes server 1 in chassis 3 and commits the transaction:

```
UCS-A#  remove server 3/1
UCS-A* # commit-buffer
UCS-A #
```

What to Do Next

If you physically re-install the blade server, you must re-acknowledge the slot for the Cisco UCS Manager to rediscover the server.

For more information, see Acknowledging a Server, on page 98.

Decommissioning a Server

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **decommission server** *chassis-num* / *server-num*	Decommissions the specified server.
Step 2	UCS-A*# **commit-buffer**	Commits the transaction to the system configuration.

The following example decommissions server 1 in chassis 3 and commits the transaction:

```
UCS-A# decommission server 3/1
UCS-A* # commit-buffer
UCS-A #
```

Turning On the Locator LED for a Server

Procedure

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

	Command or Action	Purpose
Step 2	UCS-A /chassis/server # **enable locator-led** [**multi-master** | **multi-slave**]	Turns on the server locator LED. The following command options are not applicable to Cisco UCS S3260 Server Nodes: • **multi-master—**Turns on the LED for the master node only. • **multi-slave—**Turns on the LED for the slave node only.
Step 3	UCS-A /chassis/server* # **commit-buffer**	Commits the transaction to the system configuration.

The following example turns on the locator LED on server 1 in chassis 3 and commits the transaction:

```
UCS-A# scope server 3/1
UCS-A /chassis/server # enable  locator-led
UCS-A /chassis/server* # commit-buffer
UCS-A /chassis/server #
```

The following example turns on the locator LED for the master node only on server 1 in chassis 3 and commits the transaction:

```
UCS-A# scope chassis 3/1
UCS-A /chassis/server # enable locator-led multi-master
UCS-A /chassis/server* # commit-buffer
UCS-A /chassis/server #
```

Turning Off the Locator LED for a 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 server locator LED. The following command options are not applicable to Cisco UCS S3260 Server Nodes: • **multi-master—**Turns off the LED for the master node only. • **multi-slave—**Turns off the LED for the slave node only.
Step 3	UCS-A /chassis/server* # **commit-buffer**	Commits the transaction to the system configuration.

The following example turns off the locator LED on server 1 in chassis 3 and commits the transaction:

```
UCS-A# scope chassis 3/1
UCS-A /chassis/server # disable locator-led
UCS-A /chassis/server* # commit-buffer
UCS-A /chassis/server #
```

The following example turns off the locator LED for the master node on server 1 in chassis 3 and commits the transaction:

```
UCS-A# scope chassis 3/1
UCS-A /chassis/server # disable locator-led multi-master
UCS-A /chassis/server* # commit-buffer
UCS-A /chassis/server #
```

Resetting All Memory Errors

Use this procedure to reset all correctable and uncorrectable memory errors encountered by .

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *chassis-num* / *server-num*	Enters chassis server mode for the specified server.
Step 2	UCS-A /chassis/server # **reset-all-memory-errors**	Performs a reset of the memory cards.
Step 3	UCS-A /chassis/server* # **commit-buffer**	Commits the transaction to the system configuration.

The following example performs an immediate hard reset of server 1 in chassis 3 and commits the transaction:

```
UCS-A#  scope server 3/1
UCS-A /chassis/server # reset-all-memory-errors
UCS-A /chassis/server* # commit-buffer
UCS-A /chassis/server #
```

Resetting IPMI to Factory Default Settings

Perform the following procedure if you need to reset IPMI to factory default settings.

Procedure

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

	Command or Action	Purpose
Step 2	UCS-A /chassis/server # **reset-ipmi**	Resets IPMI settings to factory default.
Step 3	UCS-A /chassis/server* # **commit-buffer**	Commits any pending transactions.

The following example resets the IPMI settings to factory default and commits the transaction:

```
UCS-A# scope server 3/1
UCS-A /chassis/server # reset-ipmi
UCS-A /chassis/server* # commit-buffer
UCS-A /chassis/server #
```

Resetting the CIMC for a Server

Sometimes, with the firmware, troubleshooting a server might require you to reset the CIMC. Resetting the CIMC is not part of the normal maintenance of a server. After you reset the CIMC, the server boots with the running version of the firmware for that server.

If the CIMC is reset, the power monitoring functions of Cisco UCS become briefly unavailable until the CIMC reboots. Typically, the reset only takes 20 seconds; however, it is possible that the peak power cap can exceed during that time. To avoid exceeding the configured power cap in a low power-capped environment, consider staggering the rebooting or activation of CIMCs.

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 # **scope cimc**	Enters chassis server CIMC mode
Step 3	UCS-A /chassis/server/cimc # **reset**	Resets the CIMC for the server.
Step 4	UCS-A /chassis/server/cimc* # **commit-buffer**	Commits the transaction to the system configuration.

The following example resets the CIMC for server 1 in chassis 3 and commits the transaction:

```
UCS-A# scope server 3/1
UCS-A /chassis/server # scope cimc
UCS-A /chassis/server/cimc # reset
UCS-A /chassis/server/cimc* # commit-buffer
UCS-A /chassis/server/cimc #
```

Resetting the CMOS for a Server

Sometimes, troubleshooting a server might require you to reset the CMOS. Resetting the CMOS is not part of the normal maintenance of a 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 # **reset-cmos**	Resets the CMOS for the server.
Step 3	UCS-A /chassis/server* # **commit-buffer**	Commits the transaction to the system configuration.

The following example resets the CMOS for server 1 in chassis 3 and commits the transaction:

```
UCS-A# scope server 3/1
UCS-A /chassis/server # reset-cmos
UCS-A /chassis/server* # commit-buffer
UCS-A /chassis/server #
```

Resetting KVM

Perform the following procedure if you need to reset and clear all KVM sessions.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *chassis-num* / *server-num*	Enters chassis server mode for the specified server.
Step 2	UCS-A /chassis/server # **reset-kvm**	Resets and clears all KVM sessions.
Step 3	UCS-A /chassis/server* # **commit-buffer**	Commits any pending transactions.

The following example resets and clears all KVM sessions and commits the transaction:

```
UCS-A# scope server 3/1
UCS-A /chassis/server # reset-kvm
UCS-A /chassis/server* # commit-buffer
UCS-A /chassis/server #
```

Issuing an NMI from a Server

Perform the following procedure if the system remains unresponsive and you need Cisco UCS Manager to issue a Non-Maskable Interrupt (NMI) to the BIOS or operating system from the CIMC. This action creates a core dump or stack trace, depending on the operating system installed on the server.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *chassis-num* / *server-num*	Enters chassis server mode for the specified server.
Step 2	UCS-A /chassis/server # **diagnostic-interrupt**	
Step 3	UCS-A /chassis/server* # **commit-buffer**	Commits any pending transactions.

The following example sends an NMI from server 1 in chassis 3 and commits the transaction:

```
UCS-A# scope server 3/1
UCS-A /chassis/server # diagnostic-interrupt
UCS-A /chassis/server* # commit-buffer
UCS-A /chassis/server #
```

Recovering a Corrupt BIOS

On rare occasions, an issue with a server may require you to recover the corrupted BIOS. This procedure is not part of the normal maintenance of a server. After you recover the BIOS, the server boots with the running version of the firmware for that 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 # **recover-bios** *version*	Loads and activates the specified BIOS version.
Step 3	UCS-A /chassis/server* # **commit-buffer**	Commits the transaction to the system configuration.

The following example shows how to recover the BIOS:

```
UCS-A# scope server 3/1
UCS-A /chassis/server # recover-bios S5500.0044.0.3.1.010620101125
UCS-A /chassis/server* # commit-buffer
UCS-A /chassis/server #
```

Health LED Alarms

The server health LED is located on the front of each server. Cisco UCS Manager allows you to view the sensor faults that cause the blade health LED to change color from green to amber or blinking amber.

The health LED alarms display the following information:

Name	Description
Severity column	The severity of the alarm. This can be one of the following: • Critical - The server health LED blinks amber. This is indicated with a red dot. • Minor - The server health LED is amber. This is indicated with an orange dot.
Description column	A brief description of the alarm.
Sensor ID column	The ID of the sensor that triggered the alarm.
Sensor Name column	The name of the sensor that triggered the alarm.

Viewing Health LED Status

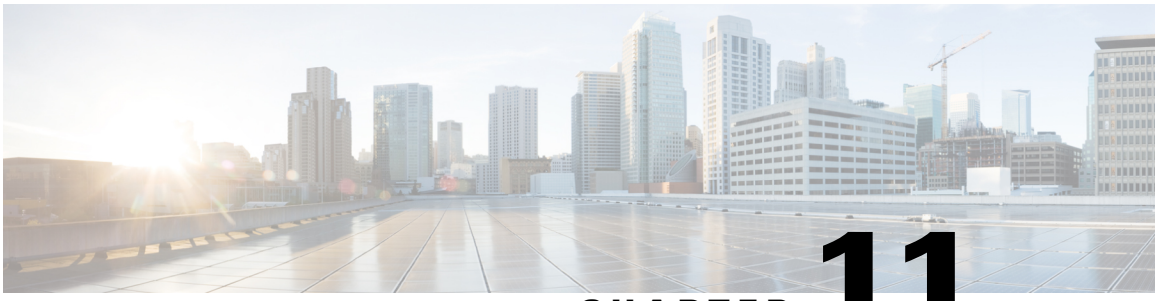
Procedure

	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 health-led expand**	Displays the health LED and sensor alarms for the selected server.

The following example shows how to display the health LED status and sensor alarms for chassis 1 server 3:

```
UCS-A# scope server 1/3
UCS-A /chassis/server # show health-led expand
Health LED:
    Severity: Normal
    Reason:
    Color: Green
    Oper State: On

UCS-A /chassis/server #
```



CHAPTER 11

Virtual Interface Management

Virtual Circuits

A virtual circuit or virtual path refers to the path that a frame takes from its source vNIC to its destination virtual switch port (vEth) or from a source virtual switch port to its destination vNIC. There are many possible virtual circuits that traverse through a physical cable.Cisco UCS Manager uses virtual network tags (VN-TAG) to identify these virtual circuits and differentiate between them. The OS decides the virtual circuit that a frame must traverse on a basis of a series of decisions.

In the server, the OS decides the Ethernet interface from which to send the frame.

Note During service profile configuration, you can select the fabric interconnect to be associated with a vNIC. You can also choose whether fabric failover is enabled for the vNIC. If fabric failover is enabled, the vNIC can access the second fabric interconnect when the default fabric interconnect is unavailable. *Cisco UCS Manager Server Management Guide* provides more details about vNIC configuration during service profile creation.

After the host vNIC is selected, the frame exits the selected vNIC and, through the host interface port (HIF), enters the IOM to which the vNIC is pinned. The frame is then forwarded to the corresponding network Interface port (NIF) and then to the Fabric Interconnect to which the IOM is pinned.

The NIF is selected based on the number of physical connections between the IOM and the Fabric Interconnect, and on the server ID from which the frame originated.

Virtual Interfaces

In a blade server environment, the number of vNICs and vHBAs configurable for a service profile is determined by adapter capability and the amount of virtual interface (VIF) namespace available on the adapter. In Cisco UCS, portions of VIF namespace are allotted in chunks called VIFs. Depending on your hardware, the maxiumum number of VIFs are allocated on a predefined, per-port basis.
The maximum number of VIFs varies based on hardware capability and port connectivity. For each configured vNIC or vHBA, one or two VIFs are allocated. Stand-alone vNICs and vHBAs use one VIF and failover vNICs and vHBAs use two.

The following variables affect the number of VIFs available to a blade server, and therefore, how many vNICs and vHBAs you can configure for a service profile.

- Maximum number of VIFs supported on your fabric interconnect
- How the fabric interconnects are cabled
- If your fabric interconnect and IOM are configured in fabric port channel mode

For more information about the maximum number of VIFs supported by your hardware configuration, see the appropriate *Cisco UCS Configuration Limits for Cisco UCS Manager* for your software release.

Virtual Interface Subscription Management and Error Handling

For fabric interconnects grouped in a port-channel, changes to the way you connect the fabric interconnect to the I/O module could result in a drastic change to the number of VIFs available to a blade server. To help you track the effect of these changes, Cisco UCS Manager maintains the following metrics:

- Maximum number of VIFs supported by hardware
- Connectivity type

If you change your configuration in a way that decreases the number of VIFs available to a blade, UCS Manager will display a warning and ask you if you want to proceed. This includes several scenarios, including times where adding or moving a connection decreases the number of VIFs.

Virtualization in Cisco UCS

Overview of Virtualization

Virtualization allows you to create multiple Virtual Machines (VMs) to run in isolation, side by side on the same physical machine.

Each virtual machine has its own set of virtual hardware (RAM, CPU, NIC) upon which an operating system and fully configured applications are loaded. The operating system sees a consistent, normalized set of hardware regardless of the actual physical hardware components.

In a virtual machine, both hardware and software are encapsulated in a single file for rapid provisioning and moving between physical servers. You can move a virtual machine, within seconds, from one physical server to another for zero-downtime maintenance and continuous workload consolidation.

The virtual hardware makes it possible for many servers, each running in an independent virtual machine, to run on a single physical server. The advantages of virtualization include better use of computing resources, greater server density, and seamless server migration.

Overview of Cisco Virtual Machine Fabric Extender

A virtualized server implementation consists of one or more VMs that run as guests on a single physical server. The guest VMs are hosted and managed by a software layer called the hypervisor or virtual machine manager (VMM). Typically, the hypervisor presents a virtual network interface to each VM and performs Layer 2 switching of traffic from a VM to other local VMs or to another interface to the external network.

Working with a Cisco virtual interface card (VIC) adapter, the Cisco Virtual Machine Fabric Extender (VM-FEX) bypasses software-based switching of VM traffic by the hypervisor for external hardware-based switching in the fabric interconnect. This method reduces the load on the server CPU, provides faster switching, and enables you to apply a rich set of network management features to local and remote traffic.

VM-FEX extends the IEEE 802.1Qbh port extender architecture to the VMs by providing each VM interface with a virtual Peripheral Component Interconnect Express (PCIe) device and a virtual port on a switch. This solution allows precise rate limiting and quality of service (QoS) guarantees on the VM interface.

Virtualization with Network Interface Cards and Converged Network Adapters

Network interface card (NIC) and converged network adapters support virtualized environments with the standard VMware integration with ESX installed on the server and all virtual machine management performed through the VC.

Portability of Virtual Machines

If you implement service profiles you retain the ability to easily move a server identity from one server to another. After you image the new server, the ESX treats that server as if it were the original.

Communication between Virtual Machines on the Same Server

These adapters implement the standard communications between virtual machines on the same server. If an ESX host includes multiple virtual machines, all communications must go through the virtual switch on the server.

If the system uses the native VMware drivers, the virtual switch is out of the network administrator's domain and is not subject to any network policies. As a result, for example, QoS policies on the network are not applied to any data packets traveling from VM1 to VM2 through the virtual switch.

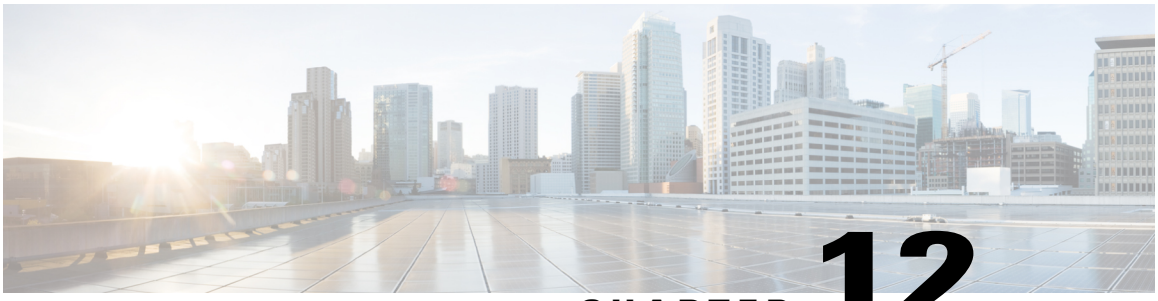
If the system includes another virtual switch, such as the Nexus 1000, that virtual switch is subject to the network policies configured on that switch by the network administrator.

Virtualization with a Virtual Interface Card Adapter

A Cisco VIC adapter is a converged network adapter (CNA) that is designed for both bare metal and VM-based deployments. The VIC adapter supports static or dynamic virtualized interfaces, which includes up to 128 virtual network interface cards (vNICs).

There are two types of vNICs used with the VIC adapter—static and dynamic. A static vNIC is a device that is visible to the OS or hypervisor. Dynamic vNICs are used for VM-FEX by which a VM is connected to a veth port on the Fabric Interconnect.

VIC adapters support VM-FEX to provide hardware-based switching of traffic to and from virtual machine interfaces.



CHAPTER 12

Troubleshoot Infrastructure

Recovering the Corrupt BIOS on a Blade Server

On rare occasions, an issue with a blade server may require you to recover the corrupted BIOS. This procedure is not part of the normal maintenance of a server. After you recover the BIOS, the blade server boots with the running version of the firmware for that server.

Before You Begin

Important Remove all attached or mapped USB storage from a server before you attempt to recover the corrupt BIOS on that server. If an external USB drive is attached or mapped from vMedia to the server, BIOS recovery fails.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *chassis-id / server-id*	Enters chassis server mode for the specified blade server in the specified chassis.
Step 2	UCS-A /chassis/server # **recover-bios** *version*	Loads and activates the specified BIOS version.
Step 3	UCS-A /chassis/server # **commit-buffer**	Commits the transaction.

The following example shows how to recover the BIOS:

```
UCS-A# scope server 1/7
UCS-A /chassis/server # recover-bios S5500.0044.0.3.1.010620101125
UCS-A /chassis/server* # commit-buffer
UCS-A /chassis/server #
```

Recovering the Corrupt BIOS on a Rack-Mount Server

On rare occasions, an issue with a rack-mount server may require you to recover the corrupted BIOS. This procedure is not part of the normal maintenance of a rack-mount server. After you recover the BIOS, the rack-mount server boots with the running version of the firmware for that server.

Before You Begin

Important Remove all attached or mapped USB storage from a server before you attempt to recover the corrupt BIOS on that server. If an external USB drive is attached or mapped from vMedia to the server, BIOS recovery fails.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *server-id*	Enters server mode for the specified rack-mount server.
Step 2	UCS-A /server # **recover-bios** *version*	Loads and activates the specified BIOS version.
Step 3	UCS-A /server # **commit-buffer**	Commits the transaction.

The following example shows how to recover the BIOS:

```
UCS-A# scope server 1
UCS-A /server # recover-bios S5500.0044.0.3.1.010620101125
UCS-A /server* # commit-buffer
UCS-A /server #
```