Cisco UCS Manager Server Management Using the CLI, Release 4.0

First Published: 2018-08-14

Last Modified: 2019-04-26

CONTENTS

Preface

-
-
-
-

Audience

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

- Server administration

- Storage administration

- Network administration

- Network security

Conventions

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

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

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

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

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

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

Warning   IMPORTANT SAFETY INSTRUCTIONS

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

SAVE THESE INSTRUCTIONS

Related Cisco UCS Documentation

Documentation Roadmaps

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

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

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

Other Documentation Resources

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

Documentation Feedback

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

CHAPTER 1

New and Changed Information

- New and Changed Information, on page 1

New and Changed Information

This section provides information on new feature and changed behavior in Cisco UCS Manager, Release 4.0(4a).

Table 1: New Features and Changed Behavior in Cisco UCS Manager, Release 4.0(4a)

Feature	Description	Where Documented
Persistent Memory Modules	Cisco UCS Manager Release 4.0(4) introduces support for the Intel® Optane™ Data Center persistent memory modules on the UCS M5 servers that are based on the Second Generation Intel® Xeon® Scalable processors. This release provides the ability to configure and manage Intel® Optane™ DC persistent memory modules through Cisco UCS Manager.	Persistent Memory Modules, on page 304
Cisco Boot Optimized M.2 RAID Controller	Cisco UCS Manager supports Cisco boot optimized M.2 RAID controller (UCS-M2-HWRAID).	Local Devices Boot, on page 140
Disk Scrub for boot drives and VDs in Cisco UCS S3260 M4 and Cisco UCS S3260 M5 Storage Servers	Cisco UCS Manager extends support for scrubbing boot drives and VDs in Cisco UCS S3260 M4 and Cisco UCS S3260 M5 Storage Servers	Scrub Policy Settings, on page 304

Feature	Description	Where Documented
Persistent Memory Scrub	Cisco UCS Manager extends support for persistent memory scrub, which enables you to preserve or remove the persistent memory configuration and data on a server.	Scrub Policy Settings, on page 304
Alternate KVM Port	Cisco UCS Manager allows you to configure a port number between 1024 and 49151 as the KVM port.	KVM Management Policy, on page 282

This section provides information on new feature and changed behavior in Cisco UCS Manager, Release 4.0(2a).

Table 2: New Features and Changed Behavior in Cisco UCS Manager, Release 4.0(2a)

Feature	Description	Where Documented
Cisco UCS C480 M5 ML Server	Cisco UCS Manager extends support for all existing features on Cisco UCS C480 M5 ML Server.	Server Management Overview, on page 3
Cisco UCS VIC 1495	Cisco UCS Manager extends support for all existing features on Cisco UCS VIC 1495.	Server Management Overview, on page 3
Cisco UCS VIC 1497	Cisco UCS Manager extends support for all existing features on Cisco UCS VIC 1497.	Server Management Overview, on page 3

This section provides information on new feature and changed behavior in Cisco UCS Manager, Release 4.0(1a).

Table 3: New Features and Changed Behavior in Cisco UCS Manager, Release 4.0(1a)

Feature	Description	Where Documented
Cisco UCS 6454 Fabric Interconnect	This release introduces Cisco UCS 6454 Fabric Interconnect that support 10/25 Gigabit ports in the fabric with 40/100 Gigabit uplink ports.	Server Management Overview, on page 3
Cisco UCS VIC 1455	In Cisco UCS Manager Release 4.0(1a), Cisco UCS VIC 1455 are supported.	Server Management Overview, on page 3
Cisco UCS C125 M5 Server	Cisco UCS Manager extends support for all existing features on Cisco UCS C125 M5 Server.	Rack-Enclosure Server Management, on page 64

Server Management Overview

Server Management Overview

Cisco UCS Manager enables you to manage general and complex server deployments. For example, you can manage a general deployment with a pair of Fabric Interconnects (FIs), which is the redundant server access layer that you get with the first chassis that can scale up to 20 chassis' and up to 160 physical servers. This can be a combination of blades and rack mount servers to support the workload in your environment. As you add more servers, you can continue to perform server provisioning, device discovery, inventory, configuration, diagnostics, monitoring, fault detection, and auditing.

Beginning with release 4.0(2a), Cisco UCS Manager extends support for all existing features on the following Cisco UCS hardware unless specifically noted:

- Cisco UCS C480 M5 ML Server

- Cisco UCS VIC 1495

- Cisco UCS VIC 1497

Beginning with release 4.0(1a), Cisco UCS Manager extends support for all existing features on the following Cisco UCS hardware unless specifically noted:

- Cisco UCS 6454 Fabric Interconnect

- Cisco UCS VIC 1455

- Cisco UCS VIC 1457

- Cisco UCS C125 M5 Server

Beginning with release 4.0(1a), Cisco UCS Manager does not support UCS M2 servers.

By default, the Cisco UCS 6454 Fabric Interconnect , the Cisco UCS 6332 FIs, the UCS Mini 6324 FIs, and the UCS 6200 Series FIs include centralized management. You can manage the UCS Blade Servers and Rack-Mount Servers that are in the same domain from one console. You can also manage the UCS Mini from the Cisco UCS Manager.

To ensure the optimum server performance, you can configure the amount of power that you allocate to servers. You can also set the server boot policy, the location from which the server boots, and the order in which the boot devices are invoked. You can create service profiles for the UCS B-Series Blade Servers and the UCS Mini to assign to servers. Service profiles enable you to assign BIOS settings, security settings, the number of vNICs and vHBAs, and anything else that you want to apply to a server.

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.
Cisco UCS Manager Server Management Guide	Discusses the new licenses, registering Cisco UCS domains with Cisco UCS Central, power capping, server boot, server profiles and server-related policies.
Cisco UCS Manager Storage Management Guide	Discusses all aspects of storage management such as SAN and VSAN in Cisco UCS Manager.
Cisco UCS Manager Network Management Guide	Discusses all aspects of network management such as LAN and VLAN connectivity in Cisco UCS Manager.
Cisco UCS Manager System Monitoring Guide	Discusses all aspects of system and health monitoring including system statistics in Cisco UCS Manager.
Cisco UCS S3260 Server Integration with Cisco UCS Manager	Discusses all aspects of management of UCS S-Series servers that are managed through Cisco UCS Manager.

CHAPTER 3

Server License Management

Licenses

Each Cisco UCS fabric interconnect comes with several port licenses that are factory installed and shipped with the hardware. You can purchase fabric interconnects fully licensed or partially licensed. You can also purchase additional licenses after delivery.

Cisco UCS 6454 Fabric Interconnect uses the following licenses:

Table 4: Cisco UCS 6454 Fabric Interconnect Licenses

Ports	Licenses
Ports 1-48	ETH_PORT_ACTIVATION_PKG and ETH_PORT_C_ACTIVATION_PKG (From 6200 Series FI) - Licenses used for 10/25 GB Ethernet ports
Ports 49-54	100G_ETH_PORT_ACTIVATION_PKG – Licenses used for 40/100 GB Ethernet ports

The following four licenses are for the 6300 Series FI and are only valid on the 6332 and 6332-16UP FIs.

- 40G_ETH_PORT_ACTIVATION_PKG – Licenses used for 40 GB Ethernet ports

- 40G_ETH_C_PORT_ACTIVATION_PKG – Licenses used for 40 GB Ethernet ports directly connected to rack servers (C-Direct)

- 10G_C_PORT_ACTIVATION_PKG – Licenses used for the first 16 10 GB unified ports on the 6332-16UP that are directly connected to rack servers (C-Direct)

• 10G_PORT_ACTIVATION_PKG – Licenses used for the first 16 10 GB unified ports on the 6332-16UP

Note The 10G_PORT_ACTIVATION_PKG and 10G_C_PORT_ACTIVATION_PKG licenses are only valid for the 6332-16UP FIs, and can only be installed on them.

The following licenses are used when S3260 system is connected to FI as appliance (appliance port) or Cisco UCS Manager managed node (server port):

Table 5: S3260 system License Requirement

FI Model	License
6200	ETH_PORT_ACTIVATION_PKG
6332	40G_ETH_PORT_ACTIVATION_PKG
6332-16UP	10G_PORT_ACTIVATION_PKG
Cisco UCS 6454 Fabric Interconnect	40G_ETH_PORT_ACTIVATION_PKG

Cisco UCS C125 M5 Servers support only Cisco UCS 6454 Fabric Interconnect and 6300 series Fabric Interconnect.

At a minimum, each fabric interconnect ships with the following counted licenses pre-installed:

Fabric Interconnect	Default Base Licenses
Cisco UCS 6248 (unified ports)	For the 12 first enabled Ethernet ports and any Fibre Channel ports in the expansion module.
Cisco UCS 6296 (unified ports)	For the first 18 enabled Ethernet ports and any Fibre Channel ports in the expansion module.
Cisco UCS 6324	For 4 non-breakout ports only. The fifth port, which does not include a license, is further broken in to four 10 GB ports.
Cisco UCS 6332 16UP	For four 40 GB ports and eight 10 GB ports. **Note** The first 16 ports are 10 GB. The remaining are 40 GB.
Cisco UCS 6332	For eight 40 GB ports.
Cisco UCS 6454 Fabric Interconnect	For 18 10/25 GB ports (ports 1-48) For 2 40/100 GB ports (ports 49-54).

Port License Consumption

Port licenses are not bound to physical ports. When you disable a licensed port, that license is retained for use with the next enabled port. To use additional fixed ports, you must purchase and install licenses for those ports. All ports, regardless of their type (fibre, ethernet) consume licenses if they are enabled.

For breakout capable ports available in the 6332 and the 6332-16UP platforms, 40 GB licenses remain applied to the main port even if that port is a breakout port, and that port continues to consume only one 40 GB license.

Note The initial configuration of a port will enable it, and consume a license.

Important Licenses are not portable across product generations. Licenses purchased for 6200 series fabric interconnects cannot be used to enable ports on 6300 series or 6400 fabric interconnects or vice-versa.

Each Cisco UCS 6324 Fabric Interconnect comes with a factory installed port license that is shipped with the hardware. This license is for the eight 40 GB unified ports, and can be used for any supported purpose. The C-direct port license is factory installed with a grace period, and can be used for Cisco UCS rack servers.

Grace Period

If you attempt to use a port that does not have an installed license, Cisco UCS initiates a 120 day grace period. The grace period is measured from the first use of the port without a license and is paused when a valid license file is installed. The amount of time used in the grace period is retained by the system.

Note Each physical port has its own grace period. Initiating the grace period on a single port does not initiate the grace period for all ports.

If a licensed port is unconfigured, that license is transferred to a port functioning within a grace period. If multiple ports are acting within grace periods, the license is moved to the port whose grace period is closest to expiring.

High Availability Configurations

To avoid inconsistencies during failover, we recommend that both fabric interconnects in the cluster have the same number of ports licensed. If symmetry is not maintained and failover occurs, Cisco UCS enables the missing licenses and initiates the grace period for each port being used on the failover node.

C-Direct Rack Licensing Support

Release 4.0(1a) and Higher

Beginning with release 4.0(1a), Cisco UCS 6454 Fabric Interconnects use the ETH_C_PORT_ACTIVATION_PKG feature pack for C-Direct port licenses for ports 1-48. There are no default ETH_C_PORT_ACTIVATION_PKG licenses shipped with the Fabric Interconnect. You may purchase them as required.

C-direct support is only applicable on ports that are connected to the rack servers. The ETH_C_PORT_ACTIVATION_PKG is added to the existing license package with all the same properties as the existing licensing feature. The Subordinate Quantity property is added to the ETH_PORT_ACTIVATION_PKG to track ports connected to rack servers.

The License Tab in the Cisco UCS Manager GUI displays the new license and the **Subordinate Quantity** for the license. You can also use the **show feature** and **show usage** commands under **scope license** to view the license feature, the vendor version type, and the grace period for each license.

Release 3.2(3k) and Earlier

Each Cisco UCS fabric interconnect is shipped with a default number of port licenses that are factory licensed and shipped with the hardware. C-direct support is only applicable on ports that are connected to the rack servers. The 10G_C_PORT_ACTIVATION_PKG and the 40G_ETH_C_PORT_ACTIVATION _PKG are added to the existing license package with all the same properties as the existing licensing feature. The **Subordinate Quantity** property is added to the 10G_PORT_ACTIVATION_PKG and 40G_ETH_PORT_ACTIVATION_PKG to track ports connected to rack servers.

The License Tab in the Cisco UCS Manager GUI displays the new license and the **Subordinate Quantity** for the license. You can also use the **show feature** and **show usage** commands under **scope license** to view the license feature, the vendor version type, and the grace period for each license.

Ports connected to rack servers can use existing 10G_PORT_ACTIVATION_PKG, 40G_ETH_PORT_ACTIVATION_PKG if the license is available or if the license is not in use. Otherwise, you must purchase a 10G_C_PORT_ACTIVATION_PKG, the 40G_ETH_C_PORT_ACTIVATION _PKG to avoid the license grace period.

There is no change in the 10 GB ports. The 10G_PORT_ACTIVATION_PKG and 10G_C_PORT_ACTIVATION_PKG license packages include all of the same properties as the existing the ETH_PORT_ACTIVATION_PKG and the ETH_PORT_C_ACTIVATION_PKG license features.

Configuration and Restrictions

- The C-Direct rack licensing feature accounts for the rack server ports that are directly connected to the FI, but not to a CIMC port. The default quantity for the 10G_C_PORT_ACTIVATION_PKG and the 40G_ETH_C_PORT_ACTIVATION _PKG is always 0.

- When a 40 GB port, or a breakout port under a 40 GB breakout port is enabled without any connections, this port is allotted a license under the 40G_ETH_PORT_ACTIVATION_PKG, if available. If this port is connected to a Direct-Connect rack server after a time lag, it triggers a complete re-allocation of licenses, then this port passes through one of the following license allocation scenarios occurs:

  When you enable a breakout port under a 40 GB breakout port, if that port is connected to a Direct-Connect rack server, and the 40G_C_PORT_ACTIVATION_PKG license files are installed on the FI, the following license allocation occurs:

  - If no other ports under the breakout port are enabled, the parent 40 GB port is allotted a license under the 40G_C_PORT_ACTIVATION_PKG, and the used quantity is incremented for this instance.

  - If other ports are enabled, and if at least one port is not connected to a Direct Connect rack server, even if the port is not being used, the parent 40 GB port is allotted a license under the 40G_ETH_PORT_ACTIVATION_PKG, and the used quantity is incremented for this instance.

- When you enable a breakout port under a 40 GB breakout port and that port is connected to a Direct-Connect rack server, and the 40G_C_PORT_ACTIVATION_PKG license files are not installed on the FI, the following license allocation occurs:

  - If no ports under the breakout port are enabled, the parent 40 GB port is allotted a license under the 40G_ETH_PORT_ACTIVATION_PKG. The subordinate quantity is increased if the

licenses are available in the 40G_ETH_PORT_ACTIVATION_PKG. If the licenses are not available, the used quantity under this feature is increased and the entire port goes in to the grace period.

- If other ports are enabled and at least one port is not connected to a Direct Connect rack server, even if the port is not being used, the parent 40 GB port is allotted a license under the 40G_ETH_PORT_ACTIVATION_PKG, and the used quantity is incremented for this instance.

Obtaining the Host ID for a Fabric Interconnect

The host ID is also known as the serial number.

Procedure

	Command or Action	**Purpose**
Step 1	UCS-A# **scope license**	Enters license mode.
Step 2	UCS-A /license # **show server-host-id**	Obtains the host ID or serial number for the fabric interconnect.
		Tip Use the entire host ID that displays after the equal (=) sign.

Example

The following example obtains the host ID for a fabric interconnect:

```
UCS-A# scope license
UCS-A /license # show server-host-id
Server host id:
    Scope Host Id
    ----- -------
    A     VDH=SSI12121212
    B     VDH=SSI13131313
UCS-A /license #
```

What to do next

Obtain the required licenses from Cisco.

Obtaining a License

✎

Note   This process may change after the release of this document. If one or more of these steps no longer applies, contact your Cisco representative for information on how to obtain a license file.

Before you begin

Obtain the following:

- Host ID or serial number for the fabric interconnect

- Claim certificate or other proof of purchase document for the fabric interconnect or expansion module

Procedure

Step 1   Obtain the product authorization key (PAK) from the claim certificate or other proof of purchase document.

Step 2   Locate the website URL in the claim certificate or proof of purchase document.

Step 3   Access the website URL for the fabric interconnect and enter the serial number and the PAK.

Cisco sends you the license file by email. The license file is digitally signed to authorize use on only the requested fabric interconnect. The requested features are also enabled once Cisco UCS Manager accesses the license file.

What to do next

Install the license on the fabric interconnect.

Installing a License

Note   In a cluster setup, Cisco recommends that you download and install licenses to both fabric interconnects in matching pairs. An individual license is only downloaded to the fabric interconnect that is used to initiate the download.

Before you begin

Obtain the required licenses from Cisco.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope license**	Enters license mode.
Step 2	UCS-A /license # **download license** *from-filesystem*	Downloads the license from its source location. For the *from-filesystem:* argument, use one of the following syntaxes: • **ftp://** *server-ip-addr* • **scp://** *username@server-ip-addr* • **sftp://** *username@server-ip-addr*

	Command or Action	Purpose
		• **tftp://** *server-ip-addr* **:** *port-num* You cannot have spaces anywhere in the path name or the file name. For example, `c:\Path\Folder_Name\License.lic` is a valid path, but `c:\Path\Folder Name\License.lic` is invalid due to the space in `"Folder Name"`.
Step 3	UCS-A /license # **install file** *license_filename*	Installs the license. **Note** There is no downtime required or impact to traffic when installing a new port license.

Example

The following example uses FTP to download and install a license:

```
UCS-A # scope license
UCS-A /license # download license ftp://192.168.10.10/license/port9.lic
UCS-A /license # install file port9.lic
UCS-A /license #
```

Viewing the Licenses Installed on a Fabric Interconnect

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope license**	Enters license mode.
Step 2	UCS-A /license # **show file** [*license_filename* | **detail**]	Displays the licenses installed on the fabric interconnect with the level of detail specified in the command.

Example

The following example displays the full details for the licenses installed on a fabric interconnect:

```
UCS-A# scope license
UCS-A /license # show file detail

License file: UCSFEAT20100928112305377.lic
    Id: 1212121212121212
    Version: 1.0
    Scope: A
    State: Installed
```

```
        Features
        Feature Name: ETH_PORT_ACTIVATION_PKG
        Vendor: cisco
        Version: 1.0
        Quantity: 24
        Lines
            Line Id: 1
            Type: Increment
            Expiry Date: Never
            Pak:
            Quantity: 24
            Signature: B10101010101

License file: UCSFEAT20100928112332175.lic
        Id: 1313131313131313
        Version: 1.0
        Scope: B
        State: Installed
        Features
        Feature Name: ETH_PORT_ACTIVATION_PKG
        Vendor: cisco
        Version: 1.0
        Quantity: 24
        Lines
            Line Id: 1
            Type: Increment
            Expiry Date: Never
            Pak:
            Quantity: 24
            Signature: F302020202020

UCS-A /license #
```

Viewing License Usage for a Fabric Interconnect

Procedure

	Command or Action	**Purpose**
Step 1	UCS-A# **scope license**	Enters license mode.
Step 2	UCS-A /license # **show usage**	Displays the license usage table for all license files installed on the fabric interconnect.
		This following are included:
		• **Feat Name**
		The name of the feature to which the license applies.
		• **Scope**
		The fabric associated with the license.
		• **Default**
		The default number of licenses provided for this Cisco UCS domain.

Command or Action	Purpose

	• **Total Quant**
	
	The total number of licenses available. This value is the sum of the number of default licenses plus the number of purchased licenses.
	
	• **Used Quant**
	
	The number of licenses currently being used by the system. If this value exceeds the total number of licenses available, then some ports will stop functioning after their associated grace period expires.
	
	• **Subordinate Quant**
	
	C-Series Rack Servers that are currently being used by the system.
	
	• **State**
	
	The operational state of the license.
	
	• **Peer Count Comparison**
	
	The number of licenses on the peer fabric interconnect compared to this fabric interconnect. This can be one of the following:
	
	   • **exceeds**—the peer fabric interconnect has more licenses installed than this fabric interconnect
	
	   • **lacks**—the peer fabric interconnect has fewer licenses installed than this fabric interconnect
	
	   • **matching**—the same number of licenses are installed on both fabric interconnects
	
	• **Grace Used**
	
	The amount of time (in seconds) used in the grace period. After the grace period ends, Cisco UCS sends alert messages until a new license is purchased.

Example

The following examples display full details of the licenses installed on a fabric interconnect:

```
UCS-A# scope license
UCS-A /license # show usage
Feat Name                    Scope Default Total Quant Used Quant Subordinate Quant
State                Peer Count Comparison  Grace Used
---------------------------- ----- ------- ----------- ---------- --------------------
-------------------- ---------------------- ----------
ETH_PORT_ACTIVATION_PKG      A     20      48          12                           0
License Ok       Matching                              0
ETH_PORT_C_ACTIVATION_PKG    A     0       0           0                            0
Not Applicable   Matching                              0
ETH_PORT_ACTIVATION_PKG      B     20      48          11                           0
License Ok       Matching                              0
ETH_PORT_C_ACTIVATION_PKG    B     0       0           0                            0
Not Applicable   Matching                              0
UCS-A /license #

UCS-A# scope license
UCS-A /license # show feature

License feature:
    Name                     Vendor Version Type              Grace Period
    ------------------------ ------ ------- ----------------- ------------
    ETH_PORT_ACTIVATION_PKG  cisco  1.0     Counted                    120
    ETH_PORT_C_ACTIVATION_PKG cisco 1.0     Counted                    120
UCS-A /license #
```

Uninstalling a License

Note
Permanent licenses cannot be uninstalled if they are in use. You can only uninstall a permanent license that is not in use. If you try to delete a permanent license that is being used, Cisco UCS Manager rejects the request and display an error message.

Before you begin

Back up the Cisco UCS Manager configuration.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope license**	Enters license mode.
Step 2	UCS-A /license # **clear file** *license-filename*	Uninstalls the specified license.

Cisco UCS Manager deactivates the license, removes the license from the list of licenses, and deletes the license from the fabric interconnect. The port is moved into unlicensed mode. In a cluster setup, you must uninstall the license from the other fabric interconnect.

Example

The following example shows the uninstallation of port9.lic:

```
UCS-A # scope license
UCS-A /license # clear file port9.lic
Clearing license port9.lic:
SERVER this_host ANY
VENDOR cisco
INCREMENT ETH_PORT_ACTIVATION_PKG cisco 1.0 permanent 1 \
        VENDOR_STRING=<LIC_SOURCE>UCS_SWIFT</LIC_SOURCE><SKU>N10-L001=</SKU> \
        HOSTID=VDH=FLC12360025 \
        NOTICE="<LicFileID>20090519200954833</LicFileID><LicLineID>1</LicLineID> \
        <PAK></PAK>" SIGN=C01FAE4E87FA

Clearing license ........done
UCS-A /license #
```

CHAPTER 4

Registering Cisco UCS Domains with Cisco UCS Central

Registration of Cisco UCS Domains

You can have Cisco UCS Central manage some or all of the Cisco UCS domains in your data center.

If you want Cisco UCS Central to manage a Cisco UCS domain, you need to register that domain. When you register, you must choose which types of policies and other configurations will be managed by Cisco UCS Central and Cisco UCS Manager. Cisco UCS Central can manage the same types of policies and configurations for all registered Cisco UCS domains. You can also choose to have different settings for each registered Cisco UCS domain.

Perform the following before registering a Cisco UCS domain with Cisco UCS Central:

- Configure an NTP server and the correct time zone in both Cisco UCS Manager and Cisco UCS Central to ensure that they are in sync. If the time and date in the Cisco UCS domain and Cisco UCS Central are out of sync, the registration might fail.

- Obtain the hostname or IP address of Cisco UCS Central

- Obtain the shared secret that was configured when Cisco UCS Central was deployed.

Policy Resolution between Cisco UCS Manager and Cisco UCS Central

For each Cisco UCS domain that you register with Cisco UCS Central, you can choose which application will manage certain policies and configuration settings. This policy resolution does not have to be the same for every Cisco UCS domain that you register with the same Cisco UCS Central.

> **Note** Unregistering a Cisco UCS domain with Cisco UCS Central will terminate all open sessions.

You have the following options for resolving these policies and configuration settings:

- **Local**—The policy or configuration is determined and managed by Cisco UCS Manager.

- **Global**—The policy or configuration is determined and managed by Cisco UCS Central.

The following table contains a list of the policies and configuration settings that you can choose to have managed by either Cisco UCS Manager or Cisco UCS Central:

Name	Description
Infrastructure & Catalog Firmware	Determines whether the Capability Catalog and infrastructure firmware policy are defined locally or come from Cisco UCS Central.
Time Zone Management	Determines whether the date and time is defined locally or comes from Cisco UCS Central.
Communication Services	Determines whether HTTP, CIM XML, Telnet, SNMP, web session limits, and Management Interfaces Monitoring Policy settings are defined locally or in Cisco UCS Central.
Global Fault Policy	Determines whether the Global Fault Policy is defined locally or in Cisco UCS Central.
User Management	Determines whether authentication and native domains, LDAP, RADIUS, TACACS+, trusted points, locales, and user roles are defined locally or in Cisco UCS Central.
DNS Management	Determines whether DNS servers are defined locally or in Cisco UCS Central.
Backup & Export Policies	Determines whether the Full State Backup Policy and All Configuration Export Policy are defined locally or in Cisco UCS Central.
Monitoring	Determines whether Call Home, Syslog, and TFTP Core Exporter settings are defined locally or in Cisco UCS Central.
SEL Policy	Determines whether managed endpoints are defined locally or in Cisco UCS Central.
Power Management	Determines whether the power management is defined locally or in Cisco UCS Central.
Power Supply Unit	Determines whether power supply units are defined locally or in Cisco UCS Central.
Port Configuration	Determines whether port configuration is defined locally or in Cisco UCS Central.

Registering a Cisco UCS Domain with Cisco UCS Central

Before you begin

Configure an NTP server and the correct time zone in both Cisco UCS Manager and Cisco UCS Central to ensure that they are in sync. If the time and date in the Cisco UCS domain and Cisco UCS Central are out of sync, the registration might fail.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system**	Enters system mode.
Step 2	UCS-A/system # **create control-ep policy** *ucs-central*	Creates the policy required to register the Cisco UCS Domain with Cisco UCS Central.
		ucs-central can be the hostname or IP address of the virtual machine where Cisco UCS Central is deployed.
		Note If you use a hostname rather than an IPv4 or IPv6 address, you must configure a DNS server. If the Cisco UCS domain is not registered with Cisco UCS Central or DNS management is set to **local**, configure a DNS server in Cisco UCS Manager. If the Cisco UCS domain is registered with Cisco UCS Central and DNS management is set to **global**, configure a DNS server in Cisco UCS Central.
Step 3	Shared Secret for Registration: *shared-secret*	Enter the shared secret (or password) that was configured when Cisco UCS Central was deployed.
Step 4	UCS-A/system/control-ep # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example registers a Cisco UCS Domain with a Cisco UCS Central system at IP address 209.165.200.233, and commits the transaction:

```
UCS-A# scope system
UCS-A /system # create control-ep policy 209.165.200.233
Shared Secret for Registration: S3cretW0rd!
```

```
UCS-A /system/control-ep* # commit-buffer
UCS-A /system/control-ep #
```

What to do next

Configure policy resolution between Cisco UCS Manager and Cisco UCS Central.

Configuring Policy Resolution between Cisco UCS Manager and Cisco UCS Central

Before you begin

You must register the Cisco UCS Domain with Cisco UCS Central before you can configure policy resolution.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system**	Enters system mode.
Step 2	UCS-A/system # **scope control-ep policy**	Enters control-ep policy mode.
Step 3	UCS-A/system/control-ep # **set backup-policy-ctrl source** {**local** | **global**}	Determines whether the Full State Backup Policy and All Configuration Export Policy are defined locally or in Cisco UCS Central.
Step 4	UCS-A/system/control-ep # **set communication-policy-ctrl source** {**local** | **global**}	Determines whether HTTP, CIM XML, Telnet, SNMP, web session limits, and Management Interfaces Monitoring Policy settings are defined locally or in Cisco UCS Central.
Step 5	UCS-A/system/control-ep # **set datetime-policy-ctrl source** {**local** | **global**}	Determines whether the date and time is defined locally or comes from Cisco UCS Central.
Step 6	UCS-A/system/control-ep # **set dns-policy-ctrl source** {**local** | **global**}	Determines whether DNS servers are defined locally or in Cisco UCS Central.
Step 7	UCS-A/system/control-ep # **set fault-policy-ctrl source** {**local** | **global**}	Determines whether the Global Fault Policy is defined locally or in Cisco UCS Central.
Step 8	UCS-A/system/control-ep # **set infra-pack-ctrl source** {**local** | **global**}	Determines whether the Capability Catalog and infrastructure firmware policy are defined locally or come from Cisco UCS Central.
Step 9	UCS-A/system/control-ep # **set mep-policy-ctrl source** {**local** | **global**}	Determines whether managed endpoints are defined locally or in Cisco UCS Central.
Step 10	UCS-A/system/control-ep # **set monitoring-policy-ctrl source** {**local** | **global**}	Determines whether Call Home, Syslog, and TFTP Core Exporter settings are defined locally or in Cisco UCS Central.

	Command or Action	**Purpose**
Step 11	UCS-A/system/control-ep # **set powermgmt-policy-ctrl source** {**local** | **global**}	Determines whether the power management is defined locally or in Cisco UCS Central.
Step 12	UCS-A/system/control-ep # **set psu-policy-ctrl source** {**local** | **global**}	Determines whether power supply units are defined locally or in Cisco UCS Central.
Step 13	UCS-A/system/control-ep # **set security-policy-ctrl source** {**local** | **global**}	Determines whether authentication and native domains, LDAP, RADIUS, TACACS+, trusted points, locales, and user roles are defined locally or in Cisco UCS Central.
Step 14	UCS-A/system/control-ep # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example configures policy resolution for a Cisco UCS Domain that is registered with Cisco UCS Central and commits the transaction:

```
UCS-A# scope system
UCS-A /system # scope control-ep policy
UCS-A /system/control-ep* # set backup-policy-ctrl source global
UCS-A /system/control-ep* # set communication-policy-ctrl source local
UCS-A /system/control-ep* # set datetime-policy-ctrl source global
UCS-A /system/control-ep* # set dns-policy-ctrl source global
UCS-A /system/control-ep* # set fault-policy-ctrl source global
UCS-A /system/control-ep* # set infra-pack-ctrl source global
UCS-A /system/control-ep* # set mep-policy-ctrl source global
UCS-A /system/control-ep* # set monitoring-policy-ctrl source global
UCS-A /system/control-ep* # set powermgmt-policy-ctrl source global
UCS-A /system/control-ep* # set psu-policy-ctrl source local
UCS-A /system/control-ep* # set security-policy-ctrl source global
UCS-A /system/control-ep* # commit-buffer
UCS-A /system/control-ep #
```

Setting Cisco UCS Central Registration Properties in Cisco UCS Manager

Procedure

	Command or Action	**Purpose**
Step 1	UCS-A# **scope system**	Enters system mode.
Step 2	UCS-A /system # **scope control-ep policy**	Enters the registration policy.
Step 3	UCS-A /system/control-ep # **set cleanupmode** {|}	This can be one of the following:

	Command or Action	Purpose
		• **Localize Global**—When a Cisco UCS domain is unregistered, all global policies in the Cisco UCS domain will be localized to Cisco UCS Manager. The policies remain in the Cisco UCS domain, policy ownership is now local to Cisco UCS Manager, and Cisco UCS Manager admin users can make changes.
		Note If you reregister the Cisco UCS domain with Cisco UCS Central, there can be policy conflicts due to the policies existing both in Cisco UCS Central and in Cisco UCS Manager. Either delete the local policies, or set the local policies to global before you try to create and associate a global service profile.
		• **Deep Remove Global**—This option should only be used after careful consideration. When a Cisco UCS domain is unregistered, all global policies in the Cisco UCS domain are removed. If there are global service profiles, they will now refer to Cisco UCS Manager local default policies, and one of the following occurs:
		• If there are local default policies present, the server will reboot.
		• If there are no local default policies, the service profile association fails with a configuration error.
		Note The deep remove global cleanup mode does not remove global VSANs and VLANs when you unregister from Cisco UCS Central. Those must be removed manually if desired.
Step 4	UCS-A /system/control-ep # **set suspendstate on**	Sets the suspend state. If set automatically, the Cisco UCS domain is temporarily removed from Cisco UCS Central, and all global policies revert to their local counterparts. All service profiles maintain their current identities. However, global pools are no longer visible and

	Command or Action	**Purpose**
		cannot be accessible by new service profiles. To turn off suspend state, you need to acknowledge the situation.
Step 5	UCS-A /system/control-ep # **set ackstate acked**	Acknowledges that inconsistencies exist between Cisco UCS Manager and Cisco UCS Central and that you are still willing to reconnect the Cisco UCS domain with Cisco UCS Central. This automatically turns off suspend state.
Step 6	UCS-A /system/control-ep # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to change the Cisco UCS Central registration cleanup mode to deep-remove-global and commit the transaction:

```
UCS-A# scope system
UCS-A /system # scope control-ep policy
UCS-A /system/control-ep* # set cleanupmode deep-remove-global
UCS-A /system/control-ep* # commit-buffer
UCS-A /system/control-ep #
```

Unregistering a Cisco UCS Domain from Cisco UCS Central

When you unregister a Cisco UCS domain from Cisco UCS Central, Cisco UCS Manager no longer receives updates to global policies.

Procedure

	Command or Action	**Purpose**
Step 1	UCS-A# **scope system**	Enters system mode.
Step 2	UCS-A/system # **delete control-ep policy**	Deletes the policy and unregisters the Cisco UCS Domain from Cisco UCS Central.
Step 3	UCS-A/system # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example unregisters a Cisco UCS Domain from Cisco UCS Central and commits the transaction:

```
UCS-A# scope system
UCS-A /system # delete control-ep policy
UCS-A /system* # commit-buffer
UCS-A /system #
```

C H A P T E R 5

Power Capping and 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 blade servers, UCS C220 and C240 M4/M5, and C480 M5/C480 M5 ML rack servers, UCS Mini, and mixed UCS domains.

UCS Manager supports power capping on the following:

• UCS 6200 Series Fabric Interconnects

• 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.

	Command or Action	Purpose
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	Required: UCS-A /org/psu-policy # **commit-buffer**	Commits the transaction to the system configuration.

Example

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

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

Starting with Cisco UCS Manager 3.2(2), chassis dynamic power rebalance mechanism is enabled by default. The mechanism continuously monitors the power usage of the blade servers and adjusts the power allocation accordingly. Chassis dynamic power rebalance mechanism operates within the overall chassis power budget set by Cisco UCS Manager, which is calculated from the available PSU power and Group power.

For mission-critical application a special priority called **no-cap** is also available. Setting the priority to **no-cap** does not guarantee that a blade server gets maximum power all the time, however, it prioritizes the blade server over other servers during the chassis dynamic power rebalance budget allocations.

Note If all the blade servers are set with no-cap priority and all of them run high power consuming loads, then there is a chance that some of the blade servers get capped under high power usage, based on the power distribution done through dynamic balance.

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 fanspeed** {*any* | **balanced**|*high-power*|*low-power*|*max-power*|*performance*}	Specifies the fan speed for the power control policy. **Note** For Cisco UCS C-Series M5 servers, if you select **performance** for this **fanspeed**, ensure that you do not select **no-cap** for **priority**. Associating a service profile with a server fails if you select **performance** for **fanspeed** policy, and **no-cap** for the **priority**.
Step 4	UCS-A /org/power-control-policy # **set priority** {*priority-num* | **no-cap**}	Specifies the priority for the power control policy. **Note** For Cisco UCS C-Series M5 servers, if you select **no-cap** for **priority**, ensure that you do not select **performance** for **fanspeed** policy. Associating a service profile with a server fails if you select **performance** for **fanspeed** policy, and **no-cap** for the **priority**.

	Command or Action	Purpose
Step 5	UCS-A /org/power-control-policy # **commit-buffer**	Commits the transaction to the system configuration.

Example

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

Example

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.

Example

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.

	Command or Action	Purpose
Step 2	UCS-A /power-cap-mgmt # **delete power-group** *power-group-name*	Deletes the specified power group.
Step 3	UCS-A /power-cap-mgmt/power-group/chassis # **commit-buffer**	Commits the transaction to the system configuration.

Example

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.

Example

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

Example

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.

Example

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.

Example

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.

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

Example

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 and if there is not enough power available for allocation, the following fault is raised:

```
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.

Example

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.
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.

Example

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.

	Command or Action	**Purpose**
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	(Optional) UCS-A /org/power-sync-policy* # **set descr** *optionall-description*	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 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.

Example

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.

Example

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.

Example

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	(Optional) UCS-A /org/service-profile/power-sync-definition* # **set** **descr** *optional-description*	Specifies the description of the power-sync-policy. You can also change the description using the descr keyword.

	Command or Action	Purpose
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.

Example

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.
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	(Optional) UCS-A /org/service-profile # **show power-sync-policy** { **detail**	**expand**	**detail expand** }	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.

Example

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.
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.

Example

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 supported for following rack servers:

- Cisco UCS C220 M4 Server

- Cisco UCS C240 M4 Server

- Cisco UCS C220 M5 Server

- Cisco UCS C240 M5 Server

- Cisco UCS C480 M5 Server

- Cisco UCS C480 M5 ML Server

Power capping is not supported for Cisco UCS C125 M5 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.

Blade Server 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.
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.

Example

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.

Example

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 #
```

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.

Example

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.

Example

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

Example

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

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.

Example

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 54.

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.

Example

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.

Example

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.

Example

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.

Example

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 CIMC reboots 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.

Example

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.

Example

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.

Example

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.

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.

Example

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 #
```

Rack-Mount Server 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.

Rack-Enclosure Server Management

Beginning with release 4.0(1a), Cisco UCS Manager extends support for all existing features on Cisco UCS C125 M5 Servers unless specifically noted in this guide.

Cisco UCS C125 M5 Servers are housed in the Cisco UCS C4200 Series Rack Server Chassis. Each Cisco UCS C4200 Series Rack Server Chassis supports up to four Cisco UCS C125 M5 Server nodes. To manage the Cisco UCS C125 M5 Server nodes, Cisco UCS Manager supports **rack-enclosure** object in CLI.

Rack enclosures can be scoped using the CLI interface. For example:

```
UCS-A # scope rack-enclosure 1
```

You can scope **rack-enclosure** for the following:

- `fan-module`
- `psu`
- `slot`

`fan-module` and `psu` can be managed the same way as other rack servers. For `slot`, see Viewing Rack Enclosure Slot Statistics, on page 79.

You can also use the **show** command to view the following in **rack-enclosure**:

- `detail`
- `event`
- `expand`
- `fan-module`
- `fault`
- `fsm`
- `psu`
- `slot`
- `stats`

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.

Example

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.

Example

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

Example

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.

Example

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.

Example

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.

Example

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.

Example

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 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.

Example

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.

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

Example

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 71.

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

Example

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

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

Turning Off the Locator LED for a Rack-Mount Server

Procedure

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

Example

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

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

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

Example

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 CIMC reboots 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.

Example

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.

	Command or Action	**Purpose**
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.

Example

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.

Example

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

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

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.

Example

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.

Example

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
```

Viewing Rack Enclosure Slot Statistics

You can see the stats for server slot in the rack enclosure housing the C125 M5 Servers.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope rack-enclosure** *rack-enclosure-num*	Enters the rack-enclosure.
Step 2	UCS-A# /rack-enclosure # **show slot**	Displays the slot stats.
Step 3	UCS-A# /rack-enclosure # **scope slot** *slot_ID*	Enters the slot.
Step 4	UCS-A# /rack-enclosure/slot # **show detail**	Displays the following stats: • Id • Slot Type • Presence State • Server ID • Server DN • Current Task

Example

The following example shows how to view slot stats in for an enclosure and individual slot stats:

```
UCS-A# scope rack-enclosure 1
UCS-A /rack-enclosure # show slot
UCS-A /rack-enclosure # show slot

Slot:
    Id        Presence State
    ---------- --------------
             1 Equipped
             2 Empty
             3 Equipped
             4 Empty
UCS-A /rack-enclosure # scope slot 1
UCS-A /rack-enclosure/slot # show detail

Slot:
    Id: 1
    Slot Type: Compute
    Presence State: Equipped
    Server ID: 4
```

```
        Server DN: sys/rack-unit-4
        Current Task:
UCS-A /rack-enclosure/slot #
```

S3X60 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.

Example

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.

Example

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.

Example

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*.
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.

Example

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.

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

Example

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.
			
			• **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.

Example

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.
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.

Example

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 82.

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.

Example

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.
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.

Example

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.

Example

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.

Example

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.
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.

Example

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 CIMC reboots 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.

Example

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.

	Command or Action	**Purpose**
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.

Example

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.

Example

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.

Example

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.

Example

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.

Example

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 #
```

Server Boot Configuration

Boot Policy

The Cisco UCS Manager enables you to create a boot policy for blade servers and rack servers.

The Cisco UCS Manager boot policy overrides the boot order in the BIOS setup menu and determines the following:

> • Selection of the boot device
>
> • Location from which the server boots
>
> • Order in which boot devices are invoked

For example, you can have associated servers boot from a local device, such as a local disk or CD-ROM (VMedia), or you can select a SAN boot or a LAN (PXE) boot.

You can either create a named boot policy to associate with one or more service profiles, or create a boot policy for a specific service profile. A boot policy must be included in a service profile, and that service profile must be associated with a server for it to take effect. If you do not include a boot policy in a service profile, Cisco UCS Manager applies the default boot policy.

Note   Changes to a boot policy might be propagated to all servers created with an updating service profile template that includes that boot policy. Re-association of the service profile with the server to rewrite the boot order information in the BIOS is automatically triggered.

You can also specify the following for the boot policy:

- Local LUN name. The name specified is the logical name in the storage profile, not the deployed name. Specify only a primary name. Specifying a secondary name results in a configuration error.

- Specific JBOD disk number for booting from JBOD disks.

- Any LUN for backward compatibility; however, we do not recommend this. Other devices must not have bootable images to ensure a successful boot.

UEFI Boot Mode

Unified Extensible Firmware Interface (UEFI) is a specification that defines a software interface between an operating system and platform firmware. Cisco UCS Manager uses UEFI to replace the BIOS firmware interfaces. This allows the BIOS to run in UEFI mode while still providing legacy support.

You can choose either legacy or UEFI boot mode when you create a boot policy. Legacy boot mode is supported for all Cisco UCS servers except Cisco UCS C125 M5 Server. UEFI boot mode is supported only on M3 and higher servers, and allows you to enable UEFI secure boot mode. Cisco UCS C125 M5 Server supports only UEFI boot mode.

UEFI PXE boot is supported with all Cisco VIC adapters on Cisco UCS rack servers integrated with Cisco UCS Manager Release 2.2(4) and later releases. Beginning with Cisco UCS Manager Release 2.2(1), UEFI PXE boot is supported on all Cisco blade servers.

The following limitations apply to the UEFI boot mode:

- UEFI boot mode is not supported with the following combinations:

    - Gen-3 Emulex and QLogic adapters on Cisco UCS blade and rack servers integrated with Cisco UCS Manager.

    - iSCSI boot for all adapters on Cisco UCS rack servers integrated with Cisco UCS Manager.

- If you want to use UEFI boot mode with two iSCSI LUNs, you must manually specify a common iSCSI initiator name in the service profile that is applied to both underlying iSCSI eNICs rather than allowing Cisco UCS Manager to select the name from an IQN suffix pool. If you do not supply a common name, Cisco UCS Manager will not be able to detect the second iSCSI LUN.

- You cannot mix UEFI and legacy boot mode on the same server.

- The server will boot correctly in UEFI mode only if the boot devices configured in the boot policy have UEFI-aware operating systems installed. If a compatible OS is not present, the boot device is not displayed on the **Actual Boot Order** tab in the **Boot Order Details** area.

- In some corner cases, the UEFI boot may not succeed because the UEFI boot manager entry was not saved correctly in the BIOS NVRAM. You can use the UEFI shell to enter the UEFI boot manager entry manually. This situation could occur in the following situations:

- If a blade server with UEFI boot mode enabled is disassociated from the service profile, and the blade is manually powered on using the **Equipment** tab or the front panel.

- If a blade server with UEFI boot mode enabled is disassociated from the service profile, and a direct VIC firmware upgrade is attempted.

- If a blade or rack server with UEFI boot mode enabled is booted off SAN LUN, and the service profile is migrated.

You can create UEFI boot parameters in Cisco UCS Manager. provides more information.

UEFI Secure Boot

Cisco UCS Manager supports UEFI secure boot on Cisco UCS B-Series M3 and higher Blade servers, Cisco UCS C-Series M3 and higher Rack servers, and Cisco UCS S-Series M4 Rack servers, and Cisco UCS C125 M5 Servers. Linux secure boot is supported on SLES 15, SLES 13 SP4, Red Hat Linux 7.6 operating systems starting with Release 4.0(4a). When UEFI secure boot is enabled, all executables, such as boot loaders and adapter drivers, are authenticated by the BIOS before they can be loaded. To be authenticated, the images must be signed by either the Cisco Certificate Authority (CA) or a Microsoft CA.

The following limitations apply to UEFI secure boot:

- UEFI boot mode must be enabled in the boot policy.

- UEFI boot mode is available only for drives.

- The Cisco UCS Manager software and the BIOS firmware must be at Release 2.2 or greater.

> **Note** UEFI boot mode is supported on Cisco UCS C-Series and S-Series rack servers beginning with Release 2.2(3a).

- User-generated encryption keys are not supported.

- UEFI secure boot can only be controlled by Cisco UCS Manager.

- If you want to downgrade to an earlier version of Cisco UCS Manager, and you have a server in secure boot mode, you must disassociate, then re-associate the server before downgrading. Otherwise, server discovery is not successful.

CIMC Secure Boot

With CIMC secure boot, only Cisco signed firmware images can be installed and run on the servers. When the CIMC 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 CIMC firmware.

Guidelines and Limitations for CIMC Secure Boot

- CIMC secure boot is supported on Cisco UCS M3 rack servers.

| Note | CIMC secure boot is enabled by default on the Cisco UCS C220 M4/M5, C240 M4/M5, and C480 M5/C480 M5 ML rack servers, and is automatically enabled on the Cisco UCS C460 M4 rack server after upgrading to CIMC firmware release 2.2(3) or higher. |

- After CIMC secure boot is enabled, you cannot disable it.

- After CIMC secure boot is enabled on a server, you cannot downgrade to a CIMC firmware image prior to 2.1(3).

Determining the CIMC Secure Boot Status

Procedure

	Command or Action	**Purpose**
Step 1	UCS-A# **scope server** *server-num*	Enters server mode for the specified server.
Step 2	UCS-A /chassis/server # **scope cimc**	Enters server CIMC mode.
Step 3	UCS-A /server/cimc # **show secure-boot**	Displays the CIMC secure boot status for the specified server. This can be one of the following: • **Unsupported**—CIMC secure boot is not supported on the server. • **Disabled**—CIMC secure boot is supported, but is disabled on the server. • **Enabling**—CIMC secure boot has been enabled, and the operation is in process. • **Enabled**—CIMC secure boot is enabled on the server.

Example

The following example shows how to display the CIMC secure boot status:

```
UCS-A# scope server 1
UCS-A /chassis/server # scope cimc
UCS-A /chassis/server/cimc # show secure-boot
Secure Boot: Disabled
UCS-A /chassis/server/cimc #
```

Enabling CIMC Secure Boot

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *server-num*	Enters server mode for the specified server.
Step 2	UCS-A /chassis/server # **scope cimc**	Enters server CIMC mode.
Step 3	UCS-A /server/cimc # **enable secure-boot**	Enables CIMC secure boot status for the specified server. CIMC secure boot is only supported on Cisco UCS M3 rack servers. **Note** Once enabled, CIMC secure boot cannot be disabled.
Step 4	UCS-A /server/cimc # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to enable CIMC secure boot and commit the transaction:

```
UCS-A# scope server 1
UCS-A /chassis/server # scope cimc
UCS-A /chassis/server/cimc # enable secure-boot
Warning: When committed, CIMC Secure Boot and Installation Feature will be enabled for the
 server.
This is an irreversible operation!!

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

Creating a Boot Policy

You can also create a local boot policy that is restricted to a service profile or service profile template. However, Cisco recommends that you create a global boot policy that can be included in multiple service profiles or service profile templates.

Before you begin

If you are creating a boot policy that boots the server from a SAN LUN and you require reliable SAN boot operations, you must first remove all local disks from servers associated with a service profile that includes the boot policy.

Note   The following example shows how to create a boot policy named boot-policy-LAN, specify that servers using this policy will not be automatically rebooted when the boot order is changed, set the UEFI boot mode, enable UEFI boot security, and commit the transaction:

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 boot-policy** *policy-name* [**purpose** {**operational**	**utility**}]	Creates a boot policy with the specified policy name, and enters organization boot policy mode. When you create the boot policy, specify the **operational** option. This ensures that the server boots from the operating system installed on the server. The **utility** options is reserved and should only be used if instructed to do so by a Cisco representative.
Step 3	(Optional) UCS-A /org/boot-policy # **set descr** *description*	Provides a description for the boot policy. **Note** If your description includes spaces, special characters, or punctuation, you must begin and end your description with quotation marks. The quotation marks do not appear in the description field of any **show** command output.
Step 4	UCS-A /org/boot-policy # **set reboot-on-update** {**no**	**yes**}	Specifies whether the servers using this boot policy are automatically rebooted after you make changes to the boot order.
Step 5	UCS-A /org/boot-policy # **set enforce-vnic-name** {**no**	**yes**}	If you choose **yes**, Cisco UCS Manager displays a configuration error and reports whether one or more of the vNICs, vHBAs, or iSCSI vNICs listed in the **Boot Order** table match the server configuration in the service profile. If you choose **no**, Cisco UCS Manager uses the vNICs, vHBAs, or iSCSI vNICs (as appropriate for the boot option) from the service profile.
Step 6	UCS-A /org/boot-policy # **set boot-mode** {**legacy**	**uefi**}	Specifies whether the servers using this boot policy are using UEFI or legacy boot mode. **Note** Cisco UCS C125 M5 Servers support only UEFI boot mode.
Step 7	UCS-A /org/boot-policy # **commit-buffer**	Commits the transaction to the system configuration.

	Command or Action	Purpose
Step 8	UCS-A /org/boot-policy # **create boot-security**	Enters boot security mode for the specified boot policy.
Step 9	UCS-A /org/boot-policy/boot-security # **set secure-boot** {**no**	**yes**}	Specifies whether secure boot is enabled for the boot policy.
Step 10	UCS-A /org/boot-policy/boot-security # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create a boot policy named boot-policy-LAN, specify that servers using this policy will not be automatically rebooted when the boot order is changed, set the UEFI boot mode, enable UEFI boot security, and commit the transaction:

```
UCS-A# scope org /
UCS-A /org* # create boot-policy boot-policy-LAN purpose operational
UCS-A /org/boot-policy* # set descr "Boot policy that boots from the LAN."
UCS-A /org/boot-policy* # set reboot-on-update no
UCS-A /org/boot-policy* # set boot-mode uefi
UCS-A /org/boot-policy* # commit-buffer
UCS-A /org/boot-policy # create boot-security
UCS-A /org/boot-policy/boot-security* # set secure-boot yes
UCS-A /org/boot-policy/boot-security* # commit-buffer
UCS-A /org/boot-policy/boot-security #
```

What to do next

Configure one or more of the following boot options for the boot policy and set their boot order:

- **LAN Boot** —Boots from a centralized provisioning server. It is frequently used to install operating systems on a server from that server. If you choose the LAN Boot option, continue to .

- **SAN Boot** —Boots from an operating system image on the SAN. You can specify a primary and a secondary SAN boot. If the primary boot fails, the server attempts to boot from the secondary.

  We recommend that you use a SAN boot policy, because it offers the most service profile mobility within the system. If you boot from the SAN, when you move a service profile from one server to another, the new server boots from exactly the same operating system image. Therefore, the new server appears to be exactly the same server to the network.

  If you choose the SAN Boot option, continue to .

- **Virtual Media Boot** —Mimics the insertion of a physical CD into a server. It is typically used to manually install operating systems on a server.

  If you choose the Virtual Media boot option, continue to .

- **NVMe Boot** —BIOS enumerates the NVMe devices present and boots to the first NVMe device having UEFI capable OS installed on it.

If you choose the NVMe boot option, continue to

- Local Devices boot—To boot from local devices, such as local disks on the server, virtual media, or remote virtual disks, continue with

Tip If you configure a local disk and a SAN LUN for the boot order storage type and the operating system or logical volume manager (LVM) is configured incorrectly, the server might boot from the local disk rather than the SAN LUN.

For example, on a server with Red Hat Linux installed, where the LVM is configured with default LV names and the boot order is configured with a SAN LUN and a local disk, Linux reports that there are two LVs with the same name and boots from the LV with the lowest SCSI ID, which could be the local disk.

Include the boot policy in a service profile and template.

SAN Boot

You can configure a boot policy to boot one or more servers from an operating system image on the SAN. The boot policy can include a primary and a secondary SAN boot. If the primary boot fails, the server attempts to boot from the secondary.

Cisco recommends using a SAN boot, because it offers the most service profile mobility within the system. If you boot from the SAN when you move a service profile from one server to another, the new server boots from the same operating system image. Therefore, the new server appears as the same server to the network.

To use a SAN boot, ensure that the following is configured:

- The Cisco UCS domain must be able to communicate with the SAN storage device that hosts the operating system image.
- A boot target LUN (Logical Unit Number) on the device where the operating system image is located.

Note SAN boot is not supported on Gen-3 Emulex adapters on Cisco UCS blade and rack servers.

Configuring a SAN Boot for a Boot Policy

Tip If you configure a local disk and a SAN LUN for the boot order storage type and the operating system or logical volume manager (LVM) is configured incorrectly, the server might boot from the local disk rather than the SAN LUN.

For example, on a server with Red Hat Linux installed, where the LVM is configured with default LV names and the boot order is configured with a SAN LUN and a local disk, Linux reports that there are two LVs with the same name and boots from the LV with the lowest SCSI ID, which could be the local disk.

This procedure continues directly from Creating a Boot Policy, on page 99.

Before you begin

Create a boot policy to contain the SAN boot configuration.

Note	If you are creating a boot policy that boots the server from a SAN LUN and you require reliable SAN boot operations, Cisco recommends that you first remove all local disks and other SAN LUNs from the boot policy in the server service profile. This does not apply to the Cisco UCS Mini Series.

Beginning with Release 2.2, all SAN boot-related CLI commands have been moved to the SAN scope. Any existing scripts from previous releases that use SAN boot under the storage scope instead of **org/boot-policy/san** or **org/service-profile/boot-definition/san** should be updated.

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 boot-policy** *policy-name*	Enters organization boot policy mode for the specified boot policy.
Step 3	UCS-A /org/boot-policy # **create san**	Creates a SAN boot for the boot policy and enters organization boot policy storage mode.
Step 4	UCS-A /org/boot-policy/san # **set order** *order_number*	Sets the boot order for the SAN boot. Enter an integer between 1 and 16.
Step 5	UCS-A /org/boot-policy/san # **create san-image** {**primary** | **secondary**}	Creates a SAN image location, and if the san-image option is specified, enters organization boot policy storage SAN image mode. When using the enhanced boot order on Cisco UCS M3 servers, or M4 servers, the boot order that you define is used. For standard boot mode using the terms "primary" or "secondary" do not imply a boot order. The effective order of boot devices within the same device class is determined by the PCIe bus scan order.
Step 6	UCS-A /org/boot-policy/ssn/san-image # **set vhba** *vhba-name*	Specifies the vHBA to be used for the SAN boot.
Step 7	UCS-A /org/boot-policy/san/san-image # **create path** {**primary** | **secondary**}	Creates a primary or secondary SAN boot path and enters organization boot policy SAN path mode.

	Command or Action	Purpose
		When using the enhanced boot order on Cisco UCS M3 servers, or M4 servers, the boot order that you define is used. For standard boot mode using the terms "primary" or "secondary" do not imply a boot order. The effective order of boot devices within the same device class is determined by the PCIe bus scan order.
Step 8	UCS-A /org/boot-policy/san/san-image/path # **set** {**lun** *lun-id* | **wwn** *wwn-num*}	Specifies the LUN or WWN to be used for the SAN path to the boot image.
Step 9	UCS-A /org/boot-policy/san/san-image/path # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to enter the boot policy named lab1-boot-policy, create a SAN boot for the policy, set the boot order to 1, create a primary SAN image, use a vHBA named vHBA2, create primary path using LUN 0, and commit the transaction:

```
UCS-A# scope org /
UCS-A /org* # scope boot-policy lab1-boot-policy
UCS-A /org/boot-policy # create san
UCS-A /org/boot-policy/san* # set order 1
UCS-A /org/boot-policy/san* # create san-image primary
UCS-A /org/boot-policy/san/san-image* # set vhba vHBA2
UCS-A /org/boot-policy/san/san-image* # create path primary
UCS-A /org/boot-policy/san/san-image/path* # set lun 0
UCS-A /org/boot-policy/san/san-image/path* # commit-buffer
UCS-A /org/boot-policy/san/san-image/path #
```

The following example shows how to create a SAN boot for the service profile SP_lab1, set the boot order to 1, create a primary SAN image, use a vHBA named vHBA2, create primary path using LUN 0, and commit the transaction:

```
UCS-A# scope org /
UCS-A /org* # scope service-profile SP_lab1
UCS-A /org/service-profile # create boot-definition
UCS-A /org/service-profile/boot-definition* # create san
UCS-A /org/service-profile/boot-definition/san* # create san-image primary
UCS-A /org/service-profile/boot-definition/san/san-image* # set vhba vHBA2
UCS-A /org/service-profile/boot-definition/san/san-image* # create path primary
UCS-A /org/service-profile/boot-definition/san/san-image/path* # set lun 0
UCS-A /org/service-profile/boot-definition/san/san-image/path* # commit-buffer
UCS-A /org/service-profile/boot-definition/san/san-image/path #
```

What to do next

Include the boot policy in a service profile and template.

iSCSI Boot

iSCSI boot enables a server to boot its operating system from an iSCSI target machine located remotely over a network.

iSCSI boot is supported on the following Cisco UCS hardware:

- Cisco UCS blade servers that have the Cisco UCS M51KR-B Broadcom BCM57711 network adapter and use the default MAC address provided by Broadcom.

- Cisco UCS M81KR Virtual Interface Card

- Cisco UCS VIC-1240 Virtual Interface Card

- Cisco UCS VIC-1280 Virtual Interface Card

- Cisco UCS VIC-1340 Virtual Interface Card

- Cisco UCS VIC 1455

- Cisco UCS rack servers that have the Cisco UCS M61KR-B Broadcom BCM57712 network adapter.

- Cisco UCS P81E Virtual Interface Card

- Cisco UCS VIC 1225 Virtual Interface Cardon Cisco UCS rack servers

There are prerequisites that must be met before you configure iSCSI boot. For a list of these prerequisites, see .

For a high-level procedure for implementing iSCSI boot, see .

iSCSI Boot Process

Cisco UCS Manager uses the iSCSI vNIC and iSCSI boot information created for the service profile in the association process to program the adapter, located on the server. After the adapter is programmed, the server reboots with the latest service profile values. After the power on self-test (POST), the adapter attempts to initialize using these service profile values. If the adapter can use the values and log in to its specified target, the adapter initializes and posts an iSCSI Boot Firmware Table (iBFT) to the host memory and a valid bootable LUN to the system BIOS. The iBFT that is posted to the host memory contains the initiator and target configuration that is programmed on the primary iSCSI VNIC.

Note   Previously, the host could see only one of the boot paths configured, depending on which path completed the LUN discovery first, and would boot from that path. Now, when there are two iSCSI boot vNICs configured, the host sees both of the boot paths. So for multipath configurations, a single IQN must be configured on both the boot vNICs. If there are different IQNs configured on the boot vNICs on a host, the host boots with the IQN that is configured on the boot vNIC with the lower PCI order.

The next step, which is the installation of the operating system (OS), requires an OS that is iBFT capable. During installation of the OS, the OS installer scans the host memory for the iBFT table and uses the information in the iBFT to discover the boot device and create an iSCSI path to the target LUN. Some OSs requires a NIC driver to complete this path. If this step is successful, the OS installer finds the iSCSI target LUN on which to install the OS.

Note The iBFT works at the OS installation software level and might not work with HBA mode (also known as TCP offload). Whether iBFT works with HBA mode depends on the OS capabilities during installation. Also, for a server that includes a Cisco UCS M51KR-B Broadcom BCM57711 adapter, the iBFT normally works at a maximum transmission unit (MTU) size of 1500, regardless of the MTU jumbo configuration. If the OS supports HBA mode, you might need to set HBA mode, dual-fabric support, and jumbo MTU size after the iSCSI installation process.

iSCSI Boot Guidelines and Prerequisites

These guidelines and prerequisites must be met before configuring iSCSI boot:

- After the iSCSI boot policies are created, a user with ls-compute privileges can include them in a service profile or service profile template. However, a user with only ls-compute privileges cannot create iSCSI boot policies.

- To set up iSCSI boot from a Windows 2008 server where the second vNIC (failover vNIC) must boot from an iSCSI LUN, consult Microsoft Knowledge Base Article 976042. Microsoft has a known issue where Windows might fail to boot from an iSCSI drive or cause a bugcheck error if the networking hardware is changed. To work around this issue, follow the resolution recommended by Microsoft.

- The storage array must be licensed for iSCSI boot and the array side LUN masking must be properly configured.

- Two IP addresses must be determined, one for each iSCSI initiator. If possible, the IP addresses should be on the same subnet as the storage array. The IP addresses are assigned statically or dynamically using the Dynamic Host Configuration Protocol (DHCP).

- You cannot configure boot parameters in the Global boot policy. Instead, after configuring boot parameters, include the boot policy in the appropriate service profile.

- The operating system (OS) must be iSCSI Boot Firmware Table (iBFT) compatible.

  - For RHEL 7.x, the kernel parameter "rd.iscsi.ibft=1" is required before the installation. If the parameter is not entered, the iSCSI boot may fail.

- For Cisco UCS M51KR-B Broadcom BCM57711 network adapters:

  - Servers that use iSCSI boot must contain the Cisco UCS M51KR-B Broadcom BCM57711 network adapter. For information on installing or replacing an adapter card, see the *Cisco UCS B250 Extended Memory Blade Server Installation and Service Note*. The service note is accessible from the *Cisco UCS B-Series Servers Documentation Roadmap* at http://www.cisco.com/go/unifiedcomputing/b-series-doc.

  - Set the MAC addresses on the iSCSI device.

  - If you are using the DHCP Vendor ID (Option 43), configure the MAC address of an iSCSI device in /etc/dhcpd.conf.

  - HBA mode (also known as TCP offload) and the boot to target setting are supported. However, only Windows OS supports HBA mode during installation.

  - Before installing the OS, disable the boot to target setting in the iSCSI adapter policy, then after installing the OS, re-enable the boot to target setting.

| **Note** | Each time you change an adapter policy setting, the adapter reboots to apply the new setting. |

- When installing the OS on the iSCSI target, the iSCSI target must be ordered *before* the device where the OS image resides. For example, if you are installing the OS on the iSCSI target from a CD, the boot order should be the iSCSI target and then the CD.

- After the server is iSCSI booted, do not modify the Initiator Name, Target name, LUN, iSCSI device IP, or Netmask/gateway using the Broadcom tool.

- Do not interrupt the POST (power on self-test) process or the Cisco UCS M51KR-B Broadcom BCM57711 network adapter will fail to initialize.

- For Cisco UCS M81KR Virtual Interface Card and Cisco UCS VIC-1240 Virtual Interface Card:

  For Cisco UCS VIC-1240 Virtual Interface Card:

  - Do not set MAC addresses on the iSCSI device.

  - HBA mode and the boot to target setting are *not* supported.

  - When installing the OS on the iSCSI target, the iSCSI target must be ordered *after* the device where the OS image resides. For example, if you are installing the OS on the iSCSI target from a CD, the boot order should be the CD and then the iSCSI target.

  - If you are using the DHCP Vendor ID (Option 43), the MAC address of the overlay vNIC must be configured in /etc/dhcpd.conf.

  - After the server is iSCSI booted, do not modify the IP details of the overlay vNIC.

- The VMware ESX/ESXi operating system does not support storing a core dump file to an iSCSI boot target LUN. Dump files must be written to a local disk.

Initiator IQN Configuration

Cisco UCS uses the following rules to determine the initiator IQN for an adapter iSCSI vNIC at the time a service profile is associated with a physical server:

- An initiator IQN at the service profile level *and* at the iSCSI vNIC level cannot be used together in a service profile.

- If an initiator IQN is specified at the service profile level, all of the adaptor iSCSI vNICs are configured to use the same initiator IQN, except in the case of DHCP Option 43, where the initiator IQN is set to empty on the adapter iSCSI vNIC.

- When an initiator IQN is set at the iSCSI vNIC level, the initiator IQN at the service profile level is removed, if one is present.

- If there are two iSCSI vNIC in a service profile and only one of them has the initiator IQN set, the second one is configured with the default IQN pool. You can change this configuration later. The only exception is if DHCP Option 43 is configured. In this case, the initiator IQN on the second iSCSI vNIC is removed during service profile association.

Note	If you change an iSCSI vNIC to use the DHCP Option 43 by setting the vendor ID, it does not remove the initiator IQN configured at the service profile level. The initiator IQN at the service profile level can still be used by another iSCSI vNIC which does not use the DHCP Option 43.

Enabling MPIO on Windows

You can enable (MPIO) to optimize connectivity with storage arrays.

Note	If you change the networking hardware, Windows might fail to boot from an iSCSI drive. For more information, see Microsoft support Article ID: 976042.

Before you begin

The server on which you enable the Microsoft Multipath I/O (MPIO) must have a Cisco VIC driver.

If there are multiple paths configured to the boot LUN, only one path should be enabled when the LUN is installed.

Procedure

Step 1	In the service profile associated with the server, configure the primary iSCSI vNIC.
	For more information, see Creating an iSCSI vNIC in a Service Profile, on page 119.
Step 2	Using the primary iSCSI vNIC, install the Windows operating system on the iSCSI target LUN.
Step 3	After Windows installation completes, enable MPIO on the host.
Step 4	In the service profile associated with the server, add the secondary iSCSI vNIC to the boot policy.
	For more information, see Creating an iSCSI Boot Policy, on page 115.

Configuring iSCSI Boot

When you configure an adapter or blade in Cisco UCS to iSCSI boot from a LUN target, complete all of the following steps.

Procedure

	Command or Action	**Purpose**
Step 1	(Optional) Configure the iSCSI boot adapter policy.	For more information, see Creating an iSCSI Adapter Policy, on page 110.

	Command or Action	Purpose
Step 2	(Optional) Configure the authentication profiles for the initiator and target.	For more information, see Creating an Authentication Profile, on page 112.
Step 3	(Optional) To configure the iSCSI initiator to use an IP address from a pool of IP addresses, add a block of IP addresses to the iSCSI initiator pool.	For more information, see Adding a Block of IP Addresses to the Initiator Pool, on page 113.
Step 4	Create a boot policy that can be used in any service profile. Alternatively, you can create a local boot policy only for the specific service policy. However, Cisco recommends that you create a boot policy that can be shared with multiple service profiles.	For more information about creating a boot policy that can be used in any service profile, see Creating an iSCSI Adapter Policy, on page 110.
Step 5	If you created a boot policy that can be used in any service profile, assign it to the service profile. Otherwise, proceed to the next step.	For more information, see Creating a Service Profile Template, on page 166.
Step 6	Configure an Ethernet vNIC in a service profile.	The Ethernet vNIC is used as the overlay vNIC for the iSCSI device. For more information, see Configuring a vNIC for a Service Profile, on page 176.
Step 7	Create an iSCSI vNIC in a service profile.	For more information, see Creating an iSCSI vNIC in a Service Profile, on page 119.
Step 8	Set the iSCSI initiator to boot using a static IP Address, an IP address from an IP pool, or DHCP.	See either Creating an iSCSI Initiator that Boots Using a Static IP Address, on page 121, Creating an iSCSI Initiator that Boots Using an IP Address from an IP Pool, on page 124, or Creating an iSCSI Initiator that Boots Using DHCP, on page 126.
Step 9	Create an iSCSI static or auto target.	For more information, see either Creating an iSCSI Static Target, on page 133 or Creating an iSCSI Auto Target, on page 136.
Step 10	Associate the service profile with a server.	For more information, see Associating a Service Profile with a Blade Server or Server Pool, on page 182.
Step 11	Verify the iSCSI boot operation.	For more information, see *Verifying iSCSI Boot*.
Step 12	Install the OS on the server.	For more information, see one of the following guides: • *Cisco UCS B-Series Blade Servers VMware Installation Guide* • *Cisco UCS B-Series Blade Servers Linux Installation Guide*

	Command or Action	Purpose
		• *Cisco UCS B-Series Blade Servers Windows Installation Guide*
Step 13	Boot the server.	

Creating an iSCSI Adapter 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 iscsi-policy** *policy-name*	Creates the iSCSI adapter policy.
Step 3	(Optional) UCS-A /org/iscsi-policy # **set descr** *description*	Provides a description for the iSCSI adapter policy.
Step 4	Required: UCS-A /org/iscsi-policy # **set iscsi-protocol-item connection-timeout** *timeout-secs*	The number of seconds to wait until Cisco UCS assumes that the initial login has failed and the iSCSI adapter is unavailable. Enter an integer between 0 and 255. If you enter 0, Cisco UCS uses the value set in the adapter firmware (default: 15 seconds).
Step 5	Required: UCS-A /org/iscsi-policy # **set iscsi-protocol-item dhcp-timeout** *timeout-secs*	The number of seconds to wait before the initiator assumes that the DHCP server is unavailable. Enter an integer between 60 and 300 (default: 60 seconds).
Step 6	Required: UCS-A /org/iscsi-policy # **set iscsi-protocol-item lun-busy-retry-count** *num*	The number of times to retry the connection in case of a failure during iSCSI LUN discovery. Enter an integer between 0 and 60. If you enter 0, Cisco UCS uses the value set in the adapter firmware (default: 15 seconds).
Step 7	Required: UCS-A /org/iscsi-policy # **set iscsi-protocol-item tcp-time-stamp** {**no** | **yes**}	Specifies whether to apply a TCP timestamp. With this setting, transmitted packets are given a time stamp of when the packet was sent so that the packet's round-trip time can be calculated, when needed. This setting applies only to Cisco UCS M51KR-B Broadcom BCM57711 adapters.

	Command or Action	**Purpose**
Step 8	Required: UCS-A /org/iscsi-policy # **set iscsi-protocol-item hbamode** {**no** | **yes**}	Specifies whether to enable HBA mode. This option should only be enabled for servers with the Cisco UCS NIC M51KR-B adapter running the Windows operating system.
Step 9	Required: UCS-A /org/iscsi-policy # **set iscsi-protocol-item boottotarget** {**no** | **yes**}	Specifies whether to boot from the iSCSI target. This option only applies to servers with the Cisco UCS NIC M51KR-B adapter. It should be disabled until you have installed an operating system on the server.
Step 10	Required: UCS-A /org/iscsi-policy # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create an iSCSI adapter policy called iscsiboot, set the connection timeout, DHCP timeout, and LUN busy retry count, apply a TCP timestamp, and commit the transaction:

```
UCS-A# scope org /
UCS-A /org # create iscsi-policy iscsiboot
UCS-A /org/iscsi-policy* # set iscsi-protocol-item connection-timeout 60
UCS-A /org/iscsi-policy* # set iscsi-protocol-item dhcp-timeout 200
UCS-A /org/iscsi-policy* # set iscsi-protocol-item lun-busy-retry-count 5
UCS-A /org/iscsi-policy* # set iscsi-protocol-item tcp-time-stamp yes
UCS-A /org/iscsi-policy* # set iscsi-protocol-item hbamode yes
UCS-A /org/iscsi-policy* # set iscsi-protocol-item boottotarget yes
UCS-A /org/iscsi-policy* # commit-buffer
UCS-A /org/iscsi-policy #
```

What to do next

Include the adapter policy in a service profile and template.

Deleting an iSCSI Adapter 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 iscsi-policy** *policy-name*	Deletes the iSCSI adapter policy.

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

Example

The following example shows how to delete an iSCSI adapter policy named iscsi-adapter-pol and commit the transaction:

```
UCS-A# scope org /
UCS-A /org # delete iscsi-policy iscsi-adapter-pol
UCS-A /org* # commit-buffer
UCS-A /org #
```

Creating an Authentication Profile

If you use authentication for iSCSI boot, you need to create an authentication profile for both the initiator and target.

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 auth-profile** *profile-name*	Creates an authentication profile with the specified name. The name can be up to 16 alphanumeric characters.
Step 3	UCS-A /org/auth-profile* # **set user-id** *id-name*	Creates a log in for authentication.
Step 4	UCS-A /org/auth-profile* # **set password**	Creates a password for authentication.
Step 5	UCS-A /org/auth-profile* # **commit-buffer**	Commits the transaction to the system configuration.
Step 6	UCS-A /org/auth-profile* # **exit**	Exits the current mode.
Step 7	Repeat steps 2 through 6 to create an authentication profile for the target.	

Example

The following example shows how to create an authentication profile for an initiator and target and commit the transaction:

```
UCS-A# scope org
```

```
UCS-A /org # create auth-profile InitAuth
UCS-A /org/auth-profile* # set user-id init
UCS-A /org/auth-profile* # set  password
Enter a password:
Confirm the password:
UCS-A /org/auth-profile* # commit-buffer
UCS-A /org/auth-profile # exit
UCS-A /org # create auth-profile TargetAuth
UCS-A /org/auth-profile* # set user-id target
UCS-A /org/auth-profile* # set  password
Enter a password:
Confirm the password:
UCS-A /org/auth-profile* # commit-buffer
UCS-A /org/auth-profile # exit
```

What to do next

Create an Ethernet vNIC to be used as the overlay vNIC for the iSCSI device, and then create an iSCSI vNIC.

Deleting an Authentication Profile

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 auth-profile** *auth-profile-name*	Deletes the specified authentication profile.
Step 3	UCS-A /org # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to delete an authentication profile called iscsi-auth and commit the transaction:

```
UCS-A# scope org
UCS-A /org # delete auth-profile iscsi-auth
UCS-A /org* # commit-buffer
UCS-A /org #
```

Adding a Block of IP Addresses to the Initiator Pool

You can create a group of IP addresses to be used for iSCSI boot. Cisco UCS Manager reserves the block of IP addresses you specify.

The IP pool must not contain any IP addresses that were assigned as static IP addresses for a server or service profile.

Procedure

	Command or Action	**Purpose**
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter / as the *org-name*.
Step 2	UCS-A /org# **scope ip-pool iscsi-initiator-pool**	Enters the mode to specify an iSCSI initiator pool.
Step 3	(Optional) UCS-A /org/ip-pool # **set descr** *description*	Provides a description for the IP pool. **Note** If your description includes spaces, special characters, or punctuation, you must begin and end your description with quotation marks. The quotation marks will not appear in the description field of any **show** command output.
Step 4	UCS-A /org/ip-pool # **set assignmentorder** {**default** | **sequential**}	This can be one of the following: • **default**—Cisco UCS Manager selects a random identity from the pool. • **sequential**—Cisco UCS Manager selects the lowest available identity from the pool.
Step 5	UCS-A /org/ip-pool# **create block** *from_ip_address to_ip_address default_gateway subnet_mask*	Creates a block of IP addresses for the iSCSI initiator.
Step 6	(Optional) UCS-A/org/ip-pool/block# **show detail expand**	Shows the block of IP addresses that you have created.
Step 7	UCS-A /org/ip-pool/block # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create an IP initiator pool for the iSCSI vNIC and commit the transaction:

```
UCS-A # scope org /
UCS-A /org # scope ip-pool iscsi-initiator-pool
UCS-A /org/ip-pool # create block 40.40.40.10 40.40.40.50 40.40.40.1 255.0.0.0
UCS-A /org/ip-pool/block # show detail expand
Block of IP Addresses:
    From: 40.40.40.10
    To: 40.40.40.50
    Default Gateway: 40.40.40.1
    Subnet Mask: 255.0.0.0
UCS-A /org/ip-pool/block # commit buffer
```

What to do next

Configure one or more service profiles or service profile templates to obtain the iSCSI initiator IP address from the iSCSI initiator IP pool.

Deleting a Block of IP Addresses from the Initiator Pool

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 ip-pool iscsi-initiator-pool**	Enters the mode to specify an iSCSI initiator pool.
Step 3	UCS-A /org/ip-pool# **delete block** *from_ip_address to_ip_address*	Deletes the specified block of IP addresses from the initiator pool.
Step 4	(Optional) UCS-A/org/ip-pool/block# **show detail expand**	Shows that the block of IP addresses has been deleted.
Step 5	UCS-A /org/ip-pool# **commit buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to delete a block of IP addresses from the initiator pool and commit the transaction:

```
UCS-A # scope org /
UCS-A /org # scope ip-pool iscsi-initiator-pool
UCS-A /org/ip-pool # delete block 40.40.40.10 40.40.40.50 40.40.40.1 255.0.0.0
UCS-A /org/ip-pool # show detail expand

IP Pool:
    Name: iscsi-initiator-pool
    Size: 0
    Assigned: 0
    Descr:
UCS-A /org/ip-pool # commit buffer
```

Creating an iSCSI Boot Policy

You can add up to two iSCSI vNICs per boot policy. One vNIC acts as the primary iSCSI boot source, and the other acts as the secondary iSCSI boot source.

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 boot-policy** *policy-name* [**purpose** {**operational**	**utility**}]	Creates a boot policy with the specified policy name, and enters organization boot policy mode.
		This name can be between 1 and 16 alphanumeric characters. You cannot use spaces or any special characters other than - (hyphen), _ (underscore), : (colon), and . (period), and you cannot change this name after the object is saved.
		When you create the boot policy, specify the **operational** option. This ensures that the server boots from the operating system installed on the server. The **utility** options is reserved and should only be used if instructed to do so by a Cisco representative.
Step 3	(Optional) UCS-A /org/boot-policy # **set descr** *description*	Provides a description for the boot policy.
		Note If your description includes spaces, special characters, or punctuation, you must begin and end your description with quotation marks. The quotation marks do not appear in the description field of any **show** command output.
Step 4	(Optional) UCS-A /org/boot-policy # **set enforce-vnic-name** {**no**	**yes**}	If you choose **yes**, Cisco UCS Manager reports whether the device name specified in the boot policy matches what is specified in the service profile.
		If you choose **no**, Cisco UCS Manager uses any vNIC, vHBA, or iSCSI device from the service profile and does not report whether the device name specified in the boot policy matches what is specified in the service profile.
Step 5	UCS-A /org/boot-policy # **set reboot-on-update** {**no**	**yes**}	Specifies whether the servers using this boot policy are automatically rebooted after you make changes to the boot order.
		In the Cisco UCS Manager GUI, if the **Reboot on Boot Order Change** check box is checked for a boot policy, and if CD-ROM or Floppy is the last device in the boot order, deleting or

	Command or Action	Purpose
		adding the device does not directly affect the boot order and the server does not reboot.
Step 6	UCS-A /org/boot-policy # **create iscsi**	Adds an iSCSI boot to the boot policy.
Step 7	UCS-A /org/boot-policy/iscsi # **create path** {**primary** | **secondary**}	Specifies the primary and secondary paths that Cisco UCS Manager uses to reach the iSCSI target .With iSCSI boot, you set up two paths. Cisco UCS Manager uses the primary path first, and if that fails, then it uses the secondary path.
Step 8	UCS-A /org/boot-policy/iscsi/path # **create iscsivnicname** *iscsi-vnic-name*	Creates an iSCSI vNIC.
Step 9	UCS-A /org/boot-policy/iscsi/path # **exit**	Exits iSCSI path mode.
Step 10	UCS-A /org/boot-policy/iscsi/path # **set order** *order-num*	Specifies the order for the iSCSI boot in the boot order.
Step 11	(Optional) Repeat steps 8-10 to create secondary iSCSI vNICs.	
Step 12	UCS-A /org/boot-policy/iscsi # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create an iSCSI boot policy named iscsi-boot-policy-LAN, provide a description for the boot policy, specify that servers using this policy are not automatically rebooted when the boot order is changed, set the boot order for iSCSI boot to 2, create an iSCSI boot and associate it with a vNIC called iscsienic1, and commit the transaction:

```
UCS-A# scope org /
UCS-A /org* # create boot-policy iscsi-boot-policy-LAN purpose operational
UCS-A /org/boot-policy* # set descr "Boot policy that boots from iSCSI."
UCS-A /org/boot-policy* # set enforce-vnic-name yes
UCS-A /org/boot-policy* # set reboot-on-update no
UCS-A /org/boot-policy* # create iscsi
UCS-A /org/boot-policy/iscsi* # create path primary
UCS-A /org/boot-policy/iscsi/path* # set iscsivnicname iscsienic1
UCS-A /org/boot-policy/iscsi/path* # exit
UCS-A /org/boot-policy/iscsi* # set order 2
UCS-A /org/boot-policy/iscsi* # commit-buffer
UCS-A /org/boot-policy #
```

What to do next

Include the boot policy in a service profile and template.

After a server is associated with a service profile that includes this boot policy, you can verify the actual boot order in the **Boot Order Details** area on the **General** tab for the server.

Deleting iSCSI Devices from a Boot 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 boot-policy** *boot-pol-name*	Enters boot policy organization mode for the specified boot policy.
Step 3	UCS-A /org/boot-policy # **delete iscsi**	Deletes the iSCSI boot from the boot policy.
Step 4	UCS-A /org/boot-policy # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to delete an iSCSI boot from the boot policy named boot-policy-iscsi and commit the transaction:

```
UCS-A# scope org /
UCS-A /org # scope boot-policy boot-policy-iscsi
UCS-A /org/boot-policy # delete iscsi
UCS-A /org/boot-policy* # commit-buffer
UCS-A /org/boot-policy #
```

Setting an Initiator IQN at the Service Profile Level

In a service profile, you can create an initiator with a specific IQN or one that is derived from a pool of IQNs.

Before you begin

You cannot delete an IQN using the CLI.

To understand the initiator IQN configuration guidelines, see Initiator IQN Configuration, on page 107.

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 service profile organization mode for the service profile.

	Command or Action	**Purpose**
Step 3	UCS-A /org/service-profile# **set iscsi-identity** {**initiator name***initiator-name*|**initiator-pool-name***pool-name*}	Creates an initiator with the specified name. The name can be up to 16 alphanumeric characters.
Step 4	UCS-A /org/service-profile* # **commit buffer**	Commits the transaction to the system configuration.
Step 5	UCS-A /org/auth-profile* # **exit**	Exits the current mode.

Example

The following example shows how to create a specific name for an iSCSI initiator and commit the transaction:

```
UCS-A# scope org /
UCS-A /org # scope service-profile accounting
UCS-A /org/service-profile # set iscsi-identity initiator-name manual:IQN
UCS-A /org/service-profile* # commit-buffer
```

Creating an iSCSI vNIC in a Service Profile

You can create an iSCSI vNIC in a service profile.

Before you begin

You must have an Ethernet vNIC in a service profile to be used as the overlay vNIC for the iSCSI device.

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 service profile organization mode for the service profile.
Step 3	UCS-A /org/service-profile # **create vnic-iscsi** *iscsi-vnic-name* .	Specifies the iSCSI vNIC name.
Step 4	(Optional) UCS-A /org/service-profile/vnic-iscsi* # **set iscsi-adaptor-policy** *iscsi-adaptor-name*	Specifies the iSCSI adapter policy that you have created for this iSCSI vNIC.
Step 5	(Optional) UCS-A /org/service-profile/vnic-iscsi* # **set auth-name** *authentication-profile-name*	Sets the authentication profile to be used by the iSCSI vNIC. The authentication profile must already exist for it to be set. For more information, see Creating an Authentication Profile, on page 112.

	Command or Action	Purpose
Step 6	UCS-A /org/service-profile/vnic-iscsi* # **set identity** { **dynamic-mac** {*dynamic-mac-address* | **derived** } | **mac-pool** *mac-pool-name* }	Specifies the MAC address for the iSCSI vNIC. **Note** The MAC address is only set for Cisco UCS NIC M51KR-B adapters.
Step 7	UCS-A /org/service-profile/vnic-iscsi* # **set iscsi-identity** {**initiator-name** *initiator-name* | **initiator-pool-name** *iqn-pool-name*}	Specifies the name of the iSCSI initiator or the name of an IQN pool from which the iSCSI initiator name will be provided. The iSCSI initiator name can be up to 223 characters.
Step 8	UCS-A /org/service-profile/vnic-iscsi* # **set overlay-vnic-name** *overlay-vnic-name*	Specifies the Ethernet vNIC that is used by the iSCSI device as the overlay vNIC. For more information, see Configuring a vNIC for a Service Profile, on page 176.
Step 9	UCS-A /org/service-profile/vnic-iscsi* # **create eth-if**	Creates an Ethernet interface for a VLAN assigned to the iSCSI vNIC.
Step 10	UCS-A /org/service-profile/vnic-iscsi/eth-if* # **set vlanname** *vlan-name*.	Specifies the VLAN name. The default VLAN is default. For the Cisco UCS M81KR Virtual Interface Card and the Cisco UCS VIC-1240 Virtual Interface Card, the VLAN that you specify must be the same as the native VLAN on the overlay vNIC. For the Cisco UCS M51KR-B Broadcom BCM57711 adapter, the VLAN that you specify can be any VLAN assigned to the overlay vNIC.
Step 11	UCS-A /org/service-profile/vnic-iscsi # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create an iSCSI vNIC called scsivnic1, add it to an existing service profile called accounting, and commit the transaction:

```
UCS-A# scope org /
UCS-A /org # scope service-profile accounting
UCS-A /org/service-profile # create vnic-iscsi iSCSI1
UCS-A /org/service-profile/vnic-iscsi* # set iscsi-adaptor-policy iscsiboot
UCS-A /org/service-profile/vnic-iscsi* # set auth-name initauth
UCS-A /org/service-profile/vnic-iscsi* # set identity dynamic-mac derived
UCS-A /org/service-profile/vnic-iscsi* # set iscsi-identity initiator-name iSCSI1
UCS-A /org/service-profile/vnic-iscsi* # set overlay-vnic-name eth1
UCS-A /org/service-profile/vnic-iscsi* # create eth-if
UCS-A /org/service-profile/vnic-iscsi/eth-if* # set vlanname default
UCS-A /org/service-profile/vnic-iscsi/eth-if* # commit buffer
```

What to do next

Configure an iSCSI initiator to boot using a static IP address, an IP address from a configured IP pool, or DHCP.

Deleting an iSCSI vNIC from a Service Profile

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 service profile organization mode for the service profile.
Step 3	UCS-A /org/service-profile # **delete vnic-iscsi** *iscsi-vnic-name*	Deletes the specified iSCSI vNIC from the specified service profile.
Step 4	UCS-A /org/service-profile # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to delete an iSCSI vNIC called scsivnic1 and commit the transaction:

```
UCS-A# scope org /
UCS-A /org # scope service-profile accounting
UCS-A /org/service-profile # delete vnic-iscsi scsivnic1
UCS-A /org/service-profile* # commit-buffer
UCS-A /org/service-profile #
```

Creating an iSCSI Initiator that Boots Using a Static IP Address

In a service profile, you can create an iSCSI initiator and configure it to boot using a static IP address.

Before you begin

You have completed the following:

- Created iSCSI overlay vNICs in a service profile.

- Created an iSCSI vNIC in a service profile.

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 service profile organization mode for the service profile.
Step 3	UCS-A /org/service-profile # **scope vnic-iscsi** *iscsi-vnic-name*	Enters the configuration mode for the specified iSCSI vNIC.
Step 4	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi # **create ip-if**	Creates an IP interface.
Step 5	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/eth-if/ip-if* # **enter static-ip-params**	Specifies that you are entering static IP boot parameters.
Step 6	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/eth-if/ip-if/static-ip-params* # **set addr** *ip-address*	Specifies the static IP address.
Step 7	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/eth-if/ip-if/static-ip-params* # **set default-gw** *ip-address*	Specifies the default gateway IP address.
Step 8	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/eth-if/ip-if/static-ip-params* # **set primary-dns** *ip-address*	Specifies the primary DNS IP address.
Step 9	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/eth-if/ip-if/static-ip-params* # **set secondary-dns** *ip-address*	Specifies the secondary DNS IP address.
Step 10	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/eth-if/ip-if/static-ip-params* # **set subnet** *subnet-ip-address*	Specifies the subnet mask.
Step 11	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/eth-if/ip-if/static-ip-params* # **commit buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to configure the initiator to boot using a static IP address and commit the transaction:

```
UCS-A # scope org
UCS-A /org # scope service-profile accounting
UCS-A /org/service-profile # scope vnic-iscsi iSCSI1
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi # create ip-if
```

```
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/ip-if* # enter static-ip-params
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/ip-if/static-ip-params* # set addr
10.104.105.193
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/ip-if/static-ip-params* # set default-gw
10.104.105.1
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/ip-if/static-ip-params* # set primary-dns
 11.11.11.100
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/ip-if/static-ip-params* # set secondary-dns
 11.11.11.100
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/ip-if/static-ip-params* # set subnet
255.255.255.0
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/ip-if/static-ip-params* # commit-buffer
```

What to do next

Create an iSCSI target.

Deleting the Static IP Address Boot Parameters from an iSCSI Initiator

In a service profile, you can delete the static IP address boot parameters from an iSCSI initiator.

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 service profile organization mode for the service profile.
Step 3	UCS-A /org/service-profile # **scope vnic-iscsi** *iscsi-vnic-name*	Enters the configuration mode for the specified iSCSI vNIC.
Step 4	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi # **scope ip-if**	Enters the configuration mode for an IP interface.
Step 5	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/ip-if* # **delete static-ip-params**	Deletes the static IP boot parameters from an initiator.
Step 6	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/ip-if/static-ip-params* # **commit buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to delete the static IP address boot parameters from the initiator and commit the transaction:

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

```
UCS-A /org/service-profile # scope vnic-iscsi iSCSI1
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi # scope ip-if
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/ip-if # delete static-ip-params
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/ip-if* # commit-buffer
```

Creating an iSCSI Initiator that Boots Using an IP Address from an IP Pool

In a service profile, you can create an iSCSI initiator and configure it to boot using an IP address from an IP pool that you have created.

Before you begin

You have completed the following:

- Created an overlay vNIC in a service profile
- Created an iSCSI vNIC in a service profile.

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 service profile organization mode for the service profile.
Step 3	UCS-A /org/service-profile # **scope iscsi-boot**	Enters the configuration mode for configuring iSCSI boot parameters.
Step 4	UCS-A /org/service-profile/iscsi-boot # **scope vnic-iscsi** *iscsi-vnic-name*	Enters the configuration mode for the specified iSCSI vNIC.
Step 5	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi* # **scope ip-if**	Enters the configuration mode for the iSCSI Ethernet interface.
Step 6	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/ip-if* # **enter pooled-ip-params**	Specifies that the iSCSI initiator boot using one of the IP addresses from the previously created iSCSI initiator IP pool.
Step 7	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/ip-if/pooled-ip-params* # **commit buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create an iSCSI initiator and configure it to boot using an IP address from an IP pool:

```
UCS-A # scope org
UCS-A /org # scope service-profile accounting
UCS-A /org/service-profile/iscsi-boot # scope vnic-iscsi iSCSI1
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi # scope ip-if
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/ip-if* # enter pooled-ip-params
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/ip-if/pooled-ip-params* # commit buffer
```

What to do next

Create an iSCSI target.

Deleting the IP Pool Boot Parameter from an iSCSI Initiator

In a service profile, you can create an iSCSI initiator and configure it to boot using an IP address from an IP pool that you have created.

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 service profile organization mode for the service profile.
Step 3	UCS-A /org/service-profile # **scope iscsi-boot**	Enters the configuration mode for configuring the iSCSI boot parameters.
Step 4	UCS-A /org/service-profile/iscsi-boot/ # **scope vnic-iscsi** *iscsi-vnic-name*	Enters the configuration mode for the specified iSCSI vNIC.
Step 5	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi # **enter ip-if**	Enters the configuration mode for an IP interface.
Step 6	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/ip-if* # **delete pooled-ip-params**	Specifies that the iSCSI initiator does not use an IP address from an IP pool to boot.
Step 7	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/ip-if/pooled-ip-params* # **commit buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to delete the boot using an IP address from an IP poo parameter and commit the transaction:

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

```
UCS-A /org/service-profile/iscsi-boot # scope vnic-iscsi iSCSI1
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi # enter ip-if
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/ip-if* # delete pooled-ip-params
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/ip-if/pooled-ip-params* # commit buffer
```

Creating an iSCSI Initiator that Boots Using DHCP

In a service profile, you can create an iSCSI initiator and configure it to boot using DHCP.

Before you begin

You have completed the following:

- Created iSCSI overlay vNICs in a service profile.

- Created an iSCSI vNIC in a service profile.

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 service profile organization mode for the service profile.
Step 3	UCS-A /org/service-profile # **scope iscsi-boot**	Enters the configuration mode for configuring iSCSI boot parameters.
Step 4	UCS-A /org/service-profile/iscsi-boot # **scope vnic-iscsi** *iscsi-vnic-name*	Enters the configuration mode for the specified iSCSI vNIC.
Step 5	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi # **create ip-if**	Creates an IP interface.
Step 6	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/ip-if* # **create dhcp-ip-params**	Specifies that you are setting the initiator to boot using DHCP.
Step 7	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/ip-if/dhcp-ip-params* # **commit buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to configure the initiator to boot using DHCP and commit the transaction:

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

```
UCS-A /org/service-profile/iscsi-boot # scope vnic-iscsi iSCSI1
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi # create ip-if
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/ip-if* # create dhcp-ip-params
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/ip-if/dhcp-ip-params* # commit-buffer
```

What to do next

Create an iSCSI target.

Deleting the DHCP Boot Parameter from an iSCSI Initiator

In a service profile, you can remove the DHCP boot parameter from an iSCSI initiator.

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 service profile organization mode for the service profile.
Step 3	UCS-A /org/service-profile # **scope iscsi-boot**	Enters the configuration mode for configuring iSCSI boot parameters.
Step 4	UCS-A /org/service-profile/iscsi-boot # **scope vnic-iscsi** *iscsi-vnic-name*	Enters the configuration mode for the specified iSCSI vNIC.
Step 5	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi # **enter ip-if**	Enters the configuration mode for an IP interface.
Step 6	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/ip-if* # **delete dhcp-ip-params**	Specifies that the initiator does not use DHCP to boot.
Step 7	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/ip-if/dhcp-ip-params* # **commit buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to delete the boot using DHCP parameter and commit the transaction:

```
UCS-A # scope org
UCS-A /org # scope service-profile accounting
UCS-A /org/service-profile # scope iscsi-boot
UCS-A /org/service-profile/iscsi-boot # scope vnic-iscsi iSCSI1
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi # enter ip-if
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/ip-if* # delete dhcp-ip-params
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/ip-if/dhcp-ip-params* # commit-buffer
```

IQN Pools

An IQN pool is a collection of iSCSI Qualified Names (IQNs) for use as initiator identifiers by iSCSI vNICs in a Cisco UCS domain.

IQN pool members are of the form *prefix***:***suffix***:***number*, where you can specify the prefix, suffix, and a block (range) of numbers.

An IQN pool can contain more than one IQN block, with different number ranges and different suffixes, but sharing the same prefix.

Creating an IQN Pool

Note   In most cases, the maximum IQN size (prefix + suffix + additional characters) is 223 characters. When using the Cisco UCS NIC M51KR-B adapter, you must limit the IQN size to 128 characters.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter / as the *org-name*.
Step 2	UCS-A /org # **create iqn-pool** *pool-name*	Creates an IQN pool with the specified pool name and enters organization IQN pool mode.
		This name can be between 1 and 32 alphanumeric characters. You cannot use spaces or any special characters other than - (hyphen), _ (underscore), : (colon), and . (period), and you cannot change this name after the object is saved.
Step 3	UCS-A /org/iqn-pool # **set iqn-prefix** *prefix*	Specifies the prefix for the IQN block members. Unless limited by the adapter card, the prefix can contain up to 150 characters.
Step 4	(Optional) UCS-A /org/iqn-pool # **set descr** *description*	Provides a description for the IQN pool. Enter up to 256 characters.
		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.

	Command or Action	Purpose
Step 5	UCS-A /org/iqn-pool # **set assignmentorder** {**default** | **sequential**}	This can be one of the following: • **default**—Cisco UCS Manager selects a random identity from the pool. • **sequential**—Cisco UCS Manager selects the lowest available identity from the pool.
Step 6	UCS-A /org/iqn-pool # **create block** *suffix from to*	Creates a block (range) of IQNs, and enters organization IQN pool block mode. You must specify the base suffix, the starting suffix number, and the ending suffix number. The resulting IQN pool members are of the form *prefix***:***suffix***:***number*. The suffix can be up to 64 characters. **Note** An IQN pool can contain more than one IQN block. To create multiple blocks, enter multiple **create block** commands from organization IQN pool mode.
Step 7	UCS-A /org/iqn-pool/block # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create an IQN pool named pool4, provide a description for the pool, specify a prefix and a block of suffixes to be used for the pool, and commit the transaction:

```
UCS-A# scope org /
UCS-A /org # create iqn-pool pool4
UCS-A /org/iqn-pool* # set iqn-prefix iqn.alpha.com
UCS-A /org/iqn-pool* # set descr "This is IQN pool 4"
UCS-A /org/iqn-pool* # create block beta 3 5
UCS-A /org/iqn-pool/block* # commit-buffer
UCS-A /org/iqn-pool/block #
```

What to do next

Include the IQN suffix pool in a service profile and template.

Adding Blocks to an IP Pool

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 ip-pool** *pool-name*	Enters organization IP pool mode for the specified pool.
Step 3	UCS-A /org/ip-pool # **create** *block first-ip-addr last-ip-addr gateway-ip-addr subnet-mask*	Creates a block (range) of IP addresses, and enters organization IP pool block mode. You must specify the first and last IP addresses in the address range, the gateway IP address, and subnet mask.

Note   An IP pool can contain more than one IP block. To create multiple blocks, enter multiple **create block** commands from organization IP pool mode. |
| **Step 4** | UCS-A /org/ip-pool/block # **commit-buffer** | Commits the transaction. |
| **Step 5** | UCS-A /org/ip-pool/block # **exit** | Exits IPv4 block configuration mode. |
| **Step 6** | UCS-A /org/ip-pool # **create ipv6-block** *first-ip6-addr last-ip6-addr gateway-ip6-addr prefix* | Creates a block (range) of IPv6 addresses, and enters organization IP pool IPv6 block mode. You must specify the first and last IPv6 addresses in the address range, the gateway IPv6 address, and network prefix.

Note   An IP pool can contain more than one IPv6 block. To create multiple IPv6 blocks, enter multiple **create ipv6-block** commands from organization IP pool mode. |
| **Step 7** | UCS-A /org/ip-pool/ ipv6-block # **commit-buffer** | Commits the transaction to the system configuration. |

Example

This example shows how to add blocks of IPv4 and IPv6 addresses to an IP pool named pool4 and commit the transaction:

```
UCS-A# scope org /
UCS-A /org # scope ip-pool pool4
UCS-A /org/ip-pool # create block 192.168.100.1 192.168.100.200 192.168.100.10 255.255.255.0
```

```
UCS-A /org/ip-pool/block* # commit-buffer
UCS-A /org/ip-pool/block #exit
UCS-A /org/ip-pool* # create ipv6-block 2001:888::10 2001:888::100 2001:888::1 64
UCS-A /org/ip-pool/ipv6-block* commit-buffer
```

Deleting a Block from an IP Pool

If you delete an address block from a pool, Cisco UCS Manager does not reallocate any addresses in that block that were assigned to vNICs or vHBAs. All assigned addresses from a deleted block remain with the vNIC or vHBA to which they are assigned until one of the following occurs:

- The associated service profiles are deleted.

- The vNIC or vHBA to which the address is assigned is deleted.

- The vNIC or vHBA is assigned to a different pool.

Note   IPv6 address blocks are not applicable to vNICs or vHBAs.

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 ip-pool***pool-name*	Enters organization IP pool mode for the specified pool.
Step 3	UCS-A /org/ip-pool # **delete** {*ip-block|ipv6-block*} {*first-ip-addr|first-ip6-addr*} {*last-ip-addr | last-ip6-addr*}	Deletes the specified block (range) of IPv4 or IPv6 addresses.
Step 4	UCS-A /org/ip-pool # **commit-buffer**	Commits the transaction to the system configuration.

Example

This example shows how to delete an IP address block from an IP pool named pool4 and commit the transaction:

```
UCS-A# scope org /
UCS-A /org # scope ip-pool pool4
UCS-A /org/ip-pool # delete block 192.168.100.1 192.168.100.200
UCS-A /org/ip-pool* # commit-buffer
UCS-A /org/ip-pool #
```

This example shows how to delete an IPv6 address block from an IP pool named pool4 and commit the transaction:

```
UCS-A# scope org /
UCS-A /org # scope ip-pool pool4
UCS-A /org/ip-pool # delete ipv6-block 2001::1 2001::10
UCS-A /org/ip-pool* # commit-buffer
UCS-A /org/ip-pool #
```

Deleting an IQN Pool

If you delete a pool, Cisco UCS Manager does not reallocate any addresses from that pool that were assigned to vNICs or vHBAs. All assigned addresses from a deleted pool remain with the vNIC or vHBA to which they are assigned until one of the following occurs:

- The associated service profiles are deleted.

- The vNIC or vHBA to which the address is assigned is deleted.

- The vNIC or vHBA is assigned to a different pool.

Procedure

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

Example

The following example shows how to delete the IQN pool named pool4 and commit the transaction:

```
UCS-A# scope org /
UCS-A /org # delete iqn-pool pool4
UCS-A /org* # commit-buffer
UCS-A /org #
```

Viewing IQN Pool Usage

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 iqn-pool** *pool-name*	Enters organization IQN pool mode for the specified pool.

	Command or Action	Purpose
Step 3	UCS-A /org/iqn-pool # **show pooled**	Displays the assignments of the IQN block members.

Example

The following example shows how to display the assignments of suffixes in the IQN pool named pool4:

```
UCS-A# scope org /
UCS-A /org # scope iqn-pool pool4
UCS-A /org/iqn-pool # show pooled
Pooled:
    Name        Assigned Assigned To Dn
    ---------- -------- --------------
    beta:3     No
    beta:4     No
    beta:5     No

UCS-A /org/iqn-pool #
```

Creating an iSCSI Static Target

You can create a static target.

Before you begin

You have already created an iSCSI vNIC.

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 service profile organization mode for the service profile to which you want to add an iSCSI target.
Step 3	UCS-A /org/service-profile # **scope iscsi-boot**	Enters the mode for configuring iSCSI boot parameters.
Step 4	UCS-A /org/service-profile/iscsi-boot # **scope vnic-iscsi** *iscsi-vnic-name*	Enters the iSCSI vNIC mode for the specified vNIC name.
Step 5	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi # **create static-target-if** {**1** | **2**}	Creates a static target for the iSCSI vNIC and assigns a priority level to it. Valid priority levels are 1 or 2.

	Command or Action	Purpose
Step 6	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/static-target-if # **set name** *name*	A regular expression that defines the iSCSI Qualified Name (IQN) or Extended Unique Identifier (EUI) name of the iSCSI target. You can enter any alphanumeric characters as well as the following special characters: • . (period) • : (colon) • - (dash) **Important** This name must be properly formatted using standard IQN or EUI guidelines. The following examples show properly formatted iSCSI target names: • iqn.2001-04.com.example • iqn.2001-04.com.example:storage:diskarrays-sn-a8675309 • iqn.2001-04.com.example:storage.tape1.sys1.xyz • iqn.2001-04.com.example:storage.disk2.sys1.xyz • eui.02004567A425678D
Step 7	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/static-target-if # **set port** *port-num*	The port associated with the iSCSI target. Enter an integer between 1 and 65535. The default is 3260.
Step 8	(Optional) UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/static-target-if # **set auth-name** *auth-profile*	If you need the target to authenticate itself and have set up an authentication profile, you need to specify the name of authentication profile. The name of the associated iSCSI authentication profile.
Step 9	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/static-target-if # **set ipaddress** *ipv4-address*	The IPv4 address assigned to the iSCSI target.
Step 10	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/static-target-if # **create lun**	Creates the LUN that corresponds to the location of the interface.
Step 11	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/static-target-if/lun* # **set id** *id-number*	Specifies the target LUN id. Valid values are from 0 to 65535.

	Command or Action	Purpose
Step 12	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/static-target-if/lun* # **exit**	Exits the current configuration mode.
Step 13	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/static-target-if # **exit**	Exits the current configuration mode.
Step 14	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi # **commit-buffer**	Commits the transaction to the system configuration.
Step 15	(Optional) Repeat steps 5 through 14 to create a second static target.	

Example

The following example shows how to create two iSCSI static target interfaces and commit the transaction:

```
UCS-A # scope org test
UCS-A /org # scope service-profile accounting
UCS-A /org/service-profile # scope iscsi-boot
UCS-A /org/service-profile/iscsi-boot # scope vnic-iscsi iSCSI1
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi # create static-target-if 1
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/static-target-if* # set name statictarget1
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/static-target-if* # set port 3260
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/static-target-if* # set auth-name
authprofile1
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/static-target-if* # set ip-address
192.168.10.10
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/static-target-if* # create lun
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/static-target-if/lun* # set id 1
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/static-target-if/lun* # exit
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/static-target-if* # exit
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi # commit-buffer
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi # create static-target-if 2
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/static-target-if* # set ipaddress
192.168.10.11
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/static-target-if* # set name statictarget2
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/static-target-if* # set port 3260
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/static-target-if* # set auth-name
authprofile1
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/static-target-if* # create lun
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/static-target-if/lun* # set id 1
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/static-target-if/lun* # exit
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/static-target-if* # exit
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi # commit-buffer
```

What to do next

To configure a second iSCSI device, repeat the steps for creating an iSCSI vNIC, initiator, and target.

Deleting an iSCSI Static Target

You can delete an iSCSI static target. However, you must have at least one iSCSI static target remaining after you delete one. Therefore, you must have two iSCSI static targets in order to delete one of them.

> **Note** If you have two iSCSI targets and you delete the first priority target, the second priority target becomes the first priority target, although the Cisco UCS Manager still shows it as the second priority target.

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 service profile organization mode for the service profile to which you want to add an iSCSI target.
Step 3	UCS-A /org/service-profile # **scope iscsi-boot**	Enters the mode for configuring iSCSI boot parameters.
Step 4	UCS-A /org/service-profile/iscsi-boot # **scope vnic-iscsi** *iscsi-vnic-name*	Enters the iSCSI vNIC mode for the specified vNIC name.
Step 5	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi # **delete static-target-if**	Deletes the static target for the iSCSI vNIC.
Step 6	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to delete an iSCSI static target and commit the transaction:

```
UCS-A # scope org test
UCS-A /org # scope service-profile sample
UCS-A /org # scope iscsi-boot
UCS-A /org/service-profile/iscsi-boot # scope vnic-iscsi trial
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi # delete static-target-if 1
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi # commit-buffer
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi #
```

Creating an iSCSI Auto Target

You can create an iSCSI auto target with or without the vendor IDs.

Before you begin

These prerequisites must be met before creating iSCSI auto target:

- You have already created an iSCSI vNIC in a service profile.

- You have considered the prerequisites for the VIC that you are using. For more information, see iSCSI Boot Guidelines and Prerequisites, on page 106

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 service profile organization mode for the service profile that you want to add an iSCSI target interface to.
Step 3	UCS-A /org # **scope iscsi-boot** **Example:**	Enters the mode for configuring iSCSI boot parameters.
Step 4	UCS-A /org/service-profile/iscsi-boot # **scope vnic-iscsi** *iscsi-vnic-name*	Enters iSCSI vNIC service profile organization mode for the specified vNIC name.
Step 5	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/ # **create auto-target-if**	Creates an auto target for the iSCSI vNIC. If you plan to use an auto target without the vendor ID, you must configure an initiator name. For more information, see Creating an iSCSI vNIC in a Service Profile, on page 119.
Step 6	(Optional) UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/auto-target-if* # **set dhcp-vendor-id** *vendor-id*	Sets a vendor ID for the auto target. The vendor ID can be up to 32 alphanumeric characters.
Step 7	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/auto-target-if* # **exit**	Exists the current configuration mode.
Step 8	UCS-A /org/service-profile/iscis-boot/vnic-iscsi # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create an iSCSI auto target *without* a vendor ID and commit the transaction:

```
UCS-A # scope org
UCS-A /org # scope service-profile accounting
UCS-A /org/service-profile # scope iscsi-boot
UCS-A /org/service-profile/iscsi-boot # scope vnic-iscsi iSCSI1
```

```
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi # create auto-target-if
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/auto-target-if* # exit
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi # commit-buffer
```

The following example shows how to create an iSCSI auto target *with* a vendor ID and commit the transaction:

```
UCS-A # scope org
UCS-A /org # scope service-profile accounting
UCS-A /org/service-profile # scope iscsi-boot
UCS-A /org/service-profile/iscsi-boot # scope vnic-iscsi iSCSI1
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi # create auto-target-if
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/auto-target-if* # set dhcp-vendor-id
iSCSI_Vendor
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi/auto-target-if* # exit
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi # commit-buffer
```

What to do next

To configure a second iSCSI device, repeat the steps for creating an iSCSI vNIC, initiator, and target.

Deleting an iSCSI Static Target

You can delete an iSCSI static target. However, you must have at least one iSCSI static target remaining after you delete one. Therefore, you must have two iSCSI static targets in order to delete one of them.

Note   If you have two iSCSI targets and you delete the first priority target, the second priority target becomes the first priority target, although the Cisco UCS Manager still shows it as the second priority target.

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 service profile organization mode for the service profile to which you want to add an iSCSI target.
Step 3	UCS-A /org/service-profile # **scope iscsi-boot**	Enters the mode for configuring iSCSI boot parameters.
Step 4	UCS-A /org/service-profile/iscsi-boot # **scope vnic-iscsi** *iscsi-vnic-name*	Enters the iSCSI vNIC mode for the specified vNIC name.
Step 5	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi # **delete static-target-if**	Deletes the static target for the iSCSI vNIC.
Step 6	UCS-A /org/service-profile/iscsi-boot/vnic-iscsi # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to delete an iSCSI static target and commit the transaction:

```
UCS-A # scope org test
UCS-A /org # scope service-profile sample
UCS-A /org # scope iscsi-boot
UCS-A /org/service-profile/iscsi-boot # scope vnic-iscsi trial
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi # delete static-target-if 1
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi # commit-buffer
UCS-A /org/service-profile/iscsi-boot/vnic-iscsi #
```

Verifying iSCSI Boot

Use the KVM console to view the boot up messages as the adapter is booting. For information on how to access the KVM console, see the *Starting the KVM Console* chapter.

This step can only be performed using the Cisco UCS Manager GUI. For more information, see the *Starting the KVM Console* chapter in the *UCS Manager GUI Configuration Guide*.

- For the Cisco UCS M51KR-B Broadcom BCM57711, the following message appears:

  Logging in the 1st iSCSI Target…. Succeeded.

- For the Cisco UCS M81KR Virtual Interface Card, the following message appears:

  Option ROM installed successfully.

LAN Boot

You can configure a boot policy to boot one or more servers from a centralized provisioning server on the LAN. A LAN (or PXE) boot is frequently used to install operating systems on a server from that LAN server.

You can add more than one type of boot device to a LAN boot policy. For example, you could add a local disk or virtual media boot as a secondary boot device.

Configuring a LAN Boot Policy for a Boot Policy

Before you begin

Create a boot policy to contain the LAN boot configuration.

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 boot-policy** *policy-name*	Enters organization boot policy mode for the specified boot policy.
Step 3	UCS-A /org/boot-policy # **create lan**	Creates a LAN boot for the boot policy and enters organization boot policy LAN mode.
Step 4	UCS-A /org/boot-policy/lan # **set order** {**1** | **2** | **3** | **4**}	Specifies the boot order for the LAN boot.
Step 5	UCS-A /org/boot-policy/lan # **create path** {**primary** | **secondary**}	Creates a primary or secondary LAN boot path and enters organization boot policy LAN path mode.
Step 6	UCS-A /org/boot-policy/lan/path # **set vnic** *vnic-name*	Specifies the vNIC to use for the LAN path to the boot image.
Step 7	UCS-A /org/boot-policy/lan/path # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example enters the boot policy named lab2-boot-policy, creates a LAN boot for the policy, sets the boot order to 2, creates primary and secondary paths using the vNICs named vNIC1 and vNIC2 , and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # scope boot-policy lab2-boot-policy
UCS-A /org/boot-policy* # create lan
UCS-A /org/boot-policy/lan* # set order 2
UCS-A /org/boot-policy/lan* # create path primary
UCS-A /org/boot-policy/lan/path* # set vnic vNIC1
UCS-A /org/boot-policy/lan/path* # exit
UCS-A /org/boot-policy/lan* # create path secondary
UCS-A /org/boot-policy/lan/path* # set vnic vNIC2
UCS-A /org/boot-policy/lan/path* # commit-buffer
UCS-A /org/boot-policy/lan/path #
```

What to do next

Include the boot policy in a service profile and template.

Local Devices Boot

Cisco UCS Manager allows you to boot from different local devices.

Note   For Cisco UCS M3 and higher blade and rack servers using enhanced boot order, you can select both top-level and second-level boot devices.

Local Disk Boot

If a server has a local drive, you can configure a boot policy to boot the server from the top-level local disk device or from any of the following second-level devices:

- Local LUN—Enables boot from local disk or local LUN.

- Local JBOD—Enables boot from a bootable JBOD.

- SD card—Enables boot from SD card.

- Internal USB—Enables boot for internal USB.

- External USB—Enables boot from external USB.

- Embedded Local LUN—Enables boot from the embedded local LUN on all Cisco UCS M4 and M5 servers.

- Embedded Local Disk—Enables boot from the embedded local disk on all Cisco UCS M4 and M5 servers.

| **Note** | For Cisco UCS C125 M5 Servers, if there is no separate PCIe storage controller, then do not use this option. Instead, use **Add Local Disk** option. |

| **Note** | Second-level devices are only available for Cisco UCS M3 and higher blade and rack servers using enhanced boot order. |

Virtual Media Boot

You can configure a boot policy to boot one or more servers from a virtual media device that is accessible from the server. A virtual media device mimics the insertion of a physical CD/DVD disk (read-only) or floppy disk (read-write) into a server. This type of server boot is typically used to manually install operating systems on a server.

| **Note** | Second-level devices are only available for Cisco UCS M3 and higher blade and rack servers using enhanced boot order. |

Remote Virtual Drive Boot

You can configure a boot policy to boot one or more servers from a remote virtual drive that is accessible from the server.

NVMe Boot

Beginning with release 3.2(1) Cisco UCS Manager provides the option of adding an NVMe device to the Boot policy for M5 blade and rack servers. BIOS enumerates the NVMe devices present and boots to the first NVMe device having UEFI capable OS installed on it.

Cisco Boot Optimized M.2 RAID Controller

Beginning with 4.0(4a), Cisco UCS Manager supports Cisco boot optimized M.2 RAID controller based off Marvell 88SE92xx PCIe to SATA 6Gb/s controller (UCS-M2-HWRAID). BIOS enumerates the M.2 SATA drives installed on this controller followed by the front panel SATA drives to boot from the first SATA device having UEFI capable OS installed on it

Configuring a Local Disk Boot for a Boot Policy

You can also create a local boot policy that is restricted to a service profile or service profile template. However, Cisco recommends that you create a global boot policy that can be included in multiple service profiles or service profile templates.

You can add more than one type of boot device to a boot policy. For example, you could add a virtual media boot as a secondary boot device.

Note Beginning with Release 2.2, if you want to add any top-level local storage device to the boot order, you must use **create local-any** after the **create local** command. If you have any policies from previous releases that contain a local storage device, they will be modified to use local-any during upgrade.

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 boot-policy** *policy-name*	Enters organization boot policy mode for the specified boot policy.
Step 3	UCS-A /org/boot-policy # **create storage**	Creates a storage boot for the boot policy and enters organization boot policy storage mode.
Step 4	UCS-A /org/boot-policy/storage # **create local**	Creates a local storage location and enters the boot policy local storage mode.
Step 5	UCS-A /org/boot-policy/storage/local/ # **create** {**embedded-local-jbod** | **embedded-local-lun** | **local-any** | **local-jbod** | **local-lun** | **nvme** | **sd-card** | **usb-extern** | **usb-intern** }	Specifies the type of local storage. This can be one of the following: • embedded-local-jbod —A local JBOD disk drive. • embedded-local-lun —A local LUN drive. **Note** In a setup with the Cisco boot optimized M.2 RAID controller (UCS-M2-HWRAID), select any to add the disk. Do not select Primary or Secondary.

	Command or Action	**Purpose**
		• local-any—Any type of local storage device. This option can be used in either legacy or UEFI boot mode.
		Note Cisco UCS M1 and M2 blade and rack servers using standard boot order can only use local-any.
		• local-lun—A local hard disk drive.
		• sd-card—An SD card.
		• usb-extern—An external USB card.
		• usb-intern—An internal USB card.
		For Cisco UCS M3 and higher blade and rack servers using enhanced boot order, you can select both top-level and second-level boot devices. For Cisco UCS M1 and M2 blade and rack servers using standard boot order, you can only select a top-level device.
Step 6	UCS-A /org/boot-policy/storage/local/*local-storage-device* # **set order** *order_number*	Sets the boot order for the specified local storage device. Enter an integer between 1 and 16. When using the enhanced boot order on Cisco UCS M3 servers, or M4 servers, the boot order that you define is used. For standard boot mode using the terms "primary" or "secondary" do not imply a boot order. The effective order of boot devices within the same device class is determined by the PCIe bus scan order.
Step 7	UCS-A /org/boot-policy/storage/local/*local-storage-device* # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create a boot policy named lab1-boot-policy, create a local hard disk drive boot for the policy, set the boot order to 3, and commit the transaction:

```
UCS-A# scope org /
UCS-A /org* # scope boot-policy lab1-boot-policy
UCS-A /org/boot-policy* # create storage
UCS-A /org/boot-policy/storage* # create local
UCS-A /org/boot-policy/storage/local* # create local-lun
UCS-A /org/boot-policy/storage/local/sd-card* # set order 3
UCS-A /org/boot-policy/storage/local/sd-card* # commit-buffer
UCS-A /org/boot-policy/storage/local/sd-card #
```

The following example shows how to create a local SD card boot for the service profile SP_lab1, set the boot order to 3, and commit the transaction:

```
UCS-A# scope org /
UCS-A /org* # scope service-profile SP_lab1
UCS-A /org/service-profile # create boot-definition
UCS-A /org/service-profile/boot-definition* # create storage
UCS-A /org/service-profile/boot-definition/storage* # create local
UCS-A /org/service-profile/boot-definition/storage/local* # create sd-card
UCS-A /org/service-profile/boot-definition/storage/local* # set order 3
UCS-A /org/service-profile/boot-definition/storage/local* # commit-buffer
UCS-A /org/service-profile/boot-definition/storage/local #
```

The following example shows how to create any top-level local device boot for the service profile SP_lab1, set the boot order to 3, and commit the transaction:

```
UCS-A# scope org /
UCS-A /org* # scope service-profile SP_lab1
UCS-A /org/service-profile # create boot-definition
UCS-A /org/service-profile/boot-definition* # create storage
UCS-A /org/service-profile/boot-definition/storage* # create local
UCS-A /org/service-profile/boot-definition/storage/local* # create local-any
UCS-A /org/service-profile/boot-definition/storage/local/local-any* # set order 3
UCS-A /org/service-profile/boot-definition/storage/local/local-any* # commit-buffer
UCS-A /org/service-profile/boot-definition/storage/local/local-any #
```

What to do next

Include the boot policy in a service profile and template.

Configuring a Virtual Media Boot for a Boot Policy

Note Virtual Media requires the USB to be enabled. If you modify the BIOS settings that affect the USB functionality, you also affect the Virtual Media. Therefore, Cisco recommends that you leave the following USB BIOS defaults for best performance:

- Make Device Non Bootable—set to **disabled**
- USB Idle Power Optimizing Setting—set to **high-performance**

Before you begin

Create a boot policy to contain the virtual media boot configuration.

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 boot-policy** *policy-name*	Enters organization boot policy mode for the specified boot policy.
Step 3	UCS-A /org/boot-policy # **create virtual-media** {**read-only**	**read-only-local**	**read-only-remote**	**read-write**	**read-write-drive**	**read-write-local**	**read-write-remote**}	Creates the specified virtual media boot for the boot policy and enters organization boot policy virtual media mode. This can be one of the following: • read-only—Local or remote CD/DVD. This option can be used in either legacy or UEFI boot mode. • read-only-local—Local CD/DVD. • read-only-remote—Remote CD/DVD. In a setup with M5 blade servers, if an ISO is mapped to the KVM console, use only **Remote CD/DVD** in Boot Order. • read-write—Local or remote floppy disk drive. This option can be used in either legacy or UEFI boot mode. • read-write-drive—Remote USB drive. • read-write-local—Local floppy disk drive. • read-write-remote—Remote floppy disk drive. **Note** For Cisco UCS M3 and higher blade and rack servers using enhanced boot order, you can select both top-level and second-level boot devices. For Cisco UCS M1 and M2 blade and rack servers using standard boot order, you can only select a top-level device.
Step 4	UCS-A /org/boot-policy/virtual-media # **set order** *order_number*	Sets the boot order for the virtual-media boot. Enter an integer between 1 and 16.
Step 5	UCS-A /org/boot-policy/virtual-media # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to enter the boot policy named lab3-boot-policy, create a CD/DVD virtual media boot, set the boot order to 3, and commit the transaction:

```
UCS-A# scope org /
UCS-A /org* # scope boot-policy lab3-boot-policy
```

```
UCS-A /org/boot-policy* # create virtual-media read-only-local
UCS-A /org/boot-policy/virtual-media* # set order 3
UCS-A /org/boot-policy/virtual-media* # commit-buffer
```

What to do next

Include the boot policy in a service profile and template.

Configuring a NVMe Boot for a Boot Policy

Note   NVMe boot policy is available only with Uefi boot mode.

Before you begin

Create a boot policy to contain the NVMe boot configuration.

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 boot-policy** *policy-name*	Enters organization boot policy mode for the specified boot policy.
Step 3	UCS-A /org/boot-policy # **scope storage**	Enters organization boot policy storage mode for the boot policy.
Step 4	UCS-A /org/boot-policy/storage # **scope local**	Enters local storage boot policy mode for the specified boot policy.
Step 5	UCS-A /org/boot-policy/storage/local # **create nvme**	Creates the NVMe boot for the boot policy.
Step 6	UCS-A /org/boot-policy/storage/local* # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to enter the boot policy named lab3-boot-policy, create a NVMe boot, and commit the transaction:

```
UCS-A# scope org /
UCS-A /org # scope boot-policy lab3-boot-policy
UCS-A /org/boot-policy/ # scope storage
UCS-A /org/boot-policy/storage # scope local
UCS-A /org/boot-policy/storage/local # create nvme
UCS-A /org/boot-policy/storage/local* # commit-buffer
```

What to do next

Include the boot policy in a service profile and template.

Creating a CIMC vMedia Boot Policy

You can also create a local boot policy that is restricted to a service profile or service profile template. However, Cisco recommends that you create a global boot policy that can be included in multiple service profiles or service profile templates.

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 boot-policy** *policy-name*	Creates a boot policy with the specified policy name, and enters organization boot policy mode.
Step 3	UCS-A /org/boot-policy* # **create virtual-media** *?*	Displays a list of local and remote devices to your can access and boot.
Step 4	UCS-A /org/boot-policy* # **create virtual-media** {**access** | **vMediaMappingName**}	Displays a list of local and remote devices to your can access and boot.
Step 5	UCS-A /org/boot-policy* # **create virtual-media** **read-write-remote-drive vMediaMap0**}	Creates vMedia Boot Device configuration for specified vMedia.
Step 6	UCS-A /org/boot-policy/virtual-media* # **commit-buffer**	Commits the transaction to the system configuration.
Step 7	UCS-A /org/boot-policy/virtual-media* # **show detail expand**	Displays the following boot order. **Boot virtual media:** **Order:** 1 **Access:** Read Write Remote vMedia Drive **Name:** vmediaMap0

Example

The following example creates a CIMC vMedia boot policy.

```
UCS-A# scope org /
UCS-A /org* # create boot-policy boot-policy vm-vmediamap-boot
UCS-A /org/boot-policy* # create virtual-media
```

Viewing a CIMC vMedia Mount

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *chassis_id*/*blade_id*	Enters chassis server mode for the specified server.
Step 2	UCS-A# /chassis/server # **scope cimc**	Enters CIMC mode.
Step 3	UCS-A /chassis/server/cimc # **show vmedia-mapping-list detail expand**	Displays the vMedia mapping details.

Example

The following example shows how to view a CIMC vMedia mount.

```
UCS-A# scope server 1/2
UCS-A /chassis/server # scope cimc
UCS-A /chassis/server/cimc # show vmedia-mapping-list detail expand

vMedia Mapping List:
vMedia Mapping:
Disk Id: 1
Mapping Name: cdd
Device Type: Cdd
Remote IP: 172.31.1.167
Image Path: cifs
Image File Name: ubunt-14.11-desktop-i386.iso
Mount Protocol: Cifs
Mount Status: Mounted
Error: None
Password:
User ID: Adminstrator

UCS-A /chassis/server/cimc #
```

Configuring the Boot Policy for a Local LUN

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 boot-policy** *policy-name*	Enters organization boot policy mode for the specified boot policy.
Step 3	UCS-A /org/boot-policy # **create storage**	Creates a storage boot for the boot policy and enters organization boot policy storage mode.

	Command or Action	Purpose
Step 4	UCS-A /org/boot-policy/storage # **create local**	Creates a local storage location and enters the boot policy local storage mode.
Step 5	UCS-A /org/boot-policy/storage/local/ # **create local-lun**	Specifies a local hard disk drive as the local storage.
Step 6	UCS-A /org/boot-policy/storage/local/local-lun # **create local-lun-image-path** {**primary** | **secondary**}	Specifies the boot order for the LUN that you specify. **Important** Cisco UCS Manager Release 2.2(4) does not support **secondary** boot order.
Step 7	UCS-A /org/boot-policy/storage/local/local-lun/local-lun-image-path # **set lunname** *lun_name*	Specifies the name of the LUN that you want to boot from.
Step 8	UCS-A /org/boot-policy/storage/local/*local-storage-device* # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create a boot policy named lab1-boot-policy, create a local hard disk drive boot for the policy, specify a boot order and a LUN to boot from, and commit the transaction:

```
UCS-A# scope org /
UCS-A /org* # scope boot-policy lab1-boot-policy
UCS-A /org/boot-policy* # create storage
UCS-A /org/boot-policy/storage* # create local
UCS-A /org/boot-policy/storage/local* # create local-lun
UCS-A /org/boot-policy/storage/local/local-lun # create local-lun-image-path primary
UCS-A /org/boot-policy/storage/local/local-lun/local-lun-image-path # set lunname luna
UCS-A /org/boot-policy/storage/local/local-lun/local-lun-image-path # commit-buffer
UCS-A /org/boot-policy/storage/local/local-lun/local-lun-image-path #
```

What to do next

Include the boot policy in a service profile and template.

Deleting a Boot 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 boot-policy** *policy-name*	Deletes the specified boot policy.
Step 3	UCS-A /org # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes the boot policy named boot-policy-LAN and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # delete boot-policy boot-policy-LAN
UCS-A /org* # commit-buffer
UCS-A /org #
```

UEFI Boot Parameters

UEFI boot mode for servers is dependent on information that is stored on the platform hardware. The boot entry, which contains information about the UEFI OS boot loader, is stored in the BIOS flash of the server. In Cisco UCS Manager releases earlier than Release 2.2(4), when a service profile is migrated from one server to another server, the boot loader information is not available on the destination server. Hence, the BIOS cannot load the boot loader information for the server to boot in UEFI boot mode.

Cisco UCSM Release 2.2(4) introduces UEFI boot parameters to provide the BIOS with information about the location of the UEFI OS boot loader on the destination server from where the BIOS loads it. Now, the server can use the boot loader information and boot in UEFI boot mode.

Guidelines and Limitations for UEFI Boot Parameters

- You can configure UEFI boot parameters only if the boot mode is UEFI.

- When you upgrade Cisco UCS Manager to Release 2.2(4), UEFI boot failure during service profile migration is not handled automatically. You must explicitly create the UEFI boot parameters in the target device to successfully boot to the UEFI-capable OS.

- UEFI boot parameters are supported on all M3 and higher servers that support second-level boot order.

- You can specify UEFI boot parameters for the following device types:

    - SAN LUN

- ISCSI LUN

- Local LUN

- UEFI boot parameters are specific to each operating system. You can specify UEFI boot parameters for the following operating systems:

  - VMware ESX

  - SuSE Linux

  - Microsoft Windows

  - Red Hat Enterprise Linux 7

Configuring UEFI Boot Parameters for a Local LUN

Before you begin

Ensure that the boot mode for the local LUN is set to UEFI.

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 boot-policy** *policy-name*	Enters organization boot policy mode for the specified boot policy.
Step 3	UCS-A /org/boot-policy # **scope storage**	Enters organization boot policy storage mode for the boot policy.
Step 4	UCS-A /org/boot-policy/storage # **scope local**	Enters the boot policy local storage mode.
Step 5	UCS-A /org/boot-policy/storage/local/ # **scope** {**local-any** | **local-lun** | **sd-card** | **usb-extern** | **usb-intern** }	Specifies the type of local storage. This can be one of the following: • local-any—Any type of local storage device. This option can be used in either legacy or UEFI boot mode. **Note** Cisco UCS M1 and M2 blade and rack servers using standard boot order can only use local-any. • local-lun—A local hard disk drive. • sd-card—An SD card. • usb-extern—An external USB card.

	Command or Action	Purpose
		• usb-intern—An internal USB card. **Important** The only type of local storage for which you can configure UEFI boot parameters is **local-lun**.
Step 6	UCS-A /org/boot-policy/storage/local/local-lun # **scope local-lun-image-path** {**primary** | **secondary**}	Enters the image path for the local LUN.
Step 7	UCS-A /org/boot-policy/storage/local/local-lun/local-lun-image-path # **create uefi-boot-param**	Creates UEFI boot parameters and enters UEFI boot parameter mode.
Step 8	UCS-A /org/boot-policy/storage/local/local-lun/local-lun-image-path/uefi-boot-param* # **set bootloader-name** *name*	Sets the name of the boot loader.
Step 9	UCS-A /org/boot-policy/storage/local/local-lun/local-lun-image-path/uefi-boot-param* # **set bootloader-path** *path*	Sets the path of the boot loader.
Step 10	UCS-A /org/boot-policy/storage/local/local-lun/local-lun-image-path/uefi-boot-param* # **set boot-description** "*description*"	Sets a description for the boot loader.
Step 11	UCS-A /org/boot-policy/storage/local/local-lun/local-lun-image-path/uefi-boot-param* # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create UEFI boot parameters for a local LUN, and commit the transaction:

```
UCS-A# scope org /
UCS-A /org* # scope boot-policy bp1
UCS-A /org/boot-policy* # scope storage
UCS-A /org/boot-policy/storage* # scope local
UCS-A /org/boot-policy/storage/local* # scope local-lun
UCS-A /org/boot-policy/storage/local/local-lun # scope local-lun-image-path primary
UCS-A /org/boot-policy/storage/local/local-lun/local-lun-image-path # create uefi-boot-param
UCS-A /org/boot-policy/storage/local/local-lun/local-lun-image-path/uefi-boot-param* # set
 bootloader-name grub.efi
UCS-A /org/boot-policy/storage/local/local-lun/local-lun-image-path/uefi-boot-param* # set
 bootloader-path EFI\redhat
UCS-A /org/boot-policy/storage/local/local-lun/local-lun-image-path/uefi-boot-param* # set
 boot-description "Red Hat Enterprise Linux"
UCS-A /org/boot-policy/storage/local/local-lun/local-lun-image-path/uefi-boot-param* #
commit-buffer
```

Configuring UEFI Boot Parameters for an iSCSI LUN

Before you begin

Ensure that the boot mode for the iSCSI LUN is set to UEFI.

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 boot-policy** *policy-name*	Enters organization boot policy mode for the specified boot policy.
Step 3	UCS-A /org/boot-policy # **scope iscsi**	Enters organization boot policy iSCSI mode for the boot policy.
Step 4	UCS-A /org/boot-policy/iscsi # **scope path** {**primary** | **secondary**}	Enters the image path for the iSCSI LUN.
Step 5	UCS-A /org/boot-policy/iscsi/path # **create uefi-boot-param**	Creates UEFI boot parameters and enters UEFI boot parameter mode.
Step 6	UCS-A /org/boot-policy/iscsi/path/uefi-boot-param* # **set bootloader-name** *name*	Sets the name of the boot loader.
Step 7	UCS-A /org/boot-policy/iscsi/path/uefi-boot-param* # **set bootloader-path** *path*	Sets the path of the boot loader.
Step 8	UCS-A /org/boot-policy/iscsi/path/uefi-boot-param* # **set boot-description** "*description*"	Sets a description for the boot loader.
Step 9	UCS-A /org/boot-policy/iscsi/path/uefi-boot-param* # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create UEFI boot parameters for an iSCSI LUN, and commit the transaction:

```
UCS-A# scope org /
UCS-A /org* # scope boot-policy bp2
UCS-A /org/boot-policy* # scope iscsi
UCS-A /org/boot-policy/iscsi # scope path primary
UCS-A /org/boot-policy/iscsi/path # create uefi-boot-param
UCS-A /org/boot-policy/iscsi/path/uefi-boot-param* # set bootloader-name grub.efi
UCS-A /org/boot-policy/iscsi/path/uefi-boot-param* # set bootloader-path EFI\redhat
UCS-A /org/boot-policy/iscsi/path/uefi-boot-param* # set boot-description "Red Hat Enterprise
```

```
    Linux"
UCS-A /org/boot-policy/iscsi/path/uefi-boot-param* # commit-buffer
```

Configuring UEFI Boot Parameters for a SAN LUN

Before you begin

Ensure that the boot mode for the SAN LUN is set to UEFI.

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 boot-policy** *policy-name*	Enters organization boot policy mode for the specified boot policy.
Step 3	UCS-A /org/boot-policy # **scope san**	Enters organization boot policy SAN mode for the boot policy.
Step 4	UCS-A /org/boot-policy/san # **scope san-image** {**primary**	**secondary**}	Enters the SAN image.
Step 5	UCS-A /org/boot-policy/san/san-image # **scope path** {**primary**	**secondary**}	Enters the image path for the SAN LUN.
Step 6	UCS-A /org/boot-policy/san/san-image/path # **create uefi-boot-param**	Creates UEFI boot parameters and enters UEFI boot parameter mode.
Step 7	UCS-A /org/boot-policy/san/san-image/path/uefi-boot-param* # **set bootloader-name** *name*	Sets the name of the boot loader.
Step 8	UCS-A /org/boot-policy/san/san-image/path/uefi-boot-param* # **set bootloader-path** *path*	Sets the path of the boot loader.
Step 9	UCS-A /org/boot-policy/san/san-image/path/uefi-boot-param* # **set boot-description** "*description*"	Sets a description for the boot loader.
Step 10	UCS-A /org/boot-policy/san/san-image/path/uefi-boot-param* # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create UEFI boot parameters for a SAN LUN, and commit the transaction:

```
UCS-A# scope org /
UCS-A /org* # scope boot-policy bp3
UCS-A /org/boot-policy* # scope san
UCS-A /org/boot-policy/san # scope san-image primary
UCS-A /org/boot-policy/san/san-image # scope path primary
UCS-A /org/boot-policy/san/san-image/path # create uefi-boot-param
UCS-A /org/boot-policy/san/san-image/path/uefi-boot-param* # set bootloader-name grub.efi
UCS-A /org/boot-policy/san/san-image/path/uefi-boot-param* # set bootloader-path EFI\redhat
UCS-A /org/boot-policy/san/san-image/path/uefi-boot-param* # set boot-description "Red Hat
 Enterprise Linux"
UCS-A /org/boot-policy/san/san-image/path/uefi-boot-param* # commit-buffer
```

Service Profile Configuration

Service Profiles in UCS Manager

A service profile defines a single server and its storage and networking characteristics. You can create a service profile for Cisco UCS Manager and UCS Mini. When a service profile is deployed to a server, UCS Manager automatically configures the server, adapters, fabric extenders, and fabric interconnects to match the configuration specified in the service profile.

A service profile includes four types of information:

- **Server definition**: Defines the resources (e.g. a specific server or a blade inserted to a specific chassis) that are required to apply to the profile.

- **Identity information**: Includes the UUID, MAC address for each virtual NIC (vNIC), and WWN specifications for each HBA.

- **Firmware revision specifications**: Used when a certain tested firmware revision is required to be installed or for some other reason a specific firmware is used.

- **Connectivity definition**: Configures network adapters, fabric extenders, and parent interconnects, however this information is abstract as it does not include the details of how each network component is configured.

The UCS system provides two types of service profiles: Service profiles that inherit server identity and service profiles that override server identity.

Service Profiles that Override Server Identity

This type of service profile provides the maximum amount of flexibility and control. This profile allows you to override the identity values that are on the server at the time of association and use the resource pools and policies set up in Cisco UCS Manager to automate some administration tasks.

You can disassociate this service profile from one server, then associate it with another server. This re-association can be done either manually or through an automated server pool policy. The burned-in settings, such as UUID and MAC address on the new server are overwritten with the configuration in the service profile. As a result, the change in the server is transparent to your network. You do not need to reconfigure any component or application on your network to begin using the new server.

This profile allows you to take advantage of and manage system resources through resource pools and policies, such as the following:

- Virtualized identity information, including pools of MAC addresses, WWN addresses, and UUIDs

- Ethernet and Fibre Channel adapter profile policies

- Firmware package policies

- Operating system boot order policies

Unless the service profile contains power management policies, a server pool qualification policy, or another policy that requires a specific hardware configuration, you can use the profile for any type of server in the Cisco UCS domain.

You can associate these service profiles with either a rack-mount server or a blade server. The ability to migrate the service profile depends upon whether you choose to restrict migration of the service profile.

Note If you choose not to restrict migration, Cisco UCS Manager does not perform any compatibility checks on the new server before migrating the existing service profile. If the hardware of both servers are not similar, the association might fail.

Service Profiles that Inherit Server Identity

This hardware-based service profile is the simplest to use and create. This profile uses the default values in the server and mimics the management of a rack-mounted server. It is tied to a specific server and cannot be moved or migrated to another server.

You do not need to create pools or configuration policies to use this service profile.

This service profile inherits and applies the identity and configuration information that is present at the time of association, such as the following:

- MAC addresses for the two NICs

- For a converged network adapter or a virtual interface card, the WWN addresses for the two HBAs

- BIOS versions

- Server UUID

☞

Important   The server identity and configuration information inherited through this service profile might not have the values burned into the server hardware at the manufacturer if those values were changed before this profile is associated with the server.

Guidelines and Recommendations for Service Profiles

In addition to any guidelines or recommendations that are specific to policies and pools included in service profiles and service profile templates, such as the local disk configuration policy, adhere to the following guidelines and recommendations that impact the ability to associate a service profile with a server:

Limit to the Number of vNICs that Can Be Configured on a Rack-Mount Server

You can configure up to 56 vNICs per supported adapter, such as the Cisco UCS P81E Virtual Interface Card (N2XX-ACPCI01), on any rack-mount server that is integrated with Cisco UCS Manager.

No Power Capping Support for Rack-Mount Servers

Power capping is not supported for rack servers. If you include a power control policy in a service profile that is associated with a rack-mount server, the policy is not implemented.

QoS Policy Guidelines for vNICs

You can only assign a QoS policy to a vNIC if the priority setting for that policy is not set to **fc**, which represents the Fibre Channel system class. You can configure the priority for the QoS policy with any other system class.

QoS Policy Guidelines for vHBAs

You can only assign a QoS policy to a vHBA if the priority setting for that policy is set to **fc**, which represents the Fibre Channel system class.

The Host Control setting for a QoS policy applies to vNICs only. It has no effect on a vHBA.

Inband Service Profiles

Configuring an Inband Service Profile

This procedure explains how to create an inband service profile.

✎

Note   All Cisco UCS M3 and M4 servers configured in Cisco UCS Manager GUI with an out-of-band configuration using the server CIMC from the **Equipment** tab, will automatically get an inband network (VLAN) and IPv4/IPv6 configuration as specified in the inband profile. Removing the network or IP pool name from the inband profile configuration will delete the inband configuration from the server, if the server inband configuration was derived from the inband profile.

Procedure

	Command or Action	Purpose
Step 1	UCS-A # **scope eth-uplink**	Enters the Ethernet uplink configuration mode.
Step 2	UCS-A /eth-uplink # **scope inband-profile**	Enters the inband profile configuration mode.
Step 3	UCS-A /eth-uplink/inband-profile # **set net-group-name** *vlan-group-name*	Sets the network group name for the inband profile.
Step 4	UCS-A /eth-uplink/inband-profile* # **set default-vlan-name** *vlan-name*	Sets the default VLAN for the inband profile.
Step 5	UCS-A /eth-uplink/inband-profile* # **set default-pool-name** *pool-name*	Sets the IP pool for the inband profile.
Step 6	UCS-A /eth-uplink/inband-profile* # **set monitor-interval** *interval*	Sets the monitor-interval after which failover to the subordinate fabric interconnect occurs. This can range from 1 to 5 seconds.
Step 7	UCS-A /eth-uplink/inband-profile* # **commit-buffer**	Commits the transaction to the system configuration.

Example

The example below creates the inband service profile inband-profile, sets the network group name to inband-vlan-group, sets the default VLAN to Inband_VLAN, sets the IP pool to inband_default, sets the monitor-interval to 5 seconds, and commits the transaction:

```
UCS-A #scope eth-uplink
UCS-A /eth-uplink # scope inband-profile
UCS-A /eth-uplink/inband-profile # set net-group-name inband-vlan-group
UCS-A /eth-uplink/inband-profile* # set default-vlan-name Inband_VLAN
UCS-A /eth-uplink/inband-profile* # set default-pool-name inband_default
UCS-A /eth-uplink/inband-profile* # set monitor-interval 5
UCS-A /eth-uplink/inband-profile* # commit-buffer
UCS-A /eth-uplink/inband-profile #
```

Configuring an Inband Management Service Profile

This procedure explains how to configure an inband management service profile.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** */*	Enters the organization configuration mode.
Step 2	UCS-A /org # **create service-profile***sp-name*	Creates the service profile specified and enters service profile configuration mode.

	Command or Action	**Purpose**
Step 3	UCS-A /org/service-profile # **create mgmt-iface***in-band*	Creates the management interface specified and enters management interface configuration mode
Step 4	UCS-A /org/service-profile/mgmt-iface # **create mgmt-vlan**	Creates a management VLAN and enters the management VLAN configuration mode.
Step 5	UCS-A/org/service-profile/mgmt-iface/mgmt-vlan # set **network-name** *network-name*	Sets the management VLAN network name.
Step 6	UCS-A /org/service-profile/mgmt-iface/mgmt-vlan # **create ext-pooled-ip**	Creates an external IP pool and enters the IP pool configuration mode.
Step 7	UCS-A /org/service-profile/mgmt-iface/mgmt-vlan/ext-pooled-ip # **set name** *pool-name*	Sets the name of the external IPv4 pool.
Step 8	UCS-A /org/service-profile/mgmt-iface/mgmt-vlan/ext-pooled-ip # **exit**	Exits IPv4 pool configuration mode.
Step 9	UCS-A /org/service-profile/mgmt-iface/mgmt-vlan #**create ext-pooled-ip6**	Creates an external IPv6 pool and enters the IPv6 pool configuration mode.
Step 10	UCS-A /org/service-profile/mgmt-iface/mgmt-vlan/ext-pooled-ip6 # **set name** *pool-name*	Sets the name of the external IPv6 pool.
Step 11	UCS-A /org/service-profile/mgmt-iface/mgmt-vlan/ext-pooled-ip6 # **commit-buffer**	Commits the transaction to the system configuration.

Example

The example below creates a service profile name inband_sp, configures a management interface named in-band, creates a management VLAN, sets the network name to Inband_VLAN, creates an external IPv4 pool and sets the name to inband_default, creates an external IP and an external IPv6 management pool, sets the name of both pools to inband_default, and commits the transaction:

```
UCS-A# scope org
UCS-A /org # create service-profile inband_sp
UCS-A /org/service-profile* # create mgmt-iface in-band
UCS-A /org/service-profile/mgmt-iface* # create mgmt-vlan
UCS-A /org/service-profile/mgmt-iface/mgmt-vlan* # set network-name Inband_VLAN
UCS-A /org/service-profile/mgmt-iface/mgmt-vlan* # create ext-pooled-ip
UCS-A /org/service-profile/mgmt-iface/mgmt-vlan/ext-pooled-ip* # set name inband_default
UCS-A /org/service-profile/mgmt-iface/mgmt-vlan/ext-pooled-ip* # exit
UCS-A /org/service-profile/mgmt-iface/mgmt-vlan* # create ext-pooled-ip6
UCS-A /org/service-profile/mgmt-iface/mgmt-vlan/ext-pooled-ip6* # set name inband_default
UCS-A /org/service-profile/mgmt-iface/mgmt-vlan/ext-pooled-ip6* # commit-buffer
UCS-A /org/service-profile/mgmt-iface/mgmt-vlan/ext-pooled-ip6 # exit
UCS-A /org/service-profile/mgmt-iface/mgmt-vlan # exit
UCS-A /org/service-profile/mgmt-iface # exit
```

What to do next

Associate the inband management interface service profile to a server.

Deleting the Inband Configuration from a Service Profile

This procedure explains how to delete the inband configuration from a service profile.

Note	If an inband profile is configured in Cisco UCS Manager with a default VLAN name and a default pool name, the server CIMC will automatically get an inband configuration from the inband profile within one minute after deleting the configuration from the service profile.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** */*	Enters the organization configuration mode.
Step 2	UCS-A/org # **scope service-profile** *blade1*	Enters the organization profile configuration mode.
Step 3	UCS-A/org/service-profile # **delete mgmt-iface***in-band*	Deletes the specified service profile.
Step 4	UCS-A/org/service-profile # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example scopes to the service profile blade1, deletes the management interface in-band, and commits the transaction:

```
UCS-A# scope org
UCS-A /org # scope service-profile blade1
UCS-A /org/service-profile # delete mgmt-iface in-band
UCS-A /org/service-profile* # commit-buffer
UCS-A /org/service-profile
```

Configuring Inband Management on the CIMC

This procedure explains how to configure inband management on a server CIMC to pooled IP.

Note Setting the inband management IP address to a static IP address is similar to setting the inband management IP address to the pooled IP address. The example below creates a management interface on chassis 1, server 1 named in-band, sets the IPv4 and IPv6 states to static, and commits the transaction. This example also creates a management VLAN, creates an external static IPv4, brings up the IPv4, creates an external static IPv6, brings up the IPv6, and commits the transaction:

```
UCS-A# scope server 1/1
UCS-A /chassis/server # scope cimc
UCS-A /chassis/server/cimc # create mgmt-iface in-band
UCS-A /chassis/server/cimc/mgmt-iface* # set ipv4state static
UCS-A /chassis/server/cimc/mgmt-iface* # set ipv6state static
UCS-A /chassis/server/cimc/mgmt-iface* # commit-buffer
UCS-A /chassis/server/cimc/mgmt-iface # show detail

External Management Interface:
    Mode: In Band
    Ip V4 State: Static
    Ip V6 State: Static
    Is Derived from Inband Profile: No
UCS-A /chassis/server/cimc/mgmt-iface # set
  ipv4state  IpV4State
  ipv6state  IpV6State
  mode       Mode


UCS-A /chassis/server/cimc/mgmt-iface # create mgmt-vlan
UCS-A /chassis/server/cimc/mgmt-iface/mgmt-vlan* # create ext-static-ip
UCS-A /chassis/server/cimc/mgmt-iface/mgmt-vlan/ext-static-ip* # set addr x.x.x.1
UCS-A /chassis/server/cimc/mgmt-iface/mgmt-vlan/ext-static-ip* # set subnet 255.255.255.0
UCS-A /chassis/server/cimc/mgmt-iface/mgmt-vlan/ext-static-ip* # set default-gw x.x.x.254
UCS-A /chassis/server/cimc/mgmt-iface/mgmt-vlan/ext-static-ip* # commit-buffer
UCS-A /chassis/server/cimc/mgmt-iface/mgmt-vlan/ext-static-ip # up
UCS-A /chassis/server/cimc/mgmt-iface/mgmt-vlan # create ext-static-ip6
UCS-A /chassis/server/cimc/mgmt-iface/mgmt-vlan/ext-static-ip6* # set addr xxxx:xxxx:xxxx:1::
UCS-A /chassis/server/cimc/mgmt-iface/mgmt-vlan/ext-static-ip6* # set default-gw
xxxx:xxxx:xxxx:1::0001
UCS-A /chassis/server/cimc/mgmt-iface/mgmt-vlan/ext-static-ip6* # set prefix 64
UCS-A /chassis/server/cimc/mgmt-iface/mgmt-vlan/ext-static-ip6* # commit-buffer
UCS-A /chassis/server/cimc/mgmt-iface/mgmt-vlan/ext-static-ip6 # up
UCS-A /chassis/server/cimc/mgmt-iface/mgmt-vlan # show detail expand

External Management Virtual LAN:
    Network Name:
    Id: 1

    External Management Static IP:
        IP Address: x.x.x.1
        Default Gateway: 10.193.1.254
        Subnet: 255.255.255.0
        Primary DNS IP: 0.0.0.0
        Secondary DNS IP: 0.0.0.0

    External Management Static IPv6:
        IP Address: xxxx:xxxx:xxxx:1::
        Default Gateway: xxxx:xxxx:xxxx:1::0001
        Prefix: 64
        Primary DNS IP: ::
        Secondary DNS IP: ::
```

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *chassi-numserver-num*	Enters chassis server mode for the specified server.
Step 2	UCS-A /chassis/server # **scope cimc**	Enters the CIMC configuration mode.
Step 3	UCS-A /chassis/server /chassis/server/cimc # **create mgmt-iface***in-band*	Creates the management interface specified and enters management interface configuration mode.
Step 4	UCS-A /chassis/server/cimc/mgmt-iface* # **set ipv4state pooled**	Sets IPv4 state to pooled.
Step 5	UCS-A /chassis/server/cimc/mgmt-iface *# **set ipv6state pooled**	Sets IPv6 state to pooled.
Step 6	UCS-A /chassis/server/cimc/mgmt-iface* # **create mgmt-vlan**	Creates a management VLAN and enters the management VLAN configuration mode.
Step 7	UCS-A /chassis/server/cimc/mgmt-iface/mgmt-vlan* # **set network-name** *network-name*	Sets the management VLAN network name.
Step 8	UCS-A /chassis/server/cimc/mgmt-iface/mgmt-vlan* # **create ext-pooled-ip**	Creates an external IPv4 pool and enters the IPv4 pool configuration mode.
Step 9	UCS-A /chassis/server/cimc/mgmt-iface/mgmt-vlan/ext-pooled-ip* # **set name** *pool-name*	Sets the name of the external IPv4 pool.
Step 10	UCS-A /chassis/server/cimc/mgmt-iface/mgmt-vlan/ext-pooled-ip* # **exit**	Exits IPv4 pool configuration mode.
Step 11	UCS-A /chassis/server/cimc/mgmt-iface/mgmt-vlan* #**create ext-pooled-ip6**	Creates an external IPv6 pool and enters the IPv6 pool configuration mode.
Step 12	UCS-A /chassis/server/cimc/mgmt-iface/mgmt-vlan/ext-pooled-ip6* # **set name** *pool-name*	Sets the name of the external IPv6 pool.
Step 13	UCS-A /chassis/server/cimc/mgmt-iface/mgmt-vlan/ext-pooled-ip6* # **commit-buffer**	Commits the transaction to the system configuration.

Example

The example below creates a management interface on chassis 1, server 1 named in-band, sets the IPv4 and IPv6 states to pooled, creates a management VLAN, sets the network name to Inband, creates an external IPv4 pool, sets the name to inband_default. Creates an external IPv6 pool, sets the name to inband_default, and commits the transaction:

```
UCS-A# scope server 1/1
UCS-A /chassis/server # scope cimc
UCS-A /chassis/server/cimc # create mgmt-iface in-band
UCS-A /chassis/server/cimc/mgmt-iface* # set ipv4state pooled
UCS-A /chassis/server/cimc/mgmt-iface* # set ipv6state pooled
UCS-A /chassis/server/cimc/mgmt-iface* # create mgmt-vlan
UCS-A /chassis/server/cimc/mgmt-iface/mgmt-vlan* # set network-name Inband
UCS-A /chassis/server/cimc/mgmt-iface/mgmt-vlan* # create ext-pooled-ip
UCS-A /chassis/server/cimc/mgmt-iface/mgmt-vlan/ext-pooled-ip* # set name Inband_default
UCS-A /chassis/server/cimc/mgmt-iface/mgmt-vlan/ext-pooled-ip* # exit
UCS-A /chassis/server/cimc/mgmt-iface/mgmt-vlan* # create ext-pooled-ip6
UCS-A /chassis/server/cimc/mgmt-iface/mgmt-vlan/ext-pooled-ip6* # set name Inband_default
UCS-A /chassis/server/cimc/mgmt-iface/mgmt-vlan/ext-pooled-ip6* # commit-buffer
UCS-A /chassis/server/cimc/mgmt-iface/mgmt-vlan/ext-pooled-ip6 #
```

Deleting the Inband Configuration from the CIMC

This procedure explains how to delete the inband configuration from a server CIMC.

Note If an inband profile is configured in Cisco UCS Manager with a default VLAN name and a default pool name, the server CIMC will automatically get an inband configuration from the inband profile within one minute after deleting the configuration from the service profile.

Procedure

	Command or Action	**Purpose**
Step 1	UCS-A# **scope server** *chassi-numserver-num*	Enters chassis server mode for the specified server.
Step 2	UCS-A /chassis/server # **scope cimc**	Enters the CIMC configuration mode.
Step 3	UCS-A /chassis/server /chassis/server/cimc # **delete mgmt-iface***in-band*	Deletes the specified service profile.
Step 4	UCS-A /chassis/server /chassis/server/cimc # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes the deletes the management interface named in-band from chassis1, server 1, and commits the transaction:

```
UCS-A# scope server 1/1
UCS-A /chassis/server # scope cimc
UCS-A /chassis/server/cimc # delete mgmt-iface in-band
UCS-A /chassis/server/cimc* # commit-buffer
UCS-A /chassis/server/cimc #
```

Service Profile Templates

Creating a Service Profile Template

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter / as the *org-name*.
Step 2	UCS-A /org # **create service-profile** *profile-name* {**initial-template** | **updating-template**}	Creates the specified service profile template and enters organization service profile mode.
		Enter a unique *profile-name* to identify this service profile template.
		This name can be between 2 and 32 alphanumeric characters. You cannot use spaces or any special characters other than - (hyphen), _ (underscore), : (colon), and . (period), and this name must be unique across all service profiles and service profile templates within the same organization.
Step 3	UCS-A /org/service-profile # **set bios-policy** *policy-name*	Associates the specified BIOS policy with the service profile.
Step 4	UCS-A /org/service-profile # **set boot-policy** *policy-name*	Associates the specified boot policy with the service profile.
Step 5	(Optional) UCS-A /org/service-profile # **set descr** *description*	Provides a description for the service profile.
		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 6	UCS-A /org/service-profile # **set dynamic-vnic-conn-policy** *policy-name*	Associates the specified dynamic vNIC connection policy with the service profile.
Step 7	UCS-A /org/service-profile # **set ext-mgmt-ip-state** {**none** | **pooled**}	Specifies how the management IP address will be assigned to the service profile.
		You can set the management IP address policy using the following options:

	Command or Action	Purpose
		• None-- The service profile is not assigned an IP address.
		• Pooled-- The service profile is assigned an IP address from the management IP pool.
		Note Setting the management IP address to static for a service profile template will result in an error.
Step 8	UCS-A /org/service-profile # **set host-fw-policy** *policy-name*	Associates the specified host firmware policy with the service profile.
Step 9	UCS-A /org/service-profile # **set identity** {**dynamic-uuid** {*uuid*	**derived**}	**dynamic-wwnn** {*wwnn*	**derived**}	**uuid-pool** *pool-name*	**wwnn-pool** *pool-name*}	Specifies how the server acquires a UUID or WWNN. You can do one of the following: • Create a unique UUID in the form *nnnnnnnn-nnnn-nnnn-nnnnnnnnnnnn* . • Derive the UUID from the one burned into the hardware at manufacture. • Use a UUID pool. • Create a unique WWNN in the form *hh* **:** *hh* **:** *hh* **:** *hh* **:** *hh* **:** *hh* **:** *hh* **:** *hh* . • Derive the WWNN from one burned into the hardware at manufacture. • Use a WWNN pool.
Step 10	UCS-A /org/service-profile # **set ipmi-access-profile** *profile-name*	Associates the specified IPMI access profile with the service profile.
Step 11	UCS-A /org/service-profile # **set lan-connectivity-policy-name** *policy-name*	Associates the specified LAN connectivity policy with the service profile. **Note** You cannot have a LAN connectivity policy and locally created vNICs in the same service profile. When you add a LAN connectivity policy to a service profile, any existing vNIC configuration is erased.
Step 12	UCS-A /org/service-profile # **set local-disk-policy** *policy-name*	Associates the specified local disk policy with the service profile.
Step 13	UCS-A /org/service-profile # **set maint-policy** *policy-name*	Associates the specified maintenance policy with the service profile.

	Command or Action	Purpose
Step 14	UCS-A /org/service-profile # **set mgmt-fw-policy** *policy-name*	Associates the specified management firmware policy with the service profile.
Step 15	UCS-A /org/service-profile # **set power-control-policy** *policy-name*	Associates the specified power control policy with the service profile.
Step 16	UCS-A /org/service-profile # **set san-connectivity-policy-name** *policy-name*	Associates the specified SAN connectivity policy with the service profile.
		Note You cannot have a SAN connectivity policy and locally created vHBAs in the same service profile. When you add a SAN connectivity policy to a service profile, any existing vHBA configuration is erased.
Step 17	UCS-A /org/service-profile # **set scrub-policy** *policy-name*	Associates the specified scrub policy with the service profile.
Step 18	UCS-A /org/service-profile # **set sol-policy** *policy-name*	Associates the specified serial over LAN policy with the service profile.
Step 19	UCS-A /org/service-profile # **set stats-policy** *policy-name*	Associates the specified statistics policy with the service profile.
Step 20	UCS-A /org/service-profile # **set user-label** *label-name*	Specifies the user label associated with the service profile.
Step 21	UCS-A /org/service-profile # **set vcon** {**1** | **2**} **selection** {**all** | **assigned-only** | **exclude-dynamic** | **exclude-unassigned**}	Specifies the selection preference for the specified vCon.
Step 22	UCS-A /org/service-profile # **set vcon-profile** *policy-name*	Associates the specified vNIC/vHBA placement profile with the service profile.
		Note You can either assign a vNIC/vHBA placement profile to the service profile, or set vCon selection preferences for the service profile, but you do not need to do both.
Step 23	UCS-A /org/service-profile # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create a service profile template and commit the transaction:

```
UCS-A# scope org /
UCS-A /org* # create service-profile ServTemp2 updating-template
UCS-A /org/service-profile* # set bios-policy biospol1
UCS-A /org/service-profile* # set boot-policy bootpol32
UCS-A /org/service-profile* # set  descr "This is a service profile example."
UCS-A /org/service-profile* # set dynamic-vnic-conn-policy mydynvnicconnpolicy
UCS-A /org/service-profile* # set ext-mgmt-ip-state pooled
UCS-A /org/service-profile* # set host-fw-policy ipmi-user987
UCS-A /org/service-profile* # set identity dynamic-uuid derived
UCS-A /org/service-profile* # set ipmi-access-profile ipmiProf16
UCS-A /org/service-profile* # set local-disk-policy localdiskpol33
UCS-A /org/service-profile* # set maint-policy maintpol4
UCS-A /org/service-profile* # set mgmt-fw-policy mgmtfwpol75
UCS-A /org/service-profile* # set power-control-policy powcontrpol13
UCS-A /org/service-profile* # set scrub-policy scrubpol55
UCS-A /org/service-profile* # set sol-policy solpol2
UCS-A /org/service-profile* # set stats-policy statspol4
UCS-A /org/service-profile* # set user-label mylabel
UCS-A /org/service-profile* # vcon-policy myvconnpolicy
UCS-A /org/service-profile* # commit-buffer
UCS-A /org/service-profile #
```

What to do next

- (Optional) Configure a boot definition for the service profile. Use this option only if you have not associated a boot policy with the service profile.

- Create a service profile instance from the service profile template.

Creating a Service Profile Instance from a Service Profile Template

Before you begin

Verify that there is a service profile template from which to create a service profile instance.

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 service-profile** *profile-name* **instance**	Creates the specified service profile instance and enters organization service profile mode.
		Enter a unique *profile-name* to identify this service profile template.
		This name can be between 2 and 32 alphanumeric characters. You cannot use spaces or any special characters other than - (hyphen), _ (underscore), : (colon), and . (period), and this name must be unique across all service profiles

	Command or Action	Purpose
		and service profile templates within the same organization.
Step 3	UCS-A /org/service-profile # **set src-templ-name** *profile-name*	Specifies the source service profile template to apply to the service profile instance. All configuration settings from the service profile template will be applied to the service profile instance.
Step 4	UCS-A /org/service-profile # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example creates a service profile instance named ServProf34, applies the service profile template named ServTemp2, and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # create service-profile ServProf34 instance
UCS-A /org/service-profile* # set src-templ-name ServTemp2
UCS-A /org/service-profile* # commit-buffer
UCS-A /org/service-profile #
```

What to do next

Associate the service profile to a server, rack server, or server pool.

Service Profile Tasks

Renaming a Service Profile

When you rename a service profile, the following occurs:

- Event logs and audit logs that reference the previous name for the service profile are retained under that name.

- A new audit record is created to log the rename operation.

- All records of faults against the service profile under its previous name are transferred to the new service profile name.

Note You cannot rename a service profile with pending changes.

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.
Step 3	UCS-A /org/service-profile # **rename-to** *new-profile-name*	Renames the specified service profile. When you enter this command, you are warned that you may lose all uncommitted changes in the CLI session. Type **y** to confirm that you want to continue. This name can be between 2 and 32 alphanumeric characters. You cannot use spaces or any special characters other than - (hyphen), _ (underscore), : (colon), and . (period), and this name must be unique across all service profiles and service profile templates within the same organization.
Step 4	UCS-A /org/service-profile/ # **commit-buffer**	Commits the transaction to the system configuration.

Example

This example shows how to change the name of a service profile from ServInst90 to ServZoned90 and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # scope service-profile ServInst90
UCS-A /org/service-profile* # rename-to ServZoned90
Rename is a standalone operation. You may lose any uncommitted changes in this CLI session.
Do you want to continue? (yes/no): y
UCS-A /org/service-profile* # commit-buffer
UCS-A /org/service-profile #
```

Creating a Hardware-Based Service Profile

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 # **create service-profile** *profile-name* **instance**	Creates the specified service profile instance and enters organization service profile mode.
		Enter a unique *profile-name* to identify this service profile.
		This name can be between 2 and 32 alphanumeric characters. You cannot use spaces or any special characters other than - (hyphen), _ (underscore), : (colon), and . (period), and this name must be unique across all service profiles and service profile templates within the same organization.
Step 3	UCS-A /org/service-profile # **set bios-policy** *policy-name*	Associates the specified BIOS policy with the service profile.
Step 4	UCS-A /org/service-profile # **set boot-policy** *policy-name*	Associates the specified boot policy with the service profile.
Step 5	(Optional) UCS-A /org/service-profile # **set descr** *description*	Provides a description for the service profile.
		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 6	UCS-A /org/service-profile # **set dynamic-vnic-conn-policy** *policy-name*	Associates the specified dynamic vNIC connection policy with the service profile.
Step 7	UCS-A /org/service-profile # **set ext-mgmt-ip-state** {**none**	**pooled**	**static**}	Specifies how the management IP address will be assigned to the service profile.
		You can set the management IP address policy using the following options:
		• None-- The service profile is not assigned an IP address.
		• Pooled-- The service profile is assigned an IP address from the management IP pool.
		• Static-- The service profile is assigned the configured static IP address.
Step 8	UCS-A /org/service-profile # **set host-fw-policy** *ipmi-user-name*	Associates the specified host forwarding policy with the service profile.

	Command or Action	Purpose
Step 9	UCS-A /org/service-profile # **set identity** {**dynamic-uuid** {*uuid* | **derived**} | **dynamic-wwnn** {*wwnn* | **derived**} | **uuid-pool** *pool-name* | **wwnn-pool** *pool-name*}	Specifies how the server acquires a UUID or WWNN. You can do one of the following: - Create a unique UUID in the form *nnnnnnnn-nnnn-nnnn-nnnnnnnnnnnn*. - Derive the UUID from the one burned into the hardware at manufacture. - Use a UUID pool. - Create a unique WWNN in the form *hh* : *hh* : *hh* : *hh* : *hh* : *hh* : *hh* : *hh* . - Derive the WWNN from one burned into the hardware at manufacture. - Use a WWNN pool.
Step 10	UCS-A /org/service-profile # **set ipmi-access-profile** *profile-name*	Associates the specified IPMI access profile with the service profile.
Step 11	UCS-A /org/service-profile # **set local-disk-policy** *policy-name*	Associates the specified local disk policy with the service profile.
Step 12	UCS-A /org/service-profile # **set maint-policy** *policy-name*	Associates the specified maintenance policy with the service profile.
Step 13	UCS-A /org/service-profile # **set mgmt-fw-policy** *policy-name*	Associates the specified management forwarding policy with the service profile.
Step 14	UCS-A /org/service-profile # **set power-control-policy** *policy-name*	Associates the specified power control policy with the service profile.
Step 15	UCS-A /org/service-profile # **set scrub-policy** *policy-name*	Associates the specified scrub policy with the service profile.
Step 16	UCS-A /org/service-profile # **set sol-policy** *policy-name*	Associates the specified serial over LAN policy with the service profile.
Step 17	UCS-A /org/service-profile # **set stats-policy** *policy-name*	Associates the specified statistics policy with the service profile.
Step 18	UCS-A /org/service-profile # **set user-label** *label-name*	Specifies the user label associated with the service profile.
Step 19	UCS-A /org/service-profile # **set vcon** {**1** | **2**} **selection** {**all** | **assigned-only** | **exclude-dynamic** | **exclude-unassigned**}	Specifies the selection preference for the specified vCon.
Step 20	UCS-A /org/service-profile # **set vcon-policy** *policy-name*	Associates the specified vNIC/vHBA placement policy with the service profile.

	Command or Action	Purpose
		Note You can either assign a vNIC/vHBA placement profile to the service profile, or set vCon selection preferences for the service profile, but you do not need to do both.
Step 21	UCS-A /org/service-profile # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create a service profile instance and commit the transaction:

```
UCS-A# scope org /
UCS-A /org* # create service-profile ServInst90 instance
UCS-A /org/service-profile* # set bios-policy biospol1
UCS-A /org/service-profile* # set boot-policy bootpol32
UCS-A /org/service-profile* # set  descr "This is a service profile example."
UCS-A /org/service-profile* # set ext-mgmt-ip-state pooled
UCS-A /org/service-profile* # set host-fw-policy ipmi-user987
UCS-A /org/service-profile* # set identity dynamic-uuid derived
UCS-A /org/service-profile* # set ipmi-access-profile ipmiProf16
UCS-A /org/service-profile* # set local-disk-policy localdiskpol33
UCS-A /org/service-profile* # set maint-policy maintpol4
UCS-A /org/service-profile* # set mgmt-fw-policy mgmtfwpol75
UCS-A /org/service-profile* # set power-control-policy powcontrpol13
UCS-A /org/service-profile* # set scrub-policy scrubpol55
UCS-A /org/service-profile* # set sol-policy solpol2
UCS-A /org/service-profile* # set stats-policy statspol4
UCS-A /org/service-profile* # set user-label mylabel
UCS-A /org/service-profile* # vcon-policy myvconnpolicy
UCS-A /org/service-profile* # commit-buffer
UCS-A /org/service-profile #
```

What to do next

- (Optional) Configure a boot definition for the service profile. Use this option only if you have not associated a boot policy with the service profile.

- Associate the service profile with a blade server, server pool, or rack server.

Creating vNIC Pairs on a Service Profile

Procedure

	Command or Action	Purpose
Step 1	UCS-A /org # **scope** *org-name* .	Enters the organization mode for the specified organization. To enter the root organization mode enter "org" as the org-name.

	Command or Action	**Purpose**
Step 2	UCS-A /org # **scope service-profile** *service profile name* .	Enters the name of the service profile where you want to create the vNIC pair.
Step 3	UCS-A /org # scope service-profile **create vnic** *eth0*.	Assigns a name to the vNIC for creating the redundancy pair.
Step 4	UCS-A /org/service-profile/vnic* # **set template-name***vNIC-primary* .	Specifies to use the Primary vNIC template that you can link to a Secondary vNIC template to create a vNIC pair at the service profile level.
Step 5	UCS-A /org/service-profile/vnic* # **exit** .	Exits the Primary vNIC template to use to create the vNIC pair.
		Note You can now create the peer vNIC to link to vNIC eth0. Ensure to commit the transaction after linking vNIC eth0 to vNIC eth1 to create the vNIC pair.
Step 6	UCS-A /org/service-profile # **create vnic** *eth1*.	Assigns a name to the vNIC for creating the peer vNIC to create the pair that you link to vNIC eth0.
Step 7	UCS-A /org/service-profile/vnic* **set template-name** *vNIC secondary* .	Specifies to use the Secondary vNIC template as the peer template to a Primary vNIC template to create a vNIC pair that you can use at the service profile level.
Step 8	UCS-A /org/service-profile/vnic* # **exit** .	Exits the Secondary vNIC template to use to create the vNIC pair.
Step 9	UCS-A /org/service-profile* # **commit-buffer** .	Commits the transaction to the system configuration.

Example

The following example creates a vNIC redundancy pair from a service profile and commits the transaction:

```
UCS-A # scope org
UCS-A /org # scope service-profile test-sp
UCS-A /org/service-profile # create vNIC eth0
UCS-A /org/service-profile/vnic* # set template-name vNIC-primary
UCS-A /org/service-profile/vnic* # exit
UCS-A /org/service-profile* # create vNIC eth1
UCS-A /org/service-profile/vnic* # set template-name vNIC-secondary
UCS-A /org/service-profile/vnic* # exit
UCS-A /org/service-profile* # commit-buffer
UCS-A /org/service-profile #
```

Configuring a vNIC for a Service Profile

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter / as the *org-name*.
Step 2	UCS-A /org # **scope service-profile** *profile-name*	Enters organization service profile mode for the specified service profile.
Step 3	UCS-A /org/service-profile # **create vnic** *vnic-name* [**eth-if** *eth-if-name*] [**fabric** {**a** | **b**}]	Creates a vNIC for the specified service profile and enters organization service profile vNIC mode.
Step 4	UCS-A /org/service-profile/vnic # **set adapter-policy** *policy-name*	Specifies the adapter policy to use for the vNIC.
Step 5	UCS-A /org/service-profile/vnic # **set fabric** {**a** | **a-b** | **b** | **b-a**}	Specifies the fabric to use for the vNIC. If you did not specify the fabric when creating the vNIC template in Step 3, you have the option to specify it with this command. If you want this vNIC to be able to access the second fabric interconnect if the default one is unavailable, choose **a-b** (A is the primary) or **b-a** (B is the primary) . **Note** Do not enable fabric failover for the vNIC under the following circumstances: • If the Cisco UCS domain is running in Ethernet Switch Mode. vNIC fabric failover is not supported in that mode. If all Ethernet uplinks on one fabric interconnect fail, the vNICs do not fail over to the other. • If you plan to associate this vNIC to a server with an adapter that does not support fabric failover, such as the Cisco UCS 82598KR-CI 10-Gigabit Ethernet Adapter. If you do so, Cisco UCS Manager generates a configuration fault when you associate the service profile with the server.

	Command or Action	**Purpose**
Step 6	UCS-A /org/service-profile/vnic # **set identity** {**dynamic-mac** {*mac-addr* | **derived**} | **mac-pool** *mac-pool-name*}	Specifies the identity (MAC address) for the vNIC. You can set the identity using one of the following options: • Create a unique MAC address in the form *nn* : *nn* : *nn* : *nn* : *nn* : *nn* . • Derive the MAC address from one burned into the hardware at manufacture. • Assign a MAC address from a MAC pool.
Step 7	UCS-A /org/service-profile/vnic # **set mtu** *size-num*	The maximum transmission unit, or packet size, that this vNIC accepts. Enter an integer between 1500 and 9216. **Note** If the vNIC has an associated QoS policy, the MTU specified here must be equal to or less than the MTU specified in the associated QoS system class. If this MTU value exceeds the MTU value in the QoS system class, packets might get dropped during data transmission.
Step 8	UCS-A /org/service-profile/vnic # **set nw-control-policy** *policy-name*	The network control policy the vNIC should use.
Step 9	UCS-A /org/service-profile/vnic # **set order** {*order-num* | **unspecified**}	Specifies the relative order for the vNIC.
Step 10	UCS-A /org/service-profile/vnic # **set pin-group** *group-name*	The LAN pin group the vNIC should use.
Step 11	UCS-A /org/service-profile/vnic # **set qos-policy** *policy-name*	The quality of service policy the vNIC should use.
Step 12	UCS-A /org/service-profile/vnic # **set stats-policy** *policy-name*	The statistics collection policy the vNIC should use.
Step 13	UCS-A /org/service-profile/vnic # **set template-name** *policy-name*	Specifies the dynamic vNIC connectivity policy to use for the vNIC.
Step 14	UCS-A /org/service-profile/vnic # **set vcon** {**1** | **2** | **3** | **4** | **any**}	Assigns the vNIC to the specified vCon. Use the **any** keyword to have Cisco UCS Manager automatically assign the vNIC.
Step 15	UCS-A /org/service-profile/vnic # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example configures a vNIC for a service profile and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # scope service-profile ServInst90
UCS-A /org/service-profile* # create vnic vnic3 fabric a
UCS-A /org/service-profile/vnic* # set adapter-policy AdaptPol2
UCS-A /org/service-profile/vnic* # set fabric a-b
UCS-A /org/service-profile/vnic* # set identity mac-pool MacPool3
UCS-A /org/service-profile/vnic* # set mtu 8900
UCS-A /org/service-profile/vnic* # set nw-control-policy ncp5
UCS-A /org/service-profile/vnic* # set order 0
UCS-A /org/service-profile/vnic* # set pin-group EthPinGroup12
UCS-A /org/service-profile/vnic* # set qos-policy QosPol5
UCS-A /org/service-profile/vnic* # set stats-policy StatsPol2
UCS-A /org/service-profile/vnic* # set template-name VnicConnPol3
UCS-A /org/service-profile/vnic* # set set vcon any
UCS-A /org/service-profile/vnic* # commit-buffer
UCS-A /org/service-profile/vnic #
```

Configuring a vHBA for a Service Profile

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.
Step 3	UCS-A /org/service-profile # **create vhba** *vhba-name* [**fabric** {**a** | **b**}] [**fc-if** *fc-if-name*]	Creates a vHBA for the specified service profile and enters organization service profile vHBA mode.
Step 4	UCS-A /org/service-profile/vhba # **set adapter-policy** *policy-name*	Specifies the adapter policy to use for the vHBA.
Step 5	UCS-A /org/service-profile/vhba # **set admin-vcon** {**1** | **2** | **any**}	Assigns the vHBA to one or all virtual network interface connections.
Step 6	UCS-A /org/service-profile/vhba # **set identity** {**dynamic-wwpn** {*wwpn* | **derived**} | **wwpn-pool** *wwn-pool-name*}	Specifies the WWPN for the vHBA. You can set the storage identity using one of the following options: • Create a unique WWPN in the form *hh***:***hh***:***hh***:***hh***:***hh***:***hh***:***hh***:***hh*. You can specify a WWPN in the range from 20:00:00:00:00:00:00:00 to 20:FF:FF:FF:FF:FF:FF:FF or from

	Command or Action	Purpose
		50:00:00:00:00:00:00:00 to 5F:FF:FF:FF:FF:FF:FF:FF.
		If you want the WWPN to be compatible with Cisco MDS Fibre Channel switches, use the WWPN template **20:00:00:25:B5:XX:XX:XX**.
		• Derive the WWPN from one burned into the hardware at manufacture.
		• Assign a WWPN from a WWN pool.
Step 7	UCS-A /org/service-profile/vhba # **set max-field-size** *size-num*	Specifies the maximum size of the Fibre Channel frame payload (in bytes) that the vHBA supports.
Step 8	UCS-A /org/service-profile/vhba # **set order** {*order-num*	**unspecified**}	Specifies the PCI scan order for the vHBA.
Step 9	UCS-A /org/service-profile/vhba # **set pers-bind** {**disabled**	**enabled**}	Disables or enables persistent binding to Fibre Channel targets.
Step 10	UCS-A /org/service-profile/vhba # **set pin-group** *group-name*	Specifies the SAN pin group to use for the vHBA.
Step 11	UCS-A /org/service-profile/vhba # **set qos-policy** *policy-name*	Specifies the QoS policy to use for the vHBA.
Step 12	UCS-A /org/service-profile/vhba # **set stats-policy** *policy-name*	Specifies the statistics threshold policy to use for the vHBA.
Step 13	UCS-A /org/service-profile/vhba # **set template-name** *policy-name*	Specifies the vHBA template to use for the vHBA.
Step 14	UCS-A /org/service-profile/vhba # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example configures a vHBA for a service profile and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # scope service-profile ServInst90
UCS-A /org/service-profile* # create vhba vhba3 fabric b
UCS-A /org/service-profile/vhba* # set adapter-policy AdaptPol2
UCS-A /org/service-profile/vhba* # set admin-vcon any
UCS-A /org/service-profile/vhba* # set identity wwpn-pool SanPool7
UCS-A /org/service-profile/vhba* # set max-field-size 2112
UCS-A /org/service-profile/vhba* # set order 0
UCS-A /org/service-profile/vhba* # set pers-bind enabled
UCS-A /org/service-profile/vhba* # set pin-group FcPinGroup12
UCS-A /org/service-profile/vhba* # set qos-policy QosPol5
UCS-A /org/service-profile/vhba* # set stats-policy StatsPol2
```

```
UCS-A /org/service-profile/vhba* # set template-name SanConnPol3
UCS-A /org/service-profile/vhba* # commit-buffer
UCS-A /org/service-profile/vhba #
```

Configuring a Local Disk for a Service Profile

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 # **create local-disk-config**	Creates a local disk configuration for the service profile and enters organization service profile local disk configuration mode.
Step 4	(Optional) UCS-A /org/service-profile/local-disk-config # **set descr** *description*	Provides a description for the local disk configuration.
Step 5	UCS-A /org/service-profile/local-disk-config # **set mode** {**any-configuration** | **no-local-storage** | **no-raid** | **raid-0-striped** | **raid-1-mirrored** | **raid-5-striped-parity** | **raid-6-striped-dual-parity** | **raid-10-mirrored-and-striped**}	Specifies the mode for the local disk.
Step 6	UCS-A /org/service-profile/local-disk-config # **create partition**	Creates a partition for the local disk and enters organization service profile local disk configuration partition mode.
Step 7	(Optional) UCS-A /org/service-profile/local-disk-config/partition # **set descr** *description*	Provides a description for the partition.
Step 8	UCS-A /org/service-profile/local-disk-config/partition # **set size** {*size-num* | **unspecified**}	Specifies the partition size in MBytes.
Step 9	UCS-A /org/service-profile/local-disk-config/partition # **set type** {**ext2** | **ext3** | **fat32** | **none** | **ntfs** | **swap**}	Specifies the partition type.
Step 10	UCS-A /org/service-profile/local-disk-config/partition # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example configures a local disk for a service profile and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # scope service-profile ServInst90
UCS-A /org/service-profile # scope boot-definition
UCS-A /org/service-profile # create local-disk-config
UCS-A /org/service-profile/local-disk-config* # set mode raid-1-mirrored
UCS-A /org/service-profile/local-disk-config* # create partition
UCS-A /org/service-profile/local-disk-config/partition* # set size 1000000
UCS-A /org/service-profile/local-disk-config/partition* # set type ntfs
UCS-A /org/service-profile/local-disk-config/partition* # commit-buffer
UCS-A /org/service-profile/local-disk-config/partition #
```

Configuring Serial over LAN for a Service Profile

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.
Step 3	UCS-A /org/service-profile # **create sol-config**	Creates a serial over LAN configuration for the service profile and enters organization service profile SoL configuration mode.
Step 4	UCS-A /org/service-profile/sol-config # {**disable** | **enable**}	Disables or enables the serial over LAN configuration for the service profile.
Step 5	(Optional) UCS-A /org/service-profile/sol-config # **set descr** *description*	Provides a description for the serial over LAN configuration.
Step 6	UCS-A /org/service-profile/sol-config # **set speed** {**115200** | **19200** | **38400** | **57600** | **9600**}	Specifies the serial baud rate.
Step 7	UCS-A /org/service-profile/sol-config # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example configures serial over LAN for the service profile named ServInst90 and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # scope service-profile ServInst90
UCS-A /org/service-profile # create sol-config
UCS-A /org/service-profile/sol-config* # enable
UCS-A /org/service-profile/sol-config* # set descr "Sets serial over LAN to 9600 baud."
UCS-A /org/service-profile/sol-config* # set speed 9600
UCS-A /org/service-profile/sol-config* # commit-buffer
UCS-A /org/service-profile/sol-config #
```

Service Profile Association

Associating a Service Profile with a Blade Server or Server Pool

Follow this procedure if you did not associate the service profile with a blade server or server pool when you created it, or to change the blade server or server pool with which a service profile is associated.

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 # **associate** {**server** *chassis-id / slot-id* | **server-pool** *pool-name qualifier*} [**restrict-migration**]	Associates the service profile with a single server, or to the specified server pool with the specified server pool policy qualifications. Adding the optional **restrict-migration** keyword prevents the service profile from being migrated to another server.
Step 4	UCS-A /org/service-profile # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example associates the service profile named ServProf34 with the server in slot 4 of chassis 1 and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # scope service-profile ServProf34
UCS-A /org/service-profile* # associate server 1/4
UCS-A /org/service-profile* # commit-buffer
UCS-A /org/service-profile #
```

Associating a Service Profile with a Rack Server

Follow this procedure if you did not associate the service profile with a rack server when you created it, or to change the rack server with which a service profile is associated.

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 # **associate server** *serv-id* [**restrict-migration**]	Associates the service profile with the specified rack server. Adding the optional the restrict-migration command prevents the service profile from being migrated to another server.
Step 4	UCS-A /org/service-profile # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example associates the service profile named ServProf34 with the rack server 1 and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # scope service-profile ServProf34
UCS-A /org/service-profile* # associate server 1
UCS-A /org/service-profile* # commit-buffer
UCS-A /org/service-profile #
```

Disassociating a Service Profile from a Server or Server Pool

This procedure covers disassociating a service profile from a blade server, rack 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 # **disassociate**	Disassociates the service profile from the server or server pool.
Step 4	UCS-A /org/service-profile # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example disassociates the service profile named ServProf34 from the server to which it was associated and commits the transaction:

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

Service Profile Boot Definition

Configuring a Boot Definition for a Service Profile

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.
Step 3	UCS-A /org/service-profile # **create boot-definition**	Creates a boot definition for the service profile and enters organization service profile boot definition mode.
Step 4	(Optional) UCS-A /org/service-profile/boot-definition # **set descr** *description*	Provides a description for the boot definition.
Step 5	(Optional) UCS-A /org/service-profile/boot-definition # **set reboot-on-update** {**no** | **yes**}	Specifies whether to automatically reboot all servers that use this boot definition after changes are made to the boot order. By default, the reboot on update option is disabled.
Step 6	UCS-A /org/service-profile/boot-definition # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example configures a boot definition for a service profile and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # scope service-profile ServInst90
UCS-A /org/service-profile* # create boot-definition
UCS-A /org/service-profile/boot-definition* # set descr "This boot definition reboots on
update."
UCS-A /org/service-profile/boot-definition* # set reboot-on-update yes
UCS-A /org/service-profile/boot-definition* # commit-buffer
UCS-A /org/service-profile/boot-definition #
```

What to do next

Configure one or more of the following boot options for the boot definition and set their boot order:

- **LAN Boot** —Boots from a centralized provisioning server. It is frequently used to install operating systems on a server from that server.

  If you choose the LAN Boot option, continue to Configuring a LAN Boot for a Service Profile Boot Definition , on page 185.

- **Storage Boot** — Boots from an operating system image on the SAN. You can specify a primary and a secondary SAN boot. If the primary boot fails, the server attempts to boot from the secondary.

  We recommend that you use a SAN boot, because it offers the most service profile mobility within the system. If you boot from the SAN, when you move a service profile from one server to another, the new server boots from exactly the same operating system image. Therefore, the new server appears to be exactly the same server to the network.

  If you choose the Storage Boot option, continue to Configuring a Storage Boot for a Service Profile Boot Definition , on page 186.

- **Virtual Media Boot** —Mimics the insertion of a physical CD into a server. It is typically used to manually install operating systems on a server.

  If you choose the Virtual Media boot option, continue to Configuring a Virtual Media Boot for a Service Profile Boot Definition , on page 188.

Configuring a LAN Boot for a Service Profile Boot Definition

Before you begin

Configure a boot definition for a service profile.

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 # **scope boot-definition**	Enters organization service profile boot definition mode.
Step 4	UCS-A /org/service-profile/boot-definition # **create lan**	Creates a LAN boot for the service profile boot definition and enters service profile boot definition LAN mode.
Step 5	UCS-A /org/service-profile/boot-definition/lan # **set order** {**1** | **2** | **3** | **4**}	Specifies the boot order for the LAN boot.
Step 6	UCS-A /org/service-profile/boot-definition/lan # **create path** {**primary** | **secondary**}	Creates a primary or secondary LAN boot path and enters service profile boot definition LAN path mode.
Step 7	UCS-A /org/service-profile/boot-definition/lan/path # **set vnic** *vnic-name*	Specifies the vNIC to use for the LAN image path.
Step 8	UCS-A /org/service-profile/boot-definition/lan/path # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example enters the service profile named ServInst90, creates a LAN boot for the service profile boot definition, sets the boot order to 2, creates a primary path, and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # scope service-profile ServInst90
UCS-A /org/service-profile* # scope boot-definition
UCS-A /org/service-profile/boot-definition* # create lan
UCS-A /org/service-profile/boot-definition/lan* # set order 2
UCS-A /org/service-profile/boot-definition/lan* # create path primary
UCS-A /org/service-profile/boot-definition/lan/path* # set vnic vnic3
UCS-A /org/service-profile/boot-definition/lan/path* # commit-buffer
UCS-A /org/service-profile/boot-definition/lan/path #
```

Configuring a Storage Boot for a Service Profile Boot Definition

Before you begin

Configure a boot definition for a service profile.

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.
Step 3	UCS-A /org/service-profile # **scope boot-definition**	Enters organization service profile boot definition mode.
Step 4	UCS-A /org/service-profile/boot-definition # **create storage**	Creates a storage boot for the service profile boot definition and enters service profile boot definition storage mode.
Step 5	UCS-A /org/service-profile/boot-definition/storage # **set order** {**1** | **2** | **3** | **4**}	Specifies the boot order for the storage boot.
Step 6	UCS-A /org/service-profile/boot-definition/storage # **create** {**local** | **san-image** {**primary** | **secondary**}}	Creates a local storage boot or a SAN image boot. If a SAN image boot is created, it enters service profile boot definition storage SAN image mode.
Step 7	UCS-A /org/service-profile/boot-definition/storage/san-image # **create path** {**primary** | **secondary**}	Creates a primary or secondary SAN image path and enters service profile boot definition storage SAN image path mode. When using the enhanced boot order on Cisco UCS M3 servers, or M4 servers, the boot order that you define is used. For standard boot mode using the terms "primary" or "secondary" do not imply a boot order. The effective order of boot devices within the same device class is determined by the PCIe bus scan order.
Step 8	UCS-A /org/service-profile/boot-definition/storage/san-image/path # **set lun** *lun-num*	Specifies the LUN used for the SAN image path.
Step 9	UCS-A /org/service-profile/boot-definition/storage/san-image/path # **set vhba** *vhba-name*	Specifies the vHBA used for the SAN image path.
Step 10	UCS-A /org/service-profile/boot-definition/storage/san-image/path # **set wwn** *wwn-num*	Specifies the WWN used for the SAN image path.
Step 11	UCS-A /org/service-profile/boot-definition/storage/san-image/path # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example enters the service profile named ServInst90, creates a storage boot for the service profile boot definition, sets the boot order to 2, creates a primary path, and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # scope service-profile ServInst90
UCS-A /org/service-profile* # scope boot-definition
UCS-A /org/service-profile/boot-definition* # create storage
UCS-A /org/service-profile/boot-definition/storage* # create san-image primary
UCS-A /org/service-profile/boot-definition/storage* # set order 2
UCS-A /org/service-profile/boot-definition/storage/san-image* # create path primary
UCS-A /org/service-profile/boot-definition/storage/san-image/path* # set lun 27512
UCS-A /org/service-profile/boot-definition/storage/san-image/path* # set vhba vhba3
UCS-A /org/service-profile/boot-definition/storage/san-image/path* # set wwn
20:00:00:00:20:00:00:23
UCS-A /org/service-profile/boot-definition/storage/san-image/path* # commit-buffer
UCS-A /org/service-profile/boot-definition/storage/san-image/path #
```

Configuring a Virtual Media Boot for a Service Profile Boot Definition

Before you begin

Configure a boot definition for a service profile.

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.
Step 3	UCS-A /org/service-profile # **scope boot-definition**	Enters organization service profile boot definition mode.
Step 4	UCS-A /org/service-profile/boot-definition # **create virtual-media** {**read-only** | **read-write**}	Creates a read-only or read-write virtual media boot for the service profile boot definition and enters service profile boot definition virtual media mode.
Step 5	UCS-A /org/service-profile/boot-definition/virtual-media # **set order** {**1** | **2** | **3** | **4**}	Specifies the boot order for the virtual media boot.
Step 6	UCS-A /org/service-profile/boot-definition/virtual-media # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example enters the service profile named ServInst90, creates a virtual media boot with read-only privileges for the service profile boot definition, sets the boot order to 3, and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # scope service-profile ServInst90
UCS-A /org/service-profile* # scope boot-definition
UCS-A /org/service-profile/boot-definition* # create virtual-media read-only
UCS-A /org/service-profile/boot-definition/virtual-media* # set order 3
UCS-A /org/service-profile/boot-definition/virtual-media* # commit-buffer
UCS-A /org/service-profile/boot-definition/virtual-media #
```

Deleting a Boot Definition for a Service Profile

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 the specified service.
Step 3	UCS-A /org/service-profile # **delete boot-definition**	Deletes the boot definition for the service profile.
Step 4	UCS-A /org/service-profile # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes the boot definition for a service profile and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # scope service-profile ServInst90
UCS-A /org/service-profile # delete boot-definition
UCS-A /org/service-profile* # commit-buffer
UCS-A /org/service-profile #
```

Fibre Channel Zoning for a Service Profile

Configuring a vHBA Initiator Group with an Existing Storage Connection Policy

This procedure assumes that you want to use an existing global Fibre Channel storage connection policy. If you want to create a storage connection policy definition just for this service profile, see Configuring a vHBA Initiator Group with a local Storage Connection Policy Definition, on page 191.

For information about how to create a global Fibre Channel storage connection policy that is available to all service profiles, see Creating a Fibre Channel Storage Connection 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** *profile-name*	Enters organization service profile mode for the specified service profile.
Step 3	UCS-A /org/service-profile # **create initiator-group** *group-name*	Creates the specified initiator group for Fibre Channel zoning and enters service profile initiator group mode.
Step 4	UCS-A /org/service-profile/initiator-group # **create initiator** *vhba-name*	Creates the specified vHBA initiator in the initiator group. If desired, repeat this step to add a second vHBA initiator to the group.
Step 5	UCS-A /org/service-profile/initiator-group # **set storage-connection-policy** *policy-name*	Associates the specified storage connection policy with the service profile.
Step 6	UCS-A /org/service-profile # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example configures a vHBA initiator group named initGroupZone1 with two vHBA initiators for a service profile named ServInst90, includes an existing Fibre Channel storage connection policy, and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # scope service-profile ServInst90
UCS-A /org/service-profile # create initiator-group initGroupZone1
UCS-A /org/service-profile/initiator-group* # create initiator vhba1
UCS-A /org/service-profile/initiator-group* # create initiator vhba2
UCS-A /org/service-profile/initiator-group* # set storage-connection-policy scpolicyZone1
UCS-A /org/service-profile* # commit-buffer
UCS-A /org/service-profile #
```

Configuring a vHBA Initiator Group with a local Storage Connection Policy Definition

This procedure assumes that you want to create a local Fibre Channel storage connection policy for a service profile. If you want to use an existing storage connection policy, see Configuring a vHBA Initiator Group with an Existing Storage Connection Policy, on page 190.

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 # **create initiator-group** *group-name*	Creates the specified initiator group for Fibre Channel zoning and enters service profile initiator group mode.
Step 4	UCS-A /org/service-profile/initiator-group # **create initiator** *vhba-name*	Creates the specified vHBA initiator in the vHBA initiator group. If desired, repeat this step to add a second vHBA initiator to the group.
Step 5	UCS-A /org/service-profile/initiator-group # **create storage-connection-def** *policy-name*	Creates the specified storage connection policy definition and enters storage connection definition mode.
Step 6	UCS-A /org/service-profile/initiator-group/storage-connection-def # **create storage-target** *wwpn*	Creates a storage target endpoint with the specified WWPN, and enters storage target mode.
Step 7	UCS-A /org/service-profile/initiator-group/storage-connection-def/storage-target # **set target-path** {**a** | **b**}	Specifies which fabric interconnect is used for communications with the target endpoint.
Step 8	UCS-A /org/service-profile/initiator-group/storage-connection-def/storage-target # **set target-vsan** *vsan*	Specifies which VSAN is used for communications with the target endpoint.
Step 9	UCS-A /org/service-profile/initiator-group # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example configures a vHBA initiator group named initGroupZone1 with two vHBA initiators for a service profile named ServInst90, configures a local storage connection policy definition named scPolicyZone1, and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # scope service-profile ServInst90
UCS-A /org/service-profile # create initiator-group initGroupZone1
UCS-A /org/service-profile/initiator-group* # create initiator vhba1
UCS-A /org/service-profile/initiator-group* # create initiator vhba2
UCS-A /org/service-profile/initiator-group* # create storage-connection-def scPolicyZone1
UCS-A /org/service-profile/initiator-group/storage-connection-def* # create storage-target

20:10:20:30:40:50:60:70
UCS-A /org/service-profile/initiator-group/storage-connection-def/storage-target* # set
target-path a
UCS-A /org/service-profile/initiator-group/storage-connection-def/storage-target* # set
target-vsan default
UCS-A /org/service-profile/initiator-group* # commit-buffer
UCS-A /org/service-profile/initiator-group #
```

Service Profile Template Management

Setting the Asset Tag Value

Procedure

	Command or Action	**Purpose**
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter / as the *org-name*.
Step 2	UCS-A /org # **scope service-profile** *sp-name*	Enters the service profile configuration mode for the specified service profile.
Step 3	UCS-A /org/service-profile # **set asset-tag value** *at-name*	Specifies the asset tag name for the server.
Step 4	UCS-A /org/service-profile # **commit buffer**	Commits the transaction to the system configuration.
Step 5	Reboot the server for the new asset tag to take effect in the BIOS.	

Example

The following example shows how to configure asset tag for a server:

```
UCS-A# scope org /
UCS-A /org* # scope service-profile sp1
UCS-A /org/service-profile* # set asset-tag value EXAMPLE
UCS-A /org/service-profile* # commit-buffer
UCS-A /org/service-profile #
```

Viewing the Server Asset Tag

Procedure

	Command or Action	**Purpose**
Step 1	UCS-A /org# **scope server** *server-name*	Enters the service name.
Step 2	UCS-A /chassis/server# **show detail**	Displays the server asset tag.

Example

The following example shows how to display the asset tag of a server:

```
UCS-A# scope server 1/5
UCS-A/chassis/server # show detail
Server:
    Slot: 5
    Name:
    User Label:
    Overall Status: Config
    Oper Qualifier: N/A
    Service Profile: asset-tag
    Asset Tag: EXAMPLE
    Association: Associated
```

Resetting the UUID Assigned to a Service Profile from a Pool in a Service Profile Template

If you change the UUID suffix pool assigned to an updating service profile template, Cisco UCS Manager does not change the UUID assigned to a service profile created with that template. If you want Cisco UCS Manager to assign a UUID from the newly assigned pool to the service profile, and therefore to the associated server, you must reset the UUID. You can only reset the UUID assigned to a service profile and its associated server under the following circumstances:

- The service profile was created from an updating service profile template and includes a UUID assigned from a UUID suffix pool.

- The UUID suffix pool name is specified in the service profile. For example, the pool name is not empty.

- The UUID value is not 0, and is therefore not derived from the server hardware.

Procedure

	Command or Action	**Purpose**
Step 1	UCS-A# **scope org** *org-name*	Enters the command mode for the organization for which you want to reset the UUID. If the system does not include multi-tenancy, type / as the *org-name* to enter the root organization.

	Command or Action	Purpose
Step 2	UCS-A /org # **scope service-profile** *profile-name*	Enters the service profile that requires the UUID for the associated server to be reset to a different UUID suffix pool.
Step 3	UCS-A /org/service-profile # **set identity dynamic-uuid derived**	Specifies that the service profile will obtain a UUID dynamically from a pool.
Step 4	UCS-A /org/service-profile # **commit-buffer**	Commits the transaction to the system configuration.

Example

This example resets the UUID of a service profile to a different UUID suffix pool:

```
UCS-A# scope org /
UCS-A /org # scope service-profile ServInst90
UCS-A /org/service-profile # set identity dynamic-uuid derived
UCS-A /org/service-profile* # commit-buffer
UCS-A /org/service-profile #
```

Resetting the MAC Address Assigned to a vNIC from a Pool in a Service Profile Template

If you change the MAC pool assigned to an updating service profile template, Cisco UCS Manager does not change the MAC address assigned to a service profile created with that template. If you want Cisco UCS Manager to assign a MAC address from the newly assigned pool to the service profile, and therefore to the associated server, you must reset the MAC address. You can only reset the MAC address assigned to a service profile and its associated server under the following circumstances:

- The service profile was created from an updating service profile template and includes a MAC address assigned from a MAC pool.

- The MAC pool name is specified in the service profile. For example, the pool name is not empty.

- The MAC address value is not 0, and is therefore not derived from the server hardware.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the command mode for the organization that contains the service profile for which you want to reset the MAC address. If the system does not include multi-tenancy, type / as the *org-name* to enter the root organization.
Step 2	UCS-A /org # **scope service-profile** *profile-name*	Enters the command mode for the service profile that requires the MAC address of the associated server to be reset to a different MAC address.

	Command or Action	Purpose
Step 3	UCS-A /org/service-profile # **scope vnic** *vnic-name*	Enters the command mode for the vNIC for which you want to reset the MAC address.
Step 4	UCS-A /org/service-profile/vnic # **set identity dynamic-mac derived**	Specifies that the vNIC will obtain a MAC address dynamically from a pool.
Step 5	UCS-A /org/service-profile/vnic # **commit-buffer**	Commits the transaction to the system configuration.

Example

This example resets the MAC address of a vNIC in a service profile:

```
UCS-A# scope org /
UCS-A /org # scope service-profile ServInst90
UCS-A /org/service-profile #  scope vnic dynamic-prot-001
UCS-A /org/service-profile/vnic # set identity dynamic-mac derived
UCS-A /org/service-profile/vnic* # commit-buffer
UCS-A /org/service-profile/vnic #
```

Resetting the WWPN Assigned to a vHBA from a Pool in a Service Profile Template

If you change the WWPN pool assigned to an updating service profile template, Cisco UCS Manager does not change the WWPN assigned to a service profile created with that template. If you want Cisco UCS Manager to assign a WWPN from the newly assigned pool to the service profile, and therefore to the associated server, you must reset the WWPN. You can only reset the WWPN assigned to a service profile and its associated server under the following circumstances:

- The service profile was created from an updating service profile template and includes a WWPN assigned from a WWPN pool.

- The WWPN pool name is specified in the service profile. For example, the pool name is not empty.

- The WWPN value is not 0, and is therefore not derived from the server hardware.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the command mode for the organization that contains the service profile for which you want to reset the WWPN. If the system does not include multi-tenancy, type / as the *org-name* to enter the root organization.
Step 2	UCS-A /org # **scope service-profile** *profile-name*	Enters the service profile of the vHBA for which you want to reset the WWPN.

	Command or Action	Purpose
Step 3	UCS-A /org/service-profile # **scope vhba** *vhba-name*	Enters the command mode for vHBA for which you want to reset the WWPN.
Step 4	UCS-A /org/service-profile/vhba # **set identity dynamic-wwpn derived**	Specifies that the vHBA will obtain a WWPN dynamically from a pool.
Step 5	UCS-A /org/service-profile/vhba # **commit-buffer**	Commits the transaction to the system configuration.

Example

This example resets the WWPN of a vHBA in a service profile:

```
UCS-A# scope org /
UCS-A /org # scope service-profile ServInst90
UCS-A /org/service-profile #  scope vhba vhba3
UCS-A /org/service-profile/vhba # set identity dynamic-wwpn derived
UCS-A /org/service-profile/vhba* # commit-buffer
UCS-A /org/service-profile/vhba #
```

C H A P T E R **11**

Server-Related Policy Configuration

BIOS Settings

Server BIOS Settings

Cisco UCS provides two methods for making global modifications to the BIOS settings on servers in an Cisco UCS domain. You can create one or more BIOS policies that include a specific grouping of BIOS settings that match the needs of a server or set of servers, or you can use the default BIOS settings for a specific server platform.

Both the BIOS policy and the default BIOS settings for a server platform enable you to fine tune the BIOS settings for a server managed by Cisco UCS Manager.

Depending upon the needs of the data center, you can configure BIOS policies for some service profiles and use the BIOS defaults in other service profiles in the same Cisco UCS domain, or you can use only one of them. You can also use Cisco UCS Manager to view the actual BIOS settings on a server and determine whether they are meeting current needs.

> **Note**   Cisco UCS Manager pushes BIOS configuration changes through a BIOS policy or default BIOS settings to the Cisco Integrated Management Controller (CIMC) buffer. These changes remain in the buffer and do not take effect until the server is rebooted.
>
> We recommend that you verify the support for BIOS settings in the server that you want to configure. Some settings, such as Mirroring Mode for RAS Memory, are not supported by all Cisco UCS servers.

Main BIOS Settings

The following table lists the main server BIOS settings that you can configure through a BIOS policy or the default BIOS settings:

Name	Description
Properties	
Reboot on BIOS Settings Change **set reboot-on-update**	When the server is rebooted after you change one or more BIOS settings. **yes**—If you enable this setting, the server is rebooted according to the maintenance policy in the server's service profile. For example, if the maintenance policy requires user acknowledgment, the server is not rebooted and the BIOS changes are not applied until a user acknowledges the pending activity. **no**—If you do not enable this setting, the BIOS changes are not applied until the next time the server is rebooted, whether as a result of another server configuration change or a manual reboot.
BIOS Setting	
Quiet Boot **set quiet-boot-config quiet-boot**	What the BIOS displays during Power On Self-Test (POST). This can be one of the following: • **disabled**—The BIOS displays all messages and Option ROM information during boot. • **enabled**—The BIOS displays the logo screen, but does not display any messages or Option ROM information during boot. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
POST error pause **set post-error-pause-config post-error-pause**	What happens when the server encounters a critical error during POST. This can be one of the following: • **disabled**—The BIOS continues to attempt to boot the server. • **enabled**—The BIOS pauses the attempt to boot the server and opens the Error Manager when a critical error occurs during POST. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Resume on AC power loss **set resume-ac-on-power-loss-config resume-action**	How the server behaves when power is restored after an unexpected power loss. This can be one of the following: • **stay-off**—The server remains off until manually powered on. • **last-state**—The server is powered on and the system attempts to restore its last state. • **reset**—The server is powered on and automatically reset. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Front panel lockout **set front-panel-lockout-config front-panel-lockout**	Whether the power and reset buttons on the front panel are ignored by the server. This can be one of the following: • **disabled**—The power and reset buttons on the front panel are active and can be used to affect the server. • **enabled**—The power and reset buttons are locked out. The server can only be reset or powered on or off from the CIMC GUI. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
CDN Control **set consistent-device-name-control cdn-name**	Consistent Device Naming allows Ethernet interfaces to be named in a consistent manner. This makes Ethernet interface names more uniform, easy to identify, and persistent when adapter or other configuration changes are made. Whether consistent device naming is enabled or not. This can be one of the following: • **disabled**—Consistent device naming is disabled for the BIOS policy. • **enabled**—Consistent device naming is enabled for the BIOS policy. This enables Ethernet interfaces to be named consistently. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Processor BIOS Settings

The following table lists the processor BIOS settings that you can configure through a BIOS policy or the default BIOS settings:

Name	Description
Intel Turbo Boost Tech **set intel-turbo-boost-config turbo-boost**	Whether the processor uses Intel Turbo Boost Technology, which allows the processor to automatically increase its frequency if it is running below power, temperature, or voltage specifications. This can be one of the following: • **disabled**—The processor does not increase its frequency automatically. • **enabled**—The processor uses Turbo Boost Technology if required. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
Enhanced Intel SpeedStep Tech **set enhanced-intel-speedstep-config speed-step**	Whether the processor uses Enhanced Intel SpeedStep Technology, which allows the system to dynamically adjust processor voltage and core frequency. This technology can result in decreased average power consumption and decreased average heat production. This can be one of the following: • **disabled**—The processor never dynamically adjusts its voltage or frequency. • **enabled**—The processor utilizes Enhanced Intel SpeedStep Technology and enables all supported processor sleep states to further conserve power. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor. We recommend that you contact your operating system vendor to make sure your operating system supports this feature.
Intel HyperThreading Tech **set hyper-threading-config hyper-threading**	Whether the processor uses Intel Hyper-Threading Technology, which allows multithreaded software applications to execute threads in parallel within each processor. This can be one of the following: • **disabled**—The processor does not permit hyperthreading. • **enabled**—The processor allows for the parallel execution of multiple threads. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor. We recommend that you contact your operating system vendor to make sure the operating system supports this feature.

Name	Description
Intel Speed Select	Allows improved CPU performance by using Intel Speed Select technology to tune the CPU to run at one of three operating profiles, based on number of logical processor cores, frequency, and TDP thread setting, to improve performance over the basic Platform Default setting. These profiles correspond to High, Medium, and Low Core settings and can be one of the following: • **basic**—The processor uses High Core Configuration settings. • **option 1**—The processor uses Medium Core Configuration settings. • **option 2** —The processor uses Low Core Configuration settings. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor. Low core profiles enable the CPU to run at maximum frequency. Refer to Intel Speed Select, on page 257 sample settings.
Core Multi Processing **set core-multi-processing-config multi-processing**	Sets the state of logical processor cores per CPU in a package. If you disable this setting, Intel Hyper Threading technology is also disabled. This can be one of the following: • **All**—Enables multiprocessing on all logical processor cores. • **1** through n—Specifies the number of logical processor cores per CPU that can run on the server. To disable multiprocessing and have only one logical processor core per CPU running on the server, choose 1. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor. We recommend that you contact your operating system vendor to make sure your operating system supports this feature.

Name	Description
Execute Disable Bit **set execute-disable bit**	Classifies memory areas on the server to specify where the application code can execute. As a result of this classification, the processor disables code execution if a malicious worm attempts to insert code in the buffer. This setting helps to prevent damage, worm propagation, and certain classes of malicious buffer overflow attacks. This can be one of the following: • **disabled**—The processor does not classify memory areas. • **enabled**—The processor classifies memory areas. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor. We recommend that you contact your operating system vendor to make sure your operating system supports this feature.
Intel Virtualization Technology **set intel-vt-config vt**	Whether the processor uses Intel Virtualization Technology, which allows a platform to run multiple operating systems and applications in independent partitions. This can be one of the following: • **disabled**—The processor does not permit virtualization. • **enabled**—The processor allows multiple operating systems in independent partitions. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor. **Note** If you change this option, you must power cycle the server before the setting takes effect.
Hardware Prefetcher **set processor-prefetch-config hardware-prefetch**	Whether the processor allows the Intel hardware prefetcher to fetch streams of data and instruction from memory into the unified second-level cache when necessary. This can be one of the following: • **disabled**—The hardware prefetcher is not used. • **enabled**—The processor uses the hardware prefetcher when cache issues are detected. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor. **Note** **CPUPerformance** must be set to **Custom** in order to specify this value. For any value other than **Custom**, this option is overridden by the setting in the selected CPU performance profile.

Name	Description
Adjacent Cache Line Prefetcher **set processor-prefetch-config adjacent-cache-line-prefetch**	Whether the processor fetches cache lines in even/odd pairs instead of fetching just the required line. This can be one of the following: • **disabled**—The processor only fetches the required line. • **enabled**—The processor fetches both the required line and its paired line. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor. **Note** **CPUPerformance** must be set to **Custom** in order to specify this value. For any value other than **Custom**, this option is overridden by the setting in the selected CPU performance profile.
DCU Streamer Prefetch **set processor-prefetch-config dcu-streamer-prefetch**	Whether the processor uses the DCU IP Prefetch mechanism to analyze historical cache access patterns and preload the most relevant lines in the L1 cache. This can be one of the following: • **disabled**—The processor does not try to anticipate cache read requirements and only fetches explicitly requested lines. • **enabled**—The DCU prefetcher analyzes the cache read pattern and prefetches the next line in the cache if it determines that it may be needed. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
DCU IP Prefetcher **set processor-prefetch-config dcu-ip-prefetch**	Whether the processor uses the DCU IP Prefetch mechanism to analyze historical cache access patterns and preload the most relevant lines in the L1 cache. This can be one of the following: • **disabled**—The processor does not preload any cache data. • **enabled**—The DCU IP prefetcher preloads the L1 cache with the data it determines to be the most relevant. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
KTI Prefetch drop-down list	KTI prefetch is a mechanism to get the memory read started early on a DDR bus. This can be one of the following: • **disabled**—The processor does not preload any cache data. • **enabled**—The KTI prefetcher preloads the L1 cache with the data it determines to be the most relevant. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
LLC Prefetch drop-down list	Whether the processor uses the LLC Prefetch mechanism to fetch the date into the LLC. This can be one of the following: • **disabled**—The processor does not preload any cache data. • **enabled**—The LLC prefetcher preloads the L1 cache with the data it determines to be the most relevant. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
XPT Prefetch drop-down list	Whether XPT prefetch is used to enable a read request sent to the last level cache to issue a copy of that request to the memory controller prefetcher. This can be one of the following: • **disabled**—The CPU does not use the XPT Prefetch option. • **enabled**—The CPU enanbles the XPT prefetcher option. • **auto**—The CPU auto enables the XPT prefetcher option. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Direct Cache Access **set direct-cache-access-config access**	Allows processors to increase I/O performance by placing data from I/O devices directly into the processor cache. This setting helps to reduce cache misses. This can be one of the following: • **auto**—The CPU determines how to place data from I/O devices into the processor cache. • **disabled**—Data from I/O devices is not placed directly into the processor cache. • **enabled**—Data from I/O devices is placed directly into the processor cache. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
Processor C State **set processor-c-state-config c-state**	Whether the system can enter a power savings mode during idle periods. This can be one of the following: • **disabled**—The system remains in a high-performance state even when idle. • **enabled**—The system can reduce power to system components such as the DIMMs and CPUs. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor. We recommend that you contact your operating system vendor to make sure your operating system supports this feature.
Processor C1E **set processor-c1e-config c1e**	Allows the processor to transition to its minimum frequency upon entering C1. This setting does not take effect until after you have rebooted the server. This can be one of the following: • **disabled**—The CPU continues to run at its maximum frequency in the C1 state. • **enabled**—The CPU transitions to its minimum frequency. This option saves the maximum amount of power in the C1 state. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Processor C3 Report **set processor-c3-report-config processor-c3-report**	Whether the processor sends the C3 report to the operating system. This can be one of the following: • **enabled**—The processor sends the C3 report to the OS. • **disabled**—The processor does not send the C3 report. • **acpi-c2**—The processor sends the C3 report using the advanced configuration and power interface (ACPI) C2 format. • **acpi-c3**—The processor sends the C3 report using the ACPI C3 format. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor. On the Cisco UCS B440 Server, the BIOS Setup menu uses enabled and disabled for these options. If you specify acpi-c2 or acpi-c2, the server sets the BIOS value for that option to enabled.

Name	Description
Processor C6 Report **set processor-c6-report-config processor-c6-report**	Whether the processor sends the C6 report to the operating system. This can be one of the following: • **disabled**—The processor does not send the C6 report. • **enabled**—The processor sends the C6 report. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Processor C7 Report drop-down list **set processor-c7-report-config processor-c7-report**	Whether the processor sends the C7 report to the operating system. This can be one of the following: • **c7**—The processor sends the report using the C7 format. • **c7s**—The processor sends the report using the C7s format. • **disabled**—The processor does not send the C7 report. • **enabled**—The processor sends the C7 report. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Processor CMCI drop-down list	Enables CMCI generation. This can be one of the following: • **disabled**—The processor disables CMCI. • **enabled**—The processor enables CMCI. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
CPU Performance **set cpu-performance-config cpu-performance**	Sets the CPU performance profile for the server. This can be one of the following: • **Custom** — • **enterprise**—For M3 and higher servers, all prefetchers and data reuse are enabled. • **high-throughput**—Data reuse and the DCU IP prefetcher are enabled, and all other prefetchers are disabled. • **hpc**—All prefetchers are enabled and data reuse is disabled. This setting is also known as high-performance computing. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
Max Variable MTRR Setting **set max-variable-mtrr-setting-config processor-mtrr**	Allows you to select the number of mean time to repair (MTRR) variables. This can be one of the following: • **auto-max**—BIOS uses the default value for the processor. • **8**—BIOS uses the number specified for the variable MTRR. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Local X2 APIC **set local-x2-apic-config localx2-apic**	Allows you to set the type of Application Policy Infrastructure Controller (APIC) architecture. This can be one of the following: • **disabled**—Processor disables Local X2 APIC. • **enabled**—Processor enables Local X2 APIC. • **xapic**—Uses the standard xAPIC architecture. • **x2apic**—Uses the enhanced x2APIC architecture to support 32 bit addressability of processors. • **auto**—Automatically uses the xAPIC architecture that is detected. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
Power Technology **set processor-energy-config cpu-power-management**	Enables you to configure the CPU power management settings for the following options: • Enhanced Intel Speedstep Technology • Intel Turbo Boost Technology • Processor Power State C6 Power Technology can be one of the following: • **disabled**—The server does not perform any CPU power management and any settings for the BIOS parameters mentioned above are ignored. • **Energy_Efficient**—The server determines the best settings for the BIOS parameters mentioned above and ignores the individual settings for these parameters. • **performance**—The server automatically optimizes the performance for the BIOS parameters mentioned above. • **custom**—The server uses the individual settings for the BIOS parameters mentioned above. You must select this option if you want to change any of these BIOS parameters. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
Energy Performance **set processor-energy-config energy-performance**	Allows you to determine whether system performance or energy efficiency is more important on this server. This can be one of the following: • **performance** — The server provides all server components with full power at all times. This option maintains the highest level of performance and requires the greatest amount of power. • **balanced-performance** — The server provides all server components with enough power to keep a balance between performance and power. • **balanced-energy** — The server provides all server components with enough power to keep a balance between performance and power. • **energy-efficient** — The server provides all server components with less power to keep reduce power consumption. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor. **Note** **CPUPowerManagement** must be set to **Custom** or the server ignores the setting for this parameter.
Frequency Floor Override **set frequency-floor-override-config cpu-frequency**	Whether the CPU is allowed to drop below the maximum non-turbo frequency when idle. This can be one of the following: • **disabled**— The CPU can drop below the maximum non-turbo frequency when idle. This option decreases power consumption but may reduce system performance. • **enabled**— The CPU cannot drop below the maximum non-turbo frequency when idle. This option improves system performance but may increase power consumption. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
P STATE Coordination **set p-state-coordination-config p-state**	Allows you to define how BIOS communicates the P-state support model to the operating system. There are 3 models as defined by the Advanced Configuration and Power Interface (ACPI) specification. • **hw-all**—The processor hardware is responsible for coordinating the P-state among logical processors with dependencies (all logical processors in a package). • **sw-all**—The OS Power Manager (OSPM) is responsible for coordinating the P-state among logical processors with dependencies (all logical processors in a physical package), and must initiate the transition on all of the logical processors. • **sw-any**—The OS Power Manager (OSPM) is responsible for coordinating the P-state among logical processors with dependencies (all logical processors in a package), and may initiate the transition on any of the logical processors in the domain. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor. **Note** **CPUPowerManagement** must be set to **Custom** or the server ignores the setting for this parameter.
DRAM Clock Throttling **set dram-clock-throttling-config dram-clock-throttling**	Allows you to tune the system settings between the memory bandwidth and power consumption. This can be one of the following: • **auto** — CPU determines the DRAM Clock Throttling settings. • **balanced**— DRAM clock throttling is reduced, providing a balance between performance and power. • **performance**—DRAM clock throttling is disabled, providing increased memory bandwidth at the cost of additional power. • **Energy_Efficient**—DRAM clock throttling is increased to improve energy efficiency. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
Channel Interleaving **set interleave-config channel-interleave**	Whether the CPU divides memory blocks and spreads contiguous portions of data across interleaved channels to enable simultaneous read operations. This can be one of the following: • **auto**—The CPU determines what interleaving is done. • **1-way**— • **2-way** • **3-way** • **4-way**—The maximum amount of channel interleaving is used. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Rank Interleaving **set interleave-config rank-interleave**	Whether the CPU interleaves physical ranks of memory so that one rank can be accessed while another is being refreshed. This can be one of the following: • **auto**—The CPU determines what interleaving is done. • **1-way**— • **2-way** • **4-way** • **8-way**—The maximum amount of rank interleaving is used. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Sub NUMA Clustering drop-down list	Whether the CPU supports sub NUMA clustering, in which the tag directory and the memory channel are always in the same region. This can be one of the following: • **disabled**— Sub NUMA clustering does not occur. • **enabled**— Sub NUMA clustering occurs. • **platform-default** — The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
Memory Interleaving **set interleave-config memory-interleave**	Whether the CPU interleaves the physical memory so that the memory can be accessed while another is being refreshed. This can be one of the following: • **2-way-node-interleave** — • **4-way-node-interleave** • **numa-1-way-node-interleave** • **8-way-node-interleave** • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Demand Scrub **set scrub-policies-config demand-scrub**	Whether the system corrects single bit memory errors encountered when the CPU or I/O makes a demand read. This can be one of the following: • **disabled**— Single bit memory errors are not corrected. • **enabled**— Single bit memory errors are corrected in memory and the corrected data is set in response to the demand read. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Patrol Scrub **set scrub-policies-config patrol-scrub**	Whether the system actively searches for, and corrects, single bit memory errors even in unused portions of the memory on the server. This can be one of the following: • **disabled**—The system checks for memory ECC errors only when the CPU reads or writes a memory address. • **enabled**—The system periodically reads and writes memory searching for ECC errors. If any errors are found, the system attempts to fix them. This option may correct single bit errors before they become multi-bit errors, but it may adversely affect performance when the patrol scrub is running. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
DCPMM Firmware Downgrade	This can be one of the following: • **disabled**— • **enabled**— • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Configurable TDP Control	Allows you to set customized value for Thermal Design Power (TDP). This can be one of the following: • **auto**— Uses the rated TDP value of the processor. • **manual**—Allows you to customize the TDP value.
Altitude **set altitude altitude-config**	The approximate number of meters above sea level at which the physical server is installed. This can be one of the following: • **auto**—The CPU determines the physical elevation. • **300-m**—The server is approximately 300 meters above sea level. • **900-m**—The server is approximately 900 meters above sea level. • **1500-m**—The server is approximately 1500 meters above sea level. • **3000-m**—The server is approximately 3000 meters above sea level. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
Package C State Limit **set package-c-state-limit-config package-c-state-limit**	The amount of power available to the server components when they are idle. This can be one of the following: **Note** Cisco recommends setting **Package C State Limit** as **no-limit** or **platform-default** for C220 M4, C240 M4, C460 M4, and S3260 M4 servers. If you are changing the **Package C State Limit** token from any other value to **no-limit**, then ensure that the **Power Technology** is set to **custom**. • **auto** —The CPU determines the physical elevation. • **no-limit**—The server may enter any available C state. • **c0**—The server provides all server components with full power at all times. This option maintains the highest level of performance and requires the greatest amount of power. • **c1**—When the CPU is idle, the system slightly reduces the power consumption. This option requires less power than C0 and allows the server to return quickly to high performance mode. • **c3**—When the CPU is idle, the system reduces the power consumption further than with the C1 option. This requires less power than C1 or C0, but it takes the server slightly longer to return to high performance mode. • **c6**—When the CPU is idle, the system reduces the power consumption further than with the C3 option. This option saves more power than C0, C1, or C3, but there may be performance issues until the server returns to full power. • **c2**—When the CPU is idle, the system reduces the power consumption further than with the C1 option. This requires less power than C1 or C0, but it takes the server slightly longer to return to high performance mode. • **c7**—When the CPU is idle, the server makes a minimal amount of power available to the components. This option saves the maximum amount of power but it also requires the longest time for the server to return to high performance mode. • **c7s**—When the CPU is idle, the server makes a minimal amount of power available to the components. This option saves more power than C7, but it also requires the longest time for the server to return to high performance mode. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
CPU Hardware Power Management drop-down list **set cpu-hardware-power-management-config cpu-hardware-power-management**	Enables processor Hardware Power Management (HWPM). This can be one of the following: • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor. • **disabled**—HWPM is disabled. • **hwpm-native-mode**—HWPM native mode is enabled. • **hwpm-oob-mode**—HWPM Out-Of-Box mode is enabled. • **Native Mode with no Legacy** (only GUI)
Energy Performance Tuning drop-down list **set power-performance-tuning-support power-performance-tuning-config**	Determines if the BIOS or Operating System can turn on the energy performance bias tuning. The options are BIOS and OS. • **bios**— • **os**— • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Workload Configuration drop-down list	This feature allows for workload optimization. The options are Balanced and I/O Sensitive: • **balanced**— • **io-sensitive**— • **NUMA** — (Only GUI) • **UMA** — (Only GUI) • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor. Cisco recommends using Balanced.
Core Performance Boost drop-down list	Whether the AMD processor increases its frequency on some cores when it is idle or not being used much. This can be one of the following: • **auto**—The CPU automatically determines how to boost performance. • **disabled**—Core performance boost is disabled. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
Global C-state Control drop-down list	Whether the AMD processors control IO-based C-state generation and DF C-states This can be one of the following: • **auto**—The CPU automatically determines how to control IO-based C-state generation. • **disabled**—Global C-state control is disabled. • **enabled**—Global C-state control is enabled. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
L1 Stream HW Prefetcher drop-down list	Whether the processor allows the AMD hardware prefetcher to speculatively fetch streams of data and instruction from memory into the L1 cache when necessary. This can be one of the following: • **auto**—The CPU determines how to place data from I/O devices into the processor cache. • **disabled**—The hardware prefetcher is not used. • **enabled**—The processor uses the hardware prefetcher when cache issues are detected. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
L2 Stream HW Prefetcher drop-down list	Whether the processor allows the AMD hardware prefetcher to speculatively fetch streams of data and instruction from memory into the L2 cache when necessary. This can be one of the following: • **auto**—The CPU determines how to place data from I/O devices into the processor cache. • **disabled**—The hardware prefetcher is not used. • **enabled**—The processor uses the hardware prefetcher when cache issues are detected. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
AMD Memory Interleaving drop-down list	Whether the AMD CPU interleaves the physical memory so that the memory can be accessed while another is being refreshed. This controls fabric level memory interleaving. Channel, die and socket have requirements based on memory populations and will be ignored if the memory does not support the selected option.This can be one of the following: • **auto**—The CPU determines how to interleave memory. • **channel**—Interleaves the physical address space over multiple channels, as opposed to each channel owning single consecutive address spaces. • **die**—Interleaves the physical address space over multiple dies, as opposed to each die owning single consecutive address spaces. • **none**—Consecutive memory blocks are accessed from the same physical memory. • **socket**—Interleaves the physical address space over multiple sockets, as opposed to each socket owning single consecutive address spaces. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
AMD Memory Interleaving Size drop-down list	Determines the size of the memory blocks to be interleaved. It also determines the starting address of the interleave (bit 8,9,10 or 11). This can be one of the following: • 1 KB • 2 KB • 256 Bytes • 512 Bytes • **auto**—The CPU determines the size of the memory block. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
Chipselect Interleaving drop-down list	Whether memory blocks across the DRAM chip selects for node 0 are interleaved. This can be one of the following: • **auto**—The CPU automatically determines how to interleave chip selects. • **disabled**—Chip selects are not interleaved within the memory controller. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Bank Group Swap drop-down list	Determines how physical addresses are assigned to applications. This can be one of the following: • **auto**—The CPU automatically determines how to assign physical addresses to applications. • **disabled**—Bank group swap is not used. • **enabled**—Bank group swap is used to improve the performance of applications. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Determinism Slider drop-down list	Allows AMD processors to determine how to operate. This can be one of the following: • **auto**—The CPU automatically uses default power determinism settings. • **performance**—Processor operates at the best performance in a consistent manner. • **power**—Processor operates at the maximum allowable performance on a per die basis. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
IOMMU drop-down list	Input Output Memory Management Unit (IOMMU) allows AMD processors to map virtual addresses to physical addresses. This can be one of the following: • **auto**—The CPU determines how map these addresses. • **disabled**—IOMMU is not used. • **enabled**—Address mapping takes place through the IOMMU. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Downcore control drop-down list	Allows AMD processors to disable cores and, thus, select how many cores to enable. This can be one of the following: • FOUR (2+2)—Two cores enabled on each CPU complex. • FOUR (4+0)—Four cores enabled on one CPU complex. • SIX (3+3)—Three cores enabled on each CPU complex. • THREE (3+0)—Three cores enabled on one CPU complex. • TWO (1+1)—Two cores enabled on each CPU complex. • TWO (2+0)—Two cores enabled on one CPU complex. • **auto**—The CPU determines how many cores need to be enabled. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
SVM Mode drop-down list	Whether the processor uses AMD Secure Virtual Machine Technology. This can be one of the following: This can be one of the following: • **disabled**—The processor does not use SVM Technology. • **enabled**—The processor uses SVM Technology. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
SMT Mode drop-down list	Whether the processor uses AMD Simultaneous MultiThreading Technology, which allows multithreaded software applications to execute threads in parallel within each processor. This can be one of the following: • **auto**—The processor allows for the parallel execution of multiple threads. • **off**—The processor does not permit multithreading. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
SMEE drop-down list	Whether the processor uses the Secure Memory Encryption Enable (SMEE) function, which provides memory encryption support. This can be one of the following: • **disabled**—The processor does not use the SMEE function. • **enabled**—The processor uses the SMEE function. • **platform-default** —The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Intel Directed I/O BIOS Settings

The following table lists the Intel Directed I/O BIOS settings that you can configure through a BIOS policy or the default BIOS settings:

Name	Description
Intel VT for directed IO **set intel-vt-directed-io-config vtd**	Whether the processor uses Intel Virtualization Technology for Directed I/O (VT-d). This can be one of the following: • **disabled**—The processor does not use virtualization technology. • **enabled**—The processor uses virtualization technology. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor. **Note** This option must be enabled if you want to change any of the other Intel Directed I/O BIOS settings.

Name	Description
Intel VTD interrupt Remapping **set intel-vt-directed-io-config interrupt-remapping**	Whether the processor supports Intel VT-d Interrupt Remapping. This can be one of the following: • **disabled**—The processor does not support remapping. • **enabled**—The processor uses VT-d Interrupt Remapping as required. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Intel VTD coherency support **set intel-vt-directed-io-config coherency-support**	Whether the processor supports Intel VT-d Coherency. This can be one of the following: • **disabled**—The processor does not support coherency. • **enabled**—The processor uses VT-d Coherency as required. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Intel VTD ATS support **set intel-vt-directed-io-config ats-support**	Whether the processor supports Intel VT-d Address Translation Services (ATS). This can be one of the following: • **disabled**—The processor does not support ATS. • **enabled**—The processor uses VT-d ATS as required. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Intel VTD pass through DMA support **set intel-vt-directed-io-config passthrough-dma**	Whether the processor supports Intel VT-d Pass-through DMA. This can be one of the following: • **disabled**—The processor does not support pass-through DMA. • **enabled**—The processor uses VT-d Pass-through DMA as required. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

RAS Memory BIOS Settings

The following table lists the RAS memory BIOS settings that you can configure through a BIOS policy or the default BIOS settings:

Name	Description
Memory RAS configuration **set memory-ras-config ras-config**	How the memory reliability, availability, and serviceability (RAS) is configured for the server. This can be one of the following: • **maximum-performance**—System performance is optimized. • **mirroring**—System reliability is optimized by using half the system memory as backup. This mode is used for UCS M4 and lower blade servers. • **lockstep**—If the DIMM pairs in the server have an identical type, size, and organization and are populated across the SMI channels, you can enable lockstep mode to minimize memory access latency and provide better performance. Lockstep is enabled by default for B440 servers. • **Mirror Mode 1LM**—Mirror Mode 1LM will set the entire 1LM memory in the system to be mirrored, consequently reducing the memory capacity by half. This mode is used for UCS M5 blade servers. • **sparing**—System reliability is optimized by holding memory in reserve so that it can be used in case other DIMMs fail. This mode provides some memory redundancy, but does not provide as much redundancy as mirroring. • **adddc sparing**—System reliability is optimized by holding memory in reserve so that it can be used in case other DIMMs fail. This mode provides some memory redundancy, but does not provide as much redundancy as mirroring. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
NUMA optimized **set numa-config numa-optimization**	Whether the BIOS supports NUMA. This can be one of the following: • **disabled**—The BIOS does not support NUMA. • **enabled**—The BIOS includes the ACPI tables that are required for NUMA-aware operating systems. If you enable this option, the system must disable Inter-Socket Memory interleaving on some platforms. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
Mirroring Mode **set memory-mirroring-mode mirroring-mode**	Memory mirroring enhances system reliability by keeping two identical data images in memory. This option is only available if you choose the **mirroring** option for **Memory RAS Config**. It can be one of the following: • **inter-socket**—Memory is mirrored between two Integrated Memory Controllers (IMCs) across CPU sockets. • **intra-socket**—One IMC is mirrored with another IMC in the same socket. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Sparing Mode **set memory-sparing-mode sparing-mode**	Sparing optimizes reliability by holding memory in reserve so that it can be used in case other DIMMs fail. This option provides some memory redundancy, but does not provide as much redundancy as mirroring. The available sparing modes depend on the current memory population. This option is only available if you choose **sparing** option for **Memory RAS Config**. It can be one of the following: • **dimm-sparing**—One DIMM is held in reserve. If a DIMM fails, the contents of a failing DIMM are transferred to the spare DIMM. • **rank-sparing**—A spare rank of DIMMs is held in reserve. If a rank of DIMMs fails, the contents of the failing rank are transferred to the spare rank. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
LV DDR Mode **set lv-dimm-support-config lv-ddr-mode**	Whether the system prioritizes low voltage or high frequency memory operations. This can be one of the following: • **auto**—The CPU determines whether to prioritize low voltage or high frequency memory operations. • **power-saving-mode**—The system prioritizes low voltage memory operations over high frequency memory operations. This mode may lower memory frequency in order to keep the voltage low. • **performance-mode**—The system prioritizes high frequency operations over low voltage operations. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
DRAM Refresh Rate **set dram-refresh-rate-config dram-refresh**	The refresh interval rate for internal memory. This can be one of the following: • **1x** • **2x** • **3x** • **4x** • **auto** • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
DDR3 Voltage Selection **set ddr3-voltage-config ddr3-voltage**	The voltage to be used by the dual-voltage RAM. This can be one of the following: • **ddr3-1500mv** • **ddr3-1350mv** • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Serial Port BIOS Settings

The following table lists the serial port BIOS settings that you can configure through a BIOS policy or the default BIOS settings:

Name	Description
Serial port A enable **set serial-port-a-config serial-port-a**	Whether serial port A is enabled or disabled. This can be one of the following: • **disabled**—The serial port is disabled. • **enabled**—The serial port is enabled. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

USB BIOS Settings

The following table lists the USB BIOS settings that you can configure through a BIOS policy or the default BIOS settings:

Name	Description
Make Device Non Bootable **set usb-boot-config make-device-non-bootable**	Whether the server can boot from a USB device. This can be one of the following: • **disabled**—The server can boot from a USB device. • **enabled**—The server cannot boot from a USB device. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Legacy USB Support **set usb-boot-config legacy-support**	Whether the system supports legacy USB devices. This can be one of the following: • **disabled**—USB devices are only available to EFI applications. • **enabled**—Legacy USB support is always available. • **auto**—Disables legacy USB support if no USB devices are connected. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
USB Idle Power Optimizing Setting **set usb-system-idle-power-optimizing-setting-config usb-idle-power-optimizing**	Whether the USB Idle Power Optimizing setting is used to reduce USB EHCI idle power consumption. Depending upon the value you choose, this setting can have an impact on performance. This can be one of the following: • **high-performance**—The USB System Idle Power Optimizing setting is disabled, because optimal performance is preferred over power savings. Selecting this option can significantly improve performance. We recommend you select this option unless your site has server power restrictions. • **lower-idle-power**—The USB System Idle Power Optimizing setting is enabled, because power savings are preferred over optimal performance. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
USB Front Panel Access Lock **set usb-front-panel-access-lock-config usb-front-panel-lock**	USB front panel access lock is configured to enable or disable the front panel access to USB ports. This can be one of the following: • **disabled** • **enabled** • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Port 60/64 Emulation **set usb-port-config usb-emulation**	Whether the system supports 60h/64h emulation for complete USB keyboard legacy support. This can be one of the following: • **disabled**—60h/64 emulation is not supported. • **enabled**—60h/64 emulation is supported. You should select this option if you are using a non-USB aware operating system on the server. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
USB Port Front **set usb-port-config usb-front**	Whether the front panel USB devices are enabled or disabled. This can be one of the following: • **disabled**—Disables the front panel USB ports. Devices connected to these ports are not detected by the BIOS and operating system. • **enabled**—Enables the front panel USB ports. Devices connected to these ports are detected by the BIOS and operating system. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
USB Port Internal **set usb-port-config usb-internal**	Whether the internal USB devices are enabled or disabled. This can be one of the following: • **disabled**—Disables the internal USB ports. Devices connected to these ports are not detected by the BIOS and operating system. • **enabled**—Enables the internal USB ports. Devices connected to these ports are detected by the BIOS and operating system. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
USB Port KVM **set usb-port-config usb-kvm**	Whether the KVM ports are enabled or disabled. This can be one of the following: • **disabled**—Disables the KVM keyboard and/or mouse devices. Keyboard and/or mouse will not work in the KVM window. • **enabled**—Enables the KVM keyboard and/or mouse devices. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
USB Port Rear **set usb-port-config usb-rear**	Whether the rear panel USB devices are enabled or disabled. This can be one of the following: • **disabled**—Disables the rear panel USB ports. Devices connected to these ports are not detected by the BIOS and operating system. • **enabled**—Enables the rear panel USB ports. Devices connected to these ports are detected by the BIOS and operating system. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
USB Port SD Card **set usb-port-config usb-sdcard**	Whether the SD card drives are enabled or disabled. This can be one of the following: • **disabled**—Disables the SD card drives. The SD card drives are not detected by the BIOS and operating system. • **enabled**—Enables the SD card drives. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
USB Port VMedia **set usb-port-config usb-vmedia**	Whether the virtual media devices are enabled or disabled. This can be one of the following: • **disabled**—Disables the vMedia devices. • **enabled**—Enables the vMedia devices. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
All USB Devices **set all-usb-devices-config all-usb**	Whether all physical and virtual USB devices are enabled or disabled. This can be one of the following: • **disabled**—All USB devices are disabled. • **enabled**—All USB devices are enabled. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
xHCI Mode **set usb-configuration-select-config xhci-enable-disable**	Whether xHCI mode is enabled or disabled. This can be one of the following: • **disabled**—xHCI mode is disabled. • **enabled**—xHCI mode is enabled. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

PCI Configuration BIOS Settings

The following table lists the PCI configuration BIOS settings that you can configure through a BIOS policy or the default BIOS settings:

Name	Description
Maximum memory below 4GB **set max-memory-below-4gb-config max-memory**	Whether the BIOS maximizes memory usage below 4GB for an operating system without PAE support, depending on the system configuration. This can be one of the following: • **disabled**—Does not maximize memory usage. Choose this option for all operating systems with PAE support. • **enabled**—Maximizes memory usage below 4GB for an operating system without PAE support. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Memory mapped IO above 4GB **set memory-mapped-io-above-4gb-config memory-mapped-io**	Whether to enable or disable memory mapped I/O of 64-bit PCI devices to 4GB or greater address space. Legacy option ROMs are not able to access addresses above 4GB. PCI devices that are 64-bit compliant but use a legacy option ROM may not function correctly with this setting enabled. This can be one of the following: • **disabled**—Does not map I/O of 64-bit PCI devices to 4GB or greater address space. • **enabled**—Maps I/O of 64-bit PCI devices to 4GB or greater address space. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
VGA Priority **set vga-priority-config vga-priority**	Allows you to set the priority for VGA graphics devices if multiple VGA devices are found in the system. This can be one of the following: • **onboard**—Priority is given to the onboard VGA device. BIOS post screen and OS boot are driven through the onboard VGA port. • **offboard**—Priority is given to the PCIE Graphics adapter. BIOS post screen and OS boot are driven through the external graphics adapter port. • **onboard-vga-disabled**—Priority is given to the PCIE Graphics adapter, and the onboard VGA device is disabled. **Note** The vKVM does not function when the onboard VGA is disabled. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor. **Note** Only onboard VGA devices are supported with Cisco UCS B-Series servers.
ASPM Support **set aspm-support-config aspm-support**	Allows you to set the level of ASPM (Active Power State Management) support in the BIOS. This can be one of the following: • **disabled**—ASPM support is disabled in the BIOS. • **auto**—The CPU determines the power state. • **forcel0**—Force all links to L0 standby (L0s) state. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
BME DMA Mitigation Support **set bme-dma-config**	Allows you to disable the PCI BME bit to mitigate the threat from an unauthorized external DMA. This can be one of the following: • **disabled**—PCI BME bit is disabled in the BIOS. • **enabled**—PCI BME bit is enabled in the BIOS. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

QPI BIOS Settings

The following table lists the QPI BIOS settings that you can configure through a BIOS policy or the default BIOS settings:

Name	Description
QPI Link Frequency Select **set qpi-link-frequency-select-config qpi-link-freqency-mt-per-sec**	The Intel QuickPath Interconnect (QPI) link frequency, in megatransfers per second (MT/s). This can be one of the following: • **6400** • **7200** • **8000** • **9600** • **auto**—The CPU determines the QPI link frequency. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
QPI Snoop Mode **set qpi-snoop-mode vpqpisnoopmode**	This can be one of the following: • **home-snoop**—The snoop is always spawned by the home agent (centralized ring stop) for the memory controller. This mode has a higher local latency than early snoop, but it provides extra resources for a larger number of outstanding transactions. • **cluster-on-die**—This mode is available only for processors that have 10 or more cores. It is the best mode for highly NUMA optimized workloads. • **home-directory-snoop-with-osb** • **early-snoop**—The distributed cache ring stops can send a snoop probe or a request to another caching agent directly. This mode has lower latency and it is best for workloads that have shared data sets across threads and can benefit from a cache-to-cache transfer, or for workloads that are not NUMA optimized. • **auto** —The CPU determines the QPI Snoop mode. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

LOM and PCIe Slots BIOS Settings

The following table lists the USB BIOS settings that you can configure through a BIOS policy or the default BIOS settings:

Name	Description
PCIe Slot SAS OptionROM **set slot-option-rom-enable-config pcie-sas**	Whether Option ROM is available on the SAS port. This can be one of the following: • **disabled**—The expansion slot is not available. • **enabled**—The expansion slot is available. • **uefi-only**—The expansion slot is available for UEFI only. • **legacy-only**—The expansion slot is available for legacy only. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
PCIe Slot *n* Link Speed **set slot-link-speed-config pcie-slot*n*-link-speed**	This option allows you to restrict the maximum speed of an adapter card installed in PCIe slot *n*. This can be one of the following: • **gen1**—2.5GT/s (gigatransfers per second) is the maximum speed allowed. • **gen2**—5GT/s is the maximum speed allowed. • **gen3**—8GT/s is the maximum speed allowed. • **auto**—The maximum speed is set automatically. • **disabled**—The maximum speed is not restricted. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
PCIe Slot *n* OptionROM **set slot-option-rom-enable-config slot*n*-option-rom-enable**	Whether Option ROM is available on the port. This can be one of the following: • **disabled**—The expansion slot is not available. • **enabled**—The expansion slot is available. • **uefi-only**—The expansion slot is available for UEFI only. • **legacy-only**—The expansion slot is available for legacy only. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
PCIe Slot HBA OptionROM **set slot-option-rom-enable-config pcie-hba**	Whether Option ROM is available on the HBA port. This can be one of the following: • **disabled**—The expansion slot is not available. • **enabled**—The expansion slot is available. • **uefi-only**—The expansion slot is available for UEFI only. • **legacy-only**—The expansion slot is available for legacy only. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
PCIe Slot MLOM OptionROM **set slot-option-rom-enable-config pcie-mlom**	Whether Option ROM is available on the MLOM port. This can be one of the following: • **disabled**—The expansion slot is not available. • **enabled**—The expansion slot is available. • **uefi-only**—The expansion slot is available for UEFI only. • **legacy-only**—The expansion slot is available for legacy only. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
PCIe Slot N*x* OptionROM **set slot-option-rom-enable-config pcie-n*x***	Whether Option ROM is available on the port. This can be one of the following: • **disabled**—The expansion slot is not available. • **enabled**—The expansion slot is available. • **uefi-only**—The expansion slot is available for UEFI only. • **legacy-only**—The expansion slot is available for legacy only. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
PCIe 10G LOM 2 Link **set lom-ports-config pcie-lom2-link**	Whether Option ROM is available on the 10G LOM port. This can be one of the following: • **disabled**—The expansion slot is not available. • **enabled**—The expansion slot is available. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
PCI ROM CLP **set pci-rom-clp-support pci-rom-clp-config**	PCI ROM Command Line Protocol (CLP) controls the execution of different Option ROMs such as PxE and iSCSI that are present in the card. By default, it is disabled. • **disabled**—The expansion slot is not available. • **enabled**—The expansion slot is available. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
SIOC1 Option ROM **set sioc1-optionrom-config sioc1-optionrom**	Whether the server can use Option ROM present in System IO Controller 1 (SIOC1). This can be one of the following: • **disabled**—The expansion slot is not available. • **enabled**—The expansion slot is available. • **uefi-only**—The expansion slot is available for UEFI only. • **legacy-only**—The expansion slot is available for legacy only. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
SIOC2 Option ROM **set sioc2-optionrom-config sioc2-optionrom**	Whether the server can use Option ROM present in System IO Controller 2 (SIOC2). This can be one of the following: • **disabled**—The expansion slot is not available. • **enabled**—The expansion slot is available. • **uefi-only**—The expansion slot is available for UEFI only. • **legacy-only**—The expansion slot is available for legacy only. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
SBMEZZ1 Option ROM **set sbmezz1-optionrom-config sbmezz1-optionrom**	Whether the server can use Option ROM present in SBMezz1 controller. This can be one of the following: • **disabled**—The expansion slot is not available. • **enabled**—The expansion slot is available. • **uefi-only**—The expansion slot is available for UEFI only. • **legacy-only**—The expansion slot is available for legacy only. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
SBMEZZ2 Option ROM **set sbmezz2-optionrom-config sbmezz2-optionrom**	Whether the server can use Option ROM present in SBMezz2 controller. This can be one of the following: • **disabled**—The expansion slot is not available. • **enabled**—The expansion slot is available. • **uefi-only**—The expansion slot is available for UEFI only. • **legacy-only**—The expansion slot is available for legacy only. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
IOESlot1 OptionROM **set** **ioeslot1-optionrom-config** **ioeslot1-optionrom**	Whether option ROM is enabled on the IOE slot 1. This can be one of the following: • **disabled**—The expansion slot is not available. • **enabled**—The expansion slot is available. • **uefi-only**—The expansion slot is available for UEFI only. • **legacy-only**—The expansion slot is available for legacy only. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
IOEMEZZ 1 OptionROM **set** **ioemezz1-optionrom-config** **ioemezz1-optionrom**	Whether option ROM is enabled on the IOE Mezz1. This can be one of the following: • **disabled**—The expansion slot is not available. • **enabled**—The expansion slot is available. • **uefi-only**—The expansion slot is available for UEFI only. • **legacy-only**—The expansion slot is available for legacy only. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
IOE Slot2 Option ROM **set** **ioeslot2-optionrom-config** **ioeslot2-optionrom**	Whether option ROM is enabled on the IOE slot 2. This can be one of the following: • **disabled**—The expansion slot is not available. • **enabled**—The expansion slot is available. • **uefi-only**—The expansion slot is available for UEFI only. • **legacy-only**—The expansion slot is available for legacy only. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
IO ENVME1 Option ROM **set** **ioenvme1-optionrom-config** **ioenvme1-optionrom**	Whether option ROM is enabled on the IOE NVMe1. This can be one of the following: • **disabled**—The expansion slot is not available. • **enabled**—The expansion slot is available. • **uefi-only**—The expansion slot is available for UEFI only. • **legacy-only**—The expansion slot is available for legacy only. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
IO ENVME2 Option ROM **set ioenvme2-optionrom-config ioenvme2-optionrom**	Whether option ROM is enabled on the IOE NVMe2. This can be one of the following: • **disabled**—The expansion slot is not available. • **enabled**—The expansion slot is available. • **uefi-only**—The expansion slot is available for UEFI only. • **legacy-only**—The expansion slot is available for legacy only. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
SBNVME1 Option ROM **set sbnvme1-optionrom-config sbnvme1-optionrom**	Whether the server can use Option ROM present in SBNVMe1 controller. This can be one of the following: • **disabled**—The expansion slot is not available. • **enabled**—The expansion slot is available. • **uefi-only**—The expansion slot is available for UEFI only. • **legacy-only**—The expansion slot is available for legacy only. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
PCIe Slot MRAID OptionROM drop-down list **set Pcie SlotMRAIDOptionROM**	Whether Option ROM is available on the MRAID port. This can be one of the following: • **disabled**—The expansion slot is not available. • **enabled**—The expansion slot is available. • **uefi-only**—The expansion slot is available for UEFI only. • **legacy-only**—The expansion slot is available for legacy only. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
PCIe Slot RAID OptionROM **set Pcie SlotRAIDOptionROM**	Whether Option ROM is available on the RAID port. This can be one of the following: • **disabled**—The expansion slot is not available. • **enabled**—The expansion slot is available. • **uefi-only**—The expansion slot is available for UEFI only. • **legacy-only**—The expansion slot is available for legacy only. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
PCIe Slot Rear Nvme*n* OptionRom **set Pcie SlotRearNvme*n*OptionROM**	Whether Option ROM is available on the Rear NVME*n* port. This can be one of the following: • **disabled**—The expansion slot is not available. • **enabled**—The expansion slot is available. • **uefi-only**—The expansion slot is available for UEFI only. • **legacy-only**—The expansion slot is available for legacy only. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Rear NVME1 Link Speed **set Pcie SlotRearNvme1LinkSpeed**	This option allows you to restrict the maximum speed of an NVME card installed in the rear PCIe slot 1. This can be one of the following: • **gen1**—2.5GT/s (gigatransfers per second) is the maximum speed allowed. • **gen2**—5GT/s is the maximum speed allowed. • **gen3**—8GT/s is the maximum speed allowed. • **auto**—The maximum speed is set automatically. • **disabled**—The maximum speed is not restricted. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Rear NVME2 Link Speed **set Pcie SlotRearNvme2LinkSpeed**	This option allows you to restrict the maximum speed of an NVME card installed in the rear PCIe slot 2. This can be one of the following: • **gen1**—2.5GT/s (gigatransfers per second) is the maximum speed allowed. • **gen2**—5GT/s is the maximum speed allowed. • **gen3**—8GT/s is the maximum speed allowed. • **auto**—The maximum speed is set automatically. • **disabled**—The maximum speed is not restricted. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
Front NVME1 Link Speed **set Pcie SlotFrontNvme1LinkSpeed**	This option allows you to restrict the maximum speed of an NVME card installed in the front PCIe slot 1. This can be one of the following: • **gen1**—2.5GT/s (gigatransfers per second) is the maximum speed allowed. • **gen2**—5GT/s is the maximum speed allowed. • **gen3**—8GT/s is the maximum speed allowed. • **auto**—The maximum speed is set automatically. • **disabled**—The maximum speed is not restricted. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Front NVME2 Link Speed **set Pcie SlotFrontNvme2LinkSpeed**	This option allows you to restrict the maximum speed of an NVME card installed in the front PCIe slot 2. This can be one of the following: • **gen1**—2.5GT/s (gigatransfers per second) is the maximum speed allowed. • **gen2**—5GT/s is the maximum speed allowed. • **gen3**—8GT/s is the maximum speed allowed. • **auto**—The maximum speed is set automatically. • **disabled**—The maximum speed is not restricted. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
HBA Link Speed **set HBALinkSpeed**	This option allows you to restrict the maximum speed of an HBA card. This can be one of the following: • **gen1**—2.5GT/s (gigatransfers per second) is the maximum speed allowed. • **gen2**—5GT/s is the maximum speed allowed. • **gen3**—8GT/s is the maximum speed allowed. • **auto**—The maximum speed is set automatically. • **disabled**—The maximum speed is not restricted. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
MLOM Link Speed **set Pcie SlotMLOMLinkSpeed**	This option allows you to restrict the maximum speed of an MLOM adapter. This can be one of the following: • **gen1**—2.5GT/s (gigatransfers per second) is the maximum speed allowed. • **gen2**—5GT/s is the maximum speed allowed. • **gen3**—8GT/s is the maximum speed allowed. • **auto**—The maximum speed is set automatically. • **disabled**—The maximum speed is not restricted. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
MRAID Link Speed **set Pcie SlotMRAIDLinkSpeed**	This option allows you to restrict the maximum speed of MRAID. This can be one of the following: • **gen1**—2.5GT/s (gigatransfers per second) is the maximum speed allowed. • **gen2**—5GT/s is the maximum speed allowed. • **gen3**—8GT/s is the maximum speed allowed. • **auto**—The maximum speed is set automatically. • **disabled**—The maximum speed is not restricted. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
RAID Link Speed **set Pcie SlotRAIDLinkSpeed**	This option allows you to restrict the maximum speed of RAID. This can be one of the following: • **gen1**—2.5GT/s (gigatransfers per second) is the maximum speed allowed. • **gen2**—5GT/s is the maximum speed allowed. • **gen3**—8GT/s is the maximum speed allowed. • **auto**—The maximum speed is set automatically. • **disabled**—The maximum speed is not restricted. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
All Onboard LOM **set AllLomPortControl**	Whether all onboard LOM ports are enabled or disabled. This can be one of the following: • **enabled**—All onboard LOM are enabled. • **disabled**—All onboard LOM are disabled. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
LOM Port 1 OptionRom **set** **LomOpromControlPort0**	Whether Option ROM is available on the LOM port 1. This can be one of the following: • **disabled**—The expansion slot is not available. • **enabled**—The expansion slot is available. • **uefi-only**—The expansion slot is available for UEFI only. • **legacy-only**—The expansion slot is available for legacy only. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
LOM Port 2 OptionRom **set** **LomOpromControlPort1**	Whether Option ROM is available on the LOM port 2. This can be one of the following: • **disabled**—The expansion slot is not available. • **enabled**—The expansion slot is available. • **uefi-only**—The expansion slot is available for UEFI only. • **legacy-only**—The expansion slot is available for legacy only. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Slot *n* State **set SlotnState**	The state of the adapter card installed in PCIe slot *n*. This can be one of the following: • **disabled**—The expansion slot is not available. • **enabled**—The expansion slot is available. • **uefi-only**—The expansion slot is available for UEFI only. • **legacy-only**—The expansion slot is available for legacy only. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
SBNVMe1 OptionROM **set** **SBNVMe1OptionROM**	Whether the server can use Option ROM present in SBNVMe1 controller. This can be one of the following: • **disabled**—The expansion slot is not available. • **enabled**—The expansion slot is available. • **uefi-only**—The expansion slot is available for UEFI only. • **legacy-only**—The expansion slot is available for legacy only. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
SBNVMe2 OptionROM **set** **SBNVMe2OptionROM**	Whether the server can use Option ROM present in SBNVMe2 controller. This can be one of the following: • **disabled**—The expansion slot is not available. • **enabled**—The expansion slot is available. • **uefi-only**—The expansion slot is available for UEFI only. • **legacy-only**—The expansion slot is available for legacy only. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
SIOCNVMe1 OptionROM **set** **SIOCNVMe1OptionROM**	Whether the server can use Option ROM present in SIOCNVMe1 controller. This can be one of the following: • **disabled**—The expansion slot is not available. • **enabled**—The expansion slot is available. • **uefi-only**—The expansion slot is available for UEFI only. • **legacy-only**—The expansion slot is available for legacy only. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
SIOCNVMe2 OptionROM **set** **SIOCNVMe2OptionROM**	Whether the server can use Option ROM present in SIOCNVMe2 controller. This can be one of the following: • **disabled**—The expansion slot is not available. • **enabled**—The expansion slot is available. • **uefi-only**—The expansion slot is available for UEFI only. • **legacy-only**—The expansion slot is available for legacy only. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
SBLom1 OptionROM **set SBLom1OptionROM**	Whether the server can use Option ROM present in the SBLom1 controller. This can be one of the following: • **disabled**—The expansion slot is not available. • **enabled**—The expansion slot is available. • **uefi-only**—The expansion slot is available for UEFI only. • **legacy-only**—The expansion slot is available for legacy only. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
SBNVMe*n* Link Speed **set SBNVMe*n*LinkSpeed**	Link speed for SBNVMe slot *n*. This can be one of the following: • **gen1**—2.5GT/s (gigatransfers per second) is the maximum speed allowed. • **gen2**—5GT/s is the maximum speed allowed. • **gen3**—8GT/s is the maximum speed allowed. • **enabled**—The maximum speed is restricted. • **disabled**—The maximum speed is not restricted. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
SIOCNVMe*n* Link Speed **set SIOCNVMe*n*LinkSpeed**	Link speed for SIOCNVMe slot *n*. This can be one of the following: • **gen1**—2.5GT/s (gigatransfers per second) is the maximum speed allowed. • **gen2**—5GT/s is the maximum speed allowed. • **gen3**—8GT/s is the maximum speed allowed. • **enabled**—The maximum speed is restricted. • **disabled**—The maximum speed is not restricted. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
SIOC*n* Link Speed **set SIOC*n*LinkSpeed**	Link speed for SIOC slot *n*. This can be one of the following: • **gen1**—2.5GT/s (gigatransfers per second) is the maximum speed allowed. • **gen2**—5GT/s is the maximum speed allowed. • **gen3**—8GT/s is the maximum speed allowed. • **enabled**—The maximum speed is restricted. • **disabled**—The maximum speed is not restricted. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
SBMezz*n* Link Speed **set SBMezz*n*LinkSpeed**	Link speed for SBMezz slot *n*. This can be one of the following: • **gen1**—2.5GT/s (gigatransfers per second) is the maximum speed allowed. • **gen2**—5GT/s is the maximum speed allowed. • **gen3**—8GT/s is the maximum speed allowed. • **enabled**—The maximum speed is restricted. • **disabled**—The maximum speed is not restricted. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
IOESlot*n* Link Speed **set IOESlot*n*LinkSpeed**	Link speed for IOE slot *n*. This can be one of the following: • **gen1**—2.5GT/s (gigatransfers per second) is the maximum speed allowed. • **gen2**—5GT/s is the maximum speed allowed. • **gen3**—8GT/s is the maximum speed allowed. • **enabled**—The maximum speed is restricted. • **disabled**—The maximum speed is not restricted. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
IOEMezz*n* Link Speed **set IOEMezz*n*LinkSpeed**	Link speed for IOEMezz slot *n*. This can be one of the following: • **gen1**—2.5GT/s (gigatransfers per second) is the maximum speed allowed. • **gen2**—5GT/s is the maximum speed allowed. • **gen3**—8GT/s is the maximum speed allowed. • **enabled**—The maximum speed is restricted. • **disabled**—The maximum speed is not restricted. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
IOENVMe*n* Link Speed **set IOENVMe*n*LinkSpeed**	Link speed for IOENVMe slot *n*. This can be one of the following: • **gen1**—2.5GT/s (gigatransfers per second) is the maximum speed allowed. • **gen2**—5GT/s is the maximum speed allowed. • **gen3**—8GT/s is the maximum speed allowed. • **enabled**—The maximum speed is restricted. • **disabled**—The maximum speed is not restricted. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
CDN Support for LOMs **set CdnSupport**	Whether the Ethernet Networking Identifier naming convention is according to Consistent Device Naming (CDN) or the traditional way of naming conventions. This can be one of the following: • **enabled**—OS Ethernet Network Identifier is named in a consistent device naming (CDN) convention according to the physical LAN on Motherboard(LOM) port numbering; LOM Port 0, LOM Port 1 and so on. • **disabled**—OS Ethernet Networking Identifier is named in a default convention as ETH0, ETH1 and so on. By default, CDN option is disabled. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
VMD Enable **set VMDEnable**	Whether NVMe SSDs that are connected to the PCIe bus can be hot swapped. It also standardizes the LED status light on these drives. This can be one of the following: • **enabled**—Hot swap of NVMe SSDs that are connected to the PCIe bus is allowed. • **disabled**—Hot swap of NVMe SSDs that are connected to the PCIe bus is not allowed. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
ACS Control SLOT-*n* **set ACSCtlSlot*n*** n = 11 to 14	Access Control Services (ACS) allow the processor to enable or disable peer-to-peer communication between multiple devices for Control Slot *n*. This can be one of the following: • **enabled**— Enables peer-to-peer communication between multiple devices for Control Slot *n*. • **disabled**— Disables peer-to-peer communication between multiple devices for Control Slot *n*. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
PCIe Slot GPU*n* OptionROM drop-down list Only for Cisco UCS C480 M5 ML Server	Whether the Option ROM is enabled on GPU slot *n*. *n* is the slot number, which can be between 1 and 8. This can be one of the following: • **disabled**—The expansion slot is not available. • **enabled**—The expansion slot is available. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
ACS Control GPU-*n* **set ACSCtlGpu*n*** n = 1 to 8	Access Control Services (ACS) allow the processor to enable or disable peer-to-peer communication between multiple devices for GPUs. This can be one of the following: • **disabled**— Enables peer-to-peer communication between multiple devices for GPUs. • **enabled**— Disables peer-to-peer communication between multiple devices for GPUs. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Graphics Configuration BIOS Settings

The following tables list the graphics configuration BIOS settings that you can configure through a BIOS policy or the default BIOS settings:

Name	Description
Integrated Graphics **set integrated-graphics-config integrated-graphics**	Enables integrated graphics. This can be one of the following: • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor. • **enabled**—Integrated graphic is enabled. • **disabled**—Integrated graphics is disabled.

Name	Description
Integrated Graphics Aperture Size **set integrated-graphics-aperture-config integrated-graphics-aperture**	Allows you to set the size of mapped memory for the integrated graphics controller. This can be one of the following: • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor. • **128mb** • **256mb** • **512mb** • **1024mb** • **2048mb** • **4096mb**
Onboard Graphics **set onboard-graphics-config onboard-graphics**	Enables onboard graphics (KVM). This can be one of the following: • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor. • **enabled**—Onboard graphics is enabled. • **disabled**—Onboard graphics is disabled.

Boot Options BIOS Settings

The following table lists the boot options BIOS settings that you can configure through a BIOS policy or the default BIOS settings:

Name	Description
Boot option retry **set boot-option-retry-config retry**	Whether the BIOS retries NON-EFI based boot options without waiting for user input. This can be one of the following: • **disabled**—Waits for user input before retrying NON-EFI based boot options. • **enabled**—Continually retries NON-EFI based boot options without waiting for user input. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
SAS RAID **set intel-entry-sas-raid-config sas-raid**	Whether the Intel SAS Entry RAID Module is enabled. This can be one of the following: • **disabled**—The Intel SAS Entry RAID Module is disabled. • **enabled**—The Intel SAS Entry RAID Module is enabled. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
SAS RAID module **set intel-entry-sas-raid-config sas-raid-module**	How the Intel SAS Entry RAID Module is configured. This can be one of the following: • **it-ir-raid**—Configures the RAID module to use Intel IT/IR RAID. • **intel-esrtii**—Configures the RAID module to use Intel Embedded Server RAID Technology II. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Onboard SCU Storage Support **set onboard-sas-storage-config onboard-sas-ctrl**	Whether the onboard software RAID controller is available to the server. This can be one of the following: • **disabled**—The software RAID controller is not available. • **enabled**—The software RAID controller is available. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Cool Down Time (sec) drop-down list	The time to wait (in seconds) before the next boot attempt. This can be one of the following: • **15**System waits for 15v seconds before the next boot attempt. • **45** • **90** • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor. This token is valid only when the Boot Option Retry token has been enabled.

Name	Description
Number of Retries drop-down list	Number of attempts to boot. This can be one of the following: • **infinite**System tries all options to boot up. • **13**System tries 5 times to boot up • **5** • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
P-SATA mode drop-down list	This options allows you to select the P-SATA mode. This can be one of the following: • **disabled**—P-SATA mode is disabled. • **lsi-sw-raid**—Sets both SATA and sSATA controllers to RAID mode for LSI SW RAID. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Power On Password drop-down list	This token requires that you set a BIOS password before using the F2 BIOS configuration. If enabled, password needs to be validated before you access BIOS functions such as IO configuration, BIOS set up, and booting to an operating system using BIOS. It can be one of the following: • **disabled**—Power On Password is disabled. • **enabled**—Power On Password is enabled. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
IPV6 PXE Support drop-down list	Enables or disables IPV6 support for PXE. This can be one of the following • **disabled**—IPV6 PXE support is not available. • **enabled**—IPV6 PXE support is always available. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
Adaptive Memory Training drop-down list	When this token is enabled, the BIOS saves the memory training results (optimized timing/voltage values) along with CPU/memory configuration information and reuses them on subsequent reboots to save boot time. The saved memory training results are used only if the reboot happens within 24 hours of the last save operation. This can be one of the following: • **disabled**—**Adaptive Memory Training** is disabled. • **enabled**—**Adaptive Memory Training** is enabled. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
BIOS Techlog Level **BIOS Tech Message Level Control** (for C125 M5)	Enabling this token allows the BIOS Tech log output to be controlled at more a granular level. This reduces the number of BIOS Tech log messages that are redundant, or of little use. This can be one of the following: • **disabled**—**BIOS Techlog Level** is disabled. • **enabled**—**BIOS Techlog Level** is enabled. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
OptionROM Launch Optimization	The Option ROM launch is controlled at the PCI Slot level, and is enabled by default. In configurations that consist of a large number of network controllers and storage HBAs having Option ROMs, all the Option ROMs may get launched if the PCI Slot Option ROM Control is enabled for all. However, only a subset of controllers may be used in the boot process. When this token is enabled, Option ROMs are launched only for those controllers that are present in boot policy. This can be one of the following: • **disabled**—**OptionROM Launch Optimization** is disabled. • **enabled**—**OptionROM Launch Optimization** is enabled. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Note BIOS parameter virtualization capability in Cisco UCS Manager maps a unified set of BIOS settings in a service profile to the actual BIOS supporting parameters. However, not all BIOS setting items are applicable to every server model/platform. When you create a custom BIOS policy and have the **Boot Option Retry** selected, and when there is no bootable option available, the reboot fails on the Cisco UCS B420 M3 or Cisco UCS B420 M4 servers and Cisco UCS Manager displays this message : *Reboot and Select proper Boot device or Insert Boot Media in selected Boot device and press a key*. You must manually set a boot option after the boot path is corrected, in order to enable the servers to reboot after a power outage. For more information about BIOS default server policies and the BIOS options and their default settings, see BIOS Policy, on page 259 and Server BIOS Settings, on page 197.

Server Management BIOS Settings

The following tables list the server management BIOS settings that you can configure through a BIOS policy or the default BIOS settings:

General Settings

Name	Description
Assert NMI on SERR **set assert-nmi-on-serr-config assertion**	Whether the BIOS generates a non-maskable interrupt (NMI) and logs an error when a system error (SERR) occurs. This can be one of the following: • **disabled**—The BIOS does not generate an NMI or log an error when a SERR occurs. • **enabled**—The BIOS generates an NMI and logs an error when a SERR occurs. You must enable this setting if you want to enable **Assert NMI on PERR**. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Assert NMI on PERR **set assert-nmi-on-perr-config assertion**	Whether the BIOS generates a non-maskable interrupt (NMI) and logs an error when a processor bus parity error (PERR) occurs. This can be one of the following: • **disabled**—The BIOS does not generate an NMI or log an error when a PERR occurs. • **enabled**—The BIOS generates an NMI and logs an error when a PERR occurs. You must enable **Assert NMI on SERR** to use this setting. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
OS Boot Watchdog Timer **set os-boot-watchdog-timer-config os-boot-watchdog-timer**	Whether the BIOS programs the watchdog timer with a predefined timeout value. If the operating system does not complete booting before the timer expires, the CIMC resets the system and an error is logged. This can be one of the following: • **disabled**—The watchdog timer is not used to track how long the server takes to boot. • **enabled**—The watchdog timer tracks how long the server takes to boot. If the server does not boot within the predefined length of time, the CIMC resets the system and logs an error. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor. This feature requires either operating system support or Intel Management software.
OS Boot Watchdog Timer Policy **set os-boot-watchdog-timer-policy-config os-boot-watchdog-timer-policy**	What action the system takes if the watchdog timer expires. This can be one of the following: • **power-off**—The server is powered off if the watchdog timer expires during OS boot. • **reset**—The server is reset if the watchdog timer expires during OS boot. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor. This option is only available if you enable the OS Boot Watchdog Timer.

Name	Description
OS Boot Watchdog Timer Timeout **set** **os-boot-watchdog-timer-timeout-config** **os-boot-watchdog-timer-timeout**	What timeout value the BIOS uses to configure the watchdog timer. This can be one of the following: • **5-minutes**—The watchdog timer expires 5 minutes after the OS begins to boot. • **10-minutes**—The watchdog timer expires 10 minutes after the OS begins to boot. • **15-minutes**—The watchdog timer expires 15 minutes after the OS begins to boot. • **20-minutes**—The watchdog timer expires 20 minutes after the OS begins to boot. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor. This option is only available if you enable the OS Boot Watchdog Timer.
FRB-2 Timer **set frb-2-timer-config frb-2-timer**	Whether the FRB-2 timer is used to recover the system if it hangs during POST. This can be one of the following: • **disabled**—The FRB-2 timer is not used. • **enabled**—The FRB-2 timer is started during POST and used to recover the system if necessary. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Console Redirection Settings

Name	Description
Console redirection **set console-redir-config console-redir**	Allows a serial port to be used for console redirection during POST and BIOS booting. After the BIOS has booted and the operating system is responsible for the server, console redirection is irrelevant and has no effect. This can be one of the following: • **disabled**—No console redirection occurs during POST. • **serial-port-a** or **COM0** or **Enabled**—Enables serial port A for console redirection during POST. This option is valid for blade servers and rack-mount servers. • **serial-port-b** or **COM1**—Enables serial port B for console redirection and allows it to perform server management tasks. This option is only valid for rack-mount servers. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor. **Note** If you enable this option, you also disable the display of the Quiet Boot logo screen during POST.
Flow Control **set console-redir-config flow-control**	Whether a handshake protocol is used for flow control. Request to Send / Clear to Send (RTS/CTS) helps to reduce frame collisions that can be introduced by a hidden terminal problem. This can be one of the following: • **none**—No flow control is used. • **rts-cts**—RTS/CTS is used for flow control. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor. **Note** This setting must match the setting on the remote terminal application.

Name	Description
Baud rate **set console-redir-config baud-rate**	What Baud rate is used for the serial port transmission speed. If you disable Console Redirection, this option is not available. This can be one of the following: • **9600**—A 9600 Baud rate is used. • **19200**—A 19200 Baud rate is used. • **38400**—A 38400 Baud rate is used. • **57600**—A 57600 Baud rate is used. • **115200**—A 115200 Baud rate is used. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor. **Note** This setting must match the setting on the remote terminal application.
Terminal type **set console-redir-config terminal-type**	What type of character formatting is used for console redirection. This can be one of the following: • **pc-ansi**—The PC-ANSI terminal font is used. • **vt100**—A supported vt100 video terminal and its character set are used. • **vt100-plus**—A supported vt100-plus video terminal and its character set are used. • **vt-utf8**—A video terminal with the UTF-8 character set is used. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor. **Note** This setting must match the setting on the remote terminal application.

Name	Description
Legacy OS redirection **set console-redir-config legacy-os-redir**	Whether redirection from a legacy operating system, such as DOS, is enabled on the serial port. This can be one of the following: • **disabled**—The serial port enabled for console redirection is hidden from the legacy operating system. • **enabled**— The serial port enabled for console redirection is visible to the legacy operating system. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Putty KeyPad **set console-redir-config putty-function-keypad**	Allows you to change the action of the PuTTY function keys and the top row of the numeric keypad. This can be one of the following: • **vt100**—The function keys generate `ESC OP` through `ESC O[`. • **linux**—Mimics the Linux virtual console. Function keys F6 to F12 behave like the default mode, but F1 to F5 generate `ESC [[A` through `ESC [[E`. • **xtermr6**—Function keys F5 to F12 behave like the default mode. Function keys F1 to F4 generate **ESC OP** through ESC OS, which are the sequences produced by the top row of the keypad on Digital terminals. • **sco**—The function keys F1 to F12 generate `ESC [M` through `ESC [X`. The function and shift keys generate `ESC [Y` through `ESC [j`. The control and function keys generate `ESC [k` through `ESC [v`. The shift, control and function keys generate `ESC [w` through `ESC [{`. • **escn**—The default mode. The function keys match the general behavior of Digital terminals. The function keys generate sequences such as `ESC [11~` and `ESC [12~`. • **vt400**—The function keys behave like the default mode. The top row of the numeric keypad generates `ESC OP` through `ESC OS`. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Name	Description
Out of Band Management drop-down list	Used for Windows Special Administration Control (SAC). This option allows you to configure the COM port 0 that can be used for Windows Emergency Management services. ACPI SPCR table is reported based on this setup option. This can be one of the following: • **disabled**—Configures the COM port 0 as a general purpose port for use with the Windows Operating System. • **enabled**—Configures the COM port 0 as a remote management port for Windows Emergency Management services. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.
Redirection After BIOS POST drop-down list **set console-redir-config putty-function-keypad**	Whether BIOS console redirection should be active after BIOS POST is complete and control given to the OS bootloader. This can be one of the following: • **always_enable**—BIOS Legacy console redirection is active during the OS boot and run time. • **bootloader**—BIOS Legacy console redirection is disabled before giving control to the OS boot loader. • **platform-default**—The BIOS uses the value for this attribute contained in the BIOS defaults for the server type and vendor.

Intel Speed Select

Intel Speed Select lets you optimize CPU performance by selecting one of three operating profiles, based on number of logical processor cores, frequency, and TDP thread setting. Speed Select is configurable with a BIOS policy based on the SKU. At boot time, the BIOS discovers the supported CPU and configures it to one of three specified profiles.

Set profiles by configuring the Speed Select BIOS policy tokens. You create policies based on performance needs. Tuning can be based on workload or VM needs, need for improved server utilization in the data center, or improved per-core performance SLAs. Low core profiles let the CPU run at maximum frequency.

The Speed Select profiles are:

- Base—High Core Configuration. Suggested for general purpose VM instances.

- Config 1—Medium Core Configuration. Suggested for high performance VM instances or Storage SKUs.

- Config 2—Low Core Configuration. Suggested for very high frequency Bare Metal servers or VM instances.

The following tables show sample profile values based on the Intel® Xeon® Gold CPU 6240Y. Base is the default profile.

Table 6: Base Config

SKU	Cores	Base Frequency (GHz)	All Core Turbo	LLC/Core (Mb)
8260Y	24	2.4	3.1	1.49
6240Y	18	2.6	3.3	1.375
4214Y	12	2.2	2.7	1.375

SKU	TDP (W)	Tj Max (ºC)
8260Y	160	104
6240Y	150	88
4214Y	85	88

Table 7: Config 1

SKU	Cores	Base Frequency (GHz)	All Core Turbo	LLC/Core (Mb)
8260Y	20	2.5	3.3	1.79
6240Y	14	2.8	3.4	1.77
4214Y	10	2.3	2.7	1.65

SKU	TDP (W)	Tj Max (ºC)
8260Y	155	TControl Spec
6240Y	150	TControl Spec
4214Y	85	TControl Spec

Table 8: Config 2

SKU	Cores	Base Frequency (GHz)	All Core Turbo	LLC/Core (Mb)
8260Y	16	2.7	3.6	2.23
6240Y	8	3.1	3.6	3.09
4214Y	8	2.4	2.9	2.06

SKU	TDP (W)	Tj Max (ºC)
8260Y	150	TControl Spec
6240Y	150	TControl Spec
4214Y	85	TControl Spec

Note   While changing the profiles can make the CPU run at a different frequency, the CIMC processor inventory page will show information for the processor Base frequency, core, and threads. It will not display running frequencies.

BIOS Policy

The BIOS policy is a policy that automates the configuration of BIOS settings for a server or group of servers. You can create global BIOS policies available to all servers in the root organization, or you can create BIOS policies in sub-organizations that are only available to that hierarchy.

To use a BIOS policy, do the following:

1. Create the BIOS policy in Cisco UCS Manager.

2. Assign the BIOS policy to one or more service profiles.

3. Associate the service profile with a server.

During service profile association, Cisco UCS Manager modifies the BIOS settings on the server to match the configuration in the BIOS policy. If you do not create and assign a BIOS policy to a service profile, the server uses the default BIOS settings for that server platform.

Default BIOS Settings

Cisco UCS Manager includes a set of default BIOS settings for each type of server supported by Cisco UCS. The default BIOS settings are available only in the root organization and are global. Only one set of default BIOS settings can exist for each server platform supported by Cisco UCS. You can modify the default BIOS settings, but you cannot create an additional set of default BIOS settings.

Each set of default BIOS settings are designed for a particular type of supported server and are applied to all servers of that specific type which do not have a BIOS policy included in their service profiles.

Unless a Cisco UCS implementation has specific needs that are not met by the server-specific settings, we recommend that you use the default BIOS settings that are designed for each type of server in the Cisco UCS domain.

Cisco UCS Manager applies these server platform-specific BIOS settings as follows:

• The service profile associated with a server does not include a BIOS policy.

• The BIOS policy is configured with the platform-default option for a specific setting.

You can modify the default BIOS settings provided by Cisco UCS Manager. However, any changes to the default BIOS settings apply to all servers of that particular type or platform. If you want to modify the BIOS settings for only certain servers, we recommend that you use a BIOS policy.

The BIOS tokens for M5 servers and later are read-only and cannot be modified. For a complete and up to date list of BIOS tokens, defaults, and values, refer Cisco UCS M5 Server BIOS Tokens .

Creating a BIOS Policy

> **Note** Cisco UCS Manager pushes BIOS configuration changes through a BIOS policy or default BIOS settings to the Cisco Integrated Management Controller (CIMC) buffer. These changes remain in the buffer and do not take effect until the server is rebooted.
>
> We recommend that you verify the support for BIOS settings in the server that you want to configure. Some settings, such as Mirroring Mode for RAS Memory, are not supported by all Cisco UCS servers.

Procedure

	Command or Action	**Purpose**
Step 1	UCS-A# **scope org** *org-name*	Enters org mode for the specified organization. To enter the default org mode, type / as the *org-name* .
Step 2	UCS-A /org # **create bios-policy** *policy-name*	Creates a BIOS policy with the specified policy name, and enters org BIOS policy mode.
Step 3	Configure the BIOS settings.	
Step 4	UCS-A /org/bios-policy # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example creates a BIOS policy under the root organization and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # create bios-policy biosPolicy3
UCS-A /org/bios-policy* # set numa-config numa-optimization enabled
UCS-A /org/bios-policy* # commit-buffer
UCS-A /org/bios-policy #
```

Modifying BIOS Defaults

You can use the following procedure to modify and configure the BIOS defaults for UCS M4 and earlier servers. The new BIOS settings that are introduced with the UCS M5 servers cannot be configured using this procedure.

We recommend that you verify the support for BIOS settings in the server that you want to configure. Some settings, such as Mirroring Mode for RAS Memory, are not supported by all Cisco UCS servers.

Unless a Cisco UCS implementation has specific needs that are not met by the server-specific settings, we recommend that you use the default BIOS settings that are designed for each type of server in the Cisco UCS domain.

Procedure

	Command or Action	**Purpose**
Step 1	UCS-A# **scope system**	Enters system mode.
Step 2	UCS-A /system # **scope server-defaults**	Enters server defaults mode.
Step 3	UCS-A /system/server-defaults # **show platform**	(Optional) Displays platform descriptions for all servers.
Step 4	UCS-A /system/server-defaults # **scope platform** *platform-description*	Enters server defaults mode for the server specified. For the *platform-description* argument, enter the server description displayed by the **show platform** command using the following format: "*vendor*" *model revision*. **Tip** You must enter the vendor exactly as shown in the **show platform** command, including all punctuation marks.
Step 5	UCS-A /system/server-defaults/platform # **scope bios-settings**	Enters server defaults BIOS settings mode for the server.
Step 6	Reconfigure the BIOS settings.	
Step 7	UCS-A /system/server-defaults/platform/bios-settings # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to change the NUMA default BIOS setting for a platform and commit the transaction:

```
UCS-A# scope system
UCS-A /system # scope server-defaults
UCS-A /system/server-defaults # show platform

Platform:
    Product Name Vendor      Model       Revision
    ------------ ---------- ---------- --------
    Cisco B200-M1
              Cisco Systems, Inc.
                        N20-B6620-1
                                  0

UCS-A /system/server-defaults # scope platform "Cisco Systems, Inc." N20-B6620-1 0
UCS-A /system/server-defaults/platform # scope bios-settings
UCS-A /system/server-defaults/platform/bios-settings # set numa-config numa-optimization
disabled
```

```
UCS-A /system/server-defaults/platform/bios-settings* # commit-buffer
UCS-A /system/server-defaults/platform/bios-settings #

UCS-A# scope system
UCS-A /system # scope server-defaults
UCS-A /system/server-defaults # show platform

Platform:
    Product Name Vendor     Model      Revision
    ------------ ---------- ---------- --------
    Cisco UCS B230-M2
                Cisco Systems, Inc.
                           B230-BASE-M2
                                       0
    Cisco UCS B440 M2
                Cisco Systems, Inc.
                           B440-BASE-M2
                                       0
    Cisco C260-M2
                Cisco Systems, Inc.
                           C260-BASE-2646
                                       0
    Cisco B200-M1
                Cisco Systems, Inc.
                           N20-B6620-1
                                       0
    Cisco B250-M1
                Cisco Systems, Inc.
                           N20-B6620-2
UCS-A /system/server-defaults # scope platform "Cisco Systems, Inc." B230-BASE-M2 0
UCS-A /system/server-defaults/platform # scope bios-settings
UCS-A /system/server-defaults/platform/bios-settings # set numa-config numa-optimization
disabled
UCS-A /system/server-defaults/platform/bios-settings* # committt-buffer
UCS-A /system/server-defaults/platform/bios-settings* #
```

Configuring BIOS Settings for M5 Servers

You can configure BIOS settings for UCS M5 and earlier servers through Cisco UCS Manager CLI. The new BIOS settings that are introduced with the UCS M5 servers can be configured only by using this procedure.

We recommend that you verify the support for BIOS settings in the server that you want to configure. Some settings, such as Mirroring Mode for RAS Memory, are not supported by all Cisco UCS servers.

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 bios-policy** *bios-policy-name*	Enters the bios-policy mode for the specified BIOS policy.
Step 3	(Optional) UCS-A /org/bios-policy # **show token-feature**	Displays the complete list of BIOS token features in the specified BIOS policy.

	Command or Action	**Purpose**
Step 4	UCS-A /org/bios-policy # scope token-feature "*bios-token-feature-name*"	Enters the token feature mode for the specified BIOS token feature.
Step 5	(Optional) UCS-A /org/bios-policy/token-feature # **show token-param**	Displays the complete list of BIOS token parameters for the specified BIOS token feature.
Step 6	UCS-A /org/bios-policy/token-feature # scope token-param *bios-token-parameter-name*	Enters the token parameter mode for the specified BIOS token parameter name.
Step 7	(Optional) UCS-A /org/bios-policy/token-feature/token-param # **show token-settings**	Displays the complete list of token settings for the specified BIOS token parameter.
Step 8	UCS-A /org/bios-policy/token-feature/token-param # **scope token-settings** *token-setting*	Enters the token settings mode for the specified BIOS token parameter name.
Step 9	UCS-A /org/bios-policy/token-feature/token-param/token-settings # **set is-selected** *yes	no*	Set the specified token setting as selected or not by using the **yes** or **no** keyword.
Step 10	UCS-A /org/bios-policy/token-feature/token-param/token-settings # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to configure BIOS token settings:

```
UCS-A# scope org
UCS-A /org # scope bios-policy bp
UCS-A /org/bios-policy # scope token-feature "Consistent Device Name Control"
UCS-A /org/bios-policy/token-feature # scope token-param cdnEnable
UCS-A /org/bios-policy/token-feature/token-param # scope token-settings Enabled
UCS-A /org/bios-policy/token-feature/token-param/token-settings # set is-selected yes
UCS-A /org/bios-policy/token-feature/token-param/token-settings* # commit-buffer
UCS-A /org/bios-policy/token-feature/token-param/token-settings #
```

Viewing the Actual BIOS Settings for M3 and M4 Servers

Follow this procedure to see the actual BIOS settings on a server.

Procedure

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

	Command or Action	**Purpose**
Step 3	UCS-A /chassis/server/bios # **scope bios-settings**	Enters BIOS settings mode for the specified server.
Step 4	UCS-A /chassis/server/bios/bios-settings # **show** *setting*	Displays the BIOS setting. Enter **show ?** to display a list of allowed values for *setting*.
		Note The **show** *setting* command is not supported on M5 and higher servers. For M5 and higher servers, see Viewing the Actual BIOS Settings for M5 and Higher Servers, on page 264.

Example

The following example displays a BIOS setting for blade 3 in chassis 1:

```
UCS-A# scope server 1/3
UCS-A /chassis/server # scope bios
UCS-A /chassis/server/bios # scope  bios-settings
UCS-A /chassis/server/bios/bios-settings # show intel-vt-config

Intel Vt Config:
    Vt
    --
    Enabled

UCS-A /chassis/server/bios/bios-settings #
```

Viewing the Actual BIOS Settings for M5 and Higher Servers

Follow this procedure to see the actual BIOS settings on a server.

Procedure

	Command or Action	**Purpose**
Step 1	UCS-A# **scope server** *chassis-id* / *server-id*	Enters chassis server mode for the specified server.
Step 2	UCS-A /chassis/server # **scope bios**	Enters BIOS mode for the specified server.
Step 3	UCS-A /chassis/server/bios # **scope bios-settings**	Enters BIOS settings mode for the specified server.
Step 4	UCS-A /chassis/server/bios/bios-settings # **show token-feature** "*BIOS_Token_Feature_Name*" **detail expand**	Displays the BIOS setting for a specific BIOS token feature name. Enter **show ?** to display a list of allowed values for *BIOS_Token_Feature_Name*.

	Command or Action	**Purpose**
Step 5	(Optional) UCS-A /chassis/server/bios/bios-settings # **show detail**	Displays the BIOS setting for all the BIOS tokens.

Example

The following example displays BIOS setting for Consistent Device Name Control on blade 4 in chassis 1:

```
UCS-A# scope server 1/4
UCS-A /chassis/server # scope bios
UCS-A /chassis/server/bios # scope  bios-settings
UCS-A /chassis/server/bios/bios-settings # show token-feature "Consistent Device Name
Control" detail expand

Token Feature:
    Bios Token Feature Name: Consistent Device Name Control

    Token Parameter:
        Bios Token Parameter Name: cdnEnable
        UI Display Name: CDN Control

        Token Settings:
            Bios Token Settings Name: Disabled
            BIOS Returned Setting Name: Disabled
            Selected: Yes
UCS-A /chassis/server/bios/bios-settings #
```

Displaying Details of BIOS Tokens in a BIOS Policy

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 bios-policy** *bios-policy-name*	Enters the bios-policy mode for the specified BIOS policy.
Step 3	UCS-A /org/bios-policy # **show detail expand**	Displays detailed information about all the BIOS token features, parameters, and settings that are configured for the specified BIOS policy.
Step 4	(Optional) UCS-A /org/bios-policy # scope token-feature "*bios-token-feature-name*"	Enters the token feature mode for the specified BIOS token feature.
Step 5	(Optional) UCS-A /org/bios-policy/token-feature # **show detail** [**expand**]	Displays the complete list of BIOS token parameters for the specified BIOS token feature.

	Command or Action	**Purpose**
Step 6	(Optional) UCS-A /org/bios-policy/token-feature # scope token-param *bios-token-parameter-name*	Enters the token parameter mode for the specified BIOS token parameter name.
Step 7	(Optional) UCS-A /org/bios-policy/token-feature/token-param # **show detail** [**expand**]	Displays the complete list of token settings for the specified BIOS token parameter.

Example

This example shows how to display detailed information about a BIOS policy, including all the BIOS token features, parameters, and settings:

```
UCS-A# scope org
UCS-A /org # scope bios-policy bp
UCS-A /org/bios-policy # show detail expand

BIOS Policy:
    Name: bp
    Description: Recommended bios settings for bp
    Reboot on BIOS Policy Change: No
    Policy Owner: Local

    Token Feature:
        Bios Token Feature Name: All USB Devices

        Token Parameter:
            Bios Token Parameter Name: AllUsbDevices
            UI Display Name: All USB Devices

            Token Settings:
                Bios Token Settings Name: Disabled
                BIOS Returned Setting Name: Disabled
                Selected: No

                Bios Token Settings Name: Enabled
                BIOS Returned Setting Name: Enabled
                Selected: No

        Bios Token Feature Name: Altitude

        Token Parameter:
            Bios Token Parameter Name: Altitude
            UI Display Name: Altitude

            Token Settings:
                Bios Token Settings Name: 1500-M
                BIOS Returned Setting Name: 1500 M
                Selected: No

                Bios Token Settings Name: 300-M
                BIOS Returned Setting Name: 300 M
                Selected: No

                Bios Token Settings Name: 3000-M
                BIOS Returned Setting Name: 3000 M
                Selected: No

                Bios Token Settings Name: 900-M
```

```
                              BIOS Returned Setting Name: 900 M
                              Selected: No

                              Bios Token Settings Name: Auto
                              BIOS Returned Setting Name: Auto
                              Selected: No

      ...
```

This example shows how to display detailed information about the BIOS token parameters for a specific BIOS token feature:

```
UCS-A# scope org
UCS-A /org # scope bios-policy bp
UCS-A /org/bios-policy # scope token-feature "Console redirection"
UCS-A /org/bios-policy/token-feature # show detail expand

Token Feature:
    Bios Token Feature Name: Console redirection

    Token Parameter:
        Bios Token Parameter Name: BaudRate
        UI Display Name: Baud rate

        Token Settings:
            Bios Token Settings Name: 115.2k
            BIOS Returned Setting Name: 115.2k
            Selected: No

            Bios Token Settings Name: 19.2k
            BIOS Returned Setting Name: 19.2k
            Selected: No

            Bios Token Settings Name: 38.4k
            BIOS Returned Setting Name: 38.4k
            Selected: No

            Bios Token Settings Name: 57.6k
            BIOS Returned Setting Name: 57.6k
            Selected: No

            Bios Token Settings Name: 9.6k
            BIOS Returned Setting Name: 9.6k
            Selected: No


        Bios Token Parameter Name: FlowCtrl
        UI Display Name: Flow Control

        Token Settings:
            Bios Token Settings Name: None
            BIOS Returned Setting Name: None
            Selected: No

            Bios Token Settings Name: RTS-CTS
            BIOS Returned Setting Name: RTS-CTS
            Selected: No
```

This example shows how to display detailed information about the BIOS token settings for a specific BIOS token parameter:

```
UCS-A# scope org
UCS-A /org # scope bios-policy bp
UCS-A /org/bios-policy # scope token-feature "Console redirection"
UCS-A /org/bios-policy/token-feature # scope token-param BaudRate
UCS-A /org/bios-policy/token-feature/token-param # show detail expand

Token Parameter:
    Bios Token Parameter Name: BaudRate
    UI Display Name: Baud rate

    Token Settings:
        Bios Token Settings Name: 115.2k
        BIOS Returned Setting Name: 115.2k
        Selected: No

        Bios Token Settings Name: 19.2k
        BIOS Returned Setting Name: 19.2k
        Selected: No

        Bios Token Settings Name: 38.4k
        BIOS Returned Setting Name: 38.4k
        Selected: No

        Bios Token Settings Name: 57.6k
        BIOS Returned Setting Name: 57.6k
        Selected: No

        Bios Token Settings Name: 9.6k
        BIOS Returned Setting Name: 9.6k
        Selected: No
```

Trusted Platform Module

Trusted Platform Module

The Trusted Platform Module (TPM) is a component that can securely store artifacts that are used to authenticate the server. These artifacts can include passwords, certificates, or encryption keys. A TPM can also be used to store platform measurements that help ensure that the platform remains trustworthy. Authentication (ensuring that the platform can prove that it is what it claims to be) and attestation (a process helping to prove that a platform is trustworthy and has not been breached) are necessary steps to ensure safer computing in all environments. It is a requirement for the Intel Trusted Execution Technology (TXT) security feature, which must be enabled in the BIOS settings for a server equipped with a TPM. Cisco UCS M4 blade and rack-mount servers include support for TPM. TPM is enabled by default on these servers.

☞

Important

- If you upgrade Cisco UCS Manager to Release 2.2(4) and higher, TPM is enabled.

- When TPM is enabled and you downgrade Cisco UCS Manager from Release 2.2(4) and higher, TPM is disabled.

Intel Trusted Execution Technology

Intel Trusted Execution Technology (TXT) provides greater protection for information that is used and stored on the business server. A key aspect of that protection is the provision of an isolated execution environment and associated sections of memory where operations can be conducted on sensitive data, invisible to the rest of the system. Intel TXT provides for a sealed portion of storage where sensitive data such as encryption keys can be kept, helping to shield them from being compromised during an attack by malicious code. Cisco UCS M4 blade and rack-mount servers include support for TXT. TXT is disabled by default on these servers.

TXT can be enabled only after TPM, Intel Virtualization technology (VT) and Intel Virtualization Technology for Directed I/O (VT-d) are enabled. When you only enable TXT, it also implicitly enables TPM, VT, and VT-d.

Enabling or Disabling TPM

Procedure

	Command or Action	**Purpose**
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter / as the *org-name*.
Step 2	UCS-A /org # **create bios-policy** *policy-name*	Creates a BIOS policy with the specified policy name, and enters org BIOS policy mode.
Step 3	UCS-A /org/bios-policy* # **set trusted-platform-module-config tpm-state** {**disabled**	**enabled**	**platform-default**}	Specifies whether TPM is **enabled** or **disabled**. **platform-default** is TPM enabled.
Step 4	UCS-A /org/bios-policy* # **commit-buffer**	Commits the transaction to the system configuration.
Step 5	UCS-A /org # **create service-profile** *sp-name*}	Creates the service profile specified and enters service profile configuration mode.
Step 6	UCS-A /org/service-profile* # **set bios-policy** *policy-name*	Associates the specified BIOS policy with the service profile.
Step 7	UCS-A /org/service-profile* # **commit-buffer**	Commits the transaction to the system configuration.
Step 8	UCS-A /org/service-profile # **associate server** *chassis-id / slot-id*	Associates the service profile with a single server.

Example

The following example shows how to enable TPM:

```
UCS-A # scope org
UCS-A /org # create bios-policy bp1
UCS-A /org/bios-policy* # set trusted-platform-module-config tpm-state enabled
UCS-A /org/bios-policy* # commit-buffer
```

```
UCS-A /org # create service-profile sp1
UCS-A /org/service-profile* # set bios-policy bp1
UCS-A /org/service-profile* # commit-buffer
UCS-A /org/service-profile # associate server 1/2
```

Viewing TPM Properties

Procedure

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

Example

The following example shows how to display the TPM properties a modular server:

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

Trusted Platform Module:
    Presence: Equipped
    Enabled Status: Enabled
    Active Status: Activated
    Ownership: Unowned
UCS-A /chassis/cartridge/server/tpm # show detail

Trusted Platform Module:
    Enabled Status: Enabled
    Active Status: Activated
    Ownership: Unowned
    Tpm Revision: 2
    Model: UCSX-TPM2-001
    Vendor: Cisco Systems Inc
    Serial: FCH19257E58
    Admin Action: Unspecified
    Config State: Not Applied
UCS-A /chassis/cartridge/server/tpm #
```

Enabling or Disabling TXT

Procedure

	Command or Action	**Purpose**
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter / as the *org-name*.
Step 2	UCS-A /org # **create bios-policy** *policy-name*	Creates a BIOS policy with the specified policy name, and enters org BIOS policy mode.
Step 3	UCS-A /org/bios-policy* # **set intel-trusted-execution-technology-config txt-support** {**enabled**	**disabled**	**platform-default**}	Specifies whether TXT is **enabled** or **disabled**. **platform-default** is TXT disabled.
Step 4	UCS-A /org/bios-policy* # **commit-buffer**	Commits the transaction to the system configuration.
Step 5	UCS-A /org # **create service-profile** *sp-name*}	Creates the service profile specified and enters service profile configuration mode.
Step 6	UCS-A /org/service-profile* # **set bios-policy** *policy-name*	Associates the specified BIOS policy with the service profile.
Step 7	UCS-A /org/service-profile* # **commit-buffer**	Commits the transaction to the system configuration.
Step 8	UCS-A /org/service-profile # **associate server** *chassis-id / slot-id*	Associates the service profile with a single server.

Example

The following example shows how to enable TXT:

```
UCS-A # scope org
UCS-A /org # create bios-policy bp1
UCS-A /org/bios-policy* # set intel-trusted-execution-technology-config txt-support enabled
UCS-A /org/bios-policy* # commit-buffer
UCS-A /org # create service-profile sp1
UCS-A /org/service-profile* # set bios-policy bp1
UCS-A /org/service-profile* # commit-buffer
UCS-A /org/service-profile # associate server 1/2
```

Consistent Device Naming

When there is no mechanism for the Operating System to label Ethernet interfaces in a consistent manner, it becomes difficult to manage network connections with server configuration changes. Consistent Device

Naming (CDN), introduced in Cisco UCS Manager Release 2.2(4), allows Ethernet interfaces to be named in a consistent manner. This makes Ethernet interface names more persistent when adapter or other configuration changes are made.

To configure CDN for a vNIC, do the following:

- Enable consistent device naming in the BIOS policy.

- Associate the BIOS policy with a service profile.

- Configure consistent naming for a vNIC.

Guidelines and Limitations for Consistent Device Naming

- CDN is supported on the following Operating Systems:

    - Windows 2012 R2

    - Windows 2016

    - Windows Server 2019

    - Red Hat Enterprise Linux (RHEL) 6.x and 7.x

    - SLES 12 SP3, SLES 12 SP4, and SLES 15 (for 4.0(4a) and later)

    - ESXi 6.7

- Consistent device naming (CDN) is supported on all M3 and higher blade and rack-mount servers.

- BIOS and adapter firmware must be part of the Release 2.2(4) or higher bundle to support CDN.

- In Cisco UCS Manager Release 3.1 and later releases, CDN is supported on the following adapters:

    - Cisco UCS VIC 1455 (UCSC-PCIE-C25Q-04)

    - Cisco UCS VIC 1457 (UCSC-MLOM-C25Q-04)

    - Cisco UCS VIC 1385 (UCSC-PCIE-C40Q-03)

    - Cisco UCS VIC 1387 (UCSC-MLOM-C40Q-03)

- If the RHEL Operating System is installed on the server, CDN will appear when running the command "**biosdevname -d**" as "**sysfs label**". CDN will not change the kernel name.

- In Cisco UCS Manager Release 2.2(4), CDN is supported only on the following adapters:

    - Cisco UCS VIC 1225 (UCSC-PCIE-CSC-02)

    - Cisco UCS MLOM 1227 (UCSC-MLOM-CSC-02)

    - Cisco UCS VIC 1225T (UCSC-PCIE-C10T-02)

    - Cisco UCS MLOM 1227T (UCSC-MLOM-C10T-02)

    - Cisco UCS VIC 1240 (UCSB-MLOM-40G-01)

    - Cisco UCS VIC 1280 (UCS-VIC-M82-8P)

    - Cisco UCS VIC 1340 (UCSB-MLOM-40G-03)

• Cisco UCS VIC 1380 (UCSB-VIC-M83-8P)

• CDN is supported for vNIC template and dynamic vNIC.

• Multiple vNICs within the same service profile cannot have the same CDN name.

• When a CDN name is not specified for a vNIC, the vNIC name is used as the CDN name.

• The CDN name that you configure for a vNIC appears as **Admin CDN Name**. The CDN name that is finally applied to the vNIC appears as **Oper CDN Name**. For example, if the **Admin CDN Name** for a vNIC called "vnic0" is cdn0, then the **Oper CDN Name** for this vNIC will be cdn0, but if the **Admin CDN Name** for the same vNIC is not specified, the **Oper CDN Name** will be vnic0.

• In Cisco UCS Manager Release 3.1 and later releases, downgrade of the adapter firmware is prevented if a CDN-enabled BIOS policy is assigned to a server.

• In Cisco UCS Manager Release 2.2(4), downgrade of Cisco UCS Manager is prevented if CDN is enabled in a BIOS policy that is assigned to an associated server.

• In Cisco UCS Manager Release 2.2(4), downgrade of the BIOS firmware is prevented if a CDN-enabled BIOS policy is assigned to a server.

• In Cisco UCS Manager Release 2.2(4), downgrade of the adapter firmware is prevented if a CDN-enabled BIOS policy is assigned to a server.

• When the applied BIOS policy is changed from CDN-disabled to CDN-enabled or from CDN-enabled to CDN-disabled, the host reboots with a warning, irrespective of whether reboot on BIOS update is enabled or not.

• It is recommended that you enable CDN in the BIOS policy and add CDN names to the vNICS before the Windows Operating System is installed.

• If the Windows Operating System is already installed on the server and CDN is then enabled in the BIOS policy, do the following:

  1. Uninstall the network drivers.

  2. Scan the system for hidden devices and uninstall them.

  3. Rescan the system for new hardware and install the network drivers again.

  If this is not done, the vNICs will not come up with the configured CDN names.

• When the applied BIOS policy is changed from CDN-disabled to CDN-enabled or from CDN-enabled to CDN-disabled on a service profile, do the following:

  1. Uninstall the network drivers.

  2. Scan the system for hidden devices and delete them.

  3. Re-scan the system for new hardware and install the network drivers again.

Note   When the BIOS policy is changed from CDN-enabled to CDN-disabled, ensure that the CDN names are removed from all the vNICs on the system.

- If any change is made to the vNICs, the BDF of all the devices on the system also changes. Following are some of the scenarios that trigger a change in the BDF of all the vNICs present on the system:

  - When a vNIC is added or deleted

  - When a vNIC is moved from one adapter on the system to another adapter on the system

When these changes are made to the system, do the following:

1. Uninstall the network driver from all the present network interfaces.

2. Scan the system for hidden devices and uninstall them.

3. Re-scan the system for new hardware and install the network driver on the network controllers again.

If the hidden devices are not deleted, the CDN names of the network adapters will not appear as configured on Cisco UCS Manager.

CDN with a Mixed Set of Adapters

When a CDN name is configured for a vNIC in a system with a mixed set of CDN-supported adapters and CDN-unsupported adapters, then system placement may not place CDN-configured vNICs on adapters that support CDN.

If CDN is enabled in the BIOS policy, and system placement places a CDN-configured vNIC (Admin CDN configured) on an adapter that does not support CDN, an info fault will be raised, but the configuration issue for the service profile will be ignored.

If CDN is enabled in the BIOS policy, and system placement places a vNIC (Admin CDN not configured) on an adapter that does not support CDN, an info fault will be raised, but the configuration issue for the service profile will be ignored. The **Oper CDN Name** in this case will be empty and will not be derived from the vNIC name.

If you want to deploy the CDN name as the host network interface name for a server, you must manually place a vNIC on a supported adapter.

Enabling Consistent Device Naming in a BIOS Policy

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter / as the *org-name*.
Step 2	UCS-A /org # **create bios-policy** *policy-name*	Creates a BIOS policy with the specified policy name, and enters org BIOS policy mode.
Step 3	UCS-A /org/bios-policy* # **set consistent-device-name-control cdn-name** {**enabled** | **disabled** | **platform-default**}	Specifies whether consistent device naming (CDN) is **enabled** or **disabled**.
Step 4	UCS-A /org/bios-policy* # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to enable CDN in a BIOS policy:

```
UCS-A # scope org
UCS-A /org # create bios-policy cdn-bios-policy
UCS-A /org/bios-policy* # set consistent-device-name-control cdn-name enabled
UCS-A /org/bios-policy* # commit-buffer
```

Associating a BIOS Policy with a Service Profile

Procedure

	Command or Action	**Purpose**
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter / as the *org-name*.
Step 2	UCS-A /org # **scope service-profile** *sp-name*}	Enters service profile configuration mode for the specified service profile.
Step 3	UCS-A /org/service-profile # **set bios-policy** *policy-name*	Associates the specified BIOS policy with the service profile.
Step 4	UCS-A /org/service-profile* # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to associate a CDN-enabled BIOS policy with a service profile:

```
UCS-A # scope org
UCS-A /org # scope service-profile sp1
UCS-A /org/service-profile # set bios-policy cdn-bios-policy
UCS-A /org/service-profile* # commit-buffer
```

Configuring Consistent Device Naming for a vNIC

Procedure

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

	Command or Action	Purpose
Step 2	UCS-A /org # **scope service-profile** *sp-name*	Enters service profile configuration mode for the specified service profile.
Step 3	UCS-A /org/service-profile # **scope vnic** *vnic-name*	Enters vNIC configuration mode for the specified vNIC.
Step 4	UCS-A /org/service-profile/vnic # **set cdn-name** *cdn-name*	Specifies the CDN name for the vNIC.
Step 5	UCS-A /org/service-profile/vnic* # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to configure CDN for a vNIC:

```
UCS-A # scope org
UCS-A /org # scope service-profile sp1
UCS-A /org/service-profile # scope vnic vn1
UCS-A /org/service-profile/vnic # set cdn-name eth0
UCS-A /org/service-profile/vnic* # commit-buffer
```

Displaying the CDN Name of a vNIC

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *server-num*	Enters server mode for the specified server.
Step 2	UCS-A /server # **scope adapter** *adapter-id*	Enters adapter mode for the specified adapter.
Step 3	UCS-A /server/adapter # **show host-eth-if** [**detail**] [**expand**]	Displays the details of the host Ethernet interface for the specified adapter.

Example

The following example shows how to display the CDN name of a vNIC:

```
UCS-A # scope server 3
UCS-A /server # scope adapter 1
UCS-A /server/adapter # show host-eth-if detail expand

Eth Interface:
    ID: 1
    Dynamic MAC Address: 00:25:B5:00:00:99
    Burned-In MAC Address: 00:00:00:00:00:00
    Model: UCSC-PCIE-CSC-02
    Name: vnic1
```

```
Cdn Name: cdn0
Admin State: Enabled
Operability: Operable
Order: 1
```

Displaying the Status of a vNIC

Procedure

	Command or Action	**Purpose**
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter / as the *org-name*.
Step 2	UCS-A /org # **scope service-profile** *sp-name*	Enters service profile configuration mode for the specified service profile.
Step 3	UCS-A /org/service-profile # **show vnic** [**detail**] [**expand**]	Displays the details of the vNIC in the specified service profile.

Example

This example shows how to display the status of a vNIC.

Note The CDN name that you configured for the vNIC appears as the **Admin CDN Name**. The CDN name that is finally applied to the BIOS policy appears as the **Oper CDN Name**.

```
UCS-A# scope org
UCS-A /org # scope service-profile sp1
UCS-A /org/service-profile # show vnic detail expand

vNIC:
    Name: vnic1
    Fabric ID: B
    Dynamic MAC Addr: 00:25:B5:17:47:01
    Desired Order: Unspecified
    Actual Order: 1
    Desired VCon Placement: 2
    Actual VCon Placement: 2
    Desired Host Port: ANY
    Actual Host Port: NONE
    Equipment: sys/chassis-2/blade-5/adaptor-3/host-eth-2
    Host Interface Ethernet MTU: 1500
    Ethernet Interface Admin CDN Name:cdn0
    Ethernet Interface Oper CDN Name:cdn0
    Template Name:
```

CIMC Security Policies

Cisco UCS Manager provides the following policies to increase security:

- KVM Management Policy

- IPMI Access Profile

IPMI Access Profile

This policy allows you to determine whether IPMI commands can be sent directly to the server, using the IP address. For example, you can send commands to retrieve sensor data from the CIMC. This policy defines the IPMI access, including a username and password that can be authenticated locally on the server, and whether the access is read-only or read-write.

You can also restrict remote connectivity by disabling or enabling IPMI over LAN in the IPMI access profile. IPMI over LAN is disabled by default on all unassociated servers, and on all servers without an IPMI access policy. When an IPMI access policy is created, the IPMI over LAN is set to enabled by default. If you do not change the value to disabled, IPMI over LAN will be enabled on all associated servers.

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

Creating an IPMI Access Profile

Before you begin

Obtain the following:

- Username with appropriate permissions that can be authenticated by the operating system of the server

- Password for the username

- Permissions associated with the username

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 ipmi-access-profile** *profile-name*	Creates the specified IPMI access profile and enters organization IPMI access profile mode.
Step 3	UCS-A /org/ipmi-access-profile # **set ipmi-over-lan** {**disable** | **enable**}	Determines whether remote connectivity can be established.

	Command or Action	Purpose
		Note IPMI over LAN is disabled by default on all unassociated servers, and on all servers without an IPMI access policy. When an IPMI access policy is created, the IPMI over LAN is set to enabled by default. If you do not change the value to disabled, IPMI over LAN will be enabled on all associated servers.
Step 4	UCS-A /org/ipmi-access-profile # **create ipmi-user** *ipmi-user-name*	Creates the specified endpoint user and enters organization IPMI access profile endpoint user mode. **Note** More than one endpoint user can be created within an IPMI access profile, with each endpoint user having its own password and privileges.
Step 5	UCS-A /org/ipmi-access-profile/ipmi-user # **set password**	Sets the password for the endpoint user. After entering the **set password** command, you are prompted to enter and confirm the password. For security purposes, the password that you type does not appear in the CLI.
Step 6	UCS-A /org/ipmi-access-profile/ipmi-user # **set privilege** {**admin**	**readonly**}	Specifies whether the endpoint user has administrative or read-only privileges.
Step 7	UCS-A /org/ipmi-access-profile/ipmi-user # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example creates an IPMI access profile named ReadOnly, creates an endpoint user named bob, sets the password and the privileges for bob, and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # create ipmi-access-profile ReadOnly
UCS-A /org/ipmi-access-profile* # create ipmi-user bob
UCS-A /org/ipmi-access-profile/ipmi-user* # set password
Enter a password:
Confirm the password:
UCS-A /org/ipmi-access-profile/ipmi-user* # set privilege readonly
UCS-A /org/ipmi-access-profile/ipmi-user* # commit-buffer
UCS-A /org/ipmi-access-profile/ipmi-user #
```

What to do next

Include the IPMI profile in a service profile and/or template.

Deleting an IPMI Access Profile

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 ipmi-access-profile** *profile-name*	Deletes the specified IPMI access profile.
Step 3	UCS-A /org # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes the IPMI access profile named ReadOnly and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # delete ipmi-access-profile ReadOnly
UCS-A /org* # commit-buffer
UCS-A /org #
```

Adding an Endpoint User to an IPMI Access Profile

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 ipmi-access-profile** *profile-name*	Enters organization IPMI access profile mode for the specified IPMI access profile.
Step 3	UCS-A /org/ipmi-access-profile # **create ipmi-user** *ipmi-user-name*	Creates the specified endpoint user and enters organization IPMI access profile endpoint user mode.
		Note More than one endpoint user can be created within an IPMI access profile, with each endpoint user having its own password and privileges.
Step 4	UCS-A /org/ipmi-access-profile/ipmi-user # **set password**	Sets the password for the endpoint user.

	Command or Action	Purpose
		After entering the **set password** command, you are prompted to enter and confirm the password. For security purposes, the password that you type does not appear in the CLI.
Step 5	UCS-A /org/ipmi-access-profile/ipmi-user # **set privilege** {**admin** | **readonly**}	Specifies whether the endpoint user has administrative or read-only privileges.
Step 6	UCS-A /org/ipmi-access-profile/ipmi-user # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example adds an endpoint user named alice to the IPMI access profile named ReadOnly and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # scope ipmi-access-profile ReadOnly
UCS-A /org/ipmi-access-profile* # create ipmi-user alice
UCS-A /org/ipmi-access-profile/ipmi-user* # set password
Enter a password:
Confirm the password:
UCS-A /org/ipmi-access-profile/ipmi-user* # set privilege readonly
UCS-A /org/ipmi-access-profile/ipmi-user* # commit-buffer
UCS-A /org/ipmi-access-profile/ipmi-user #
```

Deleting an Endpoint User from an IPMI Access Profile

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 ipmi-access-profile** *profile-name*	Enters organization IPMI access profile mode for the specified IPMI access profile.
Step 3	UCS-A /org/ipmi-access-profile # **delete ipmi-user** *epuser-name*	Deletes the specified endpoint user from the IPMI access profile.
Step 4	UCS-A /org/ipmi-access-profile # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes the endpoint user named alice from the IPMI access profile named ReadOnly and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # scope ipmi-access-profile ReadOnly
UCS-A /org/ipmi-access-profile # delete ipmi-user alice
UCS-A /org/ipmi-access-profile* # commit-buffer
UCS-A /org/ipmi-access-profile #
```

KVM Management Policy

The KVM Management policy allows you to determine whether vMedia encryption is enabled when you access a server via KVM.

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

Note After a KVM vMedia session is mapped, if you change the KVM management policy, it will result in a loss of the vMedia session. You must re-map the KVM vMedia session again.

Before Cisco UCS Manager Release 4.0(4), port 2068 was the only KVM port. Beginning with Release 4.0(4), you can configure a port number between 1024 and 49151 as the KVM port. Port 2068 continues to be the default KVM port number.

Configuring a KVM Management 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 kvm-mgmt-policy** *policy-name*	Creates the specified KVM management policy and enters organization KVM management policy mode.
Step 3	(Optional) UCS-A /org/kvm-mgmt-policy* # **set descr** *description*	Provides a description for the policy.
Step 4	UCS-A /org/kvm-mgmt-policy* # **set vmedia-encryption** {**disable** | **enable**}	Specifies vMedia encryption is enabled or disabled.
Step 5	UCS-A /org/kvm-mgmt-policy* # **set kvm-port** *port-num*	Specifies the KVM port. This can be a port number between 1024 and 49151. The default port number is 2068.
Step 6	UCS-A /org/kvm-mgmt-policy* # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create a KVM management policy named KVM_Policy1,
enable vMedia encryption, set the KVM port number, and commit the transaction:

```
UCS-A# scope org /
UCS-A /org # create kvm-mgmt-policy KVM_Policy1
UCS-A /org/kvm-mgmt-policy* # set vmedia-encryption enable
UCS-A /org/kvm-mgmt-policy* # set kvm-port 2078
UCS-A /org/kvm-mgmt-policy* # commit-buffer
UCS-A /org/kvm-mgmt-policy #
```

Modifying a KVM Management 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 kvm-mgmt-policy** *policy-name*	Enters organization KVM management policy mode for the specified KVM management policy.
Step 3	(Optional) UCS-A /org/kvm-mgmt-policy # **set descr** *description*	Provides a description for the policy.
Step 4	UCS-A /org/kvm-mgmt-policy* # **set vmedia-encryption** {**disable**	**enable**}	Specifies whether vMedia encryption is enabled or disabled.
Step 5	UCS-A /org/kvm-mgmt-policy* # **set kvm-port** *port-num*	Specifies the KVM port. This can be a port number between 1024 and 49151. The default port number is 2068.
Step 6	UCS-A /org/kvm-mgmt-policy* # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to modify a KVM management policy named KVM_Policy1,
and commit the transaction:

```
UCS-A# scope org /
UCS-A /org # scope kvm-mgmt-policy KVM_Policy1
UCS-A /org/kvm-mgmt-policy # show detail
Kvm Mgmt Policy:
    Name: KVM_Policy1
    Description:
    Vmedia Encryption: Disable
    Kvm Port: 2078
```

```
UCS-A /org/kvm-mgmt-policy* # set vmedia-encryption enable
UCS-A /org/kvm-mgmt-policy* # set kvm-port 2088
UCS-A /org/kvm-mgmt-policy* # commit-buffer
UCS-A /org/kvm-mgmt-policy # show detail
Kvm Mgmt Policy:
    Name: KVM_Policy1
    Description:
    Vmedia Encryption: Enable
    Kvm Port: 2088
```

Displaying KVM Management Policy Properties

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 kvm-mgmt-policy** *policy-name*	Enters organization KVM management policy mode for the specified KVM management policy.
Step 3	UCS-A /org/kvm-mgmt-policy # **show detail**	Displays details of the specified policy.

Example

The following example shows how to display details of a KVM management policy named KVM_Policy1:

```
UCS-A# scope org /
UCS-A /org # scope kvm-mgmt-policy KVM_Policy1
UCS-A /org/kvm-mgmt-policy # show detail
Kvm Mgmt Policy:
    Name: KVM_Policy1
    Description:
    Vmedia Encryption: Enable
    Kvm Port: 2088
UCS-A /org/kvm-mgmt-policy #
```

Graphics Card Policies

Cisco UCS Manager Release 3.1(3) extends graphics card support to include the ability to change the graphics card mode. You can now configure graphics card modes by using a graphics card policy. The graphics card modes are:

- Compute

- Graphics

• Any Configuration

Creating a Graphics Card Policy

Procedure

	Command or Action	**Purpose**
Step 1	UCS-A# **scope org**	Enters the root organization mode
Step 2	UCS-A /org # **create graphicscard-policy** *policy name*	Creates a graphics card policy with the specified policy name, and enters the graphics card policy mode.
Step 3	UCS-A /org/graphicscard-policy # **commit buffer**	Commits the transaction to the system configuration.

Example

This example shows how to create a graphics card policy:

```
UCS-A# scope org
UCS-A /org # create graphicscard-policy sample
UCS-A /org/graphicscard-policy* # commit-buffer
UCS-A /org/graphicscard-policy #
```

Setting Mode of the Graphics Card Policy

Procedure

	Command or Action	**Purpose**
Step 1	UCS-A# **scope org**	Enters the root organization mode
Step 2	UCS-A /org # **scope graphicscard-policy** *policy name*	Enters organization graphics card policy mode.
Step 3	UCS-A /org/graphicscard-policy # **set graphicscard-policy-mode** [**compute**] | [**graphic**] | [**any configuration**]	Specifies the mode for the graphics card policy.
Step 4	UCS-A /org/graphicscard-policy # **commit buffer**	Commits the transaction to the system configuration.

Example

This example shows how to set the mode of a graphics card policy:

```
UCS-A# scope org
UCS-A /org # scope graphicscard-policy sample
UCS-A /org/graphicscard-policy # set graphicscard-policy-mode graphics
UCS-A /org/graphicscard-policy* # commit-buffer
UCS-A /org/graphicscard-policy #
```

Displaying Details of the Graphics Card

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *server number*	Enters the chassis server mode for the specified server.
Step 2	UCS-A /chassis/server # **scope graphics-card** *identifier*	Enters the graphics card configuration mode for the specified server.
Step 3	UCS-A /chassis/server/graphics-card # **show graphics-card** [**detail**] [**expand**]	Displays the details of the graphics card for the specified server.

Example

This example shows how to display the details of a graphics card:

```
UCS-A# scope server 1/3
UCS-A /chassis/server # scope graphics-card 2
UCS-A /chassis/server/graphics-card* # show detail

Graphics Card:
    ID: 2
    Slot Id: 2
    Magma Expander Slot Id:
    Is Supported: Yes
    Vendor: Cisco Systems Inc
    Model: UCSB-GPU-M6
    Serial: FHH1924002B
    Mode: Graphics
    PID: UCSB-GPU-M6
    Firmware Version: 84.04.89.00.01|2754.0200.01.02
    Vendor Id: 0x10de
    Subvendor Id: 0x10de
    Device Id: 0x13f3
    Subdevice Id: 0x1143
UCS-A /chassis/server/graphics-card #
```

Displaying Details of the Graphics Card Policy

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org**	Enters the root organization mode

	Command or Action	Purpose
Step 2	UCS-A /org # **show graphicscard-policy detail**	Displays the details of the graphics card policy.

Example

This example shows how to display the details of a graphics card policy:

```
UCS-A# scope org
UCS-A /org # show graphicscard-policy detail

Graphics Card Policy:
    Name: sample
    Description:
    Graphics Card Policy Mode: Compute

    Name: default
    Description:
    Graphics Card Policy Mode: Any Configuration

    Name: graphics
    Description:
    Graphics Card Policy Mode: Graphics
UCS-A /org #
```

Configuring Local Disk Configuration Policies

Local Disk Configuration Policy

This policy configures any optional SAS local drives that have been installed on a server through the onboard RAID controller of the local drive. This policy enables you to set a local disk mode for all servers that are associated with a service profile that includes the local disk configuration policy.

The local disk modes include the following:

- **No Local Storage**—For a diskless server or a SAN only configuration. If you select this option, you cannot associate any service profile which uses this policy with a server that has a local disk.

- **RAID 0 Striped**—Data is striped across all disks in the array, providing fast throughput. There is no data redundancy, and all data is lost if any disk fails.

- **RAID 1 Mirrored**—Data is written to two disks, providing complete data redundancy if one disk fails. The maximum array size is equal to the available space on the smaller of the two drives.

- **Any Configuration**—For a server configuration that carries forward the local disk configuration without any changes.

- **No RAID**—For a server configuration that removes the RAID and leaves the disk MBR and payload unaltered.

  If you choose **No RAID** and you apply this policy to a server that already has an operating system with RAID storage configured, the system does not remove the disk contents. Therefore, there may be no visible differences on the server after you apply the **No RAID** mode. This can lead to a mismatch between

the RAID configuration in the policy and the actual disk configuration shown in the **Inventory** > **Storage** tab for the server.

To make sure that any previous RAID configuration information is removed from a disk, apply a scrub policy that removes all disk information after you apply the **No RAID** configuration mode.

• **RAID 5 Striped Parity**—Data is striped across all disks in the array. Part of the capacity of each disk stores parity information that can be used to reconstruct data if a disk fails. RAID 5 provides good data throughput for applications with high read request rates.

• **RAID 6 Striped Dual Parity**—Data is striped across all disks in the array and two parity disks are used to provide protection against the failure of up to two physical disks. In each row of data blocks, two sets of parity data are stored.

• **RAID 10 Mirrored and Striped**—RAID 10 uses mirrored pairs of disks to provide complete data redundancy and high throughput rates.

• **RAID 50 Striped Parity and Striped** —Data is striped across multiple striped parity disk sets to provide high throughput and multiple disk failure tolerance.

• **RAID 60 Striped Dual Parity and Striped** —Data is striped across multiple striped dual parity disk sets to provide high throughput and greater disk failure tolerance.

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

Note   For a Cisco UCS C-Series server integrated with Cisco UCS Manager, with an embedded on-board RAID controller, the local disk mode should always be **Any Configuration**, and the RAID must be configured directly on the controller.

Guidelines for all Local Disk Configuration Policies

Before you create a local disk configuration policy, consider the following guidelines:

No Mixed HDDs and SSDs

Do not include HDDs and SSDs in a single server or RAID configuration.

JBOD Mode Support

The B200 M3 server supports JBOD mode for local disks.

Note   Only B200 M3 and B22 M3 blade servers support the JBOD mode for local disks.

Guidelines for Local Disk Configuration Policies Configured for RAID

Configure RAID Settings in Local Disk Configuration Policy for Servers with MegaRAID Storage Controllers

If a blade server or integrated rack-mount server has a MegaRAID controller, you must configure RAID settings for the drives in the Local Disk Configuration policy included in the service profile for that server. You can do this either by configuring the local disk configuration policy in the service profile using one of the defined RAID modes for that server, or you can use the **Any Configuration** mode with the LSI Utilities toolset to create the RAID volumes.

If you do not configure your RAID LUNs before installing the OS, disk discovery failures might occur during the installation and you might see error messages such as "No Device Found."

Server May Not Boot After RAID1 Cluster Migration if Any Configuration Mode Specified in Service Profile

After RAID1 clusters are migrated, you need to associate a service profile with the server. If the local disk configuration policy in the service profile is configured with **Any Configuration** mode rather than **RAID1**, the RAID LUN remains in "inactive" state during and after association. As a result, the server cannot boot.

To avoid this issue, ensure that the service profile you associate with the server contains the identical local disk configuration policy as the original service profile before the migration and does not include the **Any Configuration** mode.

Do Not Use JBOD Mode on Servers with MegaRAID Storage Controllers

Do not configure or use JBOD mode or JBOD operations on any blade server or integrated rack-mount server with a MegaRAID storage controllers. JBOD mode and operations are not intended for nor are they fully functional on these servers.

Maximum of One RAID Volume and One RAID Controller in Integrated Rack-Mount Servers

A rack-mount server that has been integrated with Cisco UCS Manager can have a maximum of one RAID volume irrespective of how many hard drives are present on the server.

All the local hard drives in an integrated rack-mount server must be connected to only one RAID Controller. Integration with Cisco UCS Manager does not support the connection of local hard drives to multiple RAID Controllers in a single rack-mount server. We therefore recommend that you request a single RAID Controller configuration when you order rack-mount servers to be integrated with Cisco UCS Manager.

In addition, do not use third party tools to create multiple RAID LUNs on rack-mount servers. Cisco UCS Manager does not support that configuration.

Maximum of One RAID Volume and One RAID Controller in Blade Servers

A blade server can have a maximum of one RAID volume irrespective of how many drives are present in the server. All the local hard drives must be connected to only one RAID controller. For example, a B200 M3 server has an LSI controller and an Intel Patsburg controller, but only the LSI controller can be used as a RAID controller.

In addition, do not use third party tools to create multiple RAID LUNs on blade servers. Cisco UCS Manager does not support that configuration.

License Required for Certain RAID Configuration Options on Some Servers

Some Cisco UCS servers require a license for certain RAID configuration options. When Cisco UCS Manager associates a service profile containing this local disk policy with a server, Cisco UCS Manager verifies that the selected RAID option is properly licensed. If there are issues, Cisco UCS Manager displays a configuration error during the service profile association.

For RAID license information for a specific Cisco UCS server, see the *Hardware Installation Guide* for that server.

B420 M3 Server Does Not Support All Configuration Modes

The B420 M3 server does not support the following configuration modes in a local disk configuration policy:

- No RAID

- RAID 6 Striped Dual Parity

In addition, the B420 M3 does not support JBOD modes or operations.

Creating a Local Disk Configuration 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 local-disk-config-policy** *policy-name*	Creates a local disk configuration policy and enters local disk configuration policy mode.
Step 3	(Optional) UCS-A /org/local-disk-config-policy # **set descr** *description*	Provides a description for the local disk configuration policy.
Step 4	UCS-A /org/local-disk-config-policy # **set mode** {**any-configuration** | **no-local-storage** | **no-raid** | **raid-0-striped** | **raid-1-mirrored** | **raid-5-striped-parity** | **raid-6-striped-dual-parity** | **raid-10-mirrored-and-striped**}	Specifies the mode for the local disk configuration policy.
Step 5	UCS-A /org/local-disk-config-policy # **set protect** {**yes** | **no**}	Specifies whether the server retains the configuration in the local disk configuration policy even if the server is disassociated from the service profile.
		Caution Protect Configuration becomes non-functional if one or more disks in the server are defective or faulty.
		When a service profile is disassociated from a server and a new service profile associated, the

	Command or Action	Purpose
		setting for the Protect Configuration property in the new service profile takes precedence and overwrites the setting in the previous service profile.
		With this option enabled, the data on the disk is protected even after the server is decommissioned and then recommissioned. Hence, reassociation of the server with a service profile fails.
		Note If you disassociate the server from a service profile with this option enabled and then associate it with a new service profile that includes a local disk configuration policy with different properties, the server returns a configuration mismatch error and the association fails.
Step 6	UCS-A /org/local-disk-config-policy # **set flexflash-state** {**enable** | **disable**}	Specifies whether FlexFlash SD card support is enabled.
Step 7	UCS-A /org/local-disk-config-policy # **set flexflash-raid-reporting-state** {**enable** | **disable**}	Specifies whether FlexFlash RAID reporting support is enabled.
		Note If only one SD card is installed, the FlexFlash inventory displays the RAID State as Disabled and the RAID Health as NA.
Step 8	UCS-A /org/local-disk-config-policy # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example configures a local disk configuration policy and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # create local-disk-config-policy DiskPolicy7
UCS-A /org/local-disk-config-policy* # set mode raid-1-mirrored
UCS-A /org/local-disk-config-policy* # set protect yes
UCS-A /org/local-disk-config-policy* # commit-buffer
UCS-A /org/local-disk-config-policy #
```

Viewing a Local Disk Configuration 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 # **show local-disk-config-policy** *policy-name*	Displays the local disk policy. If you have not configured a local disk policy, the local disk configuration (created by the create local-disk-config command) displays.
		Displays the local disk definition (set by the create local-disk-config command). If the serial over LAN definition is not set, and if a policy is set (using the set local-disk-config-policy command), then the policy will be displayed.

Example

The following example shows how to display local disk policy information for a local disk configuration policy called DiskPolicy7:

```
UCS-A# scope org /
UCS-A /org # show local-disk-config-policy DiskPolicy7

Local Disk Config Policy:
Name: DiskPolicy7
Mode: Raid 1 Mirrored
Description:
Protect Configuration: Yes
```

Deleting a Local Disk Configuration 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 local-disk-config-policy** *policy-name*	Deletes the specified local disk configuration policy.
Step 3	UCS-A /org # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes the local disk configuration policy named DiskPolicy7 and commits
the transaction:

```
UCS-A# scope org /
UCS-A /org # delete local-disk-config-policy DiskPolicy7
UCS-A /org* # commit-buffer
UCS-A /org #
```

FlexFlash Support

Overview

Cisco UCS B-Series, C-Series M3 and higher, and S-Series M4 servers support internal Secure Digital (SD)
memory cards. The SD cards are hosted by the Cisco Flexible Flash storage controller, a PCI-based controller
which has two slots for SD cards. The cards contain a single partition called HV. When FlexFlash is enabled,
Cisco UCS Manager displays the HV partition as a USB drive to both the BIOS and the host operating system.

You can populate one or both the SD card slots that are provided. If two SD cards are populated, you can use
them in a mirrored mode.

Note Do not mix different capacity cards in the same server.

The SD cards can be used to store operating system boot images or other information. The following figure
illustrates the SD card slots.

Figure 1: SD Card Slots
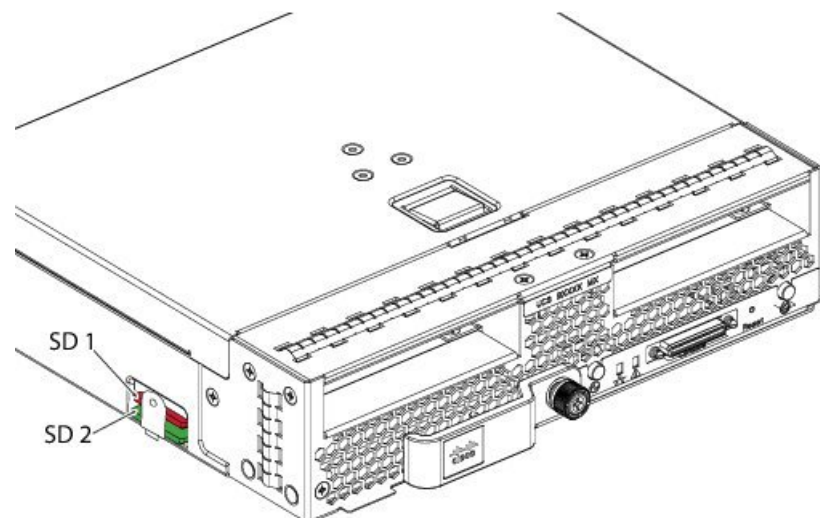
FlexFlash is disabled by default. You can enable FlexFlash in a local disk policy used in a service profile.
When FlexFlash is enabled in a local disk policy, and the server is capable of supporting SD cards, the FlexFlash

controller is enabled during service profile association. If a server is not capable of supporting SD cards or has an older CIMC version, a config failure message is displayed.

If you disable FlexFlash in a supported server, the Hypervisor or HV partition is immediately disconnected from the host. The FlexFlash controller will also be disabled as part of a related service profile disassociation.

The FlexFlash controller supports RAID-1 for dual SD cards. The FlexFlash scrub policy erases the HV partition in both cards, and brings the cards to a healthy RAID state.

You can configure new SD cards in a RAID pair and format them using one of the following methods:

- Format the SD cards.

- For an associated server, create a FlexFlash scrub policy and disassociate the service profile from the server. For an unassociated server, create a FlexFlash scrub policy and reacknowledge the server after modifying the default scrub policy.

  The *Scrub Policy Settings* section in the *Cisco UCS Manager Server Management Guide* provides more details about the usage of the scrub policy.

Note Disable the scrub policy as soon as the pairing is complete.

To boot from the HV partition, the SD card must be present in the boot policy used in the service profile.

FlexFlash Firmware Management

The FlexFlash controller firmware is bundled as part of the CIMC image. When you upgrade the CIMC, if a newer firmware version is available for the FlexFlash controller, the controller can no longer be managed, and the FlexFlash inventory displays the **Controller State** as **Waiting For User Action** and the **Controller Health** as **Old Firmware Running**. To upgrade the FlexFlash controller firmware, you need to perform a board controller update. For more information, see the appropriate *Cisco UCS B-Series Firmware Management Guide*, available at the following URL:
http://www.cisco.com/en/US/products/ps10281/products_installation_and_configuration_guides_list.html.

Limitations for the Cisco Flexible Flash Storage Controller:

- The Cisco Flexible Flash storage controller only supports 16 GB, 32 GB, and 64 GB SD cards.

  Note 16 GB and 32 GB cards are supported only on the B200-M3 blade servers, and the 64 GB SD cards are supported only on the B200-M4 and M5 blade servers.

- We do not recommend using an SD card from a rack server in a blade server, or using an SD card from a blade server in a rack server. Switching SD cards between server types might result in data loss from the SD card.

- Some Cisco UCS C-Series rack-mount servers have SD cards with four partitions: HV, HUU, SCU, and Drivers. Only the HV partition is visible in Cisco UCS Manager. You can migrate a four-partition SD card to a single HV partition card with a FlexFlash scrub policy.

- The FlexFlash controller does not support RAID-1 sync (mirror rebuild). If the SD cards are in a degraded RAID state, or if any metadata errors are reported by the controller, you must run the FlexFlash scrub

policy to pair the cards for RAID. For more information about the FlexFlash scrub policy, see Server-Related Policies. The following conditions might result in degraded RAID or metadata errors:

- Inserting a new or used SD card in one slot, when the server already has an SD card populated in the second slot.

- Inserting two SD cards from different servers.

- The server firmware version must be at 2.2(1a) or higher.

FlexFlash FX3S Support

Beginning with Release 2.2(3), Cisco UCS Manager allows additional FlexFlash support with the FX3S controller. The FX3S controller is present on the following servers:

- Cisco UCS B200 M4 and M5 blade server

- Cisco UCS C220 M4 and M5 rack server

- Cisco UCS C240 M4 and M5 rack server

- C480 M5 rack server

- C480 M5 ML blade server

- B480 M5 blade server

- Cisco UCS C125 M5 Server

FlexFlash operations with the FX3S control are similar to those with the Cisco Flexible Flash storage controller. FlexFlash is disabled by default, and is enabled using a local disk policy. You can also reset the controller, format the SD cards, and enable automatic synchronization of your paired SD cards.

The SD cards for the FX3S controller contain a single partition called Hypervisor.

Limitations for the Cisco FX3S Controller:

- The FX3S controller supports only 32 GB and 64 GB SD cards. 16 GB cards are not supported.

- The FX3S controller supports 128 GB cards on M5 blades and above.

- We do not recommend using an SD card from a rack server in a blade server, or using an SD card from a blade server in a rack server. Switching SD cards between server types might result in data loss from the SD card.

- The server firmware version must be at 2.2(3a) or higher.

Starting up Blade Servers with FlexFlash SD Cards

Use this procedure to start up blade servers using FlexFlash cards 16 GB and larger. This procedure requires that you know how to setup the blade server, software, and the associated infrastructure, and ensure that they are working. This Cisco UCS Manager controlled procedure is applicable to all blade servers, running any verison of firmware. This procedure does not apply to rack servers. Follow this procedure before you enable FlexFlash cards in a working environment.

⚠️

Caution If you use the following procedure with FlexFlash cards already in use, you will lose all data from the cards.

✎

Note This procedure does not cover FlexFlash card usage or other functions of the FlexFlash system.

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 local-disk-config-policy** *FlexFlash-name*	Enters the specified local disk configuration policy mode.
Step 3	UCS-A /org/local-disk-config-policy # **set flexflash-state** {**enable**	**disable**}	Specifies whether FlexFlash SD card support is enabled.
Step 4	UCS-A /org/local-disk-config-policy # **set flexflash-raid-reporting-state** {**enable**	**disable**}	Specifies whether FlexFlash RAID reporting support is enabled. **Note** If only one SD card is installed, the FlexFlash inventory displays the RAID State as Disabled and the RAID Health as NA.
Step 5	UCS-A /org/local-disk-config-policy # **commit-buffer**	Commits the transaction to the system.
Step 6	UCS-A/org/local-disk-config-policy # **show detail**	Displays the detailed FlexFlash controller properties. **/** as the *org-name* .
Step 7	UCS-A# **top**	
Step 8	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 9	UCS-A /org # **scope service-profile** *slot-3-name*	Enters organization service profile mode for the specified service. Slot 3 represents the particular blade's service profile.
Step 10	UCS-A /org/scope service-profile# **set local-disk-policy-state** *FlexFlash-name*	Associates the specified local disk policy with the service profile. FlexFlash represents the particular local disk policy.
Step 11	UCS-A /org/scope service-profile# **associate server** *1/1*	Associates the service profile with the specified blade server. 1 represents the blade number and the other represents the chassis number.

	Command or Action	Purpose
Step 12	UCS-A /org/local-disk-config-policy # **commit-buffer**	Commits the transaction to the system.
Step 13	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 14	UCS-A /org # **create scrub-policy** *Scrub-FF-name*	Creates a scrub policy with the specified policy name, and enters organization scrub policy mode.
Step 15	(Optional) UCS-A /org/scrub-policy # **set descr** *Scrub FlexFlash ONLY-name*	Provides a description for the scrub 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 16	UCS-A /org/scrub-policy # **set disk-scrub no**	Disables disk scrubbing on servers using this scrub policy.
Step 17	UCS-A /org/scrub-policy # **set bios-settings-scrub no**	Disables BIOS settings scrubbing on servers using this scrub policy.
Step 18	UCS-A /org/scrub-policy # **set flexflash-scrub yes**	Enables FlexFlash settings scrubbing on servers using this scrub policy.
Step 19	UCS-A /org/local-disk-config-policy # **commit-buffer**	Commits the transaction to the system.
Step 20	UCS-A# **top**	
Step 21	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 22	UCS-A /org # **scope service-profile** *slot-3-name*	Enters organization service profile mode for the specified service. Slot 3 represents the particular blade's service profile.
Step 23	UCS-A # **acknowledge server***1/3-name*	Acknowledges the specified blade server. 1 represents the chassis-num and 3 represents the server number.
Step 24	UCS-A /org/local-disk-config-policy # **commit-buffer**	Commits the transaction to the system. Wait for the blade server to complete committing the transaction.

	Command or Action	Purpose
Step 25	UCS-A # **acknowledge server***1/3-name*	Acknowledges the specified blade server. 1 represents the chassis-num and 3 represents the server number.
Step 26	UCS-A /org/local-disk-config-policy # **commit-buffer**	Commits the transaction to the system. The FlexFlash cards are now synced and ready to use.

Example

The following example shows the output from the console for starting up the FlexFlash card and creating the policies:

```
 #Creating the FlexFlash off policy

UCS-A# scope org
UCS-A /org # create local-disk-config-policy FF-off
UCS-A /org/local-disk-config-policy* # set flexflash-state disable
UCS-A/org/local-disk-config-policy* # commit-buffer
UCS-A/org/local-disk-config-policy # show detail

#Creating a Local Disk Configuration Policy

UCS-A# scope org
UCS-A /org # scope service-profile slot_4
UCS-A /org/service-profile # set local-disk-policy FF-off
UCS-A /org/service-profile* #

UCS-A/org/service-profile* # associate server 1/4
UCS-A/org/service-profile* # commit-buffer
UCS-A /org/service-profile # show detail

#Creating a FlexFlash On policy

UCS-A /org # top
UCS-A# scope org
UCS-A /org # create local-disk-config-policy FF-ON
UCS-A /org/local-disk-config-policy* # set flexflash-state enable
UCS-A /org/local-disk-config-policy* # set flexflash-raid-reporting-state  enable
UCS-A /org/local-disk-config-policy* # commit-buffer
UCS-A /org/local-disk-config-policy #
UCS-A /org/local-disk-config-policy #
UCS-A /org/local-disk-config-policy # show detail

UCS-A /org # top
UCS-A# scope org
UCS-A /org # scope service-profile slot_4
UCS-A /org/service-profile # set local-disk-policy FF-ON
UCS-A /org/service-profile* #

UCS-A /org/service-profile* # associate server 1/4
UCS-A /org/service-profile* # commit-buffer
UCS-A /org/service-profile # show detail
```

Enabling FlexFlash SD Card Support

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 local-disk-config-policy** *policy-name*	Enters the specified local disk configuration policy mode.
Step 3	UCS-A /org/local-disk-config-policy # **set flexflash-state** {**enable** | **disable**}	Specifies whether FlexFlash SD card support is enabled.
Step 4	UCS-A /org/local-disk-config-policy # **set flexflash-raid-reporting-state** {**enable** | **disable**}	Specifies whether FlexFlash RAID reporting support is enabled. **Note** If only one SD card is installed, the FlexFlash inventory displays the RAID State as Disabled and the RAID Health as NA.
Step 5	UCS-A /org/local-disk-config-policy # **commit-buffer**	Commits the transaction to the system.

Example

The following example shows how to enable FlexFlash SD card support and FlexFlash RAID reporting state on the local disk config policy default, and commits the transaction to the system:

```
UCS-A# scope org/
UCS-A /org # scope local-disk-config-policy default
UCS-A /org/local-disk-config-policy #set flexflash-state enable
UCS-A /org/local-disk-config-policy# #set flexflash-raid-reporting-state enable
UCS-A /org/local-disk-config-policy* # commit-buffer
UCS-A /org/local-disk-config-policy #
```

Disabling FlexFlash SD Card Support

This procedure describes how to disable FlexFlash support in a local disk policy.

Procedure

	Command or Action	Purpose
Step 1	UCS-A #**scope org** *{/ | org-name)*	Enters the organization mode for the organization you enter
Step 2	UCS-A /org # **scope local-disk-config-policy** *policy-name*	Enters the local disk policy configuration mode for the policy name you enter.

	Command or Action	Purpose
Step 3	UCS-A /org/local-disk-config-policy # **set flexflash-state** *disable*	Disables FlexFlash SD card support in the local disk policy.
Step 4	UCS-A /org/local-disk-config-policy # **commit-buffer**	Commits the transaction to the system.

Example

The following example scopes to the root organization, then to the default local disk configuration policy, sets the FlexFlash SD card support to disabled, and commits the transaction to the system:

```
UCS-A# scope org/
UCS-A /org # scope local-disk-config-policy default
UCS-A /org/local-disk-config-policy set flexflash-state disable
UCS-A /org/local-disk-config-policy #* commit-buffer
UCS-A /org/local-disk-config-policy #
```

Enabling Auto-Sync

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope chassis** *chassis-num*	Enters chassis mode for the specified chassis.
Step 2	UCS-A /chassis # **scope server** *server-num*	Enters server chassis mode.
Step 3	UCS-A /chassis/server # **scope flexflash-controller** *controller-id*	Enters flexflash controller server chassis mode.
Step 4	UCS-A /chassis/server/flexflash-controller # **pair** *primary_slot_number*	Resyncs the SD cards if they are out of sync, using the card in the selected slot number as the primary. This can be one of the following: • **1**—The SD card in slot 1 will be used as the primary. • **2**—The SD card in slot 2 will be used as the primary.
Step 5	UCS-A /chassis/server/flexflash-controller # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example resyncs the SD cards using the SD card in slot 2 as the primary:

```
UCS-A# scope chassis 1
UCS-A /chassis # scope server 1
UCS-A /chassis/server # scope flexflash-controller 1
UCS-A /chassis/server/flexflash-controller # pair 2
```

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

Formatting the FlexFlash Cards

Procedure

	Command or Action	**Purpose**
Step 1	UCS-A# **scope chassis** *chassis-num*	Enters chassis mode for the specified chassis.
Step 2	UCS-A /chassis # **scope server** *server-num*	Enters server chassis mode.
Step 3	UCS-A /chassis/server # **scope flexflash-controller** *controller-id*	Enters flexflash controller server chassis mode.
Step 4	UCS-A /chassis/server/flexflash-controller # **format**	Formats the SD cards.
Step 5	UCS-A /chassis/server/flexflash-controller # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to format the FlexFlash controller:

```
UCS-A# scope chassis 1
UCS-A /chassis # scope server 1
UCS-A /chassis/server # scope flexflash-controller 1
UCS-A /chassis/server/flexflash-controller # format
Warning: When commited, UCSM will format the SD Cards.
This will completely erase the data on the SD Cards!!

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

Resetting the FlexFlash Controller

Procedure

	Command or Action	**Purpose**
Step 1	UCS-A# **scope chassis** *chassis-num*	Enters chassis mode for the specified chassis.
Step 2	UCS-A /chassis # **scope server** *server-num*	Enters server chassis mode.
Step 3	UCS-A /chassis/server # **scope flexflash-controller** *controller-id*	Enters flexflash controller server chassis mode.
Step 4	UCS-A /chassis/server/flexflash-controller # **reset**	Resets the specified FlexFlash controller.
Step 5	UCS-A /chassis/server/flexflash-controller # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to reset the FlexFlash controller:

```
UCS-A# scope chassis 1
UCS-A /chassis # scope server 1
UCS-A /chassis/server # scope flexflash-controller 1
UCS-A /chassis/server/flexflash-controller # reset
Warning: When commited, UCSM will reset the FlexFlash Controller.
This will cause the host OS to lose connectivity to the SD Cards.

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

Viewing the FlexFlash Controller Status

Procedure

	Command or Action	**Purpose**
Step 1	UCS-A# **scope chassis** *chassis-num*	Enters chassis mode for the specified chassis.
Step 2	UCS-A /chassis # **scope server** *server-num*	Enters server chassis mode.
Step 3	UCS-A /chassis/server # **scope flexflash-controller** *controller-id*	Enters flexflash controller server chassis mode.
Step 4	UCS-A /chassis/server/flexflash-controller # **show detail expand**	Displays the detailed FlexFlash controller properties.

Example

The following example shows the status of the FlexFlash controller and SD cards:

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

FlexFlash Controller:
    ID: 1
    Type: SD
    FlexFlash Type: FX3S
    Vendor: Cypress
    Model: FX3S
    Serial: NA
    Firmware Version: 1.3.2 build 158
    Controller State: Connected Partition Over USB To Host
    Controller Health: Old Firmware Running
    RAID State: Enabled Paired
    RAID Health: OK
    Physical Drive Count: 2
    Virtual Drive Count: 1
    RAID Sync Support: Supported
    Operability: Operable
    Oper Qualifier Reason:
    Presence: Equipped
```

```
Current Task:

FlexFlash Card:
    Controller Index: 1
    Slot Number: 1
    Vendor: SE32G
    Model: SE32G
    HW Rev: 8.0
    Serial: 0xa2140794
    Manufacturer ID: 3
    OEM ID: SD
    Manufacturer Date: 2/14
    Size (MB): 30436
    Block Size: 512
    Card Type: FX3S configured
    Write Enabled: Not Write Protected
    Card Health: OK
    Card Mode: Secondary Active
    Operation State: Raid Partition
    Card State: Active
    Write IO Error Count: 0
    Read IO Error Count: 0
    Operability: Operable
    Oper Qualifier Reason:
    Presence: Equipped

    FlexFlash Card Drive:
        Name: Hypervisor
        Size (MB): 30432
        Removable: Yes
        Operability: Operable
        Operation State: Raid Partition

    Controller Index: 1
    Slot Number: 2
    Vendor: SE32G
    Model: SE32G
    HW Rev: 8.0
    Serial: 0xa2140742
    Manufacturer ID: 3
    OEM ID: SD
    Manufacturer Date: 2/14
    Size (MB): 30436
    Block Size: 512
    Card Type: FX3S configured
    Write Enabled: Not Write Protected
    Card Health: OK
    Card Mode: Primary
    Operation State: Raid Partition
    Card State: Active
    Write IO Error Count: 0
    Read IO Error Count: 0
    Operability: Operable
    Oper Qualifier Reason:
    Presence: Equipped

    FlexFlash Card Drive:
        Name: Hypervisor
        Size (MB): 30432
        Removable: Yes
        Operability: Operable
        Operation State: Raid Partition

Local Disk Config Definition:
```

```
        Mode: Any Configuration
        Description:
        Protect Configuration: Yes

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

Persistent Memory Modules

Cisco UCS Manager Release 4.0(4) introduces support for the Intel® Optane™ Data Center persistent memory modules on the UCS M5 servers that are based on the Second Generation Intel® Xeon® Scalable processors. These persistent memory modules can be used only with the Second Generation Intel® Xeon® Scalable processors.

Persistent memory modules are non-volatile memory modules that bring together the low latency of memory and the persistence of storage. Data stored in persistent memory modules can be accessed quickly compared to other storage devices, and is retained across power cycles.

For detailed information about configuring persistent memory modules, see the *Cisco UCS: Configuring and Managing Intel® Optane™ Data Center Persistent Memory Modules* Guide.

Scrub Policies

Scrub Policy Settings

This policy determines what happens to local data and to the BIOS settings on a server during the discovery process, when the server is re-acknowledged, or when the server is disassociated from a service profile.

> **Note** Local disk scrub policies only apply to hard drives that are managed by Cisco UCS Manager and do not apply to other devices such as USB drives.

Depending upon how you configure a scrub policy, the following can occur at those times:

Disk scrub

One of the following occurs to the data on any local drives on disassociation:

- If enabled, destroys all data on any local drives.

- If disabled (default), preserves all data on any local drives, including local storage configuration.

For a server associated with a service profile, disk scrub occurs during disassociation, based on the scrub policy used in the service profile. For an un-associated server, disk scrub occurs during the server discovery process, based on the default scrub policy.

Scrub policies are supported on all B-Series platforms and only on the following C-Series platforms:

- Cisco UCS C220 M4 Server

- Cisco UCS C240 M4 Server

- Cisco UCS C460 M4 Storage Server

- Cisco UCS C220 M5 Server

- Cisco UCS C240 M5 Server

- Cisco UCS C480 M5 Server

- Cisco UCS C480 M5 ML Server

- Cisco UCS S3260 M4 Storage Server—You can scrub only the boot drives and VDs created using the same drives.

- Cisco UCS S3260 M5 Storage Server—You can scrub only the boot drives and VDs created using the same drives.

Note You must re-acknowledge the server to see the changes related to LUN deletion if:

- you are scrubbing boot drives which have LUNs under the SAS controller in a set up with Cisco UCS S3260 M4 or Cisco UCS S3260 M5 Storage Server.

- you are scrubbing the LUNs on Cisco boot optimized M.2 RAID controller.

BIOS Settings Scrub

One of the following occurs to the BIOS settings when a service profile containing the scrub policy is disassociated from a server:

- If enabled, erases all BIOS settings for the server and resets them to the BIOS defaults for that server type and vendor.

- If disabled (default), preserves the existing BIOS settings on the server.

FlexFlash Scrub

FlexFlash Scrub enables you to pair new or degraded SD cards, resolve FlexFlash metadata configuration failures, and migrate older SD cards with 4 partitions to single partition SD cards. One of the following occurs to the SD card when a service profile containing the scrub policy is disassociated from a server, or when the server is reacknowledged:

- If enabled, the HV partition on the SD card is formatted using the PNUOS formatting utility. If two SD cards are present, the cards are RAID-1 paired, and the HV partitions in both cards are marked as valid. The card in slot 1 is marked as primary, and the card in slot 2 is marked as secondary.

- If disabled (default), preserves the existing SD card settings.

Note

- For a server associated with a service profile, FlexFlash scrub occurs during disassociation, based on the scrub policy used in the service profile. For an un-associated server, FlexFlash scrub occurs during the server discovery process, based on the default scrub policy.

- Because the FlexFlash scrub erases the HV partition on the SD cards, we recommend that you take a full backup of the SD card(s) using your preferred host operating system utilities before performing the FlexFlash scrub.

- To resolve metadata config failures in a service profile, you need to disable FlexFlash in the local disk config policy before you run the FlexFlash scrub, then enable FlexFlash after the server is reacknowledged.

- Disable the scrub policy as soon as the pairing is complete or the metadata failures are resolved.

- FlexFlash scrub is not supported for Cisco UCS S3260 Storage Server.

Persistent Memory Scrub

Persistent memory scrub enables you to preserve or remove the persistent memory configuration and data on a server.

- If enabled:

  - erases all the persistent memory data

  - resets the configuration to factory default

  - disables DIMM security

- If disabled (default), preserves the existing persistent memory configuration and data on the server. It does not change the DIMM lock state.

Creating a Scrub 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 scrub-policy** *policy-name*	Creates a scrub policy with the specified policy name, and enters organization scrub policy mode.
Step 3	(Optional) UCS-A /org/scrub-policy # **set descr** *description*	Provides a description for the scrub policy.

	Command or Action	**Purpose**
		Note If your description includes spaces, special characters, or punctuation, you must begin and end your description with quotation marks. The quotation marks will not appear in the description field of any **show** command output.
Step 4	UCS-A /org/scrub-policy # **set disk-scrub** {**no** | **yes**}	Disables or enables disk scrubbing on servers using this scrub policy as follows: • If enabled, destroys all data on any local drives. • If disabled (default), preserves all data on any local drives, including local storage configuration.
Step 5	UCS-A /org/scrub-policy # **set bios-settings-scrub** {**no** | **yes**}	Disables or enables BIOS settings scrubbing on servers using this scrub policy as follows: • If enabled, erases all BIOS settings for the server and resets them to the BIOS defaults for that server type and vendor. • If disabled (default), preserves the existing BIOS settings on the server.
Step 6	UCS-A /org/scrub-policy # **set flexflash-scrub** {**no** | **yes**}	Disables or enables flexflash scrubbing on servers using this scrub policy as follows: • If enabled, the HV partition on the SD card is formatted using the PNUOS formatting utility. If two SD cards are present, the cards are RAID-1 paired, and the HV partitions in both cards are marked as valid. The card in slot 1 is marked as primary, and the card in slot 2 is marked as secondary. • If disabled (default), preserves the existing SD card settings.
Step 7	UCS-A /org/scrub-policy # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example creates a scrub policy named ScrubPolicy2, enables disk scrubbing on servers using the scrub policy, and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # create scrub-policy ScrubPolicy2
UCS-A /org/scrub-policy* # set descr "Scrub disk but not BIOS."
UCS-A /org/scrub-policy* # set disk-scrub yes
UCS-A /org/scrub-policy* # set bios-settings-scrub no
UCS-A /org/scrub-policy* # set flexflash-scrub no
UCS-A /org/scrub-policy* # commit-buffer
UCS-A /org/scrub-policy #
```

Deleting a Scrub 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 scrub-policy** *policy-name*	Deletes the specified scrub policy.
Step 3	UCS-A /org # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes the scrub policy named ScrubPolicy2 and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # delete scrub-policy ScrubPolicy2
UCS-A /org* # commit-buffer
UCS-A /org #
```

Configuring DIMM Error Management

DIMM Correctable Error Handling

In Cisco UCS Manager, when a DIMM encounters a significant correctable error in a given predefined window, it is stated as degraded and considered as a non-functional device.

The DIMM correctable error handling feature enables you to reset all the correctable and uncorrectable memory errors on all the DIMMs in a server. When you reset the error configuration, the error count of a given DIMM is cleared, the status changes to operable, and it resets the sensor state of the given DIMM.

Resetting Memory Errors

Use this procedure to reset all correctable and uncorrectable memory errors encountered by Cisco UCS Manager and the baseboard management controller (BMC).

Procedure

	Command or Action	**Purpose**
Step 1	UCS-A# **scope chassis** *chassis-num*	Enters chassis mode for the specified chassis.
Step 2	UCS-A/chassis # **scope server** *server-num*	Enters server mode for the specified server.
Step 3	UCS-A/chassis/server # **reset-all-memory-errors**	Resets the correctable and uncorrectable errors on all the DIMMs in a server.
Step 4	UCS-A /chassis/server* # **commit-buffer**	Commits any pending transactions.

Example

This example shows how to reset the memory errors for the selected memory unit(s):

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

DIMM Blacklisting

In Cisco UCS Manager, the state of the Dual In-line Memory Module (DIMM) is based on SEL event records. When the BIOS encounters a noncorrectable memory error during memory test execution, the DIMM is marked as faulty. A faulty DIMM is a considered a nonfunctional device.

If you enable DIMM blacklisting, Cisco UCS Manager monitors the memory test execution messages and blacklists any DIMMs that encounter memory errors in the DIMM SPD data. To allow the host to map out any DIMMs that encounter uncorrectable ECC errors.

Enabling DIMM Blacklisting

The memory policy is a global policy that you can apply to existing servers on a Cisco UCS domain and also to the servers that are added after you set the memory policy.

Note
- This feature is supported both on the Cisco UCS B-Series blade servers and UCS C-Series rack servers.

Note Cisco UCS C-Series 420 M3 rack server do not support this feature.

- This global policy cannot be added to a service profile.

Before you begin
- For Cisco B-Series blade server, the server firmware must be at Release 2.2(1) or a later release.
- For Cisco C-Series rack server, the server firmware must be at Release 2.2(3).

• You must be logged in with one of the following privileges:

- • Admin

- • Server policy

- • Server profile server policy

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** /	Enters root organization mode.
Step 2	UCS-A /org # **scope memory-config-policy default**	Enters memory policy mode for the global memory policy.
Step 3	UCS-A /org/memory-config-policy # **set blacklisting enabled**	Enables DIMM blacklisting for the domain level policy and these changes applies to all the servers on that particular domain. **Note**　If the Cisco IMC of a server does not support DIMM blacklisting, an information level fault is generated.
Step 4	UCS-A /org/memory-config-policy* # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to enable DIMM blacklisting:

```
UCS-A# scope org /
UCS-A /chassis/org # scope memory-config-policy default
UCS-A /chassis/org/memory-config-policy # set blacklisting enabled
UCS-A /chassis/org/memory-config-policy* # commit-buffer
UCS-A /chassis/org/memory-config-policy #
UCS-A /chassis/org/memory-config-policy # show detail

Memory Config Policy:
    Blacklisting: enabled
```

Serial over LAN Policy

Serial over LAN Policy Overview

This policy sets the configuration for the serial over LAN connection for all servers associated with service profiles that use the policy. By default, the serial over LAN connection is disabled.

If you implement a serial over LAN policy, we recommend that you also create an IPMI profile.

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

Configuring a Serial over LAN 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 sol-policy** *policy-name*	Creates a serial over LAN policy and enters organization serial over LAN policy mode.
Step 3	(Optional) UCS-A /org/sol-policy # **set descr** *description*	Provides a description for the 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 4	UCS-A /org/sol-policy # **set speed** {**9600**| **19200** | **38400** | **57600** | **115200** }	Specifies the serial baud rate.
Step 5	UCS-A /org/sol-policy # {**disable** | **enable**}	Disables or enables the serial over LAN policy. By default, the serial over LAN policy is disabled; you must enable it before it can be applied.
Step 6	UCS-A /org/sol-policy # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example creates a serial over LAN policy named Sol115200, provides a description for the policy, sets the speed to 115200 baud, enables the policy, and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # create sol-policy Sol115200
UCS-A /org/sol-policy* # set descr "Sets serial over LAN policy to 115200 baud."
UCS-A /org/sol-policy* # set speed 115200
UCS-A /org/sol-policy* # enable
UCS-A /org/sol-policy* # commit-buffer
UCS-A /org/sol-policy #
```

Viewing a Serial over LAN 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 # **show sol-policy** *policy-name*	Displays the serial over LAN definition (set by the create sol-config command). If the serial over LAN definition is not set, and if a policy is set (using the set sol-policy command), then the policy will be displayed.

Example

The following example shows how to display serial over LAN information for a serial over LAN policy called Sol115200:

```
UCS-A# scope org /
UCS-A /org # show sol-policy Sol115200 detail

SOL Policy:
    Name: Sol115200
    SOL State: Enable
    Speed: 115200
    Description:
    Policy Owner: Local

UCS-A /org # show sol-policy Sol115200
SOL Policy:
    Name                      SOL State Speed
    --------------------------------------
    Sol115200                 Enable   115200
UCS-A /org #
```

Deleting a Serial over LAN 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 sol-policy** *policy-name*	Deletes the specified serial over LAN policy.
Step 3	UCS-A /org # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes the serial over LAN policy named Sol115200 and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # delete sol-policy Sol115200
UCS-A /org* # commit-buffer
UCS-A /org #
```

Server Autoconfiguration Policy

Server Autoconfiguration Policy Overview

Cisco UCS Manager uses this policy to determine how to configure a new server. If you create a server autoconfiguration policy, the following occurs when a new server starts:

1. The qualification in the server autoconfiguration policy is executed against the server.

2. If the server meets the required qualifications, the server is associated with a service profile created from the service profile template configured in the server autoconfiguration policy. The name of that service profile is based on the name given to the server by Cisco UCS Manager.

3. The service profile is assigned to the organization configured in the server autoconfiguration policy.

Configuring a Server Autoconfiguration 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 server-autoconfig-policy** *policy-name*	Creates a server autoconfiguration policy with the specified policy name, and enters organization server autoconfiguration policy mode.
Step 3	(Optional) UCS-A /org/server-autoconfig-policy # **set descr** *description*	Provides a description for the 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.

	Command or Action	Purpose
Step 4	(Optional) UCS-A /org/server-autoconfig-policy # **set destination org** *org-name*	Specifies the organization for which the server is to be used.
Step 5	(Optional) UCS-A /org/server-autoconfig-policy # **set qualifier** *server-qual-name*	Specifies server pool policy qualification to use for qualifying the server.
Step 6	(Optional) UCS-A /org/server-autoconfig-policy # **set template** *profile-name*	Specifies a service profile template to use for creating a service profile instance for the server.
Step 7	UCS-A /org/server-autoconfig-policy # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example creates a server autoconfiguration policy named AutoConfigFinance, provides a description for the policy, specifies finance as the destination organization, ServPoolQual22 as the server pool policy qualification, and ServTemp2 as the service profile template, and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # create server-autoconfig-policy AutoConfigFinance
UCS-A /org/server-autoconfig-policy* # set descr "Server Autoconfiguration Policy for
Finance"
UCS-A /org/server-autoconfig-policy* # set destination org finance
UCS-A /org/server-autoconfig-policy* # set qualifier ServPoolQual22
UCS-A /org/server-autoconfig-policy* # set template ServTemp2
UCS-A /org/server-autoconfig-policy* # commit-buffer
UCS-A /org/server-autoconfig-policy #
```

Deleting a Server Autoconfiguration 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 server-autoconfig-policy** *policy-name*	Deletes the specified server autoconfiguration policy.
Step 3	UCS-A /org # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes the server autoconfiguration policy named AutoConfigFinance and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # delete server-autoconfig-policy AutoConfigFinance
UCS-A /org* # commit-buffer
UCS-A /org #
```

Server Discovery Policy

Server Discovery Policy Overview

The server discovery policy determines how the UCS Manager reacts when you add a new UCS Blade Server and UCS Mini. If you create a server discovery policy, you can control whether the system conducts a deep discovery when a server is added to a chassis, or whether a user must first acknowledge the new server. By default, the system conducts a full discovery.

If you create a server discovery policy, the following occurs when a new server starts:

1. The server discovery policy qualification is executed against the server.

2. If the server meets the required qualifications, Cisco UCS Manager applies the following to the server:

   • Depending on the option that you select for the action, UCS Manager discovers the new server immediately, or waits for a user acknowledgment of the new server

   • Applies the scrub policy to the server

If automatic deep discovery is triggered by any hardware insertion, removal, or replacement, the following occurs:

1. The server is moved to a "pending activities" list.

2. A critical hardware mismatch fault is raised on the server, indicating that UCSM has detected a hardware mismatch.

3. User must explicitly acknowledge the server to trigger the deep discovery.

☞

Important In Cisco UCS Manager Release 2.2 (4), blade servers do not support drives with a block size of 4K, but rack-mount servers support such drives. If a drive with a block size of 4K is inserted into a blade server, discovery fails and the following error message appears:

```
Unable to get Scsi Device Information from the system
```

If this error occurs, do the following:

1. Remove the 4K drive.

2. Reacknowledge the server.

Reacknowledging the server causes the server to reboot and results in loss of service.

Configuring a Server Discovery Policy

Before you begin

If you plan to associate this policy with a server pool, create server pool policy qualifications.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org /**	Enters the root organization mode.
		Note Chassis discovery policies can only be accessed from the root organization.
Step 2	UCS-A /org # **create server-disc-policy** *policy-name*	Creates a server discovery policy with the specified policy name, and enters org server discovery policy mode.
Step 3	UCS-A /org/server-disc-policy # **set action** {**diag** | **immediate** | **user-acknowledged**}	Specifies when the system will attempt to discover new servers.
Step 4	(Optional) UCS-A /org/chassis-disc-policy # **set descr** *description*	Provides a description for the 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	(Optional) UCS-A /org/server-disc-policy # **set qualifier** *qualifier*	Uses the specified server pool policy qualifications to associates this policy with a server pool.
Step 6	UCS-A /org/server-disc-policy # **set scrub-policy**	Specifies the scrub policy to be used by this policy. The scrub policy defines whether the disk drive on a server should be scrubbed clean upon discovery.
Step 7	UCS-A /org/server-disc-policy # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example creates a server discovery policy named ServDiscPolExample, sets it to immediately discover new servers, provides a description for the policy, specifies the server pool policy qualifications and scrub policy, and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # create server-disc-policy ServDiscPolExample
UCS-A /org/server-disc-policy* # set action immediate
UCS-A /org/server-disc-policy* # set descr "This is an example server discovery policy."
UCS-A /org/server-disc-policy* # set qualifier ExampleQual
UCS-A /org/server-disc-policy* # set scrub-policy NoScrub
UCS-A /org/server-disc-policy # commit-buffer
```

What to do next

Include the server discovery policy in a service profile and/or template.

Deleting a Server Discovery 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 server-disc-policy** *policy-name*	Deletes the specified server discovery policy.
Step 3	UCS-A /org/server-disc-policy # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes the server discovery policy named ServDiscPolExample and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # delete server-disc-policy ServDiscPolExample
UCS-A /org* # commit-buffer
UCS-A /org #
```

Hardware Change Discovery Policy

The Hardware Change Discovery is a global policy used to set the how Cisco UCS Manager behaves when there is a hardware component change. The policy has two values:

- User Acknowledged: You must acknowledge the server to clear all the hardware inventory mismatch faults.

- Auto Acknowledged: Triggers automatic deep discovery when a hardware component change is detected.

When UCSM detects any change in the server hardware component, a critical hardware inventory mismatch fault is raised on the server. You must manually acknowledge the server to clear the fault and complete the hardware inventory. Once you have acknowledged the server, deep discovery and deep association is triggered.

For rack servers, you must decommision and recomission the server to clear the fault and complete the hardware inventory.

You cannot make changes to the policy if there is a hardware inventory mismatch fault.

Configuring a Hardware Change Discovery Policy

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org /**	Enters the root organization mode.
Step 2	UCS-A /org # **scope server-hwchange-disc-policy** *policy-name*	Enters org hardware change discovery policy mode.
Step 3	UCS-A /org/server-hwchange-disc-policy # **set action** {	**auto-acknowledged**	**user-acknowledged**}	Specifies when the system will attempt to discover new servers.
Step 4	UCS-A /org/server-hwchange-disc-policy # **set action auto-acknowledged**	Specifies the hardware change discovery policy to be used..
Step 5	UCS-A /org/server-hwchange-disc-policy # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example configures a hardware change discovery policy and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # scope server-hwchange-disc-policy
UCS-A /org/server-hwchange-disc-policy # set action
UCS-A /org/server-hwchange-disc-policy # set action auto-acknowledged
UCS-A /org/server-hwchange-disc-policy # commit-buffer
```

Viewing a Hardware Change Discovery Policy

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** .	Enters the root organization mode.
Step 2	UCS-A /org # **scope server-hwchange-disc-policy** *policy-name*	Enters org hardware change discovery policy mode.
Step 3	UCS-A /org/server-hwchange-disc-policy # **show detail**	Displays the Hardware Change Discovery Policy setting.

Example

The following example shows to to view the policy setting:

```
UCS-A# scope org /
UCS-A /org # scope server-hwchange-disc-policy
UCS-A /org/server-hwchange-disc-policy # show detail
Server Hardware Change Discovery Policy:
    Action: User Acknowledged
```

Server Inheritance Policies

Server Inheritance Policy Overview

This policy is invoked during the server discovery process to create a service profile for the server. All service profiles created from this policy use the values burned into the blade at manufacture. The policy performs the following:

- Analyzes the inventory of the server

- If configured, assigns the server to the selected organization

- Creates a service profile for the server with the identity burned into the server at manufacture

You cannot migrate a service profile created with this policy to another server.

Configuring a Server Inheritance Policy

A blade server or rack-mount server with a VIC adapter, such as the Cisco UCS M81KR Virtual Interface Card, does not have server identity values burned into the server hardware at manufacture. As a result, the identity of the adapter must be derived from default pools. If the default pools do not include sufficient entries for one to be assigned to the server, service profile association fails with a configuration error.

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 server-inherit-policy** *policy-name*	Creates a server inheritance policy with the specified policy name, and enters organization server inheritance policy mode.
Step 3	(Optional) UCS-A /org/server-inherit-policy # **set descr** *description*	Provides a description for the policy.

	Command or Action	Purpose
		Note If your description includes spaces, special characters, or punctuation, you must begin and end your description with quotation marks. The quotation marks will not appear in the description field of any **show** command output.
Step 4	(Optional) UCS-A /org/server-inherit-policy # **set destination org** *org-name*	Specifies the organization for which the server is to be used.
Step 5	(Optional) UCS-A /org/server-inherit-policy # **set qualifier** *server-qual-name*	Specifies server pool policy qualification to use for qualifying the server.
Step 6	UCS-A /org/server-inherit-policy # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example creates a server inheritance policy named InheritEngineering, provides a description for the policy, specifies engineering as the destination organization and ServPoolQual22 as the server pool policy qualification, and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # create server-inherit-policy InheritEngineering
UCS-A /org/server-inherit-policy* # set descr "Server Inheritance Policy for Engineering"
UCS-A /org/server-inherit-policy* # set destination org engineering
UCS-A /org/server-inherit-policy* # set qualifier ServPoolQual22
UCS-A /org/server-inherit-policy* # commit-buffer
UCS-A /org/server-inherit-policy #
```

Deleting a Server Inheritance 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 server-inherit-policy** *policy-name*	Deletes the specified server inheritance policy.
Step 3	UCS-A /org # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes the server inheritance policy named InheritEngineering and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # delete server-inherit-policy InheritEngineering
UCS-A /org* # commit-buffer
UCS-A /org #
```

Server Pool Policy

Server Pool Policy Overview

This policy is invoked during the server discovery process. It determines what happens if server pool policy qualifications match a server to the target pool specified in the policy.

If a server qualifies for more than one pool and those pools have server pool policies, the server is added to all those pools.

Configuring a Server Pool 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 pooling-policy** *policy-name*	Creates a server pool policy with the specified name, and enters organization pooling policy mode.
Step 3	(Optional) UCS-A /org/pooling-policy # **set descr** *description*	Provides a description for the server pool 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 4	UCS-A /org/pooling-policy # **set pool** *pool-distinguished-name*	Specifies the server pool to use with the server pool policy. You must specify the full distinguished name for the pool.

	Command or Action	Purpose
Step 5	UCS-A /org/pooling-policy # **set qualifier** *qualifier-name*	Specifies the server pool qualifier to use with the server pool policy.
Step 6	UCS-A /org/pooling-policy # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example creates a server pool policy named ServerPoolPolicy4 and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # create pooling-policy ServerPoolPolicy4
UCS-A /org/pooling-policy* # set pool org-root/compute-pool-pool3
UCS-A /org/pooling-policy* # set qualifier ServPoolQual8
UCS-A /org/pooling-policy* # commit-buffer
UCS-A /org/pooling-policy #
```

Deleting a Server Pool 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 pooling-policy** *policy-name*	Deletes the specified server pool policy.
Step 3	UCS-A /org # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes the server pool policy named ServerPoolPolicy4 and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # delete pooling-policy ServerPoolPolicy4
UCS-A /org/pooling-policy* # commit-buffer
UCS-A /org/pooling-policy #
```

Server Pool Policy Qualification

Server Pool Policy Qualification Overview

This policy qualifies servers based on the inventory of a server conducted during the discovery process. The qualifications are individual rules that you configure in the policy to determine whether a server meets the selection criteria. For example, you can create a rule that specifies the minimum memory capacity for servers in a data center pool.

Qualifications are used in other policies to place servers, not just by the server pool policies. For example, if a server meets the criteria in a qualification policy, it can be added to one or more server pools or have a service profile automatically associated with it.

You can use the server pool policy qualifications to qualify servers according to the following criteria:

- Adapter type
- Chassis location
- Memory type and configuration
- Power group
- CPU cores, type, and configuration
- Storage configuration and capacity
- Server model

Depending upon the implementation, you might need to configure several policies with server pool policy qualifications including the following:

- Autoconfiguration policy
- Chassis discovery policy
- Server discovery policy
- Server inheritance policy
- Server pool policy

Creating a Server Pool Policy Qualification

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 # **create server-qual** *server-qual-name*	Creates a server pool qualification with the specified name, and enters organization server qualification mode.
Step 3	UCS-A /org/server-qual # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example creates a server pool qualification named ServPoolQual22 and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # create server-qual ServPoolQual22
UCS-A /org/server-qual* # commit-buffer
UCS-A /org/server-qual #
```

What to do next

Configure one or more of the following server component qualifications:

- Adapter qualification
- Chassis qualification
- Memory qualification
- Power group qualification
- Processor qualification
- Storage qualification

Deleting a Server Pool Policy Qualification

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 server-qual** *server-qual-name*	Deletes the specified server pool qualification.
Step 3	UCS-A /org/server-qual # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes the server pool qualification named ServPoolQual22 and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # delete server-qual ServPoolQual22
UCS-A /org* # commit-buffer
UCS-A /org #
```

Creating an Adapter Qualification

Before you begin

Create a server pool policy qualification.

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 server-qual** *server-qual-name*	Enters organization server qualification mode for the specified server pool policy qualification.
Step 3	UCS-A /org/server-qual # **create adapter**	Creates an adapter qualification and enters organization server qualification adapter mode.
Step 4	UCS-A /org/server-qual/adapter # **create cap-qual** *adapter-type*	Creates an adapter capacity qualification for the specified adapter type and enters organization server qualification adapter capacity qualification mode. The *adapter-type* argument can be any of the following values: • **fcoe** —Fibre Channel over Ethernet • **non-virtualized-eth-if** —Non-virtualized Ethernet interface • **non-virtualized-fc-if** —Non-virtualized Fibre Channel interface • **path-encap-consolidated** —Path encapsulation consolidated • **path-encap-virtual** —Path encapsulation virtual • **protected-eth-if** —Protected Ethernet interface

	Command or Action	Purpose
		• **protected-fc-if**—Protected Fibre Channel interface
		• **protected-fcoe**—Protected Fibre Channel over Ethernet
		• **virtualized-eth-if**—Virtualized Ethernet interface
		• **virtualized-fc-if**—Virtualized Fibre Channel interface
		• **virtualized-scsi-if**—Virtualized SCSI interface
Step 5	UCS-A /org/server-qual/adapter/cap-qual # **set maximum** {*max-cap* | **unspecified**}	Specifies the maximum capacity for the selected adapter type.
Step 6	UCS-A /org/server-qual/adapter/cap-qual # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example creates and configures an adapter qualification for a non-virtualized Ethernet interface and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # scope server-qual ServPoolQual22
UCS-A /org/server-qual # create adapter
UCS-A /org/server-qual/adapter* # create cap-qual non-virtualized-eth-if
UCS-A /org/server-qual/adapter/cap-qual* # set maximum 2500000000
UCS-A /org/server-qual/adapter/cap-qual* # commit-buffer
UCS-A /org/server-qual/adapter/cap-qual #
```

Deleting an Adapter Qualification

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 server-qual** *server-qual-name*	Enters organization server qualification mode for the specified server pool policy qualification.
Step 3	UCS-A /org/server-qual # **delete adapter**	Deletes the adapter qualification from the server pool policy qualification.

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

Example

The following example deletes the adapter qualification from the server pool policy qualification named ServPoolQual22 and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # scope server-qual ServPoolQual22
UCS-A /org/server-qual # delete adapter
UCS-A /org/server-qual* # commit-buffer
UCS-A /org/server-qual #
```

Configuring a Chassis Qualification

Before you begin

Create a server pool policy qualification.

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 server-qual** *server-qual-name*	Enters organization server qualification mode for the specified server pool policy qualification.
Step 3	UCS-A /org/server-qual # **create chassis** *min-chassis-num max-chassis-num*	Creates a chassis qualification for the specified chassis range and enters organization server qualification chassis mode.
Step 4	UCS-A /org/server-qual/chassis # **create slot** *min-slot-num max-slot-num*	Creates a chassis slot qualification for the specified slot range and enters organization server qualification chassis slot mode.
Step 5	UCS-A /org/server-qual/chassis/slot # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example configures a chassis qualification for slots 1 to 4 on chassis 1 and 2 and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # scope server-qual ServPoolQual22
```

```
UCS-A /org/server-qual* # create chassis 1 2
UCS-A /org/server-qual/chassis* # create slot 1 4
UCS-A /org/server-qual/chassis/slot* # commit-buffer
UCS-A /org/server-qual/chassis/slot #
```

Deleting a Chassis Qualification

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 server-qual** *server-qual-name*	Enters organization server qualification mode for the specified server pool policy qualification.
Step 3	UCS-A /org/server-qual # **delete chassis** *min-chassis-num* *max-chassis-num*	Deletes the chassis qualification for the specified chassis range.
Step 4	UCS-A /org/server-qual # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes the chassis qualification for chassis 1 and 2 and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # scope server-qual ServPoolQual22
UCS-A /org/server-qual # delete chassis 1 2
UCS-A /org/server-qual* # commit-buffer
UCS-A /org/server-qual #
```

Creating a CPU Qualification

Before you begin

Create a server pool policy qualification.

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 server-qual** *server-qual-name*	Enters organization server qualification mode for the specified server pool policy qualification.
Step 3	UCS-A /org/server-qual # **create cpu**	Creates a CPU qualification and enters organization server qualification processor mode.
Step 4	UCS-A /org/server-qual/cpu # **set arch** {**any** | **dual-core-opteron** | **intel-p4-c** | **opteron** | **pentium-4** | **turion-64** | **xeon** | **xeon-mp**}	Specifies the processor architecture type.
Step 5	UCS-A /org/server-qual/cpu # **set maxcores** {*max-core-num* | **unspecified**}	Specifies the maximum number of processor cores.
Step 6	UCS-A /org/server-qual/cpu # **set mincores** {*min-core-num* | **unspecified**}	Specifies the minimum number of processor cores.
Step 7	UCS-A /org/server-qual/cpu # **set maxprocs** {*max-proc-num* | **unspecified**}	Specifies the maximum number of processors.
Step 8	UCS-A /org/server-qual/cpu # **set minprocs** {*min-proc-num* | **unspecified**}	Specifies the minimum number of processors.
Step 9	UCS-A /org/server-qual/cpu # **set maxthreads** {*max-thread-num* | **unspecified**}	Specifies the maximum number of threads.
Step 10	UCS-A /org/server-qual/cpu # **set minthreads** {*min-thread-num* | **unspecified**}	Specifies the minimum number of threads.
Step 11	UCS-A /org/server-qual/cpu # **set stepping** {*step-num* | **unspecified**}	Specifies the processor stepping number.
Step 12	UCS-A /org/server-qual/cpu # **set model-regex** *regex*	Specifies a regular expression that the processor name must match.
Step 13	UCS-A /org/server-qual/cpu # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example creates and configures a CPU qualification and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # scope server-qual ServPoolQual22
UCS-A /org/server-qual # create processor
UCS-A /org/server-qual/cpu* # set arch xeon
UCS-A /org/server-qual/cpu* # set maxcores 8
UCS-A /org/server-qual/cpu* # set mincores 4
UCS-A /org/server-qual/cpu* # set maxprocs 2
UCS-A /org/server-qual/cpu* # set minprocs 1
UCS-A /org/server-qual/cpu* # set maxthreads 16
UCS-A /org/server-qual/cpu* # set minthreads 8
UCS-A /org/server-qual/cpu* # set stepping 5
```

```
UCS-A /org/server-qual/cpu* # commit-buffer
UCS-A /org/server-qual/cpu #
```

Deleting a CPU Qualification

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 server-qual** *server-qual-name*	Enters organization server qualification mode for the specified server pool policy qualification.
Step 3	UCS-A /org/server-qual # **delete cpu**	Deletes the processor qualification.
Step 4	UCS-A /org/server-qual # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes the processor qualification and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # scope server-qual ServPoolQual22
UCS-A /org/server-qual # delete cpu
UCS-A /org/server-qual* # commit-buffer
UCS-A /org/server-qual #
```

Creating a Power Group Qualification

Before you begin

Create a server pool policy qualification.

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 server-qual** *server-qual-name*	Enters organization server qualification mode for the specified server pool policy qualification.
Step 3	UCS-A /org/server-qual # **create power-group** *power-group-name*	Creates a power group qualification for the specified power group name.

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

Example

The following example configures a power group qualification for a power group called powergroup1 and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # scope server-qual ServPoolQual22
UCS-A /org/server-qual # create power-group powergroup1
UCS-A /org/server-qual* # commit-buffer
UCS-A /org/server-qual #
```

Deleting a Power Group Qualification

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 server-qual** *server-qual-name*	Enters organization server qualification mode for the specified server pool policy qualification.
Step 3	UCS-A /org/server-qual # **delete power-group** *power-group-name*	Deletes the specified power group qualification.
Step 4	UCS-A /org/server-qual # **commit-buffer**	Commits the transaction to the system configuration.

Example

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

```
UCS-A# scope org /
UCS-A /org # scope server-qual ServPoolQual22
UCS-A /org/server-qual # delete power-group powergroup1
UCS-A /org/server-qual* # commit-buffer
UCS-A /org/server-qual #
```

Creating a Memory Qualification

Before you begin

Create a server pool policy qualification.

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 server-qual** *server-qual-name*	Enters organization server qualification mode for the specified server pool policy qualification.
Step 3	UCS-A /org/server-qual # **create memory**	Creates a memory qualification and enters organization server qualification memory mode.
Step 4	UCS-A /org/server-qual/memory # **set clock** {*clock-num* | **unspec**}	Specifies the memory clock speed.
Step 5	UCS-A /org/server-qual/memory # **set maxcap** {*max-cap-num* | **unspec**}	Specifies the maximum capacity of the memory array.
Step 6	UCS-A /org/server-qual/memory # **set mincap** {*min-cap-num* | **unspec**}	Specifies the minimum capacity of the memory array.
Step 7	UCS-A /org/server-qual/memory # **set speed** {*speed-num* | **unspec**}	Specifies the memory data rate.
Step 8	UCS-A /org/server-qual/memory # **set units** {*unit-num* | **unspec**}	Specifies the number of memory units (DRAM chips mounted to the memory board).
Step 9	UCS-A /org/server-qual/memory # **set width** {*width-num* | **unspec**}	Specifies the bit width of the data bus.
Step 10	UCS-A /org/server-qual/memory # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example creates and configures a memory qualification and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # scope server-qual ServPoolQual22
UCS-A /org/server-qual # create memory
UCS-A /org/server-qual/memory* # set clock 1067
UCS-A /org/server-qual/memory* # set maxcap 4096
UCS-A /org/server-qual/memory* # set mincap 2048
UCS-A /org/server-qual/memory* # set speed unspec
UCS-A /org/server-qual/memory* # set units 16
```

```
UCS-A /org/server-qual/memory* # set width 64
UCS-A /org/server-qual/memory* # commit-buffer
UCS-A /org/server-qual/memory #
```

Deleting a Memory Qualification

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 server-qual** *server-qual-name*	Enters organization server qualification mode for the specified server pool policy qualification.
Step 3	UCS-A /org/server-qual # **delete memory**	Deletes the memory qualification.
Step 4	UCS-A /org/server-qual # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes the memory qualification and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # scope server-qual ServPoolQual22
UCS-A /org/server-qual # delete memory
UCS-A /org/server-qual* # commit-buffer
UCS-A /org/server-qual #
```

Creating a Physical Qualification

Before you begin

Create a server pool policy qualification.

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 server-qual** *server-qual-name*	Enters organization server qualification mode for the specified server pool policy qualification.
Step 3	UCS-A /org/server-qual # **create physical-qual**	Creates a physical qualification and enters organization server qualification physical mode.

	Command or Action	**Purpose**
Step 4	UCS-A /org/server-qual/physical-qual # **set model-regex** *regex*	Specifies a regular expression that the model name must match.
Step 5	UCS-A /org/server-qual/physical-qual # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example creates and configures a physical qualification and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # scope server-qual ServPoolQual22
UCS-A /org/server-qual # create physical-qual
UCS-A /org/server-qual/physical-qual* # set model-regex
UCS-A /org/server-qual/physical-qual* # commit-buffer
UCS-A /org/server-qual/physical-qual #
```

Deleting a Physical Qualification

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 server-qual** *server-qual-name*	Enters organization server qualification mode for the specified server pool policy qualification.
Step 3	UCS-A /org/server-qual # **delete physical-qual**	Deletes the physical qualification.
Step 4	UCS-A /org/server-qual # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes a physical qualification and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # scope server-qual ServPoolQual22
UCS-A /org/server-qual # delete physical-qual
UCS-A /org/server-qual* # commit-buffer
UCS-A /org/server-qual #
```

Creating a Storage Qualification

Before you begin

Create a server pool policy qualification.

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 server-qual** *server-qual-name*	Enters organization server qualification mode for the specified server pool policy qualification.
Step 3	UCS-A /org/server-qual # **create storage**	Creates a storage qualification and enters organization server qualification storage mode.
Step 4	UCS-A /org/server-qual/storage # **set blocksize** {*block-size-num*	**unknown**}	Specifies the storage block size.
Step 5	UCS-A /org/server-qual/storage # **set diskless** {**no**	**unspecified**	**yes** }	Specifies whether the available storage must be diskless.
Step 6	UCS-A /org/server-qual/storage # **set disktype** {**hdd**	**ssd**	**unspecified**}	Specifies the type of disk that can be used. The options are: • **Unspecified**—Either disk type is acceptable. • **HDD**—The disk must be HDD. • **SSD**—The disk must be SSD (SATA or SAS).
Step 7	UCS-A /org/server-qual/storage # **set flexflash-num-cards** {*ff_card-num*	**unknown**}	Specifies the number of FlexFlash cards.
Step 8	UCS-A /org/server-qual/storage # **set maxcap** {*max-cap-num*	**unknown**}	Specifies the maximum capacity of the storage array.
Step 9	UCS-A /org/server-qual/storage # **set mincap** {*min-cap-num*	**unknown**}	Specifies the minimum capacity of the storage array.
Step 10	UCS-A /org/server-qual/storage # **set numberofblocks** {*block-num*	**unknown**}	Specifies the number of blocks.
Step 11	UCS-A /org/server-qual/storage # **set perdiskcap** {*disk-cap-num*	**unknown**}	Specifies the per-disk capacity.

	Command or Action	Purpose
Step 12	UCS-A /org/server-qual/storage # **set units** {*unit-num* | **unspecified**}	Specifies the number of storage units.
Step 13	UCS-A /org/server-qual/storage # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to create and configure a storage qualification and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # scope server-qual ServPoolQual22
UCS-A /org/server-qual # create storage
UCS-A /org/server-qual/storage* # set blocksize 512
UCS-A /org/server-qual/storage* # set disktype hdd
UCS-A /org/server-qual/storage* # set maxcap 420000
UCS-A /org/server-qual/storage* # set mincap 140000
UCS-A /org/server-qual/storage* # set numberofblocks 287277984
UCS-A /org/server-qual/storage* # set perdiskcap 140000
UCS-A /org/server-qual/storage* # set units 1
UCS-A /org/server-qual/storage* # set flexflash-num-cards 2
UCS-A /org/server-qual/storage* # commit-buffer
UCS-A /org/server-qual/storage #
```

Deleting a Storage Qualification

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 server-qual** *server-qual-name*	Enters organization server qualification mode for the specified server pool policy qualification.
Step 3	UCS-A /org/server-qual # **delete storage**	Deletes the storage qualification.
Step 4	UCS-A /org/server-qual/ # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes the storage qualification and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # scope server-qual ServPoolQual22
UCS-A /org/server-qual # delete storage
```

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

Configuring vNIC/vHBA Placement Policies

vNIC/vHBA Placement Policies

vNIC/vHBA placement policies are used to determine the following:

- How the virtual network interface connections (vCons) are mapped to the physical adapters on a server.

- What types of vNICs or vHBAs can be assigned to each vCon.

Each vNIC/vHBA placement policy contains four vCons that are virtual representations of the physical adapters. When a vNIC/vHBA placement policy is assigned to a service profile, and the service profile is associated with a server, the vCons in the vNIC/vHBA placement policy are assigned to the physical adapters and the vNICs and vHBAs are assigned to those vCons.

For blade or rack servers that contain one adapter, Cisco UCS assigns all vCons to that adapter. For servers that contain four adapters, Cisco UCS assigns vCon1 to Adapter1, vCon2 to Adapter2, vCon3 to Adapter3, and vCon4 to Adapter4.

For blade or rack servers that contain two or three adapters, Cisco UCS assigns the vCons based on the type of server and the selected virtual slot mapping scheme, which can be **Round Robin** or **Linear Ordered**. For details about the available mapping schemes, see vCon to Adapter Placement, on page 338.

After Cisco UCS assigns the vCons, it assigns the vNICs and vHBAs based on the **Selection Preference** for each vCon. This can be one of the following:

Note You can specify the PCI order for the vHBA; however, the desired order works within a class of devices, such as vNICs or vHBAs and not across them. Within an adapter, vNICs are always placed ahead of the vHBAs.

- **all**—All configured vNICs and vHBAs can be assigned to the vCon, whether they are explicitly assigned to it, unassigned, or dynamic. This is the default.

- **assigned-only**—vNICs and vHBAs must be explicitly assigned to the vCon. You can assign them explicitly through the service profile or the properties of the vNIC or vHBA.

- **exclude-dynamic**—Dynamic vNICs and vHBAs cannot be assigned to the vCon. The vCon can be used for all static vNICs and vHBAs, whether they are unassigned or explicitly assigned to it.

- **exclude-unassigned**—Unassigned vNICs and vHBAs cannot be assigned to the vCon. The vCon can be used for dynamic vNICs and vHBAs and for static vNICs and vHBAs that are explicitly assigned to it.

- **exclude-usnic**—Cisco usNICs cannot be assigned to the vCon. The vCon can be used for all other configured vNICs and vHBAs, whether they are explicitly assigned to it, unassigned, or dynamic.

> **Note**   An SRIOV usNIC that is explicitly assigned to a vCon set to **exclude-usnic** will
> remain assigned to that vCon.

If you do not include a vNIC/vHBA placement policy in the service profile, Cisco UCS Manager defaults to the **Round Robin** vCon mapping scheme and the **All** vNIC/vHBA selection preference, distributing the vNICs and vHBAs between the adapters based on the capabilities and relative capacities of each adapter.

vCon to Adapter Placement

Cisco UCS maps every vCon in a service profile to a physical adapter on the server. How that mapping occurs and how the vCons are assigned to a specific adapter in a server depends on the following:

- The type of server. N20-B6620-2 and N20-B6625-2 blade servers with two adapter cards use a different mapping scheme than other supported rack or blade servers.

- The number of adapters in the server.

- The setting of the virtual slot mapping scheme in the vNIC/vHBA placement policy, if applicable.

You must consider this placement when you configure the vNIC/vHBA selection preference to assign vNICs and vHBAs to vCons.

> **Note**   vCon to adapter placement is not dependent upon the PCIE slot number of the adapter. The adapter numbers
> used for the purpose of vCon placement are not the PCIE slot numbers of the adapters, but the ID assigned
> to them during server discovery.

vCon to Adapter Placement for N20-B6620-2 and N20-B6625-2 Blade Servers

In N20-B6620-2 and N20-B6625-2 blade servers, the two adapters are numbered left to right while vCons are numbered right to left. If one of these blade servers has a single adapter, Cisco UCS assigns all vCons to that adapter. If the server has two adapters, the vCon assignment depends upon the virtual slot mapping scheme:

- **round-robin**—Cisco UCS assigns vCon2 and vCon4 to Adapter1 and vCon1 and vCon3 to Adapter2. This is the default.

- **linear-ordered**—Cisco UCS assigns vCon3 and vCon4 to Adapter1 and vCon1 and vCon2 to Adapter2.

vCon to Adapter Placement for All Other Supported Servers

For all other servers supported by Cisco UCS in addition to the N20-B6620-2 and N20-B6625-2 blade servers, the vCon assignment depends on the number of adapters in the server and the virtual slot mapping scheme.

For blade or rack servers that contain one adapter, Cisco UCS assigns all vCons to that adapter. For servers that contain four adapters, Cisco UCS assigns vCon1 to Adapter1, vCon2 to Adapter2, vCon3 to Adapter3, and vCon4 to Adapter4.

For blade or rack servers that contain two or three adapters, Cisco UCS assigns the vCons based on the selected virtual slot mapping scheme: Round Robin or Linear Ordered.

Table 9: vCon to Adapter Placement Using the Round - Robin Mapping Scheme

Number of Adapters	vCon1 Assignment	vCon2 Assignment	vCon3 Assignment	vCon4 Assignment
1	Adapter1	Adapter1	Adapter1	Adapter1
2	Adapter1	Adapter2	Adapter1	Adapter2
3	Adapter1	Adapter2	Adapter3	Adapter2
4	Adapter1	Adapter2	Adapter3	Adapter4

Round Robin is the default mapping scheme.

Table 10: vCon to Adapter Placement Using the Linear Ordered Mapping Scheme

Number of Adapters	vCon1 Assignment	vCon2 Assignment	vCon3 Assignment	vCon4 Assignment
1	Adapter1	Adapter1	Adapter1	Adapter1
2	Adapter1	Adapter1	Adapter2	Adapter2
3	Adapter1	Adapter2	Adapter3	Adapter3
4	Adapter1	Adapter2	Adapter3	Adapter4

vNIC/vHBA to vCon Assignment

Cisco UCS Manager provides two options for assigning vNICs and vHBAs to vCons through the vNIC/vHBA placement policy: explicit assignment and implicit assignment.

Explicit Assignment of vNICs and vHBAs

With explicit assignment, you specify the vCon and, therefore, the adapter to which a vNIC or vHBA is assigned. Use this assignment option when you need to determine how the vNICs and vHBAs are distributed between the adapters on a server.

To configure a vCon and the associated vNICs and vHBAs for explicit assignment, do the following:

- Set the vCon configuration to any of the available options. You can configure the vCons through a vNIC/vHBA placement policy or in the service profile associated with the server. If a vCon is configured for **All**, you can still explicitly assign a vNIC or vHBA to that vCon.

- Assign the vNICs and vHBAs to a vCon. You can make this assignment through the virtual host interface placement properties of the vNIC or vHBA or in the service profile associated with the server.

If you attempt to assign a vNIC or vHBA to a vCon that is not configured for that type of vNIC or vHBA, Cisco UCS Manager displays a message advising you of the configuration error.

During service profile association, Cisco UCS Manager validates the configured placement of the vNICs and vHBAs against the number and capabilities of the physical adapters in the server before assigning the vNICs

and vHBAs according to the configuration in the policy. Load distribution is based upon the explicit assignments to the vCons and adapters configured in this policy.

If the adapters do not support the assignment of one or more vNICs or vHBAs, Cisco UCS Manager raises a fault against the service profile.

Note You can specify the PCI order for the vHBA; however, the desired order works within a class of devices, such as vNICs or vHBAs and not across them. Within an adapter, vNICs are always placed ahead of the vHBAs.

Implicit Assignment of vNICs and vHBAs

With implicit assignment, Cisco UCS Manager determines the vCon and, therefore, the adapter to which a vNIC or vHBA is assigned according to the capability of the adapters and their relative capacity. Use this assignment option if the adapter to which a vNIC or vHBA is assigned is not important to your system configuration.

To configure a vCon for implicit assignment, do the following:

- Set the vCon configuration to **All**, **Exclude Dynamic**, or **Exclude Unassigned**. You can configure the vCons through a vNIC/vHBA placement policy or in the service profile associated with the server.

- Do not set the vCon configuration to **Assigned Only**. Implicit assignment cannot be performed with this setting.

- Do not assign any vNICs or vHBAs to a vCon.

During service profile association, Cisco UCS Manager verifies the number and capabilities of the physical adapters in the server and assigns the vNICs and vHBAs accordingly. Load distribution is based upon the capabilities of the adapters, and placement of the vNICs and vHBAs is performed according to the actual order determined by the system. For example, if one adapter can accommodate more vNICs than another, that adapter is assigned more vNICs.

If the adapters cannot support the number of vNICs and vHBAs configured for that server, Cisco UCS Manager raises a fault against the service profile.

Implicit Assignment of vNICs in a Dual Adapter Environment

When you use implicit vNIC assignment for a dual slot server with an adapter card in each slot, Cisco UCS Manager typically assigns the vNICs/vHBAs as follows:

- If the server has the same adapter in both slots, Cisco UCS Manager assigns half the vNICs and half the vHBAs to each adapter.

- If the server has one non-VIC adapter and one VIC adapter, Cisco UCS Manager assigns two vNICs and two vHBAs to the non-VIC adapter and the remaining vNICs and vHBAs to the VIC adapter.

- If the server has two different VIC adapters, Cisco UCS Manager assigns the vNICs and vHBAs proportionally, based on the relative capabilities of the two adapters.

The following examples show how Cisco UCS Manager would typically assign the vNICs and vHBAs with different combinations of supported adapter cards:

- If you want to configure four vNICs and the server contains two Cisco UCS M51KR-B Broadcom BCM57711 adapters (with two vNICs each), Cisco UCS Manager assigns two vNICs to each adapter.

- If you want to configure 50 vNICs and the server contains a Cisco UCS CNA M72KR-E adapter (2 vNICs) and a Cisco UCS M81KR Virtual Interface Card adapter (128 vNICs), Cisco UCS Manager assigns two vNICs to the Cisco UCS CNA M72KR-E adapter and 48 vNICs to the Cisco UCS M81KR Virtual Interface Card adapter.

- If you want to configure 150 vNICs and the server contains a Cisco UCS M81KR Virtual Interface Card adapter (128 vNICs) and a Cisco UCS VIC-1240 Virtual Interface Card adapter (256 vNICs), Cisco UCS Manager assigns 50 vNICs to the Cisco UCS M81KR Virtual Interface Card adapter and 100 vNICs to the Cisco UCS VIC-1240 Virtual Interface Card adapter.

Note Exceptions to this implicit assignment occur if you configure the vNICs for fabric failover and if you configure dynamic vNICs for the server.

For a configuration that includes vNIC fabric failover where one adapter does not support vNIC failover, Cisco UCS Manager implicitly assigns all vNICs that have fabric failover enabled to the adapter that supports them. If the configuration includes only vNICs that are configured for fabric failover, no vNICs are implicitly assigned to the adapter that does not support them. If some vNICs are configured for fabric failover and some are not, Cisco UCS Manager assigns all failover vNICs to the adapter that supports them and a minimum of one nonfailover vNIC to the adapter that does not support them, according to the ratio above.

For a configuration that includes dynamic vNICs, the same implicit assignment would occur. Cisco UCS Manager assigns all dynamic vNICs to the adapter that supports them. However, with a combination of dynamic vNICs and static vNICs, at least one static vNIC is assigned to the adapter that does not support dynamic vNICs.

Configuring a vNIC/vHBA Placement 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 vcon-policy** *policy-name*	Creates the specified vNIC/vHBA placement profile and enters organization vcon policy mode.
Step 3	(Optional) UCS-A /org/vcon-policy # **set descr** *description*	Provides a description for the vNIC/vHBA Placement Profile.
		Enter up to 256 characters. You can use any characters or spaces except ` (accent mark), \ (backslash), ^ (carat), " (double quote), = (equal sign), > (greater than), < (less than), or ' (single quote).

	Command or Action	Purpose
		Note If your description includes spaces, special characters, or punctuation, you must begin and end your description with quotation marks. The quotation marks will not appear in the description field of any **show** command output.
Step 4	(Optional) UCS-A /org/vcon-policy # **set mapping-scheme** {**round-robin** | **linear-ordered**}	For blade or rack servers that contain one adapter, Cisco UCS assigns all vCons to that adapter. For servers that contain four adapters, Cisco UCS assigns vCon1 to Adapter1, vCon2 to Adapter2, vCon3 to Adapter3, and vCon4 to Adapter4.
		For blade or rack servers that contain two or three adapters, Cisco UCS assigns the vCons based on the selected virtual slot mapping scheme. This can be one of the following:
		• **round-robin**— In a server with two adapter cards, Cisco UCS assigns vCon1 and vCon3 to Adapter1, then assigns vCon2 and vCon4 to Adapter2.
		In a server with three adapter cards, Cisco UCS assigns vCon1 to Adapter1, vCon2 and vCon4 to Adapter2, and vCon3 to Adapter3.
		This is the default scheme.
		• **linear-ordered**— In a server with two adapter cards, Cisco UCS assigns vCon1 and vCon2 to Adapter1, then assigns vCon3 and vCon4 to Adapter2.
		In a server with three adapter cards, Cisco UCS assigns vCon1 to Adapter1 and vCon2 to Adapter2, then assigns vCon3 and vCon4 to Adapter3.
		In N20-B6620-2 and N20-B6625-2 blade servers, the two adapters are numbered left to right while vCons are numbered right to left. If one of these blade servers has a single adapter, Cisco UCS assigns all vCons to that adapter. If the server has two adapters, the vCon assignment depends upon the virtual slot mapping scheme:

	Command or Action	**Purpose**
		• **round-robin**—Cisco UCS assigns vCon2 and vCon4 to Adapter1 and vCon1 and vCon3 to Adapter2. This is the default.
		• **linear-ordered**—Cisco UCS assigns vCon3 and vCon4 to Adapter1 and vCon1 and vCon2 to Adapter2.
Step 5	UCS-A /org/vcon-policy # **set vcon** {**1** | **2** | **3** | **4**} **selection** {**all** | **assigned-only** | **exclude-dynamic** | **exclude-unassigned**}	Specifies the selection preference for the specified vCon. The options are:
		• **all**—All configured vNICs and vHBAs can be assigned to the vCon, whether they are explicitly assigned to it, unassigned, or dynamic. This is the default.
		• **assigned-only**—vNICs and vHBAs must be explicitly assigned to the vCon. You can assign them explicitly through the service profile or the properties of the vNIC or vHBA.
		• **exclude-dynamic**—Dynamic vNICs and vHBAs cannot be assigned to the vCon. The vCon can be used for all static vNICs and vHBAs, whether they are unassigned or explicitly assigned to it.
		• **exclude-unassigned**—Unassigned vNICs and vHBAs cannot be assigned to the vCon. The vCon can be used for dynamic vNICs and vHBAs and for static vNICs and vHBAs that are explicitly assigned to it.
		• **exclude-usnic**—Cisco usNICs cannot be assigned to the vCon. The vCon can be used for all other configured vNICs and vHBAs, whether they are explicitly assigned to it, unassigned, or dynamic.
		Note An SRIOV usNIC that is explicitly assigned to a vCon set to **exclude-usnic** will remain assigned to that vCon.
Step 6	UCS-A /org/vcon-policy # **commit-buffer**	Commits the transaction.

Example

The following example creates a vNIC/vHBA placement policy named Adapter1All, sets the vCon mapping scheme to Linear Ordered, specifies that only assigned vNICs and vHBAs can be placed on adapter 1, and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # create vcon-policy Adapter1
UCS-A /org/vcon-policy* # set descr "This profile places all vNICs and vHBAs on adapter 1."
UCS-A /org/vcon-policy* # set mapping-scheme linear-ordered
UCS-A /org/vcon-policy* # set vcon 1 selection assigned-only
UCS-A /org/vcon-policy* # commit-buffer
UCS-A /org/vcon-policy* #
UCS-A /org #
```

Deleting a vNIC/vHBA Placement 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 vcon-policy** *policy-name*	Deletes the specified vNIC/vHBA placement profile.
Step 3	UCS-A /org # **commit-buffer**	Commits the transaction.

Example

The following example deletes the vNIC/vHBA placement profile named Adapter1All and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # delete vcon-policy Adapter1All
UCS-A /org* # commit-buffer
UCS-A /org #
```

Explicitly Assigning a vNIC to a vCon

Before you begin

Configure the vCons through a vNIC/vHBA placement policy or in the service profile with one of the following values:

- **Assigned Only**

- **Exclude Dynamic**

- **Exclude Unassigned**

If a vCon is configured for **All**, you can explicitly assign a vNIC or vHBA to that vCon. However, there is less control with this configuration.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters organization mode for the organization which contains the service profile whose vNICs you want to explicitly assign to a vCon. 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.
Step 3	UCS-A /org/service-profile # **scope vnic** *vnic-name*	Enters organization service profile mode for the specified vnic.
Step 4	UCS-A /org/service-profile/vnic # **set vcon** {**1** | **2** | **3** | **4** | **any**}	Sets the virtual network interface connection (vCon) placement for the specified vNIC. Entering a value of any allows Cisco UCS Manager to determine the vCon to which the vNIC is assigned.
Step 5	UCS-A /org/service-profile/vnic # **set order** {*order-num* | **unspecified**}	Specifies the desired PCI order for the vNIC. Valid values include 0-128 and unspecified.
Step 6	UCS-A /org/service-profile/vnic # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example sets the vCon placement for a vNIC called vnic3 to 2, sets the desired order to 10, and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # scope service-profile accounting
UCS-A /org/service-profile # scope vnic vnic3
UCS-A /org/service-profile/vnic # set vcon 2
UCS-A /org/service-profile/vnic* # set order 10
UCS-A /org/service-profile/vnic* # commit-buffer
UCS-A /org/service-profile/vnic #
```

Explicitly Assigning a vHBA to a vCon

Before you begin

Configure the vCons through a vNIC/vHBA placement policy or in the service profile with one of the following values:

- **Assigned Only**

- **Exclude Dynamic**

- **Exclude Unassigned**

If a vCon is configured for **All**, you can explicitly assign a vNIC or vHBA to that vCon. However, there is less control with this configuration.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters organization mode for the organization which contains the service profile whose vHBAs you want to explicitly assign to a vCon. 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.
Step 3	UCS-A /org/service-profile # **scope vhba** *vhba-name*	Enters organization service profile mode for the specified vHBA.
Step 4	UCS-A /org/service-profile/vhba # **set vcon** {**1** | **2** | **3** | **4** | **any**}	Sets the virtual network interface connection (vCon) placement for the specified vHBA. Entering a value of any allows Cisco UCS Manager to determine the vCon to which the vHBA is assigned.
Step 5	UCS-A /org/service-profile/vhba # **set order** {*order-num* | **unspecified**}	Specifies the desired PCI order for the vHBA. Valid desired order number values include 0-128 and unspecified.
Step 6	UCS-A /org/service-profile/vhba # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example sets the vCon placement for a vHBA called vhba3 to 2, sets the desired order to 10, and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # scope service-profile accounting
UCS-A /org/service-profile # scope vhba vhba3
UCS-A /org/service-profile/vhba # set vcon 2
UCS-A /org/service-profile/vhba* # set order 10
UCS-A /org/service-profile/vhba* # commit-buffer
UCS-A /org/service-profile/vhba #
```

Placing Static vNICs Before Dynamic vNICs

For optimal performance, static vNICs and vHBAs should be placed before dynamic vNICs on the PCIe bus. Static vNICs refer to both static vNICs and vHBAs. Cisco UCS Manager Release 2.1 provides the following functionality regarding the order of static and dynamic vNICs:

- After upgrading to Cisco UCS Manager Release 2.1, if no change is made to existing service profiles (profiles that are defined in releases prior to Cisco UCS Manager Release 2.1), the vNIC order does not change.

- After an upgrade to Cisco UCS Manager Release 2.1, any vNIC-related change would reorder the vNIC map. As a result, all dynamic vNICs would be placed after the static vNICs.

- For newly created service profiles in Cisco UCS Manager Release 2.1, static vNICs are always ordered before dynamic vNICs.

- The above behavior is independent of the sequence of creating or deleting static or dynamic vNICs.

- For SRIOV-enabled service profiles, UCSM places the vNIC Physical Function(PF) before the corresponding Virtual Functions (VFs). This scheme guarantees that the VFs are placed close to the parent PF vNIC on the PCIe bus and BDFs are in successive incremental order for the VFs.

Example

Beginning Device Order in Cisco UCS Manager Release 2.0:

```
dyn-vNIC-1 1
dyn-vNIC-2 2
```

New Device Order in Cisco UCS Manager Release 2.0 (Add 2 static vNICs):

```
dyn-vNIC-1 1
dyn-vNIC-2 2
eth-vNIC-1 3
eth-vNIC-2 4
```

After upgrading to Cisco UCS Manager Release 2.1, (Before any vNIC-related change is made to the service profile.)

```
dyn-vNIC-1 1
dyn-vNIC-2 2
eth-vNIC-1 3
eth-vNIC-2 4
```

New Device Order in Cisco UCS Manager Release 2.1 (Add 2 dynamic vNICs by changing the policy count from 2 to 4.)

```
dyn-vNIC-1 3
dyn-vNIC-2 4
eth-vNIC-1 1
eth-vNIC-2 2
dyn-vNIC-3 5
dyn-vNIC-4 6
```

Dynamic vNICs as Multifunction PCIe Devices

Cisco UCS Manager Version 2.1 provisions static vNICs as 0-function devices (new BUS for every static vNIC). Multifunction dynamic vNICs are placed from the new Bus-slot after the last static vNIC/vHBA.

Note Cisco UCS Manager Version 2.1 supports the new StaticZero mode.

Table 11: Version Compatibility

Cisco UCS Manager		
Version 1.4 **Scheme: ZeroFunction**	**Version 2.0** **Scheme: ZeroFunction / MultiFunction**	**Version 2.1** **Scheme: ZeroFunction / MultiFunction / StaticZero**
Static and Dynamic vNICs are all on Bus [0-57], Function [0] < ZeroFunction Mode >	Static vNICs and Dynamic vNICs are on Bus [0-57], Function [0-7]. Bus 0, Function 0 Bus 0, Function 7 Bus 1, Function 0 < MultiFunction Mode >	Static vNICs or PFs will be on Bus [0-57], Function [0]. SRIOV: Corresponding VFs will be on the same Bus and Functions [1-255] No-SRIOV: Dynamic vNICs are on Bus [0-57], Function [0-7] < StaticZero Mode >
	Upgrade from Balboa will not renumber BDFs (remain in ZeroFunction mode) until Bus <= 57. Once devices exceed 58, switch to MultiFunction mode.	Upgrade from Balboa will not renumber BDFs (remain in ZeroFunction mode) until Bus <=57. Once devices exceed 58 or Platform specific maximum PCIe Bus number or change to SRIOV configuration, switch to StaticZero mode.
		Upgrade from Cisco UCS Manager Version 2.0 will not renumber BDFs (remain in ZeroFunction / MultiFunction mode). Once devices exceed 58 or Platform specific maximum PCIe Bus number OR Change to SRIOV configuration, switch to StaticZero mode.

vNIC/vHBA Host Port Placement

After a vNIC/vHBA is assigned to a vCON, it can be placed on one of the host ports of specific adapters. You can either explicitly specify the host port for placement, or allow Cisco UCS Manager to automatically assign vNICs/vHBAs to host ports.

Note You can perform vNIC/vHBA host port placement on servers that support Cisco UCS VIC 1340 and VIC 1380 adapters.

The host port placement of the vNIC/vHBA determines the order of the vNIC/vHBA on the adapter. The vNICs/vHBAs placed on the first host port will be enumerated first, followed by the vNICs/vHBAs on the second host port.

Configuring Host Port Placement

You can configure host port placement for vNICs on servers that support Cisco UCS VIC 1340 and VIC 1380 adapters.

Procedure

	Command or Action	**Purpose**
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter / as the *org-name*.
Step 2	UCS-A /org # **scope service-profile** *profile-name*	Enters service profile organization mode for the service profile.
Step 3	UCS-A /org/service-profile # **scope vnic** *vnic-name*	Enters organization service profile mode for the specified vNIC.
Step 4	UCS-A /org/service-profile/vnic # **set host-port** {1 | 2 | any}	Sets the host port for the specified vNIC. Entering a value of **any** allows Cisco UCS Manager to determine the host port to which the vNIC is assigned. If you set the host port for a vNIC on an adapter that does not support host port placement, the **Actual Host Port** parameter displays **None**.
Step 5	UCS-A /org/service-profile/vnic* # **commit-buffer**	Commits the transaction to the system configuration.
Step 6	UCS-A /org/service-profile/vnic # **show detail**	Displays details about the specified vNIC.

Example

The following example places a vNIC called vnic3 to host port 2, commits the transaction, and displays the host port information:

```
UCS-A# scope org
UCS-A /org # scope service-profile SP-2
UCS-A /org/service-profile # scope vnic vnic3
UCS-A /org/service-profile/vnic # set host-port 2
UCS-A /org/service-profile/vnic* # commit-buffer
UCS-A /org/service-profile/vnic # show detail
vNIC:
    Name: vnic3
    Fabric ID: A
    Dynamic MAC Addr: 00:25:B5:13:13:11
    Desired Order: 2
    Actual Order: 3
    Desired VCon Placement: 1
```

```
        Actual VCon Placement: 1
        Desired Host Port: 2
        Actual Host Port: 2
...

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

CIMC Mounted vMedia

Using Scriptable vMedia

Cisco UCS Manager allows provisioning of vMedia devices iso images for remote UCS servers. Using Scriptable vMedia, you can programmatically mount an IMG or an ISO image on a remote server. CIMC mounted vMedia provide communications between other mounted media inside your datacenter with no additional requirements media connection. Scriptable vMedia allows you to control virtual media devices without using a browser to manually map each UCS server individually.

Scriptable vMedia supports multiple share types including NFS, CIFS, HTTP, and HTTPS shares. **Scriptable vMedia** is enabled through BIOS configuration and configured through a Web GUI and CLI interface.

Cisco UCS Manager Scriptable vMedia supports the following functionality:

- Booting from a specific vMedia device

- Copying files from a mounted share to a local disk

- Installation and updating OS drivers

Note Cisco UCS Manager support for Scriptable vMedia is applicable for CIMC mapped devices only. Existing KVM based vMedia devices are not supported.

vMedia mount fails when the following conditions are met:

1. The remote vMedia image filename in the vMedia policy is set to **Service-Profile-Name**.

2. The service profile is renamed.

This is because the change in the name of the service profile does not change the remote vMedia image filename in the vMedia policy. The image filename still points to the older image on the remote device, which cannot be found.

Creating a CIMC vMedia Policy

Before you begin

Make sure that you have access to the following:

- Remote vMedia Server

- vMedia Devices

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 vmedia-policy** *policy-name*	Creates a vMedia policy with the specified policy name. This name can be between 1 and 16 alphanumeric characters. You cannot use spaces or any special characters other than - (hyphen), _ (underscore), : (colon), and . (period), and you cannot change this name after the object is saved.
Step 3	UCS-A /org/vmedia-policy* # **create vmedia-mapping** *mapping -name*	Creates a vMedia policy sub-directory with the specified mapping name.
Step 4	(Optional) UCS-A /org/vmedia-policy/vmedia-mapping # **set descr** *description*	Provides a description for the vMedia 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/vmedia-policy/vmedia-mapping* # **set device type** *device-type*	Specifies the remote vMedia image type you wish to mount. Options are: • **CDD** - Scriptable vMedia CD. • **HDD** - Scriptable vMedia HDD.
Step 6	UCS-A /org/vmedia-policy/vmedia-mapping* # **set image-file** *image-file-name*	Specifies the type of remote vMedia image file name. Enter the full path to the backup configuration file. This field can contain the filename [with the file extension] only. **Note** Ensure that the full path to the file begins with "/" after the share name.
Step 7	UCS-A /org/vmedia-policy/vmedia-mapping* # **set image-path** *image-path*	Specifies the remote vMedia image path. Enter the full path to the remote vMedia configuration file.
Step 8	UCS-A /org/vmedia-policy/vmedia-mapping* # **set mount-protocol** *mount-protocol*	Specifies the remote vMedia mount protocol. Options are: • **CIFS** • **NFS**

	Command or Action	**Purpose**
		• **HTTP** • **HTTPS**
Step 9	UCS-A /org/vmedia-policy/vmedia-mapping* # **set password**	Specifies the remote vMedia image password.
Step 10	UCS-A /org/vmedia-policy/vmedia-mapping* # **set remote-ip** *remote-ip*	Specifies the remote vMedia image IP address.
Step 11	UCS-A /org/vmedia-policy/vmedia-mapping* # **set user-id** *user-id*	Specifies the user id for mounting the vMedia device. Enter the username that Cisco UCS Manager should use to log in to the remote server. This field does not apply if the protocol is NFS. This field is optional if the protocol is HTTP.
Step 12	UCS-A /org/vmedia-policy/vmedia-mapping* # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example creates a vMedia policy named vMediaPolicy2, selects remote vMedia device type, mount protocol, image location, and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # create vmedia-policy vmediapolicy2
UCS-A /org/vmedia-policy* # create vmedia-mapping map1
UCS-A /org/vmedia-policy/vmedia-mapping* # set descr vmedia-map
UCS-A /org/vmedia-policy/vmedia-mapping* # set device-type cdd
UCS-A /org/vmedia-policy/vmedia-mapping* # set image-file-name win2011.iso
UCS-A /org/vmedia-policy/vmedia-mapping* # set image-path cifs
UCS-A /org/vmedia-policy/vmedia-mapping* # set image-variable-name service-profile-name
UCS-A /org/vmedia-policy/vmedia-mapping* # set mount-protocol cifs
UCS-A /org/vmedia-policy/vmedia-mapping* # set auth-option default
UCS-A /org/vmedia-policy/vmedia-mapping* # set password Password:
UCS-A /org/vmedia-policy/vmedia-mapping* # set remote-ip 172.41.1.158
UCS-A /org/vmedia-policy/vmedia-mapping* # set user-id Adminstrator
UCS-A /org/vmedia-policy/vmedia-mapping* # commit-buffer
```

Note   When vMedia policy is created the **Retry on Mount Fail** option is set to **Yes**. The following example changes the **Retry on Mount Fail** option to **No** .

```
UCS-A# scope org /
UCS-A /org # create vmedia-policy vmediapolicy2
UCS-A /org/vmedia-policy* # set retry-on-mount-fail No
UCS-A /org/vmedia-policy* # commit-buffer
```

Warning   When you set the **Retry on Mount Fail** option to **No**, a warning message appears stating: **This will disable automatic retry of mount in case of any vMedia mount failure**.

CHAPTER 12

Firmware Upgrades

Firmware Upgrades

Beginning with Cisco UCS Manager Release 4.0(1a), Cisco is releasing unified Cisco UCS Manager software and firmware upgrades for the following platforms with every release of Cisco UCS Manager:

  • Cisco UCS 6454 Fabric Interconnect with Cisco UCS B-Series, C-Series, and S-Series Servers

Beginning with Cisco UCS Manager Release 3.1, Cisco is releasing unified Cisco UCS Manager software and firmware upgrades for each of the following platforms with every release of Cisco UCS Manager:

  • Cisco UCS 6300 Series Fabric Interconnect with Cisco UCS B-Series, C-Series, and S-Series Servers

  • Cisco UCS 6200 Series Fabric Interconnect with Cisco UCS B-Series, C-Series, and S-Series

  • Cisco UCS 6324 Fabric Interconnect with Cisco UCS B-Series Servers and C-Series Servers, which is also known as UCS Mini

You can upgrade the firmware through Auto Install, packages in service profiles, using the firmware automatic synchronization server policy, and directly at endpoints. For more information on guidelines and installing firmware, see the *Cisco UCS Firmware Management Guide*.

Diagnostics Configuration

Overview of Cisco UCS Manager Diagnostics

The Cisco UCS Manager diagnostics tool enables you to verify the health of the hardware components on your servers. The diagnostics tool provides a variety of tests to exercise and stress the various hardware subsystems on the servers, such as memory and CPU. You can use the tool to run a sanity check on the state of your servers after you fix or replace a hardware component. You can also use this tool to run comprehensive burn-in tests before you deploy a new server in your production environment.

When a system is new, a default diagnostics policy is created in org scope. This default policy is named default and it cannot be deleted. The user will receive an error message if they try to delete it. The default diagnostic policy is the preferred way to execute the same set of tests across all servers. Any diagnostic policy, including the default can be customized.

The default policy only has one memory test. The default parameters of the memory test can be modified. In addition, the memory test within the default diagnostics policy can be deleted. If it does not have a memory test, the diagnostic policy will not run.

Creating a Diagnostics Policy

Before you begin

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

Procedure

	Command or Action	**Purpose**
Step 1	UCS-A # **scope org**	Enters the organization configuration mode.
Step 2	UCS-A /org # **create diag-policy** *<diag-policy>*	Creates a diagnostic policy. **Note** The diagnostic policy name can contain up to 16 characters.
Step 3	UCS-A /org/diag-policy # **commit buffer**	

Example

The following example shows how to create and set description for a diagnostic policy:

```
UCS-A# scope org
UCS-A /org # create diag-policy new-policy
UCS-A /org/diag-policy* # commit-buffer
```

Configuring a Memory Test for a Diagnostics Policy

Before you begin

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

Procedure

	Command or Action	**Purpose**
Step 1	UCS-A # **scope org**	Enters the organization configuration mode.
Step 2	UCS-A /org # **create diag-policy-name** *<diag-polic-name>*	Creates a custom diagnostic policy. The diagnostic policy can contain up to 16 characters.
Step 3	UCS-A /org/diag-policy-name* # **commit buffer**	Commits the transaction to the system configuration.
Step 4	UCS-A /org/diag-policy # **create memory-test** *<memory-test <test order>*	Creates a custom memory test for the diagnostic policy. The memory test ID can range from 1 to 64. The memory test has the following values which the user can set: | **Name**	**Description**
---	---
Order	The order in which the tests will be executed.	

	Command or Action	Purpose	
		Name	**Description**
		CPU Filter	Sets the CPU filter to all CPUs or to a specified CPU.
		Loop Count	Sets the loop count to the specified iterations. The range is from 1-1000.
		Memory Chunk Size	Sets the memory chunk to 5mb-chunk or big-chunk.
		Memory Size	Sets the memory size to a specific value.
		Pattern	Sets the memory test to butterfly, killer, prbs, prbs-addr, or prbs-killer.
Step 5	UCS-A /org/diag-policy/memory-test* # **set cpu-filter** {**all cpus** | **p0-p1-cpus**}	Sets the CPU filter to all CPUs or on the core 0 and 1 CPUs. Values are all cups or p0-p1-cpus.	
Step 6	UCS-A /org/diag-policy/memory-test* # **set memchunksize** {**5mb-chunk** | **big-chunk**}	Sets the memory chunk size to the specified value in GiB. Values are 5mb-chunk or big-chunk	
Step 7	UCS-A /org/diag-policy/memory-test* # **set memsize** {**0-4096** | **all**}	Sets the memory size to the specified value. The available values are 0-4096 or all	
Step 8	UCS-A /org/diag-policy/memory-test* # **set pattern** {**butterfly** |**killer** |**prbs** |**prbs-addr** |**prbs-killer**}	Sets the memory test to the specified pattern. Available patterns are butterfly, killer, prbs, prbs-addr, or prbs-killer.	
Step 9	UCS-A /org/diag-policy/memory-test* # **set loopcount 1-1000**	Sets the loop count to the specified iterations. The loop count can range from 1 to 1000.	
Step 10	UCS-A /org/diag-policy/memory-test* # **commit-buffer**	Commits the transaction to the system configuration.	
Step 11	UCS-A /org/diag-policy/memory-test # **exit**	Exits from the memory test scope.	
Step 12	UCS-A /org/diag-policy # **show configuration**	Displays the configuration values set for the memory test of the custom diagnostic policy.	

Example

The following example shows how to create a memory test for a diagnostic policy:

```
UCS-A# scope org
UCS-A /org # create diag-policy P2
```

```
UCS-A /org/diag-policy* # commit-buffer
UCS-A /org/diag-policy # create memory-test 1
UCS-A /org/diag-policy/memory-test* # set cpu-filter all-cpus
UCS-A /org/diag-policy/memory-test* # set memchunksize big-chunk
UCS-A /org/diag-policy/memory-test* # set memsize all
UCS-A /org/diag-policy/memory-test* # set pattern butterfly
UCS-A /org/diag-policy/memory-test* # set loopcount 1000
UCS-A /org/diag-policy/memory-test* # commit-buffer
UCS-A /org/diag-policy/memory-test # exit
UCS-A /org/diag-policy # show configuration
enter diag-policy P2
enter memory-test 1
set cpu-filter all-cpus
set loopcount 1000
set memchunksize big-chunk
set memsize all
set pattern butterfly
exit
set descr ""
set policy-owner local
exit
UCS-A /org/diag-policy #
```

Deleting a Diagnostic Policy

Before you begin

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

Procedure

	Command or Action	Purpose
Step 1	UCS-A # **scope org**	Enters the organization configuration mode.
Step 2	UCS-A /org # delete **diag-policy** <*diag-policy*>	Deletes the specified diagnostic policy.
Step 3	UCS-A /org* # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to delete a diagnostic policy:

```
UCS-A # scope org
UCS-A /org # delete diag-policy P2
UCS-A /org* # commit-buffer
UCS-A /org #
```

Running a Diagnostics Test on a Server

Before you begin

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

Procedure

	Command or Action	**Purpose**
Step 1	UCS-A # **scope server** *chassis-id /server-id*	Enters chassis server scope for the specified server.
Step 2	UCS-A /chassis/server # **scope diag**	Enters the diagnostic mode.
Step 3	UCS-A /chassis/server/diag # **set diag-policy-name** *<diag-policy-name>*	Associates the specified diagnostic policy with the server.
Step 4	UCS-A /chassis/server/diag* # **commit-buffer**	Commits the transaction to the system configuration.
Step 5	UCS-A /chassis/server/diag # **show**	Displays the server diagnostic details.
Step 6	UCS-A /chassis/server/diag # **start**	Runs the diagnostic test on the server.
Step 7	UCS-A /chassis/server/diag* # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to run a diagnostic test on server 1/7:

```
UCS-A # scope server 1/7
UCS-A /chassis/server # scope diag
UCS-A /chassis/server/diag # set diag-policy-name P1
UCS-A /chassis/server/diag* # commit-buffer
UCS-A /chassis/server/diag # show
Oper State    Diag Overall Progress    Diag Policy Name
-----------   ---------------------    ----------------
Completed     100                      P1
UCS-A /chassis/server/diag # start
UCS-A /chassis/server/diag* # commit-buffer
UCS-A /chassis/server/diag #
```

Stopping a Diagnostics Test

Before you begin

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

Procedure

	Command or Action	Purpose
Step 1	UCS-A # **scope server**	Enters the server configuration mode.
Step 2	UCS-A chassis/server # **scope diag**	Enters diagnostics configuration mode.
Step 3	UCS-A chassis/server/diag # **stop**	Stops the diagnostic policy.
Step 4	UCS-A /chassis/server/diag* # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to stop a diagnostic policy:

```
UCS-A# scope server 1/2
UCS-A /chassis/server # scope diag
UCS-A /chassis/server/diag # stop
UCS-A /chassis/server/diag* # commit-buffer
```

Diagnostics Troubleshooting

Issue	Steps to Debug
If the BIOS detects a bad DIMM, the DIMM is disabled and is not visible to the Diagnostics operation.	Refer to memory-related faults in addition to the diagnostics operation results.
If the DIMM blacklisting feature is enabled and a DIMM is blacklisted, it is not visible to the Diagnostics operation.	Refer to memory-related faults in addition to the diagnostics operation results.

The Diagnostics operation may not execute successfully, if the server has bad DIMMs which prevent the server from booting.	NA
The Diagnostics operation can fail, if an uncorrectable error causes a server reboot.	NA
A Diagnostics operation failure can occur if there are memory errors that cause the Diagnostics operation to hang.	NA

The Diagnostics operation can be interrupted by external events, such as a managed endpoint failover or a critical UCSM process restart. In these cases, the Diagnostics operation is cancelled and the Memory Tests are marked as failed.	The failure is triggered by external events. Retry the Diagnostics operation.
A Memory test fails with the error: Uncorrectable errors detected.	Check for server faults under the `Chassis/Server/Faults` tab. See the SEL logs for the DIMM errors under the `Chassis/Server/SEL Logs` tab.

A Memory test failure needs further analysis.	See the diagnostics operation logs in following log file archive on the primary FI in the /workspace partiti diagnostics/diag_log_<system-name>_<timestamp>_<chassis-id>_<blade-id>.t
	
	See the analysis file: `tmp/ServerDiags/MemoryPmem2.<id>/MemoryPmem2.analysis` in t previously mentioned log file archive.
	
	Use the following command to find the diagnostics logs with the analysis files:
	
	`# for file in `ls /workspace/diagnostics/*diag*`; do tar -tzvf $file |`
	`grep analysis && echo "IN " $file; done`