Cisco UCS Manager CLI Configuration Guide, Release 1.x

First Published: 06/25/2009

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

Text Part Number: OL-20083-01

CONTENTS

Preface

This preface includes the following:

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

Organization

This document includes the following sections:

Section	Title	Description
Part 1	Introduction	Describes the Cisco Unified Computing System (UCS), UCS Manager, and UCS Manager CLI.

Section	Title	Description
Part 2	System Configuration	Describes configuring fabric interconnects, ports, communication services, primary authentication, and role-based access control configuration, and also describes managing firmware.
Part 3	Network Configuration	Describes configuring named VLANs, LAN pin groups, MAC pools, and Quality of Service (QoS).
Part 4	Storage Configuration	Describes configuring named VSANs, SAN pin groups, and WWN pools.
Part 5	Server Configuration	Describes configuring server-related policies, server-related pools, and service profiles, and also describes installing an OS on servers.
Part 6	System Management	Describes managing chassis, servers, and I/O modules, and also describes backing up and restoring the configuration.

Conventions

This document uses the following conventions:

Convention	Indication
bold font	Commands, keywords, GUI elements, and user-entered text appear in **bold** font.
italic font	Document titles, new or emphasized terms, and arguments for which you supply values are in *italic* font.
[ ]	Elements in square brackets are optional.
{x | y | z}	Required alternative keywords are grouped in braces and separated by vertical bars.
[x | y | z]	Optional alternative keywords are grouped in brackets and separated by vertical bars.
string	A nonquoted set of characters. Do not use quotation marks around the string or the string will include the quotation marks.
`courier` font	Terminal sessions and information the system displays appear in `courier` font.
< >	Nonprinting characters such as passwords are in angle brackets.
[ ]	Default responses to system prompts are in square brackets.

Convention	Indication
!, #	An exclamation point (!) or a pound sign (#) at the beginning of a line of code indicates a comment line.

Note Means *reader take note*.

Tip Means *the following information will help you solve a problem*.

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

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

Warning Means *reader be warned*. In this situation, you might perform an action that could result in bodily injury.

Related Documentation

Documentation for Cisco UCS is available at the following URL:

http://www.cisco.com

The following are related Cisco UCS documents:

- *Cisco UCS Documentation Roadmap*
- *Cisco UCS Manager GUI Configuration Guide*
- *Cisco UCS Manager XML API Programmer's Guide*
- *Cisco UCS Manager Troubleshooting Guide*
- *Cisco UCS Site Planning Guide*
- *Cisco UCS 6100 Series Fabric Interconnect Hardware Installation Guide*
- *Cisco UCS 5108 Server Chassis Hardware Installation Guide*
- *Regulatory Compliance and Safety Information for Cisco UCS*
- *Release Notes for Cisco UCS Manager*

Documentation Feedback

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

Obtaining Documentation and Submitting a Service Request

For information on obtaining documentation, submitting a service request, and gathering additional information, see the monthly What's New in Cisco Product Documentation, which also lists all new and revised Cisco technical documentation, at:

http://www.cisco.com/en/US/docs/general/whatsnew/whatsnew.html

Subscribe to the *What's New in Cisco Product Documentation* as a Really Simple Syndication (RSS) feed and set content to be delivered directly to your desktop using a reader application. The RSS feeds are a free service and Cisco currently supports RSS version 2.0.

PART I

Introduction

CHAPTER 1

Overview of Cisco Unified Computing System

This chapter includes:

About Cisco Unified Computing System

Cisco Unified Computing System (Cisco UCS) fuses access layer networking and servers. This high-performance, next-generation server system provides a data center with a high degree of workload agility and scalability.

The hardware and software components support Cisco's unified fabric, which runs multiple types of data-center traffic over a single converged network adapter.

Architectural Simplification

The simplified architecture of Cisco UCS reduces the number of required devices and centralizes switching resources. By eliminating switching inside a chassis, network access-layer fragmentation is significantly reduced.

Cisco UCS implements Cisco unified fabric within racks and groups of racks, supporting Ethernet and Fibre Channel protocols over 10 Gigabit Cisco® Data Center Ethernet and Fibre Channel over Ethernet (FCoE) links.

The result of this radical simplification is a reduction by up to two-thirds of the switches, cables, adapters, and management points. All devices in a Cisco UCS instance remain under a single management domain, which remains highly available through the use of redundant components.

High Availability

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

Scalability

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

Flexibility

A Cisco UCS instance allows you to quickly align computing resources in the data center with rapidly changing business requirements. This built-in flexibility is determined by whether you choose to fully implement the stateless computing feature.

Pools of servers and other system resources can be applied as necessary to respond to workload fluctuations, support new applications, scale existing software and business services, and accommodate both scheduled and unscheduled downtime. Server identity can be abstracted into a mobile service profile that can be moved from server to server with minimal downtime and no need for additional network configuration.

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

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

Optimized for Server Virtualization

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

Unified Fabric

With unified fabric, multiple types of data center traffic can run over a single Data Center Ethernet (DCE) network. Instead of having a series of different host bus adapters (HBAs) and network interface cards (NICs) present in a server, unified fabric uses a single converged network adapter. This adapter can carry LAN and SAN traffic on the same cable.

Cisco UCS uses Fibre Channel over Ethernet (FCoE) to carry Fibre Channel and Ethernet traffic on the same physical Ethernet connection between the fabric interconnect and the server. This connection terminates at a converged network adapter on the server, and the unified fabric terminates on the uplink ports of the fabric interconnect. On the core network, the LAN and SAN traffic remains separated. Cisco UCS does not require that you implement unified fabric across the data center.

The converged network adapter presents an Ethernet interface and Fibre Channel interface to the operating system. At the server, the operating system is not aware of the FCoE encapsulation because it sees a standard Fibre Channel HBA.

At the fabric interconnect, the server-facing Ethernet port receives the Ethernet and Fibre Channel traffic. The fabric interconnect (using Ethertype to differentiate the frames) separates the two traffic types. Ethernet frames and Fibre Channel frames are switched to their respective uplink interfaces.

Fibre Channel over Ethernet

Cisco UCS leverages Fibre Channel over Ethernet (FCoE) standard protocol to deliver Fibre Channel. The upper Fibre Channel layers are unchanged, so the Fibre Channel operational model is maintained. FCoE network management and configuration is similar to a native Fibre Channel network.

FCoE encapsulates Fibre Channel traffic over a physical Ethernet link. FCoE is encapsulated over Ethernet with the use of a dedicated Ethertype, 0x8906, so that FCoE traffic and standard Ethernet traffic can be carried on the same link. FCoE has been standardized by the ANSI T11 Standards Committee.

Fibre Channel traffic requires a lossless transport layer. Instead of the buffer-to-buffer credit system used by native Fibre Channel, FCoE depends upon the Ethernet link to implement lossless service.

Ethernet links on the fabric interconnect provide two mechanisms to ensure lossless transport for FCoE traffic:

- Link-level flow control
- Priority flow control

Link-Level Flow Control

IEEE 802.3x link-level flow control allows a congested receiver to signal the endpoint to pause data transmission for a short time. This link-level flow control pauses all traffic on the link.

The transmit and receive directions are separately configurable. By default, link-level flow control is disabled for both directions.

On each Ethernet interface, the fabric interconnect can enable either priority flow control or link-level flow control (but not both).

Priority Flow Control

The priority flow control (PFC) feature applies pause functionality to specific classes of traffic on the Ethernet link. For example, PFC can provide lossless service for the FCoE traffic, and best-effort service for the standard Ethernet traffic. PFC can provide different levels of service to specific classes of Ethernet traffic (using IEEE 802.1p traffic classes).

PFC decides whether to apply pause based on the IEEE 802.1p CoS value. When the fabric interconnect enables PFC, it configures the connected adapter to apply the pause functionality to packets with specific CoS values.

By default, the fabric interconnect negotiates to enable the PFC capability. If the negotiation succeeds, PFC is enabled and link-level flow control remains disabled (regardless of its configuration settings). If the PFC negotiation fails, you can either force PFC to be enabled on the interface or you can enable IEEE 802.x link-level flow control.

Server Architecture and Connectivity

Overview of Service Profiles

Service profiles are the central concept of Cisco UCS. Each service profile serves a specific purpose: ensuring that the associated server hardware has the configuration required to support the applications it will host.

The service profile maintains configuration information about the server hardware, interfaces, fabric connectivity, and server and network identity. This information is stored in a format that you can manage through Cisco UCS Manager. All service profiles are centrally managed and stored in a database on the fabric interconnect.

Every server must be associated with a service profile.

Important At any given time, each server can be associated with only one service profile. Similarly, each service profile can be associated with only one server at a time.

After you associate a service profile with a server, the server is ready to have an operating system and applications installed, and you can use the service profile to review the configuration of the server. If the server associated with a service profile fails, the service profile does not automatically fail over to another server.

When a service profile is disassociated from a server, the identity and connectivity information for the server is reset to factory defaults.

Network Connectivity through Service Profiles

Each service profile specifies the LAN and SAN network connections for the server through the Cisco UCS infrastructure and out to the external network. You do not need to manually configure the network connections for Cisco UCS servers and other components. All network configuration is performed through the service profile.

When you associate a service profile with a server, the Cisco UCS internal fabric is configured with the information in the service profile. If the profile was previously associated with a different server, the network infrastructure reconfigures to support identical network connectivity to the new server.

Configuration through Service Profiles

A service profile can take advantage of resource pools and policies to handle server and connectivity configuration.

Hardware Components Configured by Service Profiles

When a service profile is associated with a server, the following components are configured according to the data in the profile:

- Server, including BIOS and BMC
- Adapters
- Fabric Interconnect

You do not need to configure these hardware components directly.

Server Identity Management through Service Profiles

You can use the network and device identities burned into the server hardware at manufacture or you can use identities that you specify in the associated service profile either directly or through identity pools, such as MAC, WWN, and UUID.

The following are examples of configuration information that you can include in a service profile:

- Profile name and description
- Unique server identity (UUID)
- LAN connectivity attributes, such as the MAC address
- SAN connectivity attributes, such as the WWN

Operational Aspects configured by Service Profiles

You can configure some of the operational functions for a server in a service profile, such as:

- Firmware packages and versions
- Operating system boot order and configuration
- IPMI and KVM access

vNIC Configuration by Service Profiles

A vNIC is a virtualized network interface that is configured on a physical network adapter and appears to be a physical NIC to the operating system of the server. The type of adapter in the system determines how many vNICs you can create. For example, a Cisco UCS CNA M71KR adapter has two NICs, which means you can create a maximum of two vNICs for each of those adapters.

A vNIC communicates over Ethernet and handles LAN traffic. At a minimum, each vNIC must be configured with a name and with fabric and network connectivity.

vHBA Configuration by Service Profiles

A vHBA is a virtualized host bus adapter that is configured on a physical network adapter and appears to be a physical HBA to the operating system of the server. The type of adapter in the system determines how many vHBAs you can create. For example, a Cisco UCS CNA M71KR has two HBAs, which means you can create a maximum of two vHBAs for each of those adapters. In contrast, a Cisco UCS 82598KR-CI does not have any HBAs, which means you cannot create any vHBAs for those adapters.

A vHBA communicates over FCoE and handles SAN traffic. At a minimum, each vHBA must be configured with a name and fabric connectivity.

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 and then associated 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 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:

- 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

Service Profiles that Inherit Server Identity

This type of service profile is the simplest to use and create. This profile mimics the management of a rack mounted server. It is tied to a specific server and cannot be moved to another server.

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

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

- MAC addresses for the two NICs
- For the Cisco UCS CNA M71KR adapters, the WWN addresses for the two HBAs
- BIOS versions
- Server UUID

Important The server identity and configuration information inherited through this service profile may not be the values burned into the server hardware at manufacture if those values have been subsequently changed before this profile is associated with the server.

Service Profile Templates

Service profile templates enable you to create a large number of similar service profiles. With a service profile template, you can quickly create several service profiles with the same basic parameters, such as the number of vNICs and vHBAs, and with identity information drawn from the same pools.

Tip If you need only one service profile with similar values to an existing service profile, you can clone a service profile in the Cisco UCS Manager GUI.

For example, if you need several service profiles with similar values to configure servers to host database software, you can create a service profile template, either manually or from an existing service profile. You then use the template to create the service profiles.

Cisco UCS supports the following types of service profile templates:

Initial template Service profiles created from an initial template inherit all of the properties of the template. However, after you create the profile, it is no longer connected to the template. If you need to make changes to one or more profiles created from this template, you must change each profile individually.

Updating template Service profiles created from an updating template inherit all properties of the template and remain connected to the template. Any changes to the template automatically update the service profiles created from the template.

Policies

Policies determine how Cisco UCS components will act in specific circumstances. You can create multiple instances of most policies. For example, you might want different boot policies, so that some servers can PXE boot, some can SAN boot, and others can boot from local storage.

Policies allow separation of functions within the system. A subject matter expert can define policies that are used in a service profile, which is created by someone without that subject matter expertise. For example, a LAN administrator can create adapter policies and quality of service policies for the system. These policies can then be used in a service profile that is created by someone who has limited or no subject matter expertise with LAN administration.

You can create and use two types of policies in Cisco UCS Manager:

- Configuration policies which configure the servers and other components.
- Operational policies which control certain management, monitoring, and access control functions.

Configuration Policies

Boot Policy

This policy determines the following:

- Configuration of the boot device
- Location from which the server boots
- Order in which boot devices are invoked

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

You must include this policy 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, the server uses the default settings in the BIOS to determine the boot order.

Important Changes to a boot policy may be propagated to all servers created with an updating service profile template that includes that boot policy. Reassociation of the service profile with the server to rewrite the boot order information in the BIOS is auto-triggered.

Guidelines

When you create a boot policy, you can add one or more of the following to the boot policy and specify their boot order:

Boot type	Description
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, 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 exact same operating system image. Therefore, the new server appears to be the exact same server to the network.
LAN boot	Boots from a centralized provisioning server. It is frequently used to install operating systems on a server from that server.
Local disk boot	If the server has a local drive, boots from that drive.
Virtual media boot	Mimics the insertion of a physical CD-ROM disk (read-only) or floppy disk (read-write) into a server. It is typically used to manually install operating systems on a server.

Note The default boot order is as follows:

1. Local disk boot
2. LAN boot
3. Virtual media read-only boot
4. Virtual media read-write boot

Chassis Discovery Policy

This discovery policy determines how the system reacts when you add a new chassis. If you create a chassis discovery policy, the system does the following:

- Automatically configures the chassis for the number of links between the chassis and the fabric interconnect specified in the policy.
- Specifies the power policy to be used by the chassis.

Ethernet and Fibre Channel Adapter Policies

These policies govern the host-side behavior of the adapter, including how the adapter handles traffic. For example, you can use these policies to change default settings for the following:

- Queues
- Interrupt handling
- Performance enhancement

- RSS hash
- Failover in an cluster configuration with two fabric interconnects

Operating systems are sensitive to the settings in these policies. The configuration and selection of the policy is driven by the type of operating system.

Host Firmware Pack

This policy enables you to specify a common set of firmware versions that make up the host firmware pack. The host firmware includes the following server and adapter components:

- BIOS
- SAS controller
- Emulex Option ROM (applicable only to Emulex-based Converged Network Adapters [CNAs])
- Emulex firmware (applicable only to Emulex-based CNAs)
- QLogic option ROM (applicable only to QLogic-based CNAs)
- Adapter firmware

The firmware pack is pushed to all servers associated with service profiles that include this policy.

This policy ensures that the host firmware is identical on all servers associated with service profiles which use the same policy. Therefore, if you move the service profile from one server to another, the firmware versions are maintained. Also, if you change the firmware version of the component in the firmware pack, new versions are applied to all the affected service profiles immediately, which could cause server reboots.

You must include this policy in a service profile, and that service profile must be associated with a server for it to take effect.

Prerequisites

This policy is not dependent upon any other policies. However, you must ensure that the appropriate firmware has been downloaded to the fabric interconnect. If the firmware image is not available while associating the service profile, UCSM will just ignore firmware update and complete association.

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 BMC. 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 must include this policy in a service profile and that service profile must be associated with a server for it to take effect.

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 the RAID mode and the way the drives are partitioned.

You must include this policy in a service profile, and that service profile must be associated with a server for it to take effect.

Management Firmware Pack

This policy enables you to specify a common set of firmware versions that make up the management firmware pack. The management firmware includes the server controller (BMC) on the server.

The firmware pack is pushed to all servers associated with service profiles that include this policy.

This policy ensures that the BMC firmware is identical on all servers associated with service profiles which use the same policy. Therefore, if you move the service profile from one server to another, the firmware versions are maintained.

You must include this policy in a service profile, and that service profile must be associated with a server for it to take effect.

Prerequisites

This policy is not dependent upon any other policies. However, you must ensure that the appropriate firmware has been downloaded to the fabric interconnect.

Quality of Service Policies

QoS policies assign a system class to the outgoing traffic for a vNIC or vHBA. This system class determines the quality of service for that traffic.

You must include a QoS policy in a vNIC policy or vHBA policy and then include that policy in a service profile to configure the vNIC or vHBA.

Server Autoconfiguration Policy

This policy determines whether one or more of the following is automatically applied to a new server:

- A server pool policy qualification that qualifies the server for one or more server pools
- An organization
- A service profile template that associates the server with a service profile created from that template

Prerequisites

Server Discovery Policy

This discovery policy determines how the system reacts when you add a new server. 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.

With this policy, an inventory of the server is conducted, then server pool policy qualifications are run to determine whether the new server qualifies for one or more server pools.

Server Inheritance Policy

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.

Server Pool Policy

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.

Server Pool Policy Qualifications

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.

Depending upon the implementation, you may include server pool policy qualifications in the following policies:

- Autoconfiguration policy
- Chassis discovery policy
- Server discovery policy
- Server inheritance policy
- Server pool policy

vHBA Template

This policy defines how a vHBA on a server connects to the SAN. This policy is also referred to as a vHBA SAN connectivity template.

You need to include this policy in a service profile for it to take effect.

vNIC Template

This policy defines how a vNIC on a server connects to the LAN. This policy is also referred to as a vNIC LAN connectivity policy.

You need to include this policy in a service profile for it to take effect.

Operational Policies

Fault Collection Policy

The fault collection policy controls the lifecycle of a fault in a Cisco UCS instance, including the length of time that each fault remains in the flapping and retention intervals.

A fault in Cisco UCS has the following lifecycle:

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

5 If the condition reoccurs during the retention interval, the fault returns to the active state. If the condition does not reoccur, the fault is deleted.

Scrub Policy

This policy determines what happens to local data on a server during the discovery process and when the server is disassociated from a service profile. This policy can ensure that the data on local drives is erased at those times.

Serial over LAN Policy

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.

Statistics Collection Policy

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

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

- Adapter—statistics related to the adapters in the fabric Interconnect
- Chassis—statistics related to the blade chassis
- Host—this policy is a placeholder for future support
- Port—statistics related to the ports, including server ports, uplink Ethernet ports, and uplink Fibre Channel ports
- Server—statistics related to servers

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

Statistics Threshold Policy

A statistics threshold policy monitors statistics about certain aspects of the system and generates an event if the threshold is crossed. You can set both minimum and maximum thresholds. For example, you can configure the policy to raise an alarm if the CPU temperature exceeds a certain value, or if a server is overutilized or underutilized.

These threshold policies do not control the hardware or device-level thresholds enforced by endpoints, such as the BMC. Those thresholds are burned in to the hardware comonents at manufacture.

Cisco UCS enables you to configure statistics threshold policies for the following components:

- Servers and server components
- Uplink Ethernet ports
- Ethernet server ports, chassis, and Fabric Interconnects

- Fibre Channel port

Note You cannot create or delete a statistics threshold policy for Ethernet server ports, uplink Ethernet ports, or uplink Fibre Channel ports. You can only configure the existing default policy.

Pools

Pools are collections of identities, or physical or logical resources, that are available in the system. All pools increase the flexibility of service profiles and allow you to centrally manage your system resources.

You can use pools to segment unconfigured servers or available ranges of server identity information into groupings that make sense for the data center. For example, if you create a pool of unconfigured servers with similar characteristics and include that pool in a service profile, you can use a policy to associate that service profile with an available, unconfigured server.

If you pool identify information, such as MAC addresses, you can pre-assign ranges for servers that will host specific applications. For example, all database servers could be configured within the same range of MAC addresses, UUIDs, and WWNs.

Server Pools

A server pool contains a set of servers. These servers typically share the same characteristics. Those characteristics can be their location in the chassis, or an attribute such as server type, amount of memory, local storage, type of CPU, or local drive configuration. You can manually assign a server to a server pool, or use server pool policies and server pool policy qualifications to automate the assignment.

If your system implements multi-tenancy through organizations, you can designate one or more server pools to be used by a specific organization. For example, a pool that includes all servers with two CPUs could be assigned to the Marketing organization, while all servers with 64GB memory could be assigned to the Finance organization.

A server pool can include servers from any chassis in the system. A given server can belong to multiple server pools.

MAC Pools

A MAC pool is a collection of network identities, or MAC addresses, that are unique in their layer 2 environment and are available to be assigned to vNICs on a server. If you use MAC pools in service profiles, you do not have to manually configure the MAC addresses to be used by the server associated with the service profile.

In a system that implements multi-tenancy, you can use the organizational hierarchy to ensure that MAC pools can only be used by specific applications or business services. Cisco UCS Manager will use the name resolution policy to assign MAC addresses from the pool.

To assign a MAC address to a server, you must include the MAC pool in a vNIC policy. The vNIC policy is then included in the service profile assigned to that server.

You can specify your own MAC addresses or use a group of MAC addresses provided by Cisco.

UUID Suffix Pools

A UUID suffix pool is a collection of SMBIOS UUIDs that are available to be assigned to servers. The first number of digits that constitute the prefix of the UUID are fixed. The remaining digits, the UUID suffix, is variable. A UUID suffix pool ensures that these variable values are unique for each server associated with a service profile which uses that particular pool to avoid conflicts.

If you use UUID suffix pools in service profiles, you do not have to manually configure the UUID of the server associated with the service profile.

WWN Pools

A WWN pool is a collection of WWNs for use by the Fibre Channel vHBAs in a Cisco UCS instance. You create separate pools for:

- WW node names assigned to the server
- WW port names assigned to the vHBA

Important A WWN pool can include only WWNNs or WWPNs in the ranges from 20:00:00:00:00:00:00:00 to 20:FF:FF:FF:FF:FF:FF:FF or from 50:00:00:00:00:00:00:00 to 5F:FF:FF:FF:FF:FF:FF:FF. All other WWN ranges are reserved.

If you use WWN pools in service profiles, you do not have to manually configure the WWNs that will be used by the server associated with the service profile. In a system that implements multi-tenancy, you can use a WWN pool to control the WWNs used by each organization.

You assign WWNs to pools in blocks. For each block or individual WWN, you can assign a boot target.

WWNN Pools

A WWNN pool is a WWN pool which contains only WW node names. If you include a pool of WWNNs in a service profile, the associated server will be assigned a WWNN from that pool

WWPN Pools

A WWPN pool is a WWN pool which contains only WW port names. If you include a pool of WWPNs in a service profile, the port on each vHBA of the associated server will be assigned a WWPN from that pool

Management IP Pool

The management IP pool is a collection of external IP addresses. Cisco UCS Manager reserves each block of IP addresses in the management IP pool for external access that terminates in the server controller (BMC) in a server.

Cisco UCS Manager uses the IP addresses in a management IP pool for external access through serial over LAN and IPMI.

Traffic Management

Oversubscription

Oversubscription occurs when multiple network devices are connected to the same fabric interconnect port. This practice optimizes fabric interconnect use, since ports rarely run at maximum speed for any length of time. As a result, when configured correctly, oversubscription allows you to take advantage of unused bandwidth. However, incorrectly configured oversubscription can result in contention for bandwidth and a lower quality of service to all services that use the oversubscribed port.

For example, oversubscription can occur if four servers share a single uplink port, and all four servers attempt to send data at a cumulative rate higher than available bandwidth of uplink port.

Oversubscription Considerations

The following elements can impact how you configure oversubscription in a Cisco UCS:

The ratio of server-facing ports to uplink ports

You need to know what how many server-facing ports and uplink ports are in the system, because that ratio can impact performance. For example, if your system has twenty ports that can communicate down to the servers and only two ports that can communicate up to the network, your uplink ports will be oversubscribed. In this situation, the amount of traffic created by the servers can also affect performance.

The number of uplink ports from the fabric interconnect to the network

You can choose to add more uplink ports between the Cisco UCS fabric interconnect and the upper layers of the LAN to increase bandwidth. In Cisco UCS, you must have at least one uplink port per fabric interconnect to ensure that all servers and NICs to have access to the LAN. The number of LAN uplinks should be determined by the aggregate bandwidth needed by all Cisco UCS servers.

FC uplink ports are available on the expansion slots only. You must add more expansion slots to increase number of available FC uplinks. Ethernet uplink ports can exist on the fixed slot and on expansion slots.

For example, if you have two Cisco UCS 5100 series chassis, that are fully populated with half width Cisco UCS B200-M1 servers you have 16 servers. In a cluster configuration, with one LAN uplink per fabric interconnect, these 16 servers share 20GbE of LAN bandwidth. If more capacity is needed, more uplinks from the fabric interconnect should be added. We recommend that you have symmetric configuration of the uplink in cluster configurations. In the same example, if 4 uplinks are used in each fabric interconnect, the 16 servers are sharing 80 Gb of bandwidth, so each has approximately 5 Gb of capacity. When multiple uplinks are used on a Cisco UCS fabric interconnect the network design team should consider using a port channel to make best use of the capacity.

The number of uplink ports from the I/O module to the fabric interconnect

You can choose to add more bandwidth between I/O module and fabric interconnect by using more uplink ports and increasing the number of cables. In Cisco UCS, you can have one, two, or four cables connecting a I/O module to a Cisco UCS fabric interconnect. The number of cables determines the number of active uplink ports and the oversubscription ratio. For example, one cable results in 8:1 oversubscription for one I/O module. If two I/O modules are in place, each with one cable, and you have 8 half width blades, the 8 blades will be sharing two uplinks (one left IOM and one right IOM). This results in 8 blades sharing an aggregate bandwidth of 20 Gb of Unified Fabric capacity. If two cables are used, this results in 4:1 oversubscription per IOM (assuming all slots populated with half width blades), and four cables result in 2:1 oversubscription. The lower oversubscription ratio gives you higher performance, but is also more costly as you consume more fabric interconnect ports.

The number of active links from the server to the fabric interconnect

Oversubscription is affected by how many servers are in a particular chassis and how bandwidth-intensive those servers are. The oversubscription ratio will be reduced if the servers which generate a large amount of traffic are not in the same chassis, but are shared between the chassis in the system. The number of cables between chassis and fabric interconnect determines the oversubscription ratio. For example, one cable results in 8:1 oversubscription, two cables result in 4:1 oversubscription, and four cables result in 2:1 oversubscription. The lower oversubscription ratio will give you higher performance, but is also more costly.

Guidelines for Estimating Oversubscription

When you estimate the optimal oversubscription ratio for a fabric interconnect port, consider the following guidelines:

Cost/performance slider

The prioritization of cost and performance is different for each data center and has a direct impact on the configuration of oversubscription. When you plan hardware usage for oversubscription, you need to know where the data center is located on this slider. For example, oversubscription can be minimized if the data center is more concerned with performance than cost. However, cost is a significant factor in most data centers, and oversubscription requires careful planning.

Bandwidth usage

The estimated bandwidth that you expect each server to actually use is important when you determine the assignment of each server to a fabric interconnect port and, as a result, the oversubscription ratio of the ports. For oversubscription, you must consider how many GBs of traffic the server will consume on average, the ratio of configured bandwidth to used bandwidth, and the times when high bandwidth use will occur.

Network type

The network type is only relevant to traffic on uplink ports, because FCoE does not exist outside . The rest of the data center network only differentiates between LAN and SAN traffic. Therefore, you do not need to take the network type into consideration when you estimate oversubscription of a fabric interconnect port.

Pinning

Pinning in Cisco UCS is only relevant to uplink ports. You can pin Ethernet or FCoE traffic from a given server to a specific uplink Ethernet port or uplink FC port.

When you pin the NIC and HBA of both physical and virtual servers to uplink ports, you give the fabric interconnect greater control over the unified fabric. This control ensures more optimal utilization of uplink port bandwidth.

Cisco UCS uses pin groups to manage which NICs, vNICs, HBAs, and vHBAs are pinned to an uplink port. To configure pinning for a server, you can either assign a pin group directly, or include a pin group in a vNIC policy, and then add that vNIC policy to the service profile assigned to that server. All traffic from the vNIC or vHBA on the server travels through the I/O module to the same uplink port.

Pinning Server Traffic to Server Ports

All server traffic travels through the IO Module to server ports on the Fabric Interconnect. The number of links for which the chassis is configured determines how this traffic is pinned.

The pinning determines which server traffic goes to which server port on the Fabric Interconnect.This pinning is fixed. You cannot modify it. As a result, you must consider the server location when you determine the appropriate allocation of bandwidth for a chassis.

Note You must review the allocation of ports to links before you allocate servers to slots. The cabled ports are not necessarily port 1 and port 2 on the IO Module. If you change the number of links between the Fabric Interconnect and the IO Module, you must reacknowledge the chassis to have the traffic rerouted.

All port numbers refer to the Fabric Interconnect-side ports on the IO Module.

Chassis with One IO Module

Links on Chassis	Servers Pinned to Link 1	Servers Pinned to Link 2	Servers Pinned to Link 3	Servers Pinned to Link 4
1 link	All server slots.	None	None	None
2 links	Slots 1, 3, 5, and 9.	Slots 2, 4, 6, and 8.	None	None
4 links	Slots 1 and 5.	Slots 2 and 6.	Slots 3 and 7.	Slots 4 and 8.

Chassis with Two IO Modules

If a chassis has two IO Modules, then traffic from one IO Module goes to one of the Fabric Interconnects and traffic from the other IO Module goes to the second Fabric Interconnect. You cannot connect two IO Modules to a single Fabric Interconnect.

Adding a second IO Module to a chassis does not improve oversubscription. The server port pinning is the same for a single IO Module. The second IO Module improves the high availability of the system through the vNIC binding to the Fabric Interconnect.

Fabric Interconnect Configured in vNIC	Server Traffic Path
A	Server traffic goes to Fabric Interconnect A. If A fails, the server traffic does not fail over to B.
B	All server traffic goes to Fabric Interconnect B. If B fails, the server traffic does not fail over to A.
A-B	All server traffic goes to Fabric Interconnect A. If A fails, the server traffic fails over to B.
B-A	All server traffic goes to Fabric Interconnect B. If B fails, the server traffic fails over to A.

Guidelines for Pinning

When you determine the optimal configuration for pin groups and pinning for an uplink port, consider the estimated bandwidth usage for the servers. If you know that some servers in the system will use a lot of bandwidth, ensure that you pin these servers to different uplink ports.

Quality of Service

Cisco UCS provides the following methods to implement quality of service:

- System classes that specify the global configuration for certain types of traffic across the entire system
- QoS policies that assign system classes for individual vNICs
- Flow control policies that determine how uplink Ethernet ports handle pause frames

System Classes

Cisco UCS uses Data Center Ethernet (DCE) to handle all traffic inside a Cisco UCS instance. This industry standard enhancement to Ethernet divides the bandwidth of the Ethernet pipe into eight virtual lanes. System classes determine how the DCE bandwidth in these virtual lanes is allocated across the entire Cisco UCS instance.

Each system class reserves a specific segment of the bandwidth for a specific type of traffic. This provides a level of traffic management, even in an oversubscribed system. For example, you can configure the Fibre Channel Priority system class to determine the percentage of DCE bandwidth allocated to FCoE traffic.

The following table describes the system classes:

System Class	Description
Platinum Priority Gold Priority Silver Priority Bronze Priority	A configurable set of system classes that you can include in the QoS policy for a service profile. Each system class manages one lane of traffic. All properties of these system classes are available for you to assign custom settings and policies.
Best Effort Priority	A system class that sets the quality of service for the lane reserved for Basic Ethernet traffic. Some properties of this system class are preset and cannot be modified. For example, this class has a drop policy that allows it to drop data packets if required.
Fibre Channel Priority	A system class that sets the quality of service for the lane reserved for Fibre Channel over Ethernet traffic. Some properties of this system class are preset and cannot be modified. For example, this class has a no-drop policy that ensures it never drops data packets.

Quality of Service Policies

QoS policies assign a system class to the outgoing traffic for a vNIC or vHBA. This system class determines the quality of service for that traffic.

You must include a QoS policy in a vNIC policy or vHBA policy and then include that policy in a service profile to configure the vNIC or vHBA.

Flow Control Policies

Flow control policies determine whether the uplink Ethernet ports in a Cisco UCS instance send and receive IEEE 802.3x pause frames when the receive buffer for a port fills. These pause frames request that the transmitting port stop sending data for a few milliseconds until the buffer clears.

For flow control to work between a LAN port and an uplink Ethernet port, you must enable the corresponding receive and send flow control parameters for both ports. For Cisco UCS, the flow control policies configure these parameters.

If you enable the send function, then the uplink Ethernet port sends a pause request to the network port if the incoming packet rate becomes too high. The pause remains in effect for a few milliseconds before traffic is reset to normal levels. If you enable the receive function, then the uplink Ethernet port will honor all pause requests from the network port. All traffic is halted on that uplink port until the network port cancels the pause request.

Because you assign the flow control policy to the port, changes to the policy have an immediate effect on how the port reacts to a pause frame or a full receive buffer.

Opt-In Features

Each Cisco UCS instance is licensed for all functionality. Depending upon how the system is configured, you can decide to opt in to some features or opt out of them for easier integration into existing environment. If a process change happens, you can change your system configuration and include one or both of the opt-in features.

The opt-in features are as follows:

- Stateless computing, which takes advantage of mobile service profiles with pools and policies where each component, such as a server or an adapter, is stateless.
- Multi-tenancy, which uses organizations and role-based access control to divide the system into smaller logical segments.

Stateless Computing

Stateless computing allows you to use a service profile to apply the personality of one server to a different server in the same Cisco UCS instance. The personality of the server includes the elements that identify that server and make it unique in the instance. If you change any of these elements, the server could lose its ability to access, use, or even achieve booted status.

The elements that make up a server's personality include:

- Firmware versions
- UUID (used for server identification)

- MAC Address (used for LAN connectivity)
- World Wide Names (used for SAN connectivity)
- Boot Settings

Stateless computing creates a dynamic server environment with highly flexible servers. Every physical server in a Cisco UCS instance remains anonymous until you associate a service profile with it, then the server gets the identity configured in the service profile. If you no longer need a business service on that server, you can shut it down, disassociate the service profile, and then associate a new service profile to create a new identity for the same physical server. The "new" server can then host another business service.

To take full advantage of the flexibility of statelessness, the optional local disks on the servers should only be used for swap or temp space and not to store operating system or application data.

You can choose to fully implement stateless computing for all physical servers in a Cisco UCS instance, to not have any stateless servers, or to have a mix of the two types.

If You Opt In to Stateless Computing

Each physical server in the Cisco UCS instance is defined through a service profile. Any server can be used to host one set of applications, then reassigned to another set of applications or business services, if required by the needs of the data center.

You create service profiles that point to policies and pools of resources that are defined in the instance. The server pools, WWN pools, and MAC pools ensure that all unassigned resources are available on an as-needed basis. For example, if a physical server fails, you can immediately assign the service profile to another server. Because the service profile provides the new server with the same identity as the original server, including WWN and MAC address, the rest of the data center infrastructure sees it as the same server and you do not need to make any configuration changes in the LAN or SAN.

If You Opt Out of Stateless Computing

Each server in the Cisco UCS instance is treated as a traditional rack mount server.

You create service profiles that inherit the identify information burned into the hardware and use these profiles to configure LAN or SAN connectivity for the server. However, if physical a server fails, you cannot reassign the service profile to a new server.

Multi-Tenancy

In Cisco UCS, you can use multi-tenancy to divide up the large physical infrastructure of an instance into logical entities known as organizations. As a result, you can achieve a logical isolation between organizations without providing a dedicated physical infrastructure for each organization.

You can assign unique resources to each tenant through the related organization, in the multi-tenant environment. These resources can include different policies, pools, and quality of service definitions. You can also implement locales to assign or restrict Cisco UCS user privileges and roles by organization, if you do not want all users to have access to all organizations.

If you set up a multi-tenant environment, all organizations are hierarchical. The top-level organization is always root. The policies and pools that you create in root are system-wide and are available to all organizations in the system. However, any policies and pools created in other organizations are only available to organizations that are above it the same hierarchy. For example, if a system has organizations named Finance and HR that are not in the same hierarchy, Finance cannot use any policies in the HR organization, and HR cannot access any policies in the Finance organization. However, both Finance and HR can use policies and pools in the root organization.

If you create organizations in a multi-tenant environment, you can also set up one or more of the following for each organization or for a sub-organization in the same hierarchy:

- Resource pools
- Policies
- Service profiles
- Service profile templates

If You Opt In to Multi-Tenancy

The Cisco UCS instance is divided into several distinct organizations. The types of organizations you create in a multi-tenancy implementation will depend upon the business needs of the company. Examples include organizations that represent the following:

- Enterprise groups or divisions within a company, such as marketing, finance, engineering, or human resources
- Different customers or name service domains, for service providers

You can create locales to ensure that users have access only to those organizations that they are authorized to administer.

If You Opt Out of Multi-Tenancy

The Cisco UCS instance remains a single logical entity with everything in the root organization. All policies and resource pools can be assigned to any server in the instance.

Virtualization

Virtualization allows the creation of multiple virtual machines to run in isolation, side-by-side on the same physical machine.

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

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

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

Virtualization with the Cisco UCS CNA M71KR and Cisco UCS 82598KR-CI Adapters

The Cisco UCS 82598KR-CI 10-Gigabit Ethernet Adapter, Cisco UCS M71KR - E Emulex Converged Network Adapter, and Cisco UCS M71KR - Q QLogic Converged Network Adapter support virtualized environments with the following VMware versions:

- VMware 3.5 update 4
- VMware 4.0

These environments support the standard VMware integration with ESX installed on the server and all virtual machine management performed through the VC.

Portability of Virtual Machines

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

Communication between Virtual Machines on the Same Server

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

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

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

CHAPTER 2

Overview of Cisco UCS Manager

This chapter includes:

About Cisco UCS Manager

Cisco UCS Manager is the management service for all components in a Cisco UCS instance. Cisco UCS Manager runs within the fabric interconnect. You can use any of the interfaces available with this management service to access, configure, administer, and monitor the network and server resources for all chassis connected to the fabric interconnect.

Multiple Management Interfaces

Cisco UCS Manager includes the following interfaces you can use to manage a Cisco UCS instance:

- Cisco UCS Manager GUI
- Cisco UCS Manager CLI
- XML API

Almost all tasks can be performed in any of the interfaces, and the results of tasks performed in one interface are automatically displayed in another.

However, you cannot do the following:

- Use Cisco UCS Manager GUI to invoke Cisco UCS Manager CLI.
- View a command that has been invoked through Cisco UCS Manager CLI in Cisco UCS Manager GUI.
- Generate CLI output from Cisco UCS Manager GUI.

Centralized Management

Cisco UCS Manager centralizes the management of resources and devices, rather than using multiple management points. This centralized management includes management of the following devices in a Cisco UCS instance:

- Fabric Interconnects
- Software switches for virtual servers
- Power and environmental management for chassis and servers
- Configuration and firmware updates for Ethernet NICs and Fibre Channel HBAs
- Firmware and BIOS settings for servers

Support for Virtual and Physical Servers

Cisco UCS Manager abstracts server state information—including server identity, I/O configuration, MAC addresses and World Wide Names, firmware revision, and network profiles—into a service profile. You can apply the service profile to any server resource in the system, providing the same flexibility and support to physical servers, virtual servers, and virtual machines connected to a virtual device provided by the Palo adapter.

Role-Based Administration and Multi-Tenancy Support

Cisco UCS Manager supports flexibly defined roles so that data centers can use the same best practices with which they manage discrete server, storage, and networks to operate a Cisco UCS instance. You can create user roles with privileges that reflect user responsibilities in the data center. For example, you can create:

- Server administrator roles with control over server-related configurations
- Storage administrator roles with control over tasks related to the SAN
- Network administrator roles with control over tasks related to the LAN

In a multi-tenancy environment, Cisco UCS Manager enables you to create locales for user roles that can limit the scope of a user to a particular organization.

Tasks You Can Perform in Cisco UCS Manager

You can use Cisco UCS Manager to perform management tasks for all physical and virtual devices within a Cisco UCS instance.

Cisco UCS Hardware Management

You can use Cisco UCS Manager to manage all hardware within a Cisco UCS instance, including the following:

- Chassis
- Servers
- Fabric interconnects
- Fans
- Ports

- Cards
- Slots
- I/O modules

Cisco UCS Resource Management

You can use Cisco UCS Manager to create and manage all resources within a Cisco UCS instance, including the following:

- Servers
- WWN addresses
- MAC addresses
- UUIDs
- Bandwidth

Server Administration in a Cisco UCS Instance

A server administrator can use Cisco UCS Manager to perform server management tasks within a Cisco UCS instance, including the following:

- Create server pools and policies related to those pools, such as qualification policies
- Create policies for the servers, such as discovery policies, scrub policies, and IPMI policies
- Create service profiles and, if desired, service profile templates
- Apply service profiles to servers
- Monitor faults, alarms, and the status of equipment

Network Administration in a Cisco UCS Instance

A network administrator can use Cisco UCS Manager to perform tasks required to create LAN configuration for a Cisco UCS instance, including the following:

- Configure uplink ports, port channels, and LAN PIN groups
- Create VLANs
- Configure the quality of service classes and definitions
- Create the pools and policies related to network configuration, such as MAC address pools and Ethernet adapter profiles

Storage Administration in a Cisco UCS Instance

A storage administrator can use Cisco UCS Manager to perform tasks required to create SAN configuration for a Cisco UCS instance, including the following:

- Configure ports, port channels, and SAN PIN groups
- Create VSANs
- Configure the quality of service classes and definitions

- Create the pools and policies related to the network configuration, such as WWN pools and Fibre Channel adapter profiles

Tasks You Cannot Perform in Cisco UCS Manager

You cannot use Cisco UCS Manager to perform certain system management tasks that are not specifically related to device management within a Cisco UCS instance

No Cross-System Management

You cannot use Cisco UCS Manager to manage systems or devices that are outside the Cisco UCS instance where Cisco UCS Manager is located. For example, you cannot manage heterogeneous environments, such as non-Cisco UCS x86 systems, SPARC systems, or PowerPC systems.

No Operating System or Application Provisioning or Management

Cisco UCS Manager provisions servers and, as a result, exists below the operating system on a server. Therefore, you cannot use it to provision or manage operating systems or applications on servers. For example, you cannot do the following:

- Deploy an OS, such as Windows or Linux
- Deploy patches for software, such as an OS or an application
- Install base software components, such as anti-virus software, monitoring agents, or backup clients
- Install software applications, such as databases, application server software, or web servers
- Perform operator actions, including restarting an Oracle database, restarting printer queues, or handling non-Cisco UCS user accounts
- Configure or manage external storage on the SAN or NAS storage

Cisco UCS Manager in a Cluster Environment

In a cluster Cisco UCS instance with two fabric interconnects, you can run a separate instance of the Cisco UCS Manager on each fabric interconnect. The Cisco UCS Manager on the primary fabric interconnect acts as the primary management instance, and the Cisco UCS Manager on the other fabric interconnect is the subordinate management instance.

The two instances of Cisco UCS Manager communicate across a private network between the L1 and L2 Ethernet ports on the fabric interconnects. Configuration and status information is communicated across this private network to ensure that all management information is replicated. This ongoing communication ensures that the management information for Cisco UCS persists even if the primary fabric interconnect fails. In addition, the "floating" management IP address that runs on the primary Cisco UCS Manager ensures a smooth transition in the event of a failover to the subordinate fabric interconnect.

CHAPTER 3

Overview of Cisco UCS Manager CLI

This chapter includes:

Managed Objects

Cisco UCS uses a managed object model, where managed objects are abstract representations of physical or logical entities that can be managed. For example, servers, chassis, I/O cards, and processors are physical entites represented as managed objects, and resource pools, user roles, service profiles, and policies are logical entites represented as managed objects.

Managed objects may have one or more associated properties that can be configured.

Complete a Command

You can use the Tab key in any mode to complete a command. Partially typing a command name and pressing Tab causes the command to be displayed in full, or to the point where another keyword must be chosen or an argument value must be entered.

Command History

The CLI stores all previously used commands in the current session. You can step through the previously used commands by using the Up Arrow or Down Arrow keys. The Up Arrow key steps to the previous command in the history, and the Down Arrow key steps to the next command in the history. If you get to the end of the history, pressing the Down Arrow key does nothing.

All commands in the history can be entered again by simply stepping through the history to recall the desired command and pressing Enter. The command is entered as if you had manually typed it. You can also recall a command and change it before you enter it.

Command Modes

The CLI is organized into a hierarchy of command modes, with the EXEC mode being the highest-level mode of the hierarchy. Higher-level modes branch into lower-level modes. You use **create**, **enter**, and **scope** commands to move from higher-level modes to modes in the next lower level , and the **exit** command to move up one level in the mode hierarchy.

Note Most command modes are associated with managed objects, so you must create an object before you can access the mode associated with that object. You use **create** and **enter** commands to create managed objects for the modes being accessed. The **scope** commands do not create managed objects, and can only access modes for which managed objects already exist.

Each mode contains a set of commands that can be entered in that mode. Most of the commands available in each mode pertain to the associated managed object. Depending on your assigned role and locale, you may have access to only a subset of the commands available in a mode; commands to which you do not have access are hidden.

The CLI prompt for each mode shows the full path down the mode hierarchy to the current mode. This helps you to determine where you are in the command mode hierarchy, and can be an invaluable tool when you need to navigate through the hierarchy.

The following table lists the main command modes, the commands used to access each mode, and the CLI prompt associated with each mode.

Table 1: Main Command Modes and Prompts

Mode Name	Commands Used to Access	Mode Prompt
EXEC	**top** command from any mode	#
chassis	**enter chassis** and **scope chassis** commands from EXEC mode	/chassis #
Ethernet server	**enter eth-server** and **scope eth-server** commands from EXEC mode	/eth-server #
Ethernet uplink	**enter eth-uplink** and **scope eth-uplink** commands from EXEC profile mode	/eth-uplink #
Fibre Channel uplink	**enter fc-uplink** and **scope fc-uplink** commands from EXEC profile mode	/fc-uplink #

Mode Name	Commands Used to Access	Mode Prompt
firmware	**enter firmware** and **scope firmware** commands from EXEC profile mode	/firmware #
monitoring	**enter monitoring** and **scope monitoring** commands from EXEC profile mode	/monitoring #
oranization	**create org**, **enter org**, and **scope org** commands from EXEC profile mode	/org #
security	**enter security** and **scope security** commands from EXEC profile mode	/security #
fabric-interconnect	**enter fabric-interconnect** and **scope fabric-interconnect** commands from EXEC profile mode	/fabric-interconnect #
system	**enter system** and **scope system** commands from EXEC profile mode	/system #

Online Help for the CLI

At any time, you can type the **?** character to display the options available at the current state of the command syntax. If you have not typed anything at the prompt, typing ? lists all available commands for the mode you are in. If you have partially typed a command, typing ? lists all available keywords and arguments available at your current position in the command syntax.

PART II

System Configuration

CHAPTER 4

Configuring the Fabric Interconnects

This chapter includes:

Initial System Setup

The first time that you access a fabric interconnect in a Cisco UCS instance, a setup wizard prompts you for the following information required to configure the system:

- Installation method (GUI or CLI)
- Setup mode (restore from full system backup or initial setup)
- System configuration type (standalone or cluster configuration)
- System name
- Admin password
- Management port IP address and subnet mask
- Default gateway IP address
- DNS Server IP address
- Default domain name

Setup Mode

You can choose to either restore the system configuration from an existing backup file, or manually setup the system by going through the setup wizard. If you choose to restore the system, the backup file must be reachable from the management network.

System Configuration Type

You can configure a Cisco UCS instance to use a single fabric interconnect in a standalone configuration, or use a redundant pair of fabric interconnects in a cluster configuration.

A cluster configuration provides high availability. If one fabric interconnect becomes unavailable, the other automatically takes over. Only one management port (Mgmt0)connection is required to support a cluster configuration; however, both Mgmt0 ports should be connected to provide link-level redundancy.

Note The cluster configuration only provides redundancy for the management plane. Data redundancy is dependent on the user configuration and may require a third-party tool to support data redundancy.

To use the cluster configuration, the two fabric interconnects must be directly connected together using Ethernet cables between the L1 (L1-to-L1) and L2 (L2-to-L2) high availability ports, with no other fabric interconnects in between. This allows the two fabric interconnects to continuously monitor the status of each other and quickly know when one has failed.

Both fabric interconnects in a cluster configuration must go through the initial setup process. The first fabric interconnect to be setup must be enabled for a cluster configuration, then when the second fabric interconnect is setup, it will automatically detect the first fabric interconnect as a peer fabric interconnect in the cluster.

For more information, refer to the *Cisco UCS Hardware Installation Guide* for your fabric interconnect.

Management Port IP Address

In a standalone configuration, you must specify only one IP address and the subnet mask for the single management port on the fabric interconnect.

In a cluster configuration, you must specify the following three IP addresses in the same subnet:

- Management port IP address for fabric interconnect A
- Management port IP address for fabric interconnect B
- Cluster IP address

Performing an Initial System Setup for a Standalone Configuration

Before You Begin

1 Verify the following physical connections on the fabric interconnect:

- The console port is physically connected to a computer terminal or console server.
- The management Ethernet port (mgmt0) is connected to an external hub, switch, or router.

For more information, refer to the *Cisco UCS Hardware Installation Guide* for your fabric interconnect.

2 Verify that the console port parameters on the computer terminal (or console server) attached to the console port are as follows:

- 9600 baud
- 8 data bits
- No parity
- 1 stop bit

3 Collect the following information that you will need to supply during the initial setup:

- System name
- Password for the admin account
- Management port IP address and subnet mask
- Default gateway IP address
- DNS server IP address (optional)
- Domain name for the system (optional)

Procedure

Step 1 Connect to the console port.

Step 2 Power on the fabric interconnect.
You will see the power on self test messages as the fabric interconnect boots.

Step 3 When the unconfigured system boots, it prompts you for the setup method to be used. Enter **console** to continue the initial setup using the console CLI.

Step 4 Enter **setup** to continue as an initial system setup.

Step 5 Enter **y** to confirm that you want to continue the initial setup.

Step 6 Enter the password for the admin account.

Step 7 To confirm, re-enter the password for the admin account.

Step 8 Enter **no** to continue the initial setup for a standalone configuration.

Step 9 Enter the system name.

Step 10 Enter the IP address for the management port on the fabric interconnect.

Step 11 Enter the subnet mask for the management port on the fabric interconnect.

Step 12 Enter the IP address for the default gateway.

Step 13 Enter **yes** if you want to specify the IP address for the DNS server, or **no** if you do not.

Step 14 (Optional) Enter the IP address for the DNS server.

Step 15 Enter **yes** if you want to specify the default domain name, or **no** if you do not.

Step 16 (Optional) Enter the default domain name.

Step 17 Review the setup summary and enter **yes** to save and apply the settings, or enter **no** to go through the setup wizard again to change some of the settings.

If you choose to go through the setup wizard again, it will automatically provide the values you previously entered, and the values will appear in brackets. To accept previously-entered values, press the Enter key.

The following example sets up a standalone configuration using the console:

```
Enter the installation method (console/gui)?  console
Enter the setup mode (restore from backup or initial setup) [restore/setup]? setup
You have chosen to setup a new switch.  Continue? (y/n): y
Enter the password for "admin": adminpassword%958
Confirm the password for "admin": adminpassword%958
Do you want to create a new cluster on this switch (select 'no' for standalone setup or if
 you want this switch to be added to an existing cluster)? (yes/no) [n]: no
Enter the system name: foo
Mgmt0 IPv4 address: 192.168.10.10
Mgmt0 IPv4 netmask: 255.255.255.0
IPv4 address of the default gateway: 192.168.10.1
Configure the DNS Server IPv4 address? (yes/no) [n]: yes
  DNS IPv4 address: 20.10.20.10
Configure the default domain name? (yes/no) [n]: yes
  Default domain name: domainname.com
Following configurations will be applied:
  Switch Fabric=A
  System Name=foo
  Management IP Address=192.168.10.10
  Management IP Netmask=255.255.255.0
  Default Gateway=192.168.10.1
  DNS Server=20.10.20.10
  Domain Name=domainname.com
Apply and save the configuration (select 'no' if you want to re-enter)? (yes/no): yes
```

Initial System Setup for a Cluster Configuration

Performing an Initial System Setup for the First Fabric Interconnect

Before You Begin

1 Verify the following physical connections on the fabric interconnect:

- A console port on the first fabric interconnect is physically connected to a computer terminal or console server.
- The management Ethernet port (mgmt0) is connected to an external hub, switch, or router.
- The L1 ports on both fabric interconnects are directly connected to each other.
- TheL2 ports on both fabric interconnects are directly connected to each other.

For more information, refer to the *Cisco UCS Hardware Installation Guide* for your fabric interconnect.

2 Verify that the console port parameters on the computer terminal (or console server) attached to the console port are as follows:

- 9600 baud
- 8 data bits
- No parity
- 1 stop bit

3 Collect the following information that you will need to supply during the initial setup:

- System name
- Password for the admin account
- Three static IP addresses: two for the management port on both fabric interconnects (one per fabric interconnect), and one for the cluster IP address used by Cisco UCS Manager
- Subnet mask for the three static IP addresses
- Default gateway IP address
- DNS server IP address (optional)
- Domain name for the system (optional)

Procedure

Step 1 Connect to the console port.

Step 2 Power on the fabric interconnect.
You will see the power on self test messages as the fabric interconnect boots.

Step 3 When the unconfigured system boots, it prompts you for the setup method to be used. Enter **console** to continue the initial setup using the console CLI.

Step 4 Enter **setup** to continue as an initial system setup.

Step 5 Enter **y** to confirm that you want to continue the initial setup.

Step 6 Enter the password for the admin account.

Step 7 To confirm, re-enter the password for the admin account.

Step 8 Enter **yes** to continue the initial setup for a cluster configuration.

Step 9 Enter the fabric interconnect fabric (either **A** or **B**).

Step 10 Enter the system name.

Step 11 Enter the IP address for the management port on the fabric interconnect.

Step 12 Enter the subnet mask for the management port on the fabric interconnect.

Step 13 Enter the IP address for the default gateway.

Step 14 Enter the virtual IP address.

Step 15 Enter **yes** if you want to specify the IP address for the DNS server, or **no** if you do not.

Step 16 (Optional) Enter the IP address for the DNS server.

Step 17 Enter **yes** if you want to specify the default domain name, or **no** if you do not.

Step 18 (Optional) Enter the default domain name.

Step 19 Review the setup summary and enter **yes** to save and apply the settings, or enter **no** to go through the setup wizard again to change some of the settings.
If you choose to go through the setup wizard again, it will automatically provide the values you previously entered, and the values will appear in brackets. To accept previously-entered values, press the Enter key.

The following example sets up the first fabric interconnect for a cluster configuration using the console:

```
Enter the installation method (console/gui)?  console
Enter the setup mode (restore from backup or initial setup) [restore/setup]? setup
You have chosen to setup a new switch.  Continue? (y/n): y
Enter the password for "admin": adminpassword%958
Confirm the password for "admin": adminpassword%958
Do you want to create a new cluster on this switch (select 'no' for standalone setup or if
 you want this switch to be added to an existing cluster)? (yes/no) [n]: yes
Enter the switch fabric (A/B): A
Enter the system name: foo
Mgmt0 IPv4 address: 192.168.10.10
Mgmt0 IPv4 netmask: 255.255.255.0
IPv4 address of the default gateway: 192.168.10.1
Virtual IPv4 address : 192.168.10.12
Configure the DNS Server IPv4 address? (yes/no) [n]: yes
  DNS IPv4 address: 20.10.20.10
Configure the default domain name? (yes/no) [n]: yes
  Default domain name: domainname.com
Following configurations will be applied:
  Switch Fabric=A
  System Name=foo
  Management IP Address=192.168.10.10
  Management IP Netmask=255.255.255.0
  Default Gateway=192.168.10.1
  Cluster Enabled=yes
  Virtual Ip Address=192.168.10.12
  DNS Server=20.10.20.10
  Domain Name=domainname.com
Apply and save the configuration (select 'no' if you want to re-enter)? (yes/no): yes
```

Performing an Initial System Setup for the Second Fabric Interconnect

Before You Begin

1 Verify the following physical connections on the fabric interconnect:

- A console port on the second fabric interconnect is physically connected to a computer terminal or console server.
- The management Ethernet port (mgmt0) is connected to an external hub, switch, or router.
- The L1 ports on both fabric interconnects are directly connected to each other.
- TheL2 ports on both fabric interconnects are directly connected to each other.

For more information, refer to the *Cisco UCS Hardware Installation Guide* for your fabric interconnect.

2 Verify that the console port parameters on the computer terminal (or console server) attached to the console port are as follows:

- 9600 baud
- 8 data bits
- No parity
- 1 stop bit

3 Collect the following information that you will need to supply during the initial setup:

- Password for the admin account of the peer fabric interconnect
- Management port IP address in the same subnet as the peer fabric interconnect

Procedure

Step 1 Connect to the console port.

Step 2 Power on the fabric interconnect.
You will see the power on self test messages as the fabric interconnect boots.

Step 3 When the unconfigured system boots, it prompts you for the setup method to be used. Enter **console** to continue the initial setup using the console CLI.

Note The fabric interconnect should detect the peer fabric interconnect in the cluster. If it does not, check the physical connections between the L1 and L2 ports, and verify that the peer fabric interconnect has been enabled for a cluster configuration.

Step 4 Enter **y** to add the subordinate fabric interconnect to the cluster.

Step 5 Enter the admin password of the peer fabric interconnect.

Step 6 Enter the IP address for the management port on the subordinate fabric interconnect.

Step 7 Review the setup summary and enter **yes** to save and apply the settings, or enter **no** to go through the setup wizard again to change some of the settings.
If you choose to go through the setup wizard again, it will automatically provide the values you previously entered, and the values will appear in brackets. To accept previously-entered values, press the Enter key.

The following example sets up the second fabric interconnect for a cluster configuration using the console:

```
Enter the installation method (console/gui)?  console
Installer has detected the presence of a peer switch. This switch will be added to the
cluster. Continue?[y/n] y
Enter the admin password of the peer switch: adminpassword%958
Mgmt0 IPv4 address: 192.168.10.11
Management Ip Address=192.168.10.11
Apply and save the configuration (select 'no' if you want to re-enter)? (yes/no): yes
```

Enabling a Standalone Fabric Interconnect for Cluster Configuration

You can add a second fabric interconnect to an existing Cisco UCS instance that uses a single standalone fabric interconnect. To do this, you must enable the standalone fabric interconnect for cluster operation, and then add the second fabric interconnect to the cluster.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **connect local-mgmt**	Enters local management mode.
Step 2	UCS-A(local-mgmt) # **enable cluster** *ip-addr*	Enables cluster operation on the standalone fabric interconnect with the specified IP address. When you enter this command, you are prompted to confirm thatyou want to enable cluster operation. Type yes to confirm.

The following example enables a standalone fabric interconnect with IP address 192.168.1.101 for cluster operation:

```
UCS-A# connect local-mgmt
UCS-A(local-mgmt)# enable cluster 192.168.1.101
```

```
This command will enable cluster mode on this setup. You cannot change it
back to stand-alone. Are you sure you want to continue? (yes/no): yes
UCS-A(local-mgmt)#
```

What to Do Next

Add the second fabric interconnect to the cluster.

Ethernet Switching Mode

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

End-Host Mode

End-host mode allows the fabric interconnect to act as an end host to the network, representing all server (hosts) connected to it through vNICs. This is achieved by pinning (either dynamically pinned or hard pinned) vNICs to uplink ports, which provides redundancy toward the network, and makes the uplink ports appear as server ports to the rest of the fabric. When in end-host mode, the fabric interconnect does not run the Spanning Tree Protocol (STP) and avoids loops by denying uplink ports from forwarding traffic to each other, and by denying egress server traffic on more than one uplink port at a time. End-host mode is the default Ethernet switching mode and should be used if either of the following are used upstream:

- Layer 2 switching for L2 Aggregation
- Virtual Switching System (VSS) aggregation layer

Note When end-host mode is enabled, if a vNIC is hard pinned to an uplink port and this uplink port goes down, the system cannot re-pin the vNIC, and the vNIC remains down

Switch Mode

Switch mode is the traditional Ethernet switching mode. The fabric interconnect runs STP to avoid loops, and broadcast and multicast packets are handled in the traditional way. Switch mode is not the default Ethernet switching mode, and should be used only if the fabric interconnect is directly connected to a router, or if either of the following are used upstream:

- Layer 3 aggregation
- VLAN in a box

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

Configuring Ethernet Switching Mode

Important When you change the Ethernet switching mode, Cisco UCS Manager logs you out and restarts the fabric interconnect. For a cluster configuration, Cisco UCS Manager restarts both fabric interconnects sequentially.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope eth-uplink**	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # **set mode** {**end-host** | **switch**}	Sets the fabric interconnect to the specified switching mode.

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

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # set mode end-host
```

CHAPTER 5

Configuring Ports

This chapter includes:

Server and Uplink Ports on the Fabric Interconnect

Each fabric interconnect has a set of ports in a fixed port module that you can configure as either server ports or uplink Ethernet ports. None of these ports are reserved. They cannot be used by a Cisco UCS instance until you configure them. You can add expansion modules to increase the number of uplink ports on the fabric interconnect, or to add uplink Fibre Channel ports to the fabric interconnect.

You need to create LAN pin groups and SAN pin groups to pin traffic from servers to an uplink port.

Each fabric interconnect can include the following types of ports:

Server Ports

Server ports handle data traffic between the fabric interconnect and the adapter cards on the servers.

You can only configure server ports on the fixed port module. Expansion modules do not include server ports.

Uplink Ethernet Ports

Uplink Ethernet ports handle Ethernet traffic between the fabric interconnect and the next layer of the network. All network-bound Ethernet traffic is pinned to one of these ports.

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

Uplink Fibre Channel Ports

Uplink Fibre Channel ports handle FCoE traffic between the fabric interconnect and the next layer of the network. All network-bound FCoE traffic is pinned to one of these ports.

You can only configure uplink Fibre Channel ports on an expansion module. The fixed module does not include uplink Fibre Channel ports.

Server Ports

Configuring a Server Port

You can only configure server ports on the fixed port module. Expansion modules do not include server ports.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope eth-server**	Enters Ethernet server mode.
Step 2	UCS-A /eth-server # **scope fabric-interconnect** {**a** | **b**}	Enters Ethernet server fabric interconnect mode for the specified fabric-interconnect.
Step 3	UCS-A /eth-server/fabric-interconnect # **create interface** *slot-num port-num*	Creates an interface for the specified Ethernet server port.

The following example creates an interface for Ethernet server port 35 on slot 3 of fabric-interconnect B.

```
UCS-A# scope eth-server
UCS-A /eth-server # scope fabric-interconnect b
UCS-A /eth-server/fabric-interconnect # create interface a 1 12
```

Deleting a Server Port

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope eth-server**	Enters Ethernet server mode.
Step 2	UCS-A /eth-server # **scope fabric-interconnect** {**a** | **b**}	Enters Ethernet server fabric interconnect mode for the specified fabric-interconnect.
Step 3	UCS-A /eth-server/fabric-interconnect # **delete interface** *slot-num port-num*	Deletes the interface for the specified Ethernet server port.

The following example deletes the interface for Ethernet server port 35 on slot 3 of fabric-interconnect B.

```
UCS-A# scope eth-server
UCS-A /eth-server # scope fabric-interconnect b
UCS-A /eth-server/fabric-interconnect # delete interface 1 12
```

Uplink Ethernet Ports

Configuring an Uplink Ethernet Port

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

Procedure

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

The following example creates an interface for Ethernet uplink port 3 on slot 2 of fabric interconnect B.

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # scope fabric-interconnect B
UCS-A /eth-uplink/fabric-interconnect # create interface 2 3
UCS-A /eth-uplink/fabric-interconnect* # commit-buffer
UCS-A /eth-uplink/fabric-interconnect #
```

Deleting an Uplink Ethernet Port

Procedure

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

The following example deletes the interface for Ethernet uplink port 3 on slot 2 of fabric-interconnect B.

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

Uplink Ethernet Port Channels

Configuring an Uplink Ethernet Port Channel

Procedure

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

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

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink* # scope fabric-interconnect a
UCS-A /eth-uplink/fabric-interconnect* # create port-channel 13
UCS-A /eth-uplink/fabric-interconnect/port-channel* # set name portchan13a
UCS-A /eth-uplink/fabric-interconnect/port-channel* # enable
UCS-A /eth-uplink/fabric-interconnect/port-channel* # set flow-control-policy flow-con-pol432
UCS-A /eth-uplink/fabric-interconnect/port-channel* # commit-buffer
UCS-A /eth-uplink/fabric-interconnect/port-channel #
```

Deleting an Uplink Ethernet Port Channel

Procedure

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

The following example deletes the port channel on port 13 of fabric interconnect A.

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

Adding a Member Port to an Uplink Ethernet Port Channel

Procedure

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

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

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

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

Deleting a Member Port to an Uplink Ethernet Port Channel

Procedure

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

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

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

CHAPTER 6

Configuring Communication Services

This chapter includes:

Communication Services

You can use the following communication services to interface third-party applications with Cisco UCS:

Communication Service	Description
CIM XML	This service is disabled by default and is only available in read-only mode. If enabled, the default port is 5988. This common information model is one of the standards defined by the Distributed Management Task Force.
HTTP	This service is enabled on port 80 by default. You must enable either HTTP or HTTPS to run Cisco UCS Manager GUI. If you select HTTP, all data is exchanged in clear text mode. For security purposes, we recommend that you enable HTTPS and disable HTTP.
HTTPS	This service is enabled on port 443 by default. You must enable either HTTP or HTTPS to run Cisco UCS Manager GUI. If you select HTTPS, all data is exchanged in encrypted mode through a secure server.

Communication Service	Description
	For security purposes, we recommend that you enable HTTPS and disable HTTP.
SMASH CLP	This service is enabled for read-only access and supports a limited subset of the protocols, such as the show command. You cannot disable it. This shell service is one of the standards defined by the Distributed Management Task Force.
SNMP	This service is disabled by default. If enabled, the service uses port 161. You must configure the community and at least one SNMP trap. Only enable this service if your system includes integration with an SNMP server.
SSH	This service is enabled on port 22. You cannot disable it, nor can you change the default port. This service provides access to the Cisco UCS Manager CLI.
Telnet	This service is disabled by default. This service provides access to the Cisco UCS Manager CLI.

Configuring CIM XML

Note Cisco recommends that you enable only the communication services that are required to interface with other network applications.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system**	Enters system mode.
Step 2	UCS-A /system # **scope services**	Enters system services mode.
Step 3	UCS-A /system/services # **enable cimxml**	Enables the CIM XLM service.
Step 4	UCS-A /system/services # **set cimxml port** *port-num*	Specifies the port to be used for the CIM XML connection.
Step 5	UCS-A /system/services # **commit-buffer**	Commits the transaction to the system configuration.

The following example enables CIM XML, sets the port number to 5988, and commits the transaction:

```
UCS-A# scope system
UCS-A /system # scope services
UCS-A /system/services # enable cimxml
UCS-A /system/services* # set cimxml port 5988
```

```
UCS-A /system/services* # commit-buffer
UCS-A /system/services #
```

Configuring HTTP

Note Cisco recommends that you enable only the communication services that are required to interface with other network applications.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system**	Enters system mode.
Step 2	UCS-A /system # **scope services**	Enters system services mode.
Step 3	UCS-A /system/services # **enable http**	Enables the HTTP service.
Step 4	UCS-A /system/services # **set http port** *port-num*	Specifies the port to be used for the HTTP connection.
Step 5	UCS-A /system/services # **commit-buffer**	Commits the transaction to the system configuration.

The following example enables HTTP, sets the port number to 80, and commits the transaction:

```
UCS-A# scope system
UCS-A /system # scope services
UCS-A /system/services # enable http
UCS-A /system/services* # set http port 80
UCS-A /system/services* # commit-buffer
UCS-A /system/services #
```

Configuring HTTPS

Note Cisco recommends that you enable only the communication services that are required to interface with other network applications.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system**	Enters system mode.
Step 2	UCS-A /system # **scope services**	Enters system services mode.
Step 3	UCS-A /system/services # **enable https**	Enables the HTTPS service.

	Command or Action	Purpose
Step 4	UCS-A /system/services # **set https port** *port-num*	Specifies the port to be used for the HTTPS connection.
Step 5	UCS-A /system/services # **set https keyring** *keyring-name*	Specifies the name for the HTTPS keyring. **Caution** When the HTTPS keyring is modified using the **set https keyring** command, all current HTTP and HTTPS sessions will be closed without any warning.
Step 6	UCS-A /system/services # **commit-buffer**	Commits the transaction to the system configuration.

The following example enables HTTPS, sets the port number to 443, sets the key ring name to kring7984, and commits the transaction:

```
UCS-A# scope system
UCS-A /system # scope services
UCS-A /system/services # enable https
UCS-A /system/services* # set https port 443
UCS-A /system/services* # set https keyring kring7984
UCS-A /system/services* # commit-buffer
UCS-A /system/services #
```

Configuring SNMP

Note Cisco recommends that you enable only the communication services that are required to interface with other network applications.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope monitoring**	Enters monitoring mode.
Step 2	UCS-A /monitoring # **set snmp community** *community-name*	Creates the specified SNMP community name. The community name can be any alphanumeric string up to 32 characters.
Step 3	UCS-A /monitoring # **create snmp-trap** *trap-name*	Creates the specified SNMP trap.
Step 4	UCS-A /snmp-trap # **set community** *community-name*	Specifies the SNMP community name to be used for the SNMP trap.
Step 5	UCS-A /snmp-trap # **set port** *port-num*	Specifies the port to be used for the SNMP trap.
Step 6	UCS-A /snmp-trap # **commit-buffer**	Commits the transaction to the system configuration.

The following example creates the community name, SnmpCommSystem2, creates an SNMP trap named sys-trap2, specifies that the trap will use the SnmpCommSystem2 community on port 2, and commits the transaction:

```
UCS-A# scope monitoring
UCS-A /monitoring # set snmp community SnmpCommSystem2
UCS-A /monitoring* # create snmp-trap sys-trap2
UCS-A /monitoring/snmp-trap* # set community SnmpCommSystem2
UCS-A /monitoring/snmp-trap* # set port 2
UCS-A /monitoring/snmp-trap* # commit-buffer
UCS-A /monitoring/snmp-trap #
```

Configuring Telnet

Note Cisco recommends that you enable only the communication services that are required to interface with other network applications.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system**	Enters system mode.
Step 2	UCS-A /system # **scope services**	Enters system services mode.
Step 3	UCS-A /services # **enable telnet-server**	Enables the Telnet service.
Step 4	UCS-A /services # **commit-buffer**	Commits the transaction to the system configuration.

The following example enables Telnet and commits the transaction:

```
UCS-A# scope system
UCS-A /system # scope services
UCS-A /services # enable telnet-server
UCS-A /services* # commit-buffer
UCS-A /services #
```

Disabling Communication Services

Note Cisco recommends that you disable all communication services that are not required to interface with other network applications.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system**	Enters system mode.
Step 2	UCS-A /system # **scope services**	Enters system services mode.

	Command or Action	Purpose
Step 3	UCS-A /system/services # **disable** *service-name*	Disables the specified service, where the *service-name* argument is one of the following keywords: • **cimxml**—Disables CIM XML service • **http**—Disables HTTP service • **https**—Disables HTTPS service • **telnet-server**—Disables Telnet service
Step 4	UCS-A /system/services # **commit-buffer**	Commits the transaction to the system configuration.

The following example disables CIM XML and commits the transaction:

```
UCS-A# scope system
UCS-A# scope services
UCS-A /system/services # disable cimxml
UCS-A /system/services* # commit-buffer
UCS-A /system/services #
```

CHAPTER 7

Configuring Primary Authentication

This chapter includes:

Primary Authentication

Cisco UCS supports two methods to authenticate user logins:

- Local to Cisco UCS Manager
- Remote through one of the following protocols:
    - LDAP
    - RADIUS
    - TACACS+

Note You can only use one authentication method. For example, if you select LDAP as your authentication provider, you cannot use local, RADIUS, or TACACS+ for authentication.

Remote Authentication Providers

If a system is configured for one of the supported remote authentication services, you must create a provider for that service to ensure that Cisco UCS Manager can communicate with it. In addition, you need to be aware of the following guidelines that impact user authorization:

User Accounts in Remote Authentication Services

You can create user accounts in Cisco UCS Manager or in the remote authentication server.

The temporary sessions for users who log in through remote authentication services can be viewed under **Remotely Authenticated Users** in the following location on the **Admin tab**: **All ➤ User Management ➤ User Services** .

User Roles in Remote Authentication Services

If you create user accounts in the remote authentication server, you must ensure that the accounts include the roles those users require for working in Cisco UCS Manager and that the names of those roles match the names used in Cisco UCS Manager. If an account does not have the required roles, the user will have only read-only privileges.

The following table contains the name of the attribute that contains the value of the roles. Cisco UCS Manager checks for the value of this attribute when it queries the remote authentication service during login.

Remote Authentication Protocol	Attribute Name
LDAP	CiscoAvPair
RADIUS	cisco-av-pair
TACACS+	cisco-av-pair

Creating a Remote Authentication Provider

Creating an LDAP Provider

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **scope ldap**	Enters security LDAP mode.
Step 3	UCS-A /security/ldap # **set attribute** *attribute*	Restricts database searches to records that contain the specified attribute.
Step 4	UCS-A /security/ldap # **set basedn** *distinguished-name*	Restricts database searches to records that contain the specified distinguished name.
Step 5	UCS-A /security/ldap # **set filter** *filter*	Restricts database searches to records that contain the specified filter.
Step 6	UCS-A /security/ldap # **set timeout** *seconds*	(Optional) Sets the time interval the system waits for a response from the LDAP server before noting the server as down.

	Command or Action	Purpose
Step 7	UCS-A /security/ldap # **create server** *server-name*	Creates an LDAP server instance and enters security LDAP server mode
Step 8	UCS-A /security/ldap/server # **set ssl** {**yes** | **no**}	Enables or disables the use of SSL when communicating with the LDAP server.
Step 9	UCS-A /security/ldap/server # **set key**	(Optional) Sets the LDAP server key. To set the key value, press Return after typing the **set key** command and enter the key value at the prompt.
Step 10	UCS-A /security/ldap/server # **set port** *port-num*	Specifies the port used to communicate with the LDAP server.
Step 11	UCS-A /security/ldap/server # **set rootdn** *root-dist-name*	Specifies the distinguished name for the LDAP database superuser account.

The following example sets the LDAP attribute to CiscoAvPair, the base distinguished name to "DC=nuova-sam-aaa3,DC=qalab,DC=com", the filter to sAMAccountName=$userid, the timeout interval to 5 seconds, creates a server instance named 10.193.169.246, disables SSL, sets the key, sets the authentication port to 389, and sets the root distinguished name to "cn=Administrator,cn=Users,DC=nuova-sam-aaa3,DC=qalab,DC=com":

```
UCS-A# scope security
UCS-A /security # scope ldap
UCS-A /security/ldap # set attribute CiscoAvPair
UCS-A /security/ldap* # set basedn "DC=cisco-ucsm-aaa3,DC=qalab,DC=com"
UCS-A /security/ldap* # set filter sAMAccountName=$userid
UCS-A /security/ldap* # set timeout 5
UCS-A /security/ldap* # create server 10.193.169.246
UCS-A /security/ldap/server* # set ssl no
UCS-A /security/ldap/server* # set key
Enter the key:
Confirm the key:
UCS-A /security/ldap/server* # set port 389
UCS-A /security/ldap/server* # set rootdn
"cn=Administrator,cn=Users,DC=cisco-ucsm-aaa3,DC=qalab,DC=com"
UCS-A /security/ldap/server* # commit-buffer
UCS-A /security/ldap/server #
```

Creating a RADIUS Provider

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **scope radius**	Enters security RADIUS mode.
Step 3	UCS-A /security/radius # **set retries** *retry-num*	(Optional) Sets the number of times to retry communicating with the RADIUS server before noting the server as down.

	Command or Action	Purpose
Step 4	UCS-A /security/radius # **set timeout** *seconds*	(Optional) Sets the time interval the system waits for a response from the RADIUS server before noting the server as down.
Step 5	UCS-A /security/radius # **create server** *server-name*	Creates a RADIUS server instance and enters security RADIUS server mode
Step 6	UCS-A /security/radius/server # **set authport** *authport-num*	Specifies the port used to communicate with the RADIUS server.
Step 7	UCS-A /security/radius/server # **set key**	(Optional) Sets the RADIUS server key. To set the key value, press Return after typing the **set key** command and enter the key value at the prompt.

The following example sets the RADIUS retries to 4, the timeout interval to 30 seconds, creates a server instance named radiusserv7, sets the authentication port to 5858, and sets the key to radiuskey321:

```
UCS-A# scope security
UCS-A /security # scope radius
UCS-A /security/radius # set retries 4
UCS-A /security/radius # set timeout 30
UCS-A /security/radius # create server radiusserv7
UCS-A /security/radius/server # set authport 5858
UCS-A /security/radius/server # set key
Enter the key: radiuskey321
Confirm the key: radiuskey321
```

Creating a TACACS+ Provider

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **scope tacacs**	Enters security TACACS+ mode.
Step 3	UCS-A /security/tacacs # **set timeout** *seconds*	(Optional) Sets the time interval the system waits for a response from the TACACS+ server before noting the server as down.
Step 4	UCS-A /security/tacacs # **create server** *server-name*	Creates an TACACS+ server instance and enters security TACACS+ server mode
Step 5	UCS-A /security/tacacs/server # **set key**	(Optional) Sets the TACACS+ server key. To set the key value, press Return after typing the **set key** command and enter the key value at the prompt.

	Command or Action	Purpose
Step 6	UCS-A /security/tacacs/server # **set port** *port-num*	Specifies the port used to communicate with the TACACS+ server.

The following example sets the TACACS+ timeout interval to 45 seconds, creates a server instance named tacacsserv680, sets the key to tacacskey321, and the authentication port to 5859:

```
UCS-A# scope security
UCS-A /security # scope tacacs
UCS-A /security/tacacs # set timeout 45
UCS-A /security/tacacs # create server tacacsserv680
UCS-A /security/tacacs/server # set key
Enter the key: tacacskey321
Confirm the key: tacacskey321
UCS-A /security/tacacs/server # set port 5859
```

Selecting a Primary Authentication Service

Selecting the Console Authentication Service

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **set authentication console** *auth-type*	Specifies the console authentication, where the *auth-type* argument is one of the following keywords: • **ldap**—Specifies LDAP authentication • **local**—Specifies local authentication • **radius**—Specifies RADIUS authentication • **tacacs**—Specifies TACACS+ authentication

The following example sets the console to use local authentication:

```
UCS-A# scope security
UCS-A /security # set authentication console local
```

Selecting the Default Authentication Service

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **set authentication default** *auth-type*	Specifies the default authentication, where the *auth-type* argument is one of the following keywords:

	Command or Action	Purpose
		• **ldap**—Specifies LDAP authentication • **local**—Specifies local authentication • **radius**—Specifies RADIUS authentication • **tacacs**—Specifies TACACS+ authentication

The following example sets the default authentication to LDAP:

```
UCS-A# scope security
UCS-A /security # set authentication default ldap
```

CHAPTER 8

Configuring Organizations

This chapter includes the following sections:

Organizations in a Multi-Tenancy Environment

In Cisco UCS, you can use multi-tenancy to divide up the large physical infrastructure of an instance into logical entities known as organizations. As a result, you can achieve a logical isolation between organizations without providing a dedicated physical infrastructure for each organization.

You can assign unique resources to each tenant through the related organization, in the multi-tenant environment. These resources can include different policies, pools, and quality of service definitions. You can also implement locales to assign or restrict Cisco UCS user privileges and roles by organization, if you do not want all users to have access to all organizations.

If you set up a multi-tenant environment, all organizations are hierarchical. The top-level organization is always root. The policies and pools that you create in root are system-wide and are available to all organizations in the system. However, any policies and pools created in other organizations are only available to organizations that are above it the same hierarchy. For example, if a system has organizations named Finance and HR that are not in the same hierarchy, Finance cannot use any policies in the HR organization, and HR cannot access any policies in the Finance organization. However, both Finance and HR can use policies and pools in the root organization.

If you create organizations in a multi-tenant environment, you can also set up one or more of the following for each organization or for a sub-organization in the same hierarchy:

- Resource pools
- Policies
- Service profiles

- Service profile templates

The root organization is always the top level organization.

Hierarchical Name Resolution in a Multi-Tenancy Environment

In a multi-tenant environment, Cisco UCS uses the hierarchy of an organization to resolve the names of policies and resource pools. When Cisco UCS Manager searches for details of a policy or a resource assigned to a pool, the following occurs:

1. Cisco UCS Manager checks for policies and pools with the specified name within the organization assigned to the service profile or policy.
2. If a policy is found or an available resource is inside a pool, Cisco UCS Manager uses that policy or resource. If the pool does not have any available resources at the local level, Cisco UCS Manager moves up in the hierarchy to the parent organization and searches for a pool with the same name. Cisco UCS Manager repeats this step until the search reaches the root organization.
3. If the search reaches the root organization and has not found an available resource or policy, Cisco UCS Manager returns to the local organization and begins to search for a default policy or available resource in the default pool.
4. If an applicable default policy or available resource in a default pool is found, Cisco UCS Manager uses that policy or resource. If the pool does not have any available resources, Cisco UCS Manager moves up in the hierarchy to the parent organization and searches for a default pool. Cisco UCS Manager repeats this step until the search reaches the root organization.
5. If Cisco UCS Manager cannot find an applicable policy or available resource in the hierarchy, it returns an allocation error.

Example: Server Pool Name Resolution in a Single-Level Hierarchy

In this example, all organizations are at the same level below the root organization. For example, a service provider creates separate organizations for each customer. In this configuration, organizations only have access to the policies and resource pools assigned to that organization and to the root organization.

In this example, a service profile in the XYZcustomer organization is configured to use servers from the XYZcustomer server pool. When resource pools and policies are assigned to the service profile, the following occurs:

1. Cisco UCS Manager checks for an available server in the XYZcustomer server pool.
2. If the XYZcustomer server pool has an available server, Cisco UCS Manager associates that server with the service profile and discontinues the search. If the pool does not have an available server, Cisco UCS Manager checks the root organization for a server pool with the same name.
3. If the root organization includes an XYZcustomer server pool and that pool has an available server, Cisco UCS Manager associates that server with the service profile and discontinues the search. If the pool does not have an available server, Cisco UCS Manager returns to the XYZcustomer organization to check the default server pool.
4. If the default pool in the XYZcustomer organization has an available server, Cisco UCS Manager associates that server with the service profile and discontinues the search. If the default pool does not have an available server, Cisco UCS Manager checks the default server pool in the root organization.

5 If the default server pool in the root organization has an available server, Cisco UCS Manager associates that server with the service profile and discontinues the search. If the default pool does not have an available server, Cisco UCS Manager returns an allocation error.

Example: Server Pool Name Resolution in a Multi-Level Hierarchy

In this example, each organization includes at least one suborganization. For example, a company could create organizations for each major division in the company and for subdivisions of those divisions. In this configuration, each organization has access to its local policies and resource pools and to the resource pools in the parent hierarchy.

In this example, the Finance organization includes two sub-organizations, AccountsPayable and AccountsReceivable. A service profile in the AccountsPayable organization is configured to use servers from the AP server pool. When resource pools and policies are assigned to the service profile, the following occurs:

1 Cisco UCS Manager checks for an available server in the AP server pool defined in the service profile.

2 If the AP server pool has an available server, Cisco UCS Manager associates that server with the service profile and discontinues the search. If the pool does not have an available server, Cisco UCS Manager moves one level up the hierarchy and checks the Finance organization for a pool with the same name.

3 If the Finance organization includes a pool with the same name and that pool has an available server, Cisco UCS Manager associates that server with the service profile and discontinues the search. If the pool does not have an available server, Cisco UCS Manager moves one level up in the hierarchy and checks the root organization for a pool with the same name.

4 If the root organization includes a pool with the same name and that pool has an available server, Cisco UCS Manager associates that server with the service profile and discontinues the search. If the pool does not have an available server, Cisco UCS Manager returns to the AccountsPayable organization to check the default server pool.

5 If the default pool in the AccountsPayable organization has an available server, Cisco UCS Manager associates that server with the service profile and discontinues the search. If the default pool does not have an available server, Cisco UCS Manager moves one level up in the hierarchy and checks the default server pool in the Finance organization.

6 If the default pool in the Finance organization has an available server, Cisco UCS Manager associates that server with the service profile and discontinues the search. If the default pool does not have an available server, Cisco UCS Manager moves one level up in the hierarchy and checks the default server pool in the root organization.

7 If the default server pool in the root organization has an available server, Cisco UCS Manager associates that server with the service profile and discontinues the search. If the default pool does not have an available server, Cisco UCS Manager returns an allocation error.

Configuring an Organization Under the Root Organization

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 # **create org** *org-name*	Creates the specified organization under the root organization and enters organization mode for the specified organization. **Note** When you move from one organization mode to another, the command prompt does not change.
Step 3	UCS-A /org # **commit-buffer**	Commits the transaction to the system configuration.

The following example creates an organization named Finance under the root organization:

```
UCS-A# scope org /
UCS-A /org* # create org Finance
UCS-A /org* # commit-buffer
UCS-A /org #
```

Configuring an Organization Under an Organization that is not Root

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org /**	Enters the root organization mode.
Step 2	UCS-A /org # **scope org** *org-name*	Enters organization mode for the specified organization. **Note** When you move from one organization mode to another, the command prompt does not change.
Step 3	UCS-A /org # **create org** *org-name*	Creates the specified organization under the previously configured non-root organization and enters organization mode for the specified organization.
Step 4	UCS-A /org # **commit-buffer**	Commits the transaction to the system configuration.

The following example creates an organization named Finance under the NorthAmerica organization:

```
UCS-A# scope org /
UCS-A /org* # scope org NorthAmerica
UCS-A /org* # create org Finance
UCS-A /org* # commit-buffer
UCS-A /org #
```

Deleting an Organization

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org /**	Enters the root organization mode.
Step 2	UCS-A /org # **delete org** *org-name*	Deletes the specified organization.

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

The following example deletes the organization under the root organization named Finance:

```
UCS-A# scope org /
UCS-A /org* # delete org Finance
UCS-A /org* # commit-buffer
UCS-A /org #
```

CHAPTER 9

Configuring Role-Based Access Control

This chapter includes:

Role-Based Access Control

Role-Based Access Control (RBAC) is a method of restricting or authorizing system access for users based on user roles and locales. A role defines the privileges of a user in the system and the locale defines the organizations (domains) that a user is allowed access. Because users are not directly assigned privileges, management of individual user privileges is simply a matter of assigning the appropriate roles and locales.

A user is granted write access to desired system resources only if the assigned role grants the access privileges and the assigned locale allows access. For example, a user with the Server Administrator role in the Engineering organization could update server configurations in the Engineering organization, but would not be able to update server configurations in the Finance organization unless the locales assigned to the user include the Finance organization.

User Accounts

User accounts are used to access the system. Up to 48 user accounts can be configured in each Cisco UCS instance. Each user account must have a unique user name and password.

The system has a default user account, admin, which cannot be modified or deleted. This account is the system administrator or superuser account and has full privileges. There is no default password assigned to the admin account; you must choose the password during the initial system setup.

Each user account must have a unique user name that is not all-numeric and does not start with a number. If an all-numeric user name exists on an AAA server (RADIUS or TACACS+) and is entered during login, the user is not logged in. Local users with all numeric names cannot be created.

For authentication purposes, a password is required for each user account. To prevent users from choosing insecure passwords, each password must meet the following requirements:

- At least eight characters long
- Does not contain more than three consecutive characters, such as abcd
- Does not contain more than two repeating characters, such as aaabbb
- Does not contain dictionary words
- Does not contain common proper names

A user account can also be set with a SSH public key. The public key can be set in one of the two formats: OpenSSH and SECSH.

User accounts can be configured to expire at a predefined time. When the expiration time is reached the user account is disabled. By default, user accounts do not expire.

User Roles

User roles contain one or more privileges that define the operations allowed for the user who is assigned the role. A user can be assigned one or more roles. A user assigned multiple roles has the combined privileges of all assigned roles. For example, if Role1 has storage related privileges, and Role2 has server related privileges, then users who are assigned to both Role1 and Role2 have storage and server related privileges.

All roles include read access to all configuration on the system. The difference between the read-only role and other roles is that a user who is only assigned the read-only role cannot modify the system state. A user assigned another role can modify the system state in that user's assigned area or areas.

The system contains the following default user roles:

AAA Administrator

Read-and-write access to users, roles, and AAA configuration. Read access to the rest of the system.

Administrator

Complete read-and-write access to the entire system. The default admin account is assigned this role by default and it cannot be changed.

Network Administrator

Read-and-write access to fabric interconnect infrastructure and network security operations. Read access to the rest of the system.

Operations

Read-and-write access to systems logs, including the syslog servers, and faults. Read access to the rest of the system.

Read-Only

Read-only access to system configuration with no privileges to modify the system state.

Server Equipment Administrator

Read-and-write access to physical server related operations. Read access to the rest of the system.

Server Profile Administrator

Read-and-write access to logical server related operations. Read access to the rest of the system.

Server Security Administrator	Read-and-write access to server security related operations. Read access to the rest of the system.
Storage Administrator	Read-and-write access to storage operations. Read access to the rest of the system.

Roles can be created, modified to add new or remove existing privileges, or deleted. When a role is modified, the new privileges are applied to all users assigned to that role. Privilege assignment is not restricted to the privileges defined for the default roles. That is, you can use a custom set of privileges to create a unique role. For example, the default Server Administrator and Storage Administrator roles have different set of privileges, but a new Server and Storage Administrator role can be created that combines the privileges of both roles.

If a role is deleted after it has been assigned to users, it is also deleted from those user accounts.

User profiles on AAA servers (RADIUS or TACACS+) should be modified to add the roles corresponding to the privileges granted to that user. The cisco-av-pair vendor-specific attribute is used to store the role information. The AAA servers return this attribute with the request and parse it to get the roles. LDAP servers return the roles in the user profile attributes.

Privileges

Privileges give users assigned to user roles access to specific system resources and permission to perform specific tasks. The following table lists each privilege and the user role given that privilege by default.

Table 2: Privileges and Default Role Assignments

Privilege	Description	Default Role Assignment
aaa	System security and AAA	AAA Administrator
admin	System administration	Administrator
ext-lan-config	External LAN configuration	Network Administrator
ext-lan-policy	External LAN policy	Network Administrator
ext-lan-qos	External LAN QoS	Network Administrator
ext-lan-security	External LAN security	Network Administrator
ext-san-config	External SAN configuration	Storage Administrator
ext-san-policy	External SAN policy	Storage Administrator
ext-san-qos	External SAN QoS	Storage Administrator
ext-san-security	External SAN security	Storage Administrator
fault	Alarms and alarm policies	Operations
operations	Logs and Smart Call Home	Operations

Privilege	Description	Default Role Assignment
pod-config	Pod configuration	Network Administrator
pod-policy	Pod policy	Network Administrator
pod-qos	Pod QoS	Network Administrator
pod-security	Pod security	Network Administrator
read-only	Read-only access Read-only is not a selectable privilege; it is always present.	Read-Only
server-equipment	Server hardware management	Server Equipment Administrator
server-maintenance	Server maintenance	Server Equipment Administrator
server-policy	Server policy	Server Equipment Administrator
server-security	Server security	Server Security Administrator
service-profile-config	Service profile configuration	Server Profile Administrator
service-profile-config-policy	Service profile configuration policy	Server Profile Administrator
service-profile-ext-access	Service profile end point access	Server Profile Administrator
service-profile-network	Service profile network	Network Administrator
service-profile-network-policy	Service profile network policy	Network Administrator
service-profile-qos	Service profile QoS	Network Administrator
service-profile-qos-policy	Service profile QoS policy	Network Administrator
service-profile-security	Service profile security	Server Security Administrator
service-profile-security-policy	Service profile security policy	Server Security Administrator
service-profile-server	Service profile server management	Server Security Administrator
service-profile-server-policy	Service profile pool policy	Server Security Administrator
service-profile-storage	Service profile storage	Storage Administrator
service-profile-storage-policy	Service profile storage policy	Storage Administrator

User Locales

A user can be assigned one or more locales. Each locale defines one or more organizations (domains) the user is allowed access, and access would be limited to the organizations specified in the locale. One exception to this rule is a locale without any organizations, which gives unrestricted access to system resources in all organizations.

Users with AAA Administrator privileges (AAA Administrator role) can assign organizations to the locale of other users. The assignment of organizations is restricted to only those in the locale of the user assigning the organizations. For example, if a locale contains only the Engineering organization then a user assigned that locale can only assign the Engineering organization to other users.

You can hierarchically manage organizations. A user that is assigned at a top level organization has automatic access to all organizations under it. For example, an Engineering organization can contain a Software Engineering organization and a Hardware Engineering organization. A locale containing only the Software Engineering organization has access to system resources only within that organization; however, a locale that contains the Engineering organization has access to the resources for both the Software Engineering and Hardware Engineering organizations.

Configuring User Roles

Creating a User Role

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **create role** *name*	Creates the user role and enters security role mode.
Step 3	UCS-A /security/role # **add privilege** *privilege-name*	Adds one or more privileges to the role. **Note** You can specify more than one *privilege-name* on the same command line to add multiple privileges to the role, or you can add privileges to the same role using multiple **add** commands.
Step 4	UCS-A /security/role # **commit-buffer**	Commits the transaction to the system configuration.

The following example creates the service-profile-security-admin role, adds the service profile security and service profile security policy privileges to the role, and commits the transaction:

```
UCS-A# scope security
UCS-A /security # create role ls-security-admin
UCS-A /security/role* # add privilege service-profile-security service-profile-security-policy
UCS-A /security/role* # commit-buffer
UCS-A /security/role #
```

Adding Privileges to a User Role

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **scope role** *name*	Enters security role mode for the specified role.
Step 3	UCS-A /security/role # **add privilege** *privilege-name*	Adds one or more privileges to the existing privileges of the user role. **Note** You can specify more than one *privilege-name* on the same command line to add multiple privileges to the role, or you can add privileges to the same role using multiple **add privilege** commands.
Step 4	UCS-A /security/role # **commit-buffer**	Commits the transaction to the system configuration.

The following example adds the server security and server policy privileges to the service-profile-security-admin role and commits the transaction:

```
UCS-A# scope security
UCS-A /security # scope role service-profile-security-admin
UCS-A /security/role # add privilege server-security server-policy
UCS-A /security/role* # commit-buffer
UCS-A /security/role #
```

Removing Privileges from a User Role

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **scope role** *name*	Enters security role mode for the specified role.
Step 3	UCS-A /security/role # **remove privilege** *privilege-name*	Removes one or more privileges from the existing user role privileges. **Note** You can specify more than one *privilege-name* on the same command line to remove multiple privileges from the role, or you can remove privileges from the same role using multiple **remove privilege** commands.
Step 4	UCS-A /security/role # **commit-buffer**	Commits the transaction to the system configuration.

The following example removes the server security and server policy privileges from the service-profile-security-admin role:

```
UCS-A# scope security
UCS-A /security # scope role service-profile-security-admin
UCS-A /security/role # remove privilege server-security server-policy
UCS-A /security/role* # commit-buffer
UCS-A /security/role #
```

Deleting a User Role

Procedure

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

The following example deletes the service-profile-security-admin role and commits the transaction:

```
UCS-A# scope security
UCS-A /security # delete role service-profile-security-admin
UCS-A /security* # commit-buffer
UCS-A /security #
```

Configuring Locales

Creating a Locale

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **create locale** *locale-name*	Creates a locale and enters security locale mode.
Step 3	UCS-A /security/locale # **create org-ref** *org-ref-name* **orgdn** *orgdn-name*	References (binds) an organization to the locale. The *org-ref-name* argument is the name used to identify the organization reference, and the *orgdn-name* argument is the distinguished name of the organization being referenced.
Step 4	UCS-A /security/locale # **commit-buffer**	Commits the transaction to the system configuration.

The following example creates the western locale, references the finance organization to the locale, names the reference finance-ref, and commits the transaction:

```
UCS-A# scope security
UCS-A /security # create locale western
UCS-A /security/locale* # create org-ref finance-ref orgdn finance
UCS-A /security/locale* # commit-buffer
UCS-A /security/locale #
```

Adding an Organization to a Locale

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A# **scope locale** *locale-name*	Enters security locale mode.
Step 3	UCS-A /security/locale # **create org-ref** *org-ref-name* **orgdn** *orgdn-name*	References (binds) an organization to the locale. The *org-ref-name* argument is the name used to identify the organization reference, and the *orgdn-name* argument is the distinguished name of the organization being referenced.
Step 4	UCS-A /security/locale # **commit-buffer**	Commits the transaction to the system configuration.

The following example enters the western locale, adds (references) the marketing organization to the locale, names the reference marketing-ref, and commits the transaction:

```
UCS-A# scope security
UCS-A /security # scope locale western
UCS-A /security/locale* # create org-ref marketing-ref orgdn marketing
UCS-A /security/locale* # commit-buffer
UCS-A /security/locale #
```

Deleting an Organization from a Locale

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **scope locale** *locale-name*	Enters security locale mode.
Step 3	UCS-A /security/locale # **delete org-ref** *org-ref-name*	Deletes the organization from the locale.
Step 4	UCS-A /security/locale # **commit-buffer**	Commits the transaction to the system configuration.

The following example deletes the finance organization from the western locale and commits the transaction:

```
UCS-A# scope security
UCS-A /security # scope locale western
UCS-A /security/locale # delete org-ref finance-ref
UCS-A /security/locale* # commit-buffer
UCS-A /security/locale #
```

Deleting a Locale

Procedure

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

The following example deletes the western locale and commits the transaction:

```
UCS-A# scope security
UCS-A /security # delete locale western
UCS-A /security* # commit-buffer
UCS-A /security #
```

Configuring User Accounts

Creating a User Account

At a minimum, we recommend that you create the following users:

- Server administrator account
- Network administrator account
- Storage administrator

Before You Begin

If the system includes:

- Remote authentication services, ensure the users exist in the remote authentication server with the appropriate roles and privileges.
- Multi-tenancy with organizations, create one or more locales. If you do not have any locales, all users will be created in root and will have roles and privileges in all organizations.
- SSH authentication, obtain the SSH key.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **create local-user** *local-user-name*	Creates the user account and enters security local user mode.
Step 3	UCS-A /security/local-user # **set password** *password*	Sets the password for the user account
Step 4	UCS-A /security/local-user # **set firstname** *first-name*	(Optional) Specifies the first name of the user.
Step 5	UCS-A /security/local-user # **set lastname** *last-name*	(Optional) Specifies the last name of the user.
Step 6	UCS-A /security/local-user # **set expiration** *month day-of-month year*	(Optional) Specifies the date that the user account expires. The *month* argument is the first three letters of the month name.
Step 7	UCS-A /security/local-user # **set email** *email-addr*	(Optional) Specifies the user e-mail address.
Step 8	UCS-A /security/local-user # **set phone** *phone-num*	(Optional) Specifies the user phone number.
Step 9	UCS-A /security/local-user # **set sshkey** *ssh-key*	(Optional) Specifies the SSH key used for passwordless access.
Step 10	UCS-A security/local-user # **commit-buffer**	Commits the transaction.

The following example creates the user account named kikipopo, sets the password to foo12345, commits the transaction, and commits the transaction:

```
UCS-A# scope security
UCS-A /security # create local-user kikipopo
UCS-A /security/local-user* # set password
Enter a password:
Confirm the password:
UCS-A /security/local-user* # commit-buffer
UCS-A /security/local-user #
```

The following example creates the user account named lincey, sets an OpenSSH key for passwordless access, and commits the transaction.

```
UCS-A# scope security
UCS-A /security # create local-user lincey
UCS-A /security/local-user* #  set sshkey "ssh-rsa
AAAAB3NzaC1yc2EAAAABIwAAAIEAuo9VQ2CmWBI9/S1f30klCWjnV3lgdXMzO0WUl5iPw85lkdQqap+NFuNmHcb4K
iaQB8X/PDdmtlxQQcawclj+k8f4VcOelBxlsGk5luq5ls1ob1VOIEwcKEL/h5lrdbNlI8y3SS9I/gGiBZ9ARlop9LDpD
m8HPh2LOgyH7Ei1MI8="
UCS-A /security/local-user* # commit-buffer
UCS-A /security/local-user #
```

The following example creates the user account named jforlenz, sets a Secure SSH key for passwordless access, and commits the transaction.

```
UCS-A# scope security
UCS-A /security # create local-user jforlenz
UCS-A /security/local-user* #  set sshkey
Enter lines one at a time. Enter ENDOFBUF to finish. Press ^C to abort.
User's SSH key:
> ---- BEGIN SSH2 PUBLIC KEY ----
>AAAAB3NzaC1yc2EAAAABIwAAAIEAuo9VQ2CmWBI9/S1f30klCWjnV3lgdXMzO0WUl5iPw8
>5lkdQqap+NFuNmHcb4KiaQB8X/PDdmtlxQQcawclj+k8f4VcOelBxlsGk5luq5ls1ob1VO
>IEwcKEL/h5lrdbNlI8y3SS9I/gGiBZ9ARlop9LDpDm8HPh2LOgyH7Ei1MI8=
> ---- END SSH2 PUBLIC KEY ----
> ENDOFBUF
UCS-A /security/local-user* # commit-buffer
UCS-A /security/local-user #
```

Deleting a User Account

Procedure

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

The following example deletes the foo user account and commits the transaction:

```
UCS-A# scope security
UCS-A /security # delete local-user foo
UCS-A /security* # commit-buffer
UCS-A /security #
```

Monitoring User Sessions

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **show user-session** {**local** | **remote**} [**detail**]	Displays session information for all users currently logged in to the system.

The following example lists all local users currently logged in to the system:

```
UCS-A# scope security
UCS-A /security # show user-session local
Session Id      User            Host                 Login Time
--------------- --------------- -------------------- ----------
pts_25_1_31264  steve           192.168.100.111      2009-05-09T14:06:59
ttyS0_1_3532    jeff            console              2009-05-02T15:11:08
web_25277_A     faye            192.168.100.112      2009-05-15T22:11:25
```

The following example displays detailed information on all local users currently logged in to the system:

```
UCS-A# scope security
UCS-A /security # show user-session local detail
Session Id pts_25_1_31264:
    Fabric Id: A
    Term: pts/25
    User: steve
    Host: 64.101.53.93
    Pid: 31264
    Login Time: 2009-05-09T14:06:59

Session Id ttyS0_1_3532:
    Fabric Id: A
    Term: ttyS0
    User: jeff
    Host: console
    Pid: 3532
    Login Time: 2009-05-02T15:11:08

Session Id web_25277_A:
    Fabric Id: A
    Term: web_25277
    User: faye
    Host: 192.168.100.112
    Pid: 3518
    Login Time: 2009-05-15T22:11:25
```

CHAPTER 10

Firmware Management

This chapter includes:

Overview of Firmware

Cisco UCS uses firmware obtained from and certified by Cisco to upgrade firmware on the following components:

- Servers, including the BIOS, storage controller, and server controller (BMC)
- Adapters, including NIC and HBA firmware, and Option ROM (where applicable)
- I/O modules
- Fabric interconnects
- Cisco UCS Manager

Image Management

Cisco delivers all firmware updates or packages to Cisco UCS components in images. These images can be the following:

- Component image, which contains the firmware for one component
- Package, which is a collection of component images

Cisco also provides release notes with each image, which you can obtain from the same website from which you obtained the image.

Cisco UCS Manager provides mechanisms to download both component images and packages to the fabric interconnect.

Image Headers

Every image has a header, which includes the following:

- Checksum
- Version information
- Compatibility information that the system can use to verify the compatibility of component images and any dependencies

Image Catalog

Cisco UCS Manager provides you with two views of the catalog of firmware images and their contents that have been downloaded to the fabric interconnect. These views are:

Packages This view provides you with a read-only representation of the packages that have been downloaded onto the fabric interconnect. This view is sorted by image, not by the contents of the image. For packages, you can use this view to see which component images are (were) in each downloaded package.

You can use this view to identify the firmware updates available for each component. You can also use this view to delete obsolete and unneeded images. A package is automatically deleted after all images in the package are deleted.

Images The images view lists the component images available on the system. You cannot use this view to see packages. The information available about each component image includes the name of the component, the image size, the image version, and the vendor and model of the component.

Tip Cisco UCS Manager stores the images in bootflash on the fabric interconnect. In a cluster system, space usage in bootflash on both fabric interconnects is the same, because all images are synchronized between them. If Cisco UCS Manager reports that the bootflash is out of space, delete obsolete images to free up space.

Firmware Updates

You can use any of the Cisco UCS Manager interfaces to update firmware in the system, including Cisco UCS Manager GUI and the Cisco UCS Manager CLI.

You can use either of the following methods to update the firmware:

- Direct update at the endpoints.
- Updates to server components through service profiles that include a host firmware package policy and a management firmware package policy.

Note Direct update is not available for some server components, such as BIOS and storage controller.

Firmware Versions

The firmware versions on a component depend upon the type of component.

Firmware Versions in BMC, I/O Modules, and Adapters

Each BMC, I/O module, and adapter has two slots for firmware in flash. Each slot holds a version of firmware. One slot is active and the other is the backup slot. A component boots from whichever slot is designated as active.

The following firmware version terminology is used in the GUI and CLI:

Running Version The running version is the firmware that is currently active and in use by the component.

Startup Version The startup version is the firmware that will be used when the component next boots up. Cisco UCS Manager provides the activate operation to change the startup version.

Backup Version The backup version is the firmware that is sitting in the other slot and is not in use by the component. This can be firmware that you have updated to the component but have not yet activated, or it can be an older firmware version that was replaced by a recent activate. Cisco UCS Manager provides the update operation to replace the image in the backup slot.

If the component cannot boot from the startup version, the component boots from the backup version.

Firmware Versions in the Fabric Interconnect and Cisco UCS Manager

You can update the fabric interconnect firmware and Cisco UCS Manager on the fabric interconnect. The fabric interconnect and Cisco UCS Manager firmware do not have backup versions, because all the images are stored on the fabric interconnect. As a result, the number of bootable fabric interconnect images is not limited to two, like the server BMC and adapters. Instead, the number of bootable fabric interconnect images is limited by the available space in the flash memory.

Note There are running and startup versions of the fabric interconnect and Cisco UCS Manager firmware, but there are no backup versions.

Direct Firmware Update at Endpoints

You can perform direct firmware updates on the following endpoints:

- Fabric interconnects
- Cisco UCS Manager
- I/O modules
- BMC
- Adapters

Note You cannot update the BIOS firmware directly. You must perform the BIOS firmware update through a host firmware package in a service profile.

Stages of a Direct Firmware Update

Cisco UCS Manager separates the direct update process into stages to ensure that you can push the firmware to a component while the system is running without affecting uptime on the server or other components. Because you do not need to reboot the server until after you activate, you can perform that task overnight or during other maintenance periods.

When you manually update firmware, the following stages occur:

Update

During this stage, the system pushes the selected firmware version to the component. The update process always overwrites the firmware in the backup slot on the component. The update stage applies only to I/O modules, BMCs, and adapters.

Activate

During this stage, the system sets the specified image version (normally the backup version) as active and reboots the endpoint. When the endpoint is rebooted, the backup slot becomes the active slot, and the active slot becomes the backup slot. The firmware in the new active slot becomes the startup version and the running version.

If the component cannot boot from the startup firmware, it defaults to the backup version and raises an alarm.

Recommended Order of Components for Firmware Activation

If you upgrade firmware by individual components in a Cisco UCS instance, we recommend that you activate the updates in the following order for quicker activation:

1. Adapter
2. BMC
3. I/O module
4. Fabric interconnect

Note Consider the following when activating the firmware:

- You can update all components in parallel.
- While activating adapter and I/O Moduless, you can use the set-startup-only option to set the startup version and skip the reset.
- Activating a fabric interconnect resets all I/O Moldules connected to it in addition to resetting itself.

Firmware Updates through Service Profiles

You can use service profiles to update the server and adapter firmware, including the BIOS on the server, by defining the following policies and including them in the service profile associated with a server:

- Host Firmware Package policy
- Management Firmware Package policy

Note You cannot update the firmware on an I/O module, fabric interconnect, or Cisco UCS Manager through service profiles. You must update the firmware on those components directly.

Host Firmware Pack

This policy enables you to specify a common set of firmware versions that make up the host firmware pack. The host firmware includes the following server and adapter components:

- BIOS
- SAS controller
- Emulex Option ROM (applicable only to Emulex-based Converged Network Adapters [CNAs])
- Emulex firmware (applicable only to Emulex-based CNAs)
- QLogic option ROM (applicable only to QLogic-based CNAs)
- Adapter firmware

The firmware pack is pushed to all servers associated with service profiles that include this policy.

This policy ensures that the host firmware is identical on all servers associated with service profiles which use the same policy. Therefore, if you move the service profile from one server to another, the firmware versions are maintained. Also, if you change the firmware version of the component in the firmware pack, new versions are applied to all the affected service profiles immediately, which could cause server reboots.

You must include this policy in a service profile, and that service profile must be associated with a server for it to take effect.

Prerequisites

This policy is not dependent upon any other policies. However, you must ensure that the appropriate firmware has been downloaded to the fabric interconnect. If the firmware image is not available while associating the service profile, UCSM will just ignore firmware update and complete association.

Management Firmware Pack

This policy enables you to specify a common set of firmware versions that make up the management firmware pack. The management firmware includes the server controller (BMC) on the server.

The firmware pack is pushed to all servers associated with service profiles that include this policy.

This policy ensures that the BMC firmware is identical on all servers associated with service profiles which use the same policy. Therefore, if you move the service profile from one server to another, the firmware versions are maintained.

You must include this policy in a service profile, and that service profile must be associated with a server for it to take effect.

Prerequisites

This policy is not dependent upon any other policies. However, you must ensure that the appropriate firmware has been downloaded to the fabric interconnect.

Stages of a Firmware Update through Service Profiles

If you use policies in service profiles to update server and adapter firmware, you must complete the following stages:

Firmware Package Policy Creation

During this stage, you create the host and/or management firmware packages and include them in the appropriate firmware policies.

Associate

During this stage, you include a firmware policy in a service profile, and then associate the service profile with a server. The system pushes the selected firmware versions to the endpoints and reboots to ensure that the endpoints are running the versions specified in the firmware pack.

When the firmware versions in the policies change, the system automatically performs firmware updates (wherever necessary), activates, and reboot the endpoints.

Caution This can be disruptive as endpoints reboot.

Firmware Downgrades

You downgrade firmware in a Cisco UCS instance in the same way that you upgrade firmware. The package or version that you select when you update the firmware determines whether you are performing an upgrade or a downgrade.

Obtaining Images from Cisco

Procedure

Step 1 In a web browser, navigate to the web link provided by Cisco to obtain firmware images for Cisco UCS.

Step 2 Select one or more firmware images and copy them to a network server.

Step 3 Read the release notes provided with the image or images.

What to Do Next

Download the firmware image to the fabric interconnect.

Note In a cluster setup, the firmware image is automatically downloaded to both fabric interconnects, regardless of which fabric interconnect is used to initiate the download. Cisco UCS Manager always keeps the images in both fabric interconnects in sync. If one fabric interconnect is down while dowloading, the download still finishes successfully. The images are synced to the other fabric interconnect when it comes back online.

Downloading a Firmware Package to the Fabric Interconnect

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope firmware**	Enters firmware mode.
Step 2	UCS-A /firmware # **download image** *URL*	Downloads the firmware package for Cisco UCS. Using the download path provided by Cisco, specify the URL using one of the following syntax: • **ftp:**//*server-ip-addr* /*path* • **scp:**//*username@server-ip-addr*/*path* • **sftp:**//*username@server-ip-addr*/*path* • **tftp:**//*server-ip-addr* **:***port-num*/*path*
Step 3	UCS-A /firmware # **show download-task**	Displays the status for your download task. When your image is completely downloaded, the task state changes from Downloading to Downloaded. The CLI does not automatically refresh, so you may have to enter the **show download-task** command multiple times until the task state displays Downloaded.

The following example uses SCP to download the ucs-k9-bundle.1.0.0.988.gbin firmware package.

```
UCS-A# scope firmware
UCS-A /firmware # download image scp://user1@192.168.10.10/images/ucs-k9-bundle.1.0.1.100.gbin
```

What to Do Next

Display the download status to confirm that the firmware package has completely downloaded, and then directly update the firmware at the endpoints.

Display Firmware Package Download Status

After a firmware download operation has been started, you can check the download status to see if the package is still downloading, or if it has completely downloaded.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope firmware**	Enters firmware mode.
Step 2	UCS-A /firmware # **show download-task**	Displays the status for your download task. When your image is completely downloaded, the task state changes from Downloading to Downloaded. The CLI does not automatically refresh, so you may have to enter the **show download-task** command multiple times until the task state displays Downloaded.

The following example displays the download status for the ucs-k9-bundle.1.0.0.988.gbin firmware package. The **show download-task** is entered multiple times until the download state indicates that the firmware package has been downloaded:

```
UCS-A# scope firmware
UCS-A /firmware # show download-task

Download task:
File Name Protocol Server          Userid          State
--------- -------- --------------- --------------- -----
ucs-k9-bundle.1.0.0.988.gbin
          Scp      10.193.32.11    user1           Downloading

UCS-A /firmware # show download-task

Download task:
File Name Protocol Server          Userid          State
--------- -------- --------------- --------------- -----
ucs-k9-bundle.1.0.0.988.gbin
          Scp      10.193.32.11    user1           Downloading

UCS-A /firmware # show download-task

Download task:
File Name Protocol Server          Userid          State
--------- -------- --------------- --------------- -----
ucs-k9-bundle.1.0.0.988.gbin
          Scp      10.193.32.11    user1           Downloaded
```

Directly Updating Firmware at Endpoints

Display All Available Software Images on the Fabric Interconnect

This procedure is optional and displays the available software images on the fabric interconnect for all endpoints.. You can also use the **show image** command in each endpoint mode to display the available software images for that endpoint.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope firmware**	Enters firmware mode
Step 2	UCS-A /firmware # **show image**	Displays all software images downloaded onto the fabric interconnect. **Note** You must provide the software version number when directly updating an endpoint. If you intend to directly update firmware at an endpoint, note its version number in the right column.

The following example displays all available software images on the fabric interconnect:

```
UCS-A# scope firmware
UCS-A /firmware # show image
Name                                                    Type                 Version
------------------------------------------------------- -------------------- -------
ucs-2100.1.0.0.988.gbin                                 Iom                  1.0(0.988)
ucs-6100-k9-kickstart.4.0.1a.N2.1.0.988.gbin            Switch Kernel
4.0(1a)N2(1.0.988)
ucs-6100-k9-system.4.0.1a.N2.1.0.988.gbin               Switch Software
4.0(1a)N2(1.0.988)
ucs-b200-m1-bios.S5500.86B.01.00.0030-978a.021920.gbin  Server Bios
S5500.86B.01.00.0030-978a.021920
ucs-b200-m1-k9-bmc.1.0.0.988.gbin                       Bmc                  1.0(0.988)
ucs-b200-m1-sasctlr.2009.02.09.gbin                     Storage Controller   2009.02.09
ucs-m71kr-e-cna.1.0.0.988.gbin                          Adapter              1.0(0.988)
ucs-m71kr-e-hba.zf280a4.gbin                            Host Hba             zf280a4
ucs-m71kr-e-optionrom.ZN502N5.gbin                      Host Hba Optionrom   ZN502N5
ucs-m71kr-q-cna.1.0.0.988.gbin                          Adapter              1.0(0.988)
ucs-m71kr-q-optionrom.1.69.gbin                         Host Hba Optionrom   1.69
ucs-m81kr-vic.1.0.0.988.gbin                            Adapter              1.0(0.988)
ucs-manager-k9.1.0.0.988.gbin                           System               1.0(0.988)
```

Updating an Adapter

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope adapter** *chassis-id*/ *blade-id*/*adapter-id*	Enters chassis server adapter mode for the specified adapter.
Step 2	UCS-A /chassis/server/adapter # **show image**	Displays the available software images for the adapter.

	Command or Action	Purpose
Step 3	UCS-A /chassis/server/adapter # **update firmware** *version-num*	Updates the selected firmware version on the adapter.
Step 4	UCS-A /chassis/server/adapter # **activate firmware** *version-num*	Activates the selected firmware version on the adapter.
Step 5	UCS-A /chassis/server/adapter # **commit-buffer**	Commits the transaction.

The following example updates and activates the adapter firmware to version 1.0(0.988):

```
UCS-A# scope adapter 1/3/1
UCS-A# /chassis/server/adapter # show image
Name                                               Type                 Version      State
-------------------------------------------------- -------------------- ------------ -----
ucs-m71kr-e-cna.1.0.0.988.gbin                    Adapter              1.0(0.988)   Active
ucs-m71kr-q-cna.1.0.0.988.gbin                    Adapter              1.0(0.988)   Active
ucs-m81kr-vic.1.0.0.988.gbin                           Adapter         1.0(0.988)   Active
UCS-A# /chassis/server/adapter # update firmware 1.0(0.988)
UCS-A# /chassis/server/adapter* # activate firmware 1.0(0.988)
UCS-A# /chassis/server/adapter* # commit-buffer
UCS-A# /chassis/server/adapter #
```

Updating a BMC

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 bmc**	Enters chassis server BMC mode.
Step 3	UCS-A /chassis/server/bmc # **update firmware** *version-num*	Updates the selected firmware version on the BMC in the server.
Step 4	UCS-A /chassis/server/bmc # **activate firmware** *version-num*	Activates the selected firmware version on the BMC in the server.
Step 5	UCS-A /chassis/server/bmc # **commit-buffer**	Commits the transaction.

The following example updates and activates the BMC firmware to version 1.0(0.988):

```
UCS-A# scope server 1/3
UCS-A# /chassis/server # scope bmc
UCS-A# /chassis/server/bmc* # update firmware 1.0(0.988)
UCS-A# /chassis/server/bmc* # activate firmware 1.0(0.988)
UCS-A# /chassis/server/bmc* # commit-buffer
UCS-A# /chassis/server/bmc #
```

Updating an I/O Module

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope chassis** *chassis-id*	Enters chassis mode for the specified chassis.
Step 2	UCS-A /chassis # **scope iom** *iom-id*	Enters chassis I/O module mode for the selected I/O module.
Step 3	UCS-A /chassis/iom # **show image**	Displays the available software images for the I/O module.
Step 4	UCS-A /chassis/iom # **update firmware** *version-num*	Updates the selected firmware version on the I/O module.
Step 5	UCS-A /chassis/iom # **activate firmware** *version-num*	Activates the selected firmware version on the I/O module.
Step 6	UCS-A /chassis/iom # **commit-buffer**	Commits the transaction.

The following example upgrades the I/O module to version 1.0(0.988):

```
UCS-A# scope chassis 1
UCS-A# /chassis # scope iom 1
UCS-A# /chassis/iom # show image
Name                                              Type                Version       State
------------------------------------------------- -------------------- ------------- -----
ucs-2100.1.0.0.988.gbin                           Iom                 1.0(0.988)    Active
UCS-A# /chassis/iom # update firmware 1.0(0.988)
UCS-A# /chassis/iom* # activate firmware 1.0(0.988)
UCS-A# /chassis/iom* # commit-buffer
UCS-A# /chassis/iom #
```

Updating a Fabric Interconnect

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope fabric-interconnect** {**a** | **b**}	Enters fabric interconnect mode for the specified fabric interconnect.
Step 2	UCS-A /fabric-interconnect # **show image**	Displays the available software images for the fabric interconnect.
Step 3	UCS-A /fabric-interconnect # **activate firmware** {**kernel-version** *kernel-ver-num* | **system-version** *system-ver-num*}	Activates the selected firmware version on the fabric interconnect.
Step 4	UCS-A /fabric-interconnect # **commit-buffer**	Commits the transaction.

The following example upgrades the fabric interconnect to version 4.0(1a)N2(1.0.988):

```
UCS-A# scope fabric-interconnect a
UCS-A /fabric-interconnect # show image
Name                                              Type                 Version     State
------------------------------------------------- -------------------- ----------- -----
ucs-6100-k9-kickstart.4.0.1a.N2.1.0.988.gbin      Switch Kernel        4.0(1a)N2(1.0.988)
                                                                                   Active
ucs-6100-k9-system.4.0.1a.N2.1.0.988.gbin         Switch Software      4.0(1a)N2(1.0.988)
                                                                                   Active
UCS-A /fabric-interconnect # activate firmware kernel-version 4.0(1a)N2(1.0.988)
system-version 4.0(1a)N2(1.0.988)
UCS-A /fabric-interconnect* # commit-buffer
```

Updating the UCS Manager

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system**	Enters system mode.
Step 2	UCS-A /system # **show image**	Displays the available software images for the UCS Manager (system).
Step 3	UCS-A /system # **activate firmware** *version-num* [**ignorecompcheck**]	Activates the selected firmware version on the system. Use the optional **ignorecompcheck** keyword to have the system ignore the compatibility check. **Note** Activating the UCS Manager does not require rebooting the fabric interconnect, however, management services will briefly go down and all VSH shells will be terminated as part of the activation.
Step 4	UCS-A /system # **commit-buffer**	Commits the transaction.

The following example upgrades the UCS Manager to version 1.0(0.988):

```
UCS-A# scope system
UCS-A# /system # show image
Name                                              Type             Version     State
------------------------------------------------- ---------------- ----------- -----
ucs-manager-k9.1.0.0.988.gbin                     System           1.0(0.988)  Active
UCS-A# /system # activate firmware 1.0(0.988)
UCS-A# /system* # commit-buffer
UCS-A# /system #
```

Updating Firmware through Service Profiles

Configuring a Host Firmware Pack

Before You Begin

Ensure that the appropriate firmware has been downloaded to the fabric interconnect.

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 fw-host-pack** *pack-name*	Creates a host firmware pack with the specified package name and enters organization firmware host pack mode.
Step 3	UCS-A /org/fw-host-pack # **set descr** *description*	(Optional) Provides a description for the host firmware pack. **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/fw-host-pack # **create pack-image** *hw-vendor-name hw-model*{**adapter** | **host-hba** | **host-hba-combined** | **host-hba-optionrom** | **host-nic** | **server-bios** | **storage-controller** | **unspecified**} *version-num*	Creates a package image for the host firmware pack and enters organization firmware host pack package image mode. The *hw-vendor-name* and *hw-model* values are labels that help you easily identify the package image when you enter the **show image detail** command. The *version-num* value specifies the version number of the firmware being used for the package image.
Step 5	UCS-A org/fw-host-pack/pack-image # **set version** *version-num*	(Optional) Specifies the package image version number. Changing this number triggers firmware updates on all components using the firmware through a service profile. Use this step is only when updating a host firmware pack, not when creating a pack. **Note** The host firmware pack can contain multiple package images. Repeat steps 5 and 6 create additional package images for other components.
Step 6	UCS-A org/fw-host-pack/pack-image # **commit-buffer**	Commits the transaction.

The following example creates the app1 host firmware pack, creates a storage controller package image with version 2009.02.09 firmware, and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # create fw-host-pack app1
UCS-A /org/fw-host-pack* # set descr "This is a host firmware pack example."
UCS-A /org/fw-host-pack* # create pack-image Cisco UCS storage-controller 2009.02.09
UCS-A /org/fw-host-pack/pack-image* # commit-buffer
UCS-A /org/fw-host-pack/pack-image #
```

What to Do Next

Include the policy in a service profile and/or template.

Configuring a Management Firmware Pack

Before You Begin

Ensure that the appropriate firmware has been downloaded to the fabric interconnect.

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 fw-mgmt-pack** *pack-name*	Creates a management firmware pack with the specified package name and enters organization firmware management pack mode.
Step 3	UCS-A /org/fw-mgmt-pack # **set descr** *description*	(Optional) Provides a description for the management firmware pack. **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/fw-mgmt-pack # **create pack-image** *hw-vendor-name hw-model* **bmc** *version-num*	Creates a package image for the management firmware pack and enters organization firmware management pack package image mode. The *hw-vendor-name* and *hw-model* values are labels that help you easily identify the package image when you enter the **show image detail** command. The *version-num* value specifies the version number of the firmware being used for the package image.
Step 5	UCS-A org/fw-mgmt-pack/pack-image # **set version** *version-num*	(Optional) Specifies the package image version number. Changing this number triggers firmware updates on all components using the firmware through a service profile. Use this step is only when updating a host firmware pack, not when creating a pack.
Step 6	UCS-A org/fw-mgmt-pack/pack-image # **commit-buffer**	Commits the transaction.

The following example creates the bmc1 host firmware pack, creates a BMC package image with version 1.0(0.988) firmware, and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # create fw-mgmt-pack bmc1
UCS-A /org/fw-mgmt-pack* # set descr "This is a management firmware pack example."
UCS-A /org/fw-mgmt-pack* # create pack-image Cisco UCS bmc 1.0(0.988)
UCS-A /org/fw-mgmt-pack/pack-image* # commit-buffer
UCS-A /org/fw-mgmt-pack/pack-image #
```

What to Do Next

Include the policy in a service profile and/or template.

PART III

Network Configuration

CHAPTER 11

Configuring Named VLANs

This chapter includes:

Named VLANs

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

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

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

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

Creating a Named VLAN Accessible to Both Fabric Interconnects

Important You cannot create VLANs with IDs from 3968 to 4048. This range of VLAN IDs is reserved.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope eth-uplink**	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # **create vlan** *vlan-name vlan-id*	Creates a named VLAN, specifies the VLAN name and VLAN ID, and enters Ethernet uplink VLAN mode.
Step 3	UCS-A /eth-uplink/vlan # **set default-net**	(Optional) Sets the VLAN as the default VLAN. **Note** Only one VLAN can exist as the default VLAN. If multiple VLANs are set as the default, the most recently set VLAN is the default.
Step 4	UCS-A /eth-uplink/vlan # **commit-buffer**	Commits the transaction to the system configuration.

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

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

Creating a Named VLAN Accessible to One Fabric Interconnect

Important You cannot create VLANs with IDs from 3968 to 4048. This range of VLAN IDs is reserved.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope eth-uplink**	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # **scope fabric** {**a** | **b**}	Enters Ethernet uplink fabric interconnect mode for the specified fabric interconnect (A or B).
Step 3	UCS-A /eth-uplink/fabric # **create vlan** *vlan-name vlan-id*	Creates a named VLAN, specifies the VLAN name and VLAN ID, and enters Ethernet uplink fabric interconnect VLAN mode.
Step 4	UCS-A /eth-uplink/fabric/vlan # **set default-net**	(Optional) Sets the VLAN as the native VLAN. **Note** Only one VLAN can exist as the native VLAN. If you set multiple VLANs as the native VLAN, the last one to be set becomes the native VLAN.

	Command or Action	Purpose
Step 5	UCS-A /eth-uplink/fabric/vlan # **commit-buffer**	Commits the transaction to the system configuration.

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

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

Deleting a Named VLAN

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

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope eth-uplink**	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # **delete vlan** *vlan-name*	Deletes the specified named VLAN.
Step 3	UCS-A /eth-uplink # **commit-buffer**	Commits the transaction to the system configuration.

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

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

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

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

CHAPTER 12

Configuring LAN Pin Groups

This chapter includes:

LAN Pin Groups

Cisco UCS uses LAN pin groups to pin Ethernet traffic from a vNIC on a server to an uplink Ethernet port or port channel on the fabric interconnect. You can use this pinning to manage the distribution of traffic from the servers.

To configure pinning for a server, you must include the LAN pin group in a vNIC policy. The vNIC policy is then included in the service profile assigned to that server. All traffic from the vNIC travels through the I/O module to the specified uplink Ethernet port.

Configuring an Ethernet Pin Group

In a system with two fabric interconnects, you can associate the pin group with only one fabric interconnect or with both fabric interconnects.

Before You Begin

Configure the ports and port channels with which you want to configure the pin group. You can only include ports and port channels configured as uplink ports in a LAN pin group

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope eth-uplink**	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # **create pin-group** *pin-group-name*	Creates an Ethernet pin group with the specified name, and enters Ethernet uplink pin group mode.

	Command or Action	Purpose
Step 3	UCS-A /eth-uplink/pin-group # **set descr** *description*	(Optional) Provides a description for the pin group. **Note** If your description includes spaces, special characters, or punctuation, you must begin and end your description with quotation marks. The quotation marks will not appear in the description field of any **show** command output.
Step 4	UCS-A /eth-uplink/pin-group # **set target** {**a** | **b** | **dual**} {**port** *slot-num* / *port-num* | **port-channel** *port-num*}	(Optional) Sets the Ethernet pin target to the specified fabric and port, or fabric and port channel.
Step 5	UCS-A /eth-uplink/pin-group # **commit-buffer**	Commits the transaction to the system configuration.

The following example creates an Ethernet pin group named pingroup54 on fabric A, provides a description for the pin group, sets the pin group target to port channel 28, and commits the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink # create pin-group pingroup54
UCS-A /eth-uplink/pin-group* # set descr "This is my pin group #54"
UCS-A /eth-uplink/pin-group* # set target a port-channel 28
UCS-A /eth-uplink/pin-group* # commit-buffer
UCS-A /eth-uplink/pin-group #
```

What to Do Next

Include the pin group in a vNIC template.

CHAPTER 13

Configuring MAC Pools

This chapter includes:

MAC Pools

A MAC pool is a collection of network identities, or MAC addresses, that are unique in their layer 2 environment and are available to be assigned to vNICs on a server. If you use MAC pools in service profiles, you do not have to manually configure the MAC addresses to be used by the server associated with the service profile.

In a system that implements multi-tenancy, you can use the organizational hierarchy to ensure that MAC pools can only be used by specific applications or business services. Cisco UCS Manager will use the name resolution policy to assign MAC addresses from the pool.

To assign a MAC address to a server, you must include the MAC pool in a vNIC policy. The vNIC policy is then included in the service profile assigned to that server.

You can specify your own MAC addresses or use a group of MAC addresses provided by Cisco.

Configuring a MAC 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 # **create mac-pool** *mac-pool-name*	Creates a MAC pool with the specified name, and enters organization MAC pool mode.
Step 3	UCS-A /org/mac-pool # **set descr** *description*	(Optional) Provides a description for the MAC pool.

	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/mac-pool # **create block** *first-mac-addr last-mac-addr*	Creates a block (range) of MAC addresses, and enters organization MAC pool block mode. You must specify the first and last MAC addresses in the address range using the form *nn* **:** *nn* **:** *nn* **:** *nn* **:** *nn* **:** *nn*, with the addresses separated by a space. **Note** A MAC pool can contain more than one MAC address block. To create multiple MAC address blocks, you must enter multiple **create block** commands from organization MAC pool mode.
Step 5	UCS-A /org/mac-pool # **commit-buffer**	Commits the transaction to the system configuration.

The following example creates a MAC pool named pool37, provides a description for the pool, defines a MAC address block by specifying the first and last MAC addresses in the block, and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # create mac-pool pool37
UCS-A /org/mac-pool* # set descr "This is my MAC pool"
UCS-A /org/mac-pool* # create block 00:A0:D7:42:00:01 00:A0:D7:42:01:00
UCS-A /org/mac-pool/block* # commit-buffer
UCS-A /org/mac-pool/block #
```

What to Do Next

Include the MAC pool in a vNIC template.

CHAPTER 14

Configuring Quality of Service

This chapter includes:

Quality of Service

Cisco UCS provides the following methods to implement quality of service:

- System classes that specify the global configuration for certain types of traffic across the entire system
- QoS policies that assign system classes for individual vNICs
- Flow control policies that determine how uplink Ethernet ports handle pause frames

System Classes

Cisco UCS uses Data Center Ethernet (DCE) to handle all traffic inside a Cisco UCS instance. This industry standard enhancement to Ethernet divides the bandwidth of the Ethernet pipe into eight virtual lanes. System classes determine how the DCE bandwidth in these virtual lanes is allocated across the entire Cisco UCS instance.

Each system class reserves a specific segment of the bandwidth for a specific type of traffic. This provides a level of traffic management, even in an oversubscribed system. For example, you can configure the Fibre Channel Priority system class to determine the percentage of DCE bandwidth allocated to FCoE traffic.

The following table describes the system classes:

System Class	Description
Platinum Priority Gold Priority Silver Priority Bronze Priority	A configurable set of system classes that you can include in the QoS policy for a service profile. Each system class manages one lane of traffic. All properties of these system classes are available for you to assign custom settings and policies.
Best Effort Priority	A system class that sets the quality of service for the lane reserved for Basic Ethernet traffic. Some properties of this system class are preset and cannot be modified. For example, this class has a drop policy that allows it to drop data packets if required.
Fibre Channel Priority	A system class that sets the quality of service for the lane reserved for Fibre Channel over Ethernet traffic. Some properties of this system class are preset and cannot be modified. For example, this class has a no-drop policy that ensures it never drops data packets.

Quality of Service Policies

QoS policies assign a system class to the outgoing traffic for a vNIC or vHBA. This system class determines the quality of service for that traffic.

You must include a QoS policy in a vNIC policy or vHBA policy and then include that policy in a service profile to configure the vNIC or vHBA.

Flow Control Policies

Flow control policies determine whether the uplink Ethernet ports in a Cisco UCS instance send and receive IEEE 802.3x pause frames when the receive buffer for a port fills. These pause frames request that the transmitting port stop sending data for a few milliseconds until the buffer clears.

For flow control to work between a LAN port and an uplink Ethernet port, you must enable the corresponding receive and send flow control parameters for both ports. For Cisco UCS, the flow control policies configure these parameters.

If you enable the send function, then the uplink Ethernet port sends a pause request to the network port if the incoming packet rate becomes too high. The pause remains in effect for a few milliseconds before traffic is reset to normal levels. If you enable the receive function, then the uplink Ethernet port will honor all pause requests from the network port. All traffic is halted on that uplink port until the network port cancels the pause request.

Because you assign the flow control policy to the port, changes to the policy have an immediate effect on how the port reacts to a pause frame or a full receive buffer.

Configuring a System Class

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope eth-server**	Enters Ethernet server mode.
Step 2	UCS-A /eth-server # **scope qos**	Enters Ethernet server QoS mode.
Step 3	UCS-A /eth-server/qos # **scope eth-classified** {**bronze** | **gold** | **platinum silver**}	Enters Ethernet server QoS Ethernet classified mode for the specified system class.
Step 4	UCS-A /eth-server/qos/eth-classified # **set adminstate** {**enable** | **disable**}	Enables or disables the administrative state of the specified system class.
Step 5	UCS-A /eth-server/qos/eth-classified # **set cos** *cos-value*	Specifies the class of service for the specified system class. Valid class of service values are 0 to 6; higher values indicate more important traffic.
Step 6	UCS-A /eth-server/qos/eth-classified # **set drop** {**drop** | **no-drop**}	Specifies whether the channel can drop packets or not. **Note** Only one system class can use the no-drop option.
Step 7	UCS-A /eth-server/qos/eth-classified # **set mtu** {*mtu-value* | **fc** | **normal**}	Specifies the maximum transmission unit (MTU) for the specified system class. Valid MTU values are 1538 to 9216.
Step 8	UCS-A /eth-server/qos/eth-classified # **set weight** {*weight-value* | **best-effort** | **none**}	Specifies the relative weight for the specified system class. Valid weight values are 0 to 10.
Step 9	UCS-A /eth-server/qos/eth-classified # **commit-buffer**	Commits the transaction to the system configuration.

The following example enables the platinum system class, allows the channel to drop packets, sets the class of service to 6, the MTU to normal, the weight to 5, and commits the transaction:

```
UCS-A# scope eth-server
UCS-A /eth-server # scope qos
UCS-A /eth-server/qos # scope eth-classified platinum
UCS-A /eth-server/qos/eth-classified # set adminstate enable
UCS-A /eth-server/qos/eth-classified* # set drop drop
UCS-A /eth-server/qos/eth-classified* # set cos 6
UCS-A /eth-server/qos/eth-classified* # set mtu normal
UCS-A /eth-server/qos/eth-classified* # set weight 5
UCS-A /eth-server/qos/eth-classified* # commit-buffer
UCS-A /eth-server/qos/eth-classified #
```

Configuring a QoS 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 qos-policy** *policy-name*	Creates a QoS policy with the specified policy name, and enters organization QoS policy mode.
Step 3	UCS-A /org/qos-policy # **create vnic-egress-policy**	Creates a vNIC egress policy to be used for the QoS policy and enters organization QoS policy vNIC egress policy mode.
Step 4	UCS-A /org/qos-policy/vnic-egress-policy # **set prio** {**best-effort** | **bronze** | **gold** | **platinum** | **silver**}	Specifies the name of the system class to be used for the vNIC egress policy.
Step 5	UCS-A /org/qos-policy/vnic-egress-policy # **commit-buffer**	Commits the transaction to the system configuration.

The following example assigns the best effort system class to the vNIC egress policy of the QoS policy and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # create qos-policy QosPolicy34
UCS-A /org/qos-policy* # create vnic-egress-policy
UCS-A /org/qos-policy/vnic-egress-policy* # set prio best-effort
UCS-A /org/qos-policy/vnic-egress-policy* # commit-buffer
UCS-A /org/qos-policy/vnic-egress-policy #
```

What to Do Next

Include the QoS policy in a vNIC template.

Deleting a QoS Policy

If you delete a QoS policy that is in use or disable a system class that is used in a QoS policy, any vNIC which uses that QoS policy will be assigned to the Best Effort Priority system class. In a system that implements multi-tenancy, Cisco UCS Manager will first attempt to find a matching QoS policy in the organization hierarchy.

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 # **delete qos-policy** *policy-name*	Deletes the specified QoS policy.
Step 3	UCS-A /org # **commit-buffer**	Commits the transaction to the system configuration.

The following deletes the QoS policy named QosPolicy34 and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # delete qos-policy QosPolicy34
UCS-A /org* # commit-buffer
UCS-A /org #
```

Configuring a Flow Control Policy

Before You Begin

Configure the network port with the corresponding setting for the flow control that you need. For example, if you enable the send setting for flow-control pause frames in the policy, make sure that the receive parameter in the network port is set to on or desired. If you want the Cisco UCS port to receive flow-control frames, make sure that the network port has a send parameter set to on or desired. If you do not want to use flow control, you can set the send and receive parameters on the network port to off.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope eth-uplink**	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # **scope flow-control**	Enters Ethernet uplink flow control mode.
Step 3	UCS-A /eth-uplink/flow-control # **create policy** *policy-name*	Creates the specified flow control policy.
Step 4	UCS-A /eth-uplink/flow-control/policy # **set prio** *prio-option*	Specifies one of the following flow control priority options: • **auto**—The Cisco UCS system and the network negotiate whether PPP will be used on this fabric interconnect. • **on**—PPP is enabled on this fabric interconnect.
Step 5	UCS-A /eth-uplink/flow-control/policy # **set receive** *receive-option*	Specifies one of the following flow control receive options: • **off**—Pause requests from the network are ignored and traffic flow continues as normal. • **on**—Pause requests are honored and all traffic is halted on that uplink port until the network cancels the pause request.

	Command or Action	Purpose
Step 6	UCS-A /eth-uplink/flow-control/policy # **set send** *send-option*	Specifies one of the following flow control send options: • **off**—Traffic on the port flows normally regardless of the packet load. • **on**—The Cisco UCS system sends a pause request to the network if the incoming packet rate becomes too high. The pause remains in effect for a few milliseconds before traffic is reset to normal levels.
Step 7	UCS-A /org/qos-policy/vnic-egress-policy # **commit-buffer**	Commits the transaction to the system configuration.

The following configures a flow control policy and commits the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink* # scope flow-control
UCS-A /eth-uplink/flow-control* # create policy FlowControlPolicy23
UCS-A /eth-uplink/flow-control/policy* # set prio auto
UCS-A /eth-uplink/flow-control/policy* # set receive on
UCS-A /eth-uplink/flow-control/policy* # set send on
UCS-A /eth-uplink/flow-control/policy* # commit-buffer
UCS-A /eth-uplink/flow-control/policy #
```

What to Do Next

Associate the flow control policy with an uplink Ethernet port or port channel.

Deleting a Flow Control Policy

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope eth-uplink**	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # **scope flow-control**	Enters Ethernet uplink flow control mode.
Step 3	UCS-A /eth-uplink/flow-control # **delete policy** *policy-name*	Deletes the specified flow control policy.
Step 4	UCS-A /eth-uplink/flow-control # **commit-buffer**	Commits the transaction to the system configuration.

The following example deletes the flow control policy named FlowControlPolicy23 and commits the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink* # scope flow-control
UCS-A /eth-uplink/flow-control* # delete policy FlowControlPolicy23
UCS-A /eth-uplink/flow-control* # commit-buffer
UCS-A /eth-uplink/flow-control #
```

CHAPTER 15

Configuring Network-Related Policies

This chapter includes:

Configuring vNIC Templates

vNIC Template

This policy defines how a vNIC on a server connects to the LAN. This policy is also referred to as a vNIC LAN connectivity policy.

You need to include this policy in a service profile for it to take effect.

Configuring a vNIC Template

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 vnic-templ** *vnic-templ-name* [**eth-if** *vlan-name*] [**fabric** {**a** | **b**}] [**target** [**adapter** | **vm**]]	Creates a vNIC template and enters organization vNIC template mode.
Step 3	UCS-A /org/vnic-templ # **set descr** *description*	(Optional) Provides a description for the vNIC template.
Step 4	UCS-A /org/vnic-templ # **set fabric** {**a** | **b**}	(Optional) Specifies the fabric to use for the vNIC. If you did not specify the fabric when creating the vNIC template in

	Command or Action	Purpose
		Step 2, then you have the option to specify it with this command.
Step 5	UCS-A /org/vnic-templ # **set mac-pool** *mac-pool-name*	Specifies the MAC pool to use for the vNIC.
Step 6	UCS-A /org/vnic-templ # **set pin-group** *group-name*	Specifies the LAN pin group to use for the vNIC.
Step 7	UCS-A /org/vnic-templ # **set qos-policy** *policy-name*	Specifies the QoS policy to use for the vNIC.
Step 8	UCS-A /org/vnic-templ # **set stats-policy** *policy-name*	Specifies the server and server component statistics threshold policy to use for the vNIC.
Step 9	UCS-A /org/vnic-templ # **set type** {**initial-template** | **updating-template**}	Specifies the vNIC template update type. If you do not want vNIC instances created from this template to be automatically updated when the template is updated, use the **initial-template** keyword; otherwise, use the **updating-template** keyword to ensure that all vNIC instance are updated when the vNIC template is updated.
Step 10	UCS-A /org/vnic-templ # **commit-buffer**	Commits the transaction to the system configuration.

The following example configures a vNIC template and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # create vnic template VnicTempFoo
UCS-A /org/vnic-templ* # set descr "This is a vNIC template example."
UCS-A /org/vnic-templ* # set fabric a
UCS-A /org/vnic-templ* # set mac-pool pool137
UCS-A /org/vnic-templ* # set pin-group PinGroup54
UCS-A /org/vnic-templ* # set stats-policy ServStatsPolicy
UCS-A /org/vnic-templ* # set type updating-template
UCS-A /org/vnic-templ* # commit-buffer
UCS-A /org/vnic-templ #
```

Deleting a vNIC Template

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 vnic-templ** *vnic-templ-name*	Deletes the specified vNIC template.
Step 3	UCS-A /org # **commit-buffer**	Commits the transaction to the system configuration.

The following example deletes the vNIC template named VnicTempFoo and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # delete vnic template VnicTempFoo
UCS-A /org* # commit-buffer
UCS-A /org #
```

Configuring Ethernet Adapter Policies

Ethernet and Fibre Channel Adapter Policies

These policies govern the host-side behavior of the adapter, including how the adapter handles traffic. For example, you can use these policies to change default settings for the following:

- Queues
- Interrupt handling
- Performance enhancement
- RSS hash
- Failover in an cluster configuration with two fabric interconnects

Operating systems are sensitive to the settings in these policies. The configuration and selection of the policy is driven by the type of operating system.

Configuring an Ethernet 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 eth-profile** *policy-name*	Creates the specified Ethernet adapter policy and enters organization Ethernet profile mode.
Step 3	UCS-A /org/eth-profile # **set comp-queue count** *count*	(Optional) Configures the Ethernet completion queue profile.
Step 4	UCS-A /org/eth-profile # **set descr** *description*	(Optional) 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 5	UCS-A /org/eth-profile # **set ext-ipv6-rss-hash** {**ip-hash** {**disabled** | **enabled**} | **tcp-hash** {**disabled** | **enabled**}}	(Optional) Configures the external IPv6 RSS hash profile.
Step 6	UCS-A /org/eth-profile # **set failover timeout** *timeout-sec*	(Optional) Configures the Ethernet failover profile.
Step 7	UCS-A /org/eth-profile # **set interrupt** {**coalescing-time** *sec* | **coalescing-type** {**idle** | **min**} | **count** *count*}	(Optional) Configures the Ethernet interrupt profile.
Step 8	UCS-A /org/eth-profile # **set ipv4-rss-hash** {**ip-hash** {**disabled** | **enabled**} | **tcp-hash** {**disabled** | **enabled**}}	(Optional) Configures the IPv4 RSS hash profile.
Step 9	UCS-A /org/eth-profile # **set ipv6-rss-hash** {**ip-hash** {**disabled** | **enabled**} | **tcp-hash** {**disabled** | **enabled**}}	(Optional) Configures the IPv6 RSS hash profile.
Step 10	UCS-A /org/eth-profile # **set offload** {**large-receive** | **tcp-rx-checksum** | **tcp-segment** | **tcp-tx-checksum**} {**disabled** | **enabled**}	(Optional) Configures the Ethernet offload profile.
Step 11	UCS-A /org/eth-profile # **set recv-queue** {**count** *count* | **ring-size** *size-num*}	(Optional) Configures the Ethernet receive queue profile.
Step 12	UCS-A /org/eth-profile # **set rss receivesidescaling** {**disabled** | **enabled**}	(Optional) Configures the RSS profile.
Step 13	UCS-A /org/rth-profile # **set work-queue** {**count** *count* | **ring-size** *size-num*}	(Optional) Configures the Ethernet work queue profile.
Step 14	UCS-A /org/eth-profile # **commit-buffer**	Commits the transaction to the system configuration.

The following example configures an Ethernet adapter policy:

```
UCS-A# scope org /
UCS-A /org* # create eth-profile EthPolicy19
UCS-A /org/eth-profile* # set comp-queue count 16
UCS-A /org/eth-profile* # set descr "This is an Ethernet adapter policy example."
UCS-A /org/eth-profile* # set ext-ipv6-rss-hash ip-hash disabled
UCS-A /org/eth-profile* # set failover timeout 300
UCS-A /org/eth-profile* # set interrupt count 64
UCS-A /org/eth-profile* # set ipv4-rss-hash ip-hash disabled
UCS-A /org/eth-profile* # set ipv6-rss-hash ip-hash disabled
UCS-A /org/eth-profile* # set offload large-receive disabled
UCS-A /org/eth-profile* # set recv-queue count 32
UCS-A /org/eth-profile* # set rss receivesidescaling enabled
UCS-A /org/eth-profile* # set work-queue
UCS-A /org/eth-profile* # commit-buffer
UCS-A /org/eth-profile #
```

Deleting an Ethernet 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 eth-profile** *policy-name*	Deletes the specified Ethernet adapter policy.
Step 3	UCS-A /org # **commit-buffer**	Commits the transaction to the system configuration.

The following example deletes the Ethernet adapter policy named EthPolicy19:

```
UCS-A# scope org /
UCS-A /org* # delete eth-profile EthPolicy19
UCS-A /org* # commit-buffer
UCS-A /org #
```

PART IV

Storage Configuration

CHAPTER 16

Configuring Named VSANs

This chapter includes:

Named VSANs

A named VSAN creates a connection to a specific external SAN. The VSAN isolates traffic to that external SAN, including broadcast traffic. The traffic on one named VSAN knows that the traffic on another named VSAN exists, but cannot read or access that traffic.

Like a named VLAN, the name that you assign to a VSAN ID adds a layer of abstraction that allows you to globally update all servers associated with service profiles that use the named VSAN. You do not need to reconfigure the servers individually to maintain communication with the external SAN. You can create more than one named VSAN with the same VSAN ID.

In a cluster configuration, a named VSAN can be configured to be accessible only to the FC uplinks on one fabric interconnect or to the FC Uplinks on both fabric interconnects.

Creating a Named VSAN Accessible to Both Fabric Interconnects

You can create a named VSAN with IDs from 1 to 4093.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope fc-uplink**	Enters Fibre Channel uplink mode.
Step 2	UCS-A /fc-uplink # **create vsan** *vsan-name vsan-id fcoe-id*	Creates the specified named VSAN, specifies the VSAN name, VSAN ID and FCoE VLAN ID, and enters Fibre Channel uplink VSAN mode.

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

The following example creates a named VSAN for both fabric interconnects, names the VSAN accounting, assigns the VSAN ID 2112, assigns the FCoE VLAN ID 4021, and commits the transaction:

```
UCS-A# scope fc-uplink
UCS-A /eth-uplink* # create vsan accounting 2112 4021
UCS-A /eth-uplink/vsan* # commit-buffer
UCS-A /eth-uplink/vsan #
```

Creating a Named VSAN Accessible to One Fabric Interconnect

You can create a named VSAN with IDs from 1 to 4093.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope fc-uplink**	Enters Fibre Channel uplink mode.
Step 2	UCS-A /fc-uplink # **scope fabric** {**a** | **b**}	Enters Fibre Channel uplink fabric interconnect mode for the specified fabric interconnect (A or B).
Step 3	UCS-A /fc-uplink/fabric # **create vsan** *vsan-name vsan-id fcoe-id*	Creates the specified named VSAN, specifies the VSAN name, VSAN ID and FCoE VLAN ID, and enters Fibre Channel uplink VSAN mode.
Step 4	UCS-A /fc-uplink/fabric/vsan # **commit-buffer**	Commits the transaction to the system configuration.

The following example creates a named VSAN for fabric interconnect A, names the VSAN finance, assigns the VSAN ID 3955, assigns the FCoE VLAN ID 2221, and commits the transaction:

```
UCS-A# scope fc-uplink
UCS-A /fc-uplink # scope fabric a
UCS-A /fc-uplink/fabric # create vsan finance 3955 2221
UCS-A /fc-uplink/fabric/vsan* # commit-buffer
UCS-A /fc-uplink/fabric/vsan #
```

Deleting a Named VSAN

If Cisco UCS includes a named VSAN with the same VSAN ID as the one you delete, the VSAN will not be removed from the fabric interconnect configuration until all named VSANs with that ID are deleted.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope eth-uplink**	Enters Ethernet uplink mode.
Step 2	UCS-A /fc-uplink # **delete vsan** *vsan-name*	Deletes the specified named VSAN.

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

The following example deletes a named VSAN and commits the transaction:

```
UCS-A# scope fc-uplink
UCS-A /fc-uplink* # delete vsan finance
UCS-A /fc-uplink* # commit-buffer
UCS-A /fc-uplink #
```

CHAPTER 17

Configuring SAN Pin Groups

This chapter includes:

SAN Pin Groups

Cisco UCS uses SAN pin groups to pin Fibre Channel traffic from a vHBA on a server to an uplink Fibre Channel port on the fabric interconnect. You can use this pinning to manage the distribution of traffic from the servers.

To configure pinning for a server, you must include the SAN pin group in a vHBA policy. The vHBA policy is then included in the service profile assigned to that server. All traffic from the vHBA will travel through the I/O module to the specified uplink Fibre Channel port.

You can assign the same pin group to multiple vHBA policies. As a result, you do not need to manually pin the traffic for each vHBA.

Important Changing the target interface for an existing SAN pin group disrupts traffic for all vHBAs which use that pin group. The fabric interconnect performs a log in and log out for the Fibre Channel protocols to re-pin the traffic.

Configuring a Fibre Channel Pin Group

In a system with two fabric interconnects, you can associate the pin group with only one fabric interconnect or with both fabric interconnects.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope fc-uplink**	Enters Fibre Channel uplink mode.

	Command or Action	Purpose
Step 2	UCS-A /fc-uplink # **create pin-group** *pin-group-name*	Creates a Fibre Channel pin group with the specified name, and enters Fibre Channel uplink pin group mode.
Step 3	UCS-A /fc-uplink/pin-group # **set descr** *description*	(Optional) Provides a description for the pin group. **Note** If your description includes spaces, special characters, or punctuation, you must begin and end your description with quotation marks. The quotation marks will not appear in the description field of any **show** command output.
Step 4	UCS-A /fc-uplink/pin-group # **set target** {**a** | **b** | **dual**} **port** *slot-num* / *port-num*	(Optional) Sets the Fibre Channel pin target to the specified fabric and port.

The following example creates a Fibre Channel pin group named fcpingroup12, provides a description for the pin group, and sets the pin group target to slot 2, port 1.

```
UCS-A# scope fc-uplink
UCS-A /fc-uplink # create pin-group fcpingroup12
UCS-A /fc-uplink/pin-group* # set descr "This is my pin group #12"
UCS-A /fc-uplink/pin-group* # set target a port 2/1
```

What to Do Next

Include the pin group in a vHBA template.

CHAPTER 18

Configuring WWN Pools

This chapter includes:

WWN Pools

A WWN pool is a collection of WWNs for use by the Fibre Channel vHBAs in a Cisco UCS instance. You create separate pools for:

- WW node names assigned to the server
- WW port names assigned to the vHBA

Important A WWN pool can include only WWNNs or WWPNs in the ranges from 20:00:00:00:00:00:00:00 to 20:FF:FF:FF:FF:FF:FF:FF or from 50:00:00:00:00:00:00:00 to 5F:FF:FF:FF:FF:FF:FF:FF. All other WWN ranges are reserved.

If you use WWN pools in service profiles, you do not have to manually configure the WWNs that will be used by the server associated with the service profile. In a system that implements multi-tenancy, you can use a WWN pool to control the WWNs used by each organization.

You assign WWNs to pools in blocks. For each block or individual WWN, you can assign a boot target.

WWNN Pools

A WWNN pool is a WWN pool which contains only WW node names. If you include a pool of WWNNs in a service profile, the associated server will be assigned a WWNN from that pool

WWPN Pools

A WWPN pool is a WWN pool which contains only WW port names. If you include a pool of WWPNs in a service profile, the port on each vHBA of the associated server will be assigned a WWPN from that pool

Configuring a WWN Pool

Important A WWN pool can include only WWNNs or WWPNs in the ranges from 20:00:00:00:00:00:00:00 to 20:FF:FF:FF:FF:FF:FF:FF or from 50:00:00:00:00:00:00:00 to 5F:FF:FF:FF:FF:FF:FF:FF. All other WWN ranges are reserved.

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 wwn-pool** *wwn-pool-name* {**node-wwn-assignment** | **port-wwn-assignment**}	Creates a WWN pool with the specified name and purpose, and enters organization WWN pool mode. The purpose of the WWN pool can be one of the following: • To assign world wide node names (WWNNs) and world wide port names (WWPNs) • To assign only WWNNs • To assign only WWPNs
Step 3	UCS-A /org/wwn-pool # **set descr** *description*	(Optional) Provides a description for the WWN 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/wwn-pool # **create block** *first-wwn last-wwn*	(Optional) Creates a block (range) of WWNs, and enters organization WWN pool block mode. You must specify the first and last WWN in the block using the form *nn* **:** *nn* **:** *nn* **:** *nn* **:** *nn* **:** *nn* **:** *nn* **:** *nn*, with the WWNs separated by a space. **Note** A WWN pool can contain more than one WWN block. To create multiple WWN blocks, you must enter multiple **create block** commands from organization WWN pool mode.
Step 5	UCS-A /org/wwn-pool # **create initiator** *wwn wwn*	(Optional) Creates a single initiator, and enters organization WWN pool initiator mode. You must specify the initiator using the form *nn* **:** *nn* **:** *nn* **:** *nn* **:** *nn* **:** *nn* **:** *nn* **:** *nn*. **Note** A WWN pool can contain more than one initiator. To create multiple initiators, you must enter multiple **create initiator** commands from organization WWN pool mode.

	Command or Action	Purpose
Step 6	UCS-A /org/wwn-pool/block **commit-buffer**	Commits the transaction to the system configuration.

The following example creates a WWN pool named sanpool, provides a description for the pool, and specifies a block of WWNs and an initiator to be used for the pool:

```
UCS-A# scope org
UCS-A /org # create wwn-pool sanpool node-wwn-assignment
UCS-A /org/wwn-pool* # set descr "This is my WWNN pool"
UCS-A /org/wwn-pool* # create block 23:00:00:05:AD:1E:00:01 23:00:00:05:AD:1E:01:00
UCS-A /org/wwn-pool/block* # exit
UCS-A /org/wwn-pool* # create initiator 23:00:00:05:AD:1E:02:00
UCS-A /org/wwn-pool/initiator* # commit-buffer
UCS-A /org/wwn-pool/initiator #
```

What to Do Next

Include the WWN pool in a vHBA template.

CHAPTER 19

Configuring Storage-Related Policies

This chapter includes:

Configuring vHBA Templates

vHBA Template

This policy defines how a vHBA on a server connects to the SAN. This policy is also referred to as a vHBA SAN connectivity template.

You need to include this policy in a service profile for it to take effect.

Configuring a vHBA Template

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 vhba-templ** *vhba-templ-name* [**fabric** {**a** | **b**}] [**fc-if** *vsan-name*]	Creates a vHBA template and enters organization vHBA template mode.
Step 3	UCS-A /org/vhba-templ # **set descr** *description*	(Optional) Provides a description for the vHBA template.
Step 4	UCS-A /org/vhba-templ # **set fabric** {**a** | **b**}	(Optional) Specifies the fabric to use for the vHBA. If you did not specify the fabric when creating the vHBA template in

	Command or Action	Purpose
		Step 2, then you have the option to specify it with this command.
Step 5	UCS-A /org/vhba-templ # **set fc-if** *vsan-name*	(Optional) Specifies the Fibre Channel interface (named VSAN) to use for the vHBA. If you did not specify the Fibre Channel interface when creating the vHBA template in Step 2, then you have the option to specify it with this command.
Step 6	UCS-A /org/vhba-templ # **set mac-pool** *mac-pool-name*	Specifies the MAC pool to use for the vHBA.
Step 7	UCS-A /org/vhba-templ # **set pin-group** *group-name*	Specifies the pin group to use for the vHBA.
Step 8	UCS-A /org/vhba-templ # **set stats-policy** *policy-name*	Specifies the server and server component statistics threshold policy to use for the vHBA.
Step 9	UCS-A /org/vhba-templ # **set type** {**initial-template** | **updating-template**}	Specifies the vHBA template update type. If you do not want vHBA instances created from this template to be automatically updated when the template is updated, use the **initial-template** keyword; otherwise, use the **updating-template** keyword to ensure that all vHBA instance are updated when the vHBA template is updated.
Step 10	UCS-A /org/vhba-templ # **commit-buffer**	Commits the transaction to the system configuration.

The following example configures a vHBA template and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # create vhba template VhbaTempFoo
UCS-A /org/vhba-templ* # set descr "This is a vHBA template example."
UCS-A /org/vhba-templ* # set fabric a
UCS-A /org/vhba-templ* # set fc-if accounting
UCS-A /org/vhba-templ* # set mac-pool pool137
UCS-A /org/vhba-templ* # set pin-group FcPinGroup12
UCS-A /org/vhba-templ* # set stats-policy ServStatsPolicy
UCS-A /org/vhba-templ* # set type updating-template
UCS-A /org/vhba-templ* # commit-buffer
UCS-A /org/vhba-templ #
```

Deleting a vHBA Template

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 # **delete vhba-templ** *vhba-templ-name*	Deletes the specified vHBA template.
Step 3	UCS-A /org # **commit-buffer**	Commits the transaction to the system configuration.

The following example deletes the vHBA template named VhbaTempFoo and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # delete vhba template VhbaTempFoo
UCS-A /org* # commit-buffer
UCS-A /org #
```

Configuring Fibre Channel Adapter Policies

Ethernet and Fibre Channel Adapter Policies

These policies govern the host-side behavior of the adapter, including how the adapter handles traffic. For example, you can use these policies to change default settings for the following:

- Queues
- Interrupt handling
- Performance enhancement
- RSS hash
- Failover in an cluster configuration with two fabric interconnects

Operating systems are sensitive to the settings in these policies. The configuration and selection of the policy is driven by the type of operating system.

Configuring a Fibre Channel 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 fc-profile** *policy-name*	Creates the specified Fibre Channel adapter policy and enters organization Fibre Channel profile mode.
Step 3	UCS-A /org/fc-profile # **set descr** *description*	(Optional) 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	UCS-A /org/fc-profile # **set error-recovery** {**error-detect-timeout** *timeout-msec* | **fcp-error-recovery** {**disabled** | **enabled**} | **link-down-timeout** *timeout-msec* | **port-down-io-retry-count** *retry-count* | **port-down-timeout** *timeout-msec* | **resource-allocation-timeout** *timeout-msec*}	(Optional) Configures the Fibre Channel error recovery profile.
Step 5	UCS-A /org/fc-profile # **set port** {**io-throttle-count** *throttle-count* | **max-field-size** *size-num* | **max-luns** *max-num*}	(Optional) Configures the Fibre Channel port profile.
Step 6	UCS-A /org/fc-profile # **set port-f-logi** {**retries** *retry-count* | **timeout** *timeout-sec*}	(Optional) Configures the Fibre Channel port fabric login (FLOGI) profile.
Step 7	UCS-A /org/fc-profile # **set port-p-logi** {**retries** *retry-count* | **timeout** *timeout-secs*}	(Optional) Configures the Fibre Channel port-to-port login (PLOGI) profile.
Step 8	UCS-A /org/fc-profile # **set recv-queue** {**count** *count* | **ring-size** *size-num*}	(Optional) Configures the Fibre Channel receive queue profile.
Step 9	UCS-A /org/fc-profile # **set scsi-io** {**count** *count* | **ring-size** *size-num*}	(Optional) Configures the Fibre Channel SCSI I/O profile.
Step 10	UCS-A /org/fc-profile # **set work-queue** {**count** *count* | **ring-size** *size-num*}	(Optional) Configures the Fibre Channel work queue profile.
Step 11	UCS-A /org/fc-profile # **commit-buffer**	Commits the transaction to the system configuration.

The following example configures a Fibre Channel adapter policy:

```
UCS-A# scope org /
UCS-A /org* # create fc-profile FcPolicy42
UCS-A /org/fc-profile* # set descr "This is a Fibre Channel adapter policy example."
UCS-A /org/fc-profile* # set error-recovery error-detect-timeout 2500
UCS-A /org/fc-profile* # set port max-luns 4
UCS-A /org/fc-profile* # set port-f-logi retries 250
UCS-A /org/fc-profile* # set port-p-logi timeout 5000
UCS-A /org/fc-profile* # set recv-queue count 1
UCS-A /org/fc-profile* # set scsi-io ring-size 256
UCS-A /org/fc-profile* # set work-queue ring-size 256
UCS-A /org/fc-profile* # commit-buffer
UCS-A /org/fc-profile #
```

Deleting a Fibre Channel 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 fc-profile** *policy-name*	Deletes the specified Fibre Channel adapter policy.
Step 3	UCS-A /org # **commit-buffer**	Commits the transaction to the system configuration.

The following example deletes the Fibre Channel adapter policy named FcPolicy42:

```
UCS-A# scope org /
UCS-A /org* # delete fc-profile FcPplicy42
UCS-A /org* # commit-buffer
UCS-A /org #
```

PART V

Server Configuration

CHAPTER 20

Configuring Server-Related Pools

This chapter includes:

Server Pool Configuration

Server Pools

A server pool contains a set of servers. These servers typically share the same characteristics. Those characteristics can be their location in the chassis, or an attribute such as server type, amount of memory, local storage, type of CPU, or local drive configuration. You can manually assign a server to a server pool, or use server pool policies and server pool policy qualifications to automate the assignment.

If your system implements multi-tenancy through organizations, you can designate one or more server pools to be used by a specific organization. For example, a pool that includes all servers with two CPUs could be assigned to the Marketing organization, while all servers with 64GB memory could be assigned to the Finance organization.

A server pool can include servers from any chassis in the system. A given server can belong to multiple server pools.

Configuring a Server Pool

Procedure

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

	Command or Action	Purpose
Step 2	UCS-A /org # **create server-pool** *server-pool-name*	Creates a server pool with the specified name, and enters organization server pool mode.
Step 3	UCS-A /org/server-pool # **create server** *chassis-num* / *slot-num*	Creates a server for the server pool. **Note** A server pool can contain more than one server. To create multiple servers for the pool, you must enter multiple **create server** commands from organization server pool mode.
Step 4	UCS-A /org/server-pool # **commit-buffer**	Commits the transaction to the system configuration.

The following example creates a server pool named ServPool2, creates two servers for the server pool, and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # create server-pool ServPool2
UCS-A /org/server-pool* # create server 1/1
UCS-A /org/server-pool* # create server 1/4
UCS-A /org/server-pool* # commit-buffer
UCS-A /org/server-pool #
```

Deleting a 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 # **delete server-pool** *server-pool-name*	Deletes the specified server pool.
Step 3	UCS-A /org # **commit-buffer**	Commits the transaction to the system configuration.

The following example deletes the server pool named ServPool2 and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # delete server-pool ServPool2
UCS-A /org* # commit-buffer
UCS-A /org #
```

UUID Suffix Pool Configuration

UUID Suffix Pools

A UUID suffix pool is a collection of SMBIOS UUIDs that are available to be assigned to servers. The first number of digits that constitute the prefix of the UUID are fixed. The remaining digits, the UUID suffix, is variable. A UUID suffix pool ensures that these variable values are unique for each server associated with a service profile which uses that particular pool to avoid conflicts.

If you use UUID suffix pools in service profiles, you do not have to manually configure the UUID of the server associated with the service profile.

Configuring a UUID Suffix 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 # **create uuid-suffix-pool** *pool-name*	Creates a UUID suffix pool with the specified pool name and enters organization UUID suffix pool mode.
Step 3	UCS-A /org/uuid-suffix-pool # **set descr** *description*	(Optional) Provides a description for the UUID suffix 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/uuid-suffix-pool # **create block** *first-uuid last-uuid*	Creates a block (range) of UUID suffixes, and enters organization UUID suffix pool block mode. You must specify the first and last UUID suffixes in the block using the form *nnnn-nnnnnnnnnnnn*, with the UUID suffixes separated by a space. **Note** A UUID suffix pool can contain more than one UUID suffix block. To create multiple blocks, you must enter multiple **create block** commands from organization UUID suffix pool mode.
Step 5	UCS-A /org/uuid-suffix-pool/block # **commit-buffer**	Commits the transaction to the system configuration.

The following example creates a UUID suffix pool named pool4, provides a description for the pool, and specifies a block of UUID suffixes to be used for the pool:

```
UCS-A# scope org /
UCS-A /org* # create uuid-suffix-pool pool4
UCS-A /org/uuid-suffix-pool* # set descr "This is UUID suffix pool 4"
UCS-A /org/uuid-suffix-pool* # create block 1000-000000000001 1000-000000000010
UCS-A /org/uuid-suffix-pool/block* # commit-buffer
UCS-A /org/uuid-suffix-pool/block #
```

What to Do Next

Include the UUID suffix pool in a service profile and/or template.

Deleting a UUID Suffix 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 # **delete uuid-suffix-pool** *pool-name*	Deletes the specified UUID suffix pool.
Step 3	UCS-A /org # **commit-buffer**	Commits the transaction to the system configuration.

The following example deletes the UUID suffix pool named pool4 and commits the transaction:

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

Management IP Pool Configuration

Management IP Pool

The management IP pool is a collection of external IP addresses. Cisco UCS Manager reserves each block of IP addresses in the management IP pool for external access that terminates in the server controller (BMC) in a server.

Cisco UCS Manager uses the IP addresses in a management IP pool for external access through serial over LAN and IPMI.

Configuring an IP Address Block for the Management IP 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 ip-pool ext-mgmt**	Enters organization IP pool mode. **Note** You cannot create (or delete) a management IP pool. You can only enter (scope to) the existing default pool.
Step 3	UCS-A /org/mac-pool # **set descr** *description*	(Optional) Provides a description for the management IP pool.

	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/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** A IP pool can contain more than one IP address block. To create multiple IP address blocks, you must enter multiple **create block** commands from organization IP pool mode.
Step 5	UCS-A /org/ip-pool/block # **commit-buffer**	Commits the transaction to the system configuration.

The following example configures an IP address block for the management IP pool:

```
UCS-A# scope org /
UCS-A /org # scope ip-pool ext-mgmt
UCS-A /org/ip-pool* # set descr "This is a management IP pool example."
UCS-A /org/ip-pool* # create block 192.168.100.1 192.168.200.1 192.168.100.10 255.255.248.0
UCS-A /org/ip-pool/block* # commit-buffer
UCS-A /org/ip-pool/block #
```

Deleting an IP Address Block from the Management IP 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 ip-pool ext-mgmt**	Enters the management IP pool.
Step 3	UCS-A /org/ip-pool # **delete block** *first-ip-addr last-ip-addr*	Deletes the specified block (range) of IP addresses.
Step 4	UCS-A /org/ip-pool # **commit-buffer**	Commits the transaction to the system configuration.

The following example configures an IP address block for the management IP pool:

```
UCS-A# scope org /
UCS-A /org # scope ip-pool ext-mgmt
UCS-A /org/ip-pool* # delete block 192.168.100.1 192.168.200.1
UCS-A /org/ip-pool* # commit-buffer
UCS-A /org/ip-pool #
```

CHAPTER 21

Configuring Server-Related Policies

This chapter includes:

Server Autoconfiguration Policy Configuration

Server Autoconfiguration Policy

This policy determines whether one or more of the following is automatically applied to a new server:

- A server pool policy qualification that qualifies the server for one or more server pools
- An organization
- A service profile template that associates the server with a service profile created from that template

Prerequisites

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	UCS-A /org/server-autoconfig-policy # **set descr** *description*	(Optional) 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/server-autoconfig-policy # **set destination org** *org-name*	(Optional) Specifies the organization for which the server is to be used.
Step 5	UCS-A /org/server-autoconfig-policy # **set qualifier** *server-qual-name*	(Optional) Specifies server pool policy qualification to use for qualifying the server.
Step 6	UCS-A /org/server-autoconfig-policy # **set template** *profile-name*	(Optional) 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.

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.

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 #
```

Boot Policy Configuration

Boot Policy

This policy determines the following:

- Configuration of the boot device
- Location from which the server boots
- Order in which boot devices are invoked

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

You must include this policy 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, the server uses the default settings in the BIOS to determine the boot order.

Important Changes to a boot policy may be propagated to all servers created with an updating service profile template that includes that boot policy. Reassociation of the service profile with the server to rewrite the boot order information in the BIOS is auto-triggered.

Guidelines

When you create a boot policy, you can add one or more of the following to the boot policy and specify their boot order:

Boot type	Description
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, 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 exact same operating system image. Therefore, the new server appears to be the exact same server to the network.
LAN boot	Boots from a centralized provisioning server. It is frequently used to install operating systems on a server from that server.
Local disk boot	If the server has a local drive, boots from that drive.
Virtual media boot	Mimics the insertion of a physical CD-ROM disk (read-only) or floppy disk (read-write) into a server. It is typically used to manually install operating systems on a server.

Note The default boot order is as follows:

1. Local disk boot
2. LAN boot
3. Virtual media read-only boot
4. Virtual media read-write boot

Configuring 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 # **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

	Command or Action	Purpose
		and should only be used if instructed to do so by a Cisco representative.
Step 3	UCS-A /org/boot-policy # **set descr** *description*	(Optional) 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 will 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 # **commit-buffer**	Commits the transaction to the system configuration.

The following example creates a boot policy named boot-policy-LAN, provides a description for the boot policy, specifies that servers using this policy will not be automatically rebooted when the boot order is changed, and commits 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* # commit-buffer
UCS-A /org/boot-policy #
```

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 "Configuring a LAN Boot for a Boot Policy, page 150."

- **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.

  Cisco recommends 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 will boot from the exact same operating system image. Therefore, the new server will appear to be the exact same server to the network.

  If you choose the Storage Boot option, continue to "Configuring a Storage Boot for a Boot Policy, page 150."

- **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 Boot Policy, page 152."

Include the boot policy in a service profile and/or template.

Configuring a LAN Boot 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*.
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.

The following example creates a boot policy named boot-policy-LAN, 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 boot-policy-LAN
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/or template.

Configuring a Storage Boot for a Boot Policy

Before You Begin

Create a boot policy to contain the storage 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 # **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 # **set order** {**1** | **2** | **3** | **4**}	Sets the boot order for the storage boot.
Step 5	UCS-A /org/boot-policy/storage # **create** {**local** | **san-image** {**primary** | **secondary**}	Creates a local or SAN image storage location, and if the san-image option is specified, enters organization boot policy storage SAN image mode.
Step 6	UCS-A /org/boot-policy/storage/san-image # **set vhba** *vhba-name*	Specifies the vHBA to be used for the storage boot.
Step 7	UCS-A /org/boot-policy/storage/san-image # **create path** {**primary** | **secondary**}	Creates a primary or secondary storage boot path and enters organization boot policy LAN path mode.
Step 8	UCS-A /org/boot-policy/storage/san-image/path # **set** {**lun** *lun-id* | **wwn** *wwn-num*}	Specifies the LUN or WWN to be used for the storage path to the boot image.
Step 9	UCS-A /org/boot-policy/storage/san-image/path # **commit-buffer**	Commits the transaction to the system configuration.

The following example creates a boot policy named boot-policy-storage, creates a storage boot for the policy, sets the boot order to 1, creates a primary SAN image, uses a vHBA named vHBA2, creates primary path using LUN 967295200, and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # scope boot-policy boot-policy-storage
UCS-A /org/boot-policy* # create storage
UCS-A /org/boot-policy/storage* # set order 1
UCS-A /org/boot-policy/storage* # create san-image primary
UCS-A /org/boot-policy/storage* # set vhba vHBA2
UCS-A /org/boot-policy/storage/san-image* # create path primary
UCS-A /org/boot-policy/storage/san-image/path* # set lun 967295200
UCS-A /org/boot-policy/storage/san-image/path* # commit-buffer
UCS-A /org/boot-policy/storage/san-image/path #
```

What to Do Next

Include the boot policy in a service profile and/or template.

Configuring a Virtual Media Boot for a Boot Policy

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*.
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-write**}	Creates a virtual media boot for the boot policy, specifies whether the virtual media is has read-only or read-write privileges, and enters organization boot policy virtual media mode.
Step 4	UCS-A /org/boot-policy/virtual-media # **set order** {**1** | **2** | **3** | **4**}	Sets the boot order for the virtual-media boot.
Step 5	UCS-A /org/boot-policy/virtual-media # **commit-buffer**	Commits the transaction to the system configuration.

The following example

```
UCS-A# scope org /
UCS-A /org* # scope boot-policy boot-policy-vm
UCS-A /org/boot-policy* # create virtual-media read-only
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/or 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.

The following example deletes the boot policy named boot-policy-LAN:

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

Chassis Discover Configuration

Chassis Discovery Policy

This discovery policy determines how the system reacts when you add a new chassis. If you create a chassis discovery policy, the system does the following:

- Automatically configures the chassis for the number of links between the chassis and the fabric interconnect specified in the policy.
- Specifies the power policy to be used by the chassis.

Configuring a Chassis Discovery Policy

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org /**	Enters the root organization mode. **Note** The chassis discovery policy can only be accessed from the root organization.
Step 2	UCS-A /org # **scope chassis-disc-policy**	Enters organization chassis discovery policy mode.
Step 3	UCS-A /org/chassis-disc-policy # **set action** {**1-link** | **2-link** | **4-link**}	Specifies the number of links to the fabric interconnect that the chassis must have before it can be discovered.
Step 4	UCS-A /org/chassis-disc-policy # **set descr** *description*	(Optional) Provides a description for the chassis discovery policy. **Note** If your description includes spaces, special characters, or punctuation, you must begin and end your description with quotation marks. The quotation marks will not appear in the description field of any **show** command output.
Step 5	UCS-A /org/chassis-disc-policy # **set qualifier** *qualifier*	(Optional) Uses the specified server pool policy qualifications to associates this policy with a server pool.
Step 6	UCS-A /org/chassis-disc-policy # **commit-buffer**	Commits the transaction to the system configuration.

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

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

IPMI Access Profile Configuration

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 BMC. 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 must include this policy in a service profile and that service profile must be associated with a server for it to take effect.

Configuring an IPMI Access Profile

Before You Begin

- 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 # **create epuser** *epuser-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/epuser # **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/epuser # **set privilege** {**admin** | **readonly**}	Specifies whether the endpoint user has administrative or read-only privileges.
Step 6	UCS-A /org/ipmi-access-profile/epuser # **commit-buffer**	Commits the transaction to the system configuration.

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 epuser bob
UCS-A /org/ipmi-access-profile/epuser* # set password
Enter a password:
Confirm the password:
UCS-A /org/ipmi-access-profile/epuser* # set privilege readonly
UCS-A /org/ipmi-access-profile/epuser* # commit-buffer
UCS-A /org/ipmi-access-profile/epuser #
```

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.

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 epuser** *epuser-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/epuser # **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 5	UCS-A /org/ipmi-access-profile/epuser # **set privilege** {**admin** | **readonly**}	Specifies whether the endpoint user has administrative or read-only privileges.
Step 6	UCS-A /org/ipmi-access-profile/epuser # **commit-buffer**	Commits the transaction to the system configuration.

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 epuser alice
UCS-A /org/ipmi-access-profile/epuser* # set password
Enter a password:
Confirm the password:
UCS-A /org/ipmi-access-profile/epuser* # set privilege readonly
UCS-A /org/ipmi-access-profile/epuser* # commit-buffer
UCS-A /org/ipmi-access-profile/epuser #
```

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 epuser** *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.

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 epuser alice
UCS-A /org/ipmi-access-profile* # commit-buffer
UCS-A /org/ipmi-access-profile #
```

Local Disk Configuration Policy Configuration

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 the RAID mode and the way the drives are partitioned.

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 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*.

	Command or Action	Purpose
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	UCS-A /org/local-disk-config-policy # **set descr** *description*	(Optional) Provides a description for the local disk configuration policy.
Step 4	UCS-A /org/local-disk-config-policy # **set mode** {**no-local-storage** | **raid-mirrored** | **raid-striped** | **unspecified**}	Specifies the mode for the local disk configuration policy.
Step 5	UCS-A /org/local-disk-config-policy # **commit-buffer**	Commits the transaction to the system configuration.

The following example configures a local disk configuration policy:

```
UCS-A# scope org /
UCS-A /org* # create local-disk-config-policy DiskPolicy7
UCS-A /org/local-disk-config-policy* # set mode raid-mirrored
UCS-A /org/local-disk-config-policy* # commit-buffer
UCS-A /org/local-disk-config-policy #
```

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.

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 #
```

Scrub Policy Configuration

Scrub Policy

This policy determines what happens to local data on a server during the discovery process and when the server is disassociated from a service profile. This policy can ensure that the data on local drives is erased at those times.

Configuring 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	UCS-A /org/scrub-policy # **set descr** *description*	(Optional) 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 4	UCS-A /org/scrub-policy # **set disk-scrub** {**no** | **yes**}	Disables or enables disk scrubbing on servers using this scrub policy.
Step 5	UCS-A /org/scrub-policy # **commit-buffer**	Commits the transaction to the system configuration.

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 policy set to yes."
UCS-A /org/scrub-policy* # set disk-scrub yes
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.

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 #
```

Server Discovery Configuration

Server Discovery Policy

This discovery policy determines how the system reacts when you add a new server. 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.

With this policy, an inventory of the server is conducted, then server pool policy qualifications are run to determine whether the new server qualifies for one or more server pools.

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.

	Command or Action	Purpose
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	UCS-A /org/chassis-disc-policy # **set descr** *description*	(Optional) 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	UCS-A /org/server-disc-policy # **set qualifier** *qualifier*	(Optional) 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.

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.

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

The following example deletes the server discovery policy named ServDiscPolExample:

```
UCS-A# scope org /
UCS-A /org* # delete server-disc-policy ServDiscPolExample
UCS-A /org* # commit-buffer
UCS-A /org #
```

Server Pool Policy Configuration

Server Pool Policy

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	UCS-A /org/pooling-policy # **set descr** *description*	(Optional) 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.
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.

The following example creates a server pool policy named PoolPolicy4 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.

The following example deletes the server pool policy named PoolPolicy4 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 Configuration

Server Pool Policy Qualifications

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.

Depending upon the implementation, you may include server pool policy qualifications in the following policies:

- 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*.
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.

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
- 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.

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 #
```

Configuring 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 • 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	Specifies the maximum capacity for the selected adapter type.

	Command or Action	Purpose
	# **set maximum** {*max-cap* | **unspecified**}	
Step 6	UCS-A /org/server-qual/adapter/cap-qual # **commit-buffer**	Commits the transaction to the system configuration.

The following example 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.
Step 4	UCS-A /org/server-qual # **commit-buffer**	Commits the transaction to the system configuration.

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.

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.

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 #
```

Configuring 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 PCB).
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.

The following example 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.

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 #
```

Configuring a Processor 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 processor**	Creates a processor qualification and enters organization server qualification processor mode.
Step 4	UCS-A /org/server-qual/processor # **set arch** {**any** | **dual-core-opteron** | **intel-p4-c** | **opteron** | **pentium-4** | **turion-64** | **xeon** | **xeon-mp**}	Specifies the processor architecture type.

	Command or Action	Purpose
Step 5	UCS-A /org/server-qual/processor # **set maxcores** {*max-core-num* | **unspecified**}	Specifies the maximum number of processor cores.
Step 6	UCS-A /org/server-qual/processor # **set mincores** {*min-core-num* | **unspecified**}	Specifies the minimum number of processor cores.
Step 7	UCS-A /org/server-qual/processor # **set maxprocs** {*max-proc-num* | **unspecified**}	Specifies the maximum number of processors.
Step 8	UCS-A /org/server-qual/processor # **set minprocs** {*min-proc-num* | **unspecified**}	Specifies the minimum number of processors.
Step 9	UCS-A /org/server-qual/processor # **set maxthreads** {*max-thread-num* | **unspecified**}	Specifies the maximum number of threads.
Step 10	UCS-A /org/server-qual/processor # **set minthreads** {*min-thread-num* | **unspecified**}	Specifies the minimum number of threads.
Step 11	UCS-A /org/server-qual/processor # **set stepping** {*step-num* | **unspecified**}	Specifies the processor stepping number.
Step 12	UCS-A /org/server-qual/processor # **commit-buffer**	Commits the transaction to the system configuration.

The following example configures a processor 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/processor* # set arch xeon
UCS-A /org/server-qual/processor* # set maxcores 8
UCS-A /org/server-qual/processor* # set mincores 4
UCS-A /org/server-qual/processor* # set maxprocs 2
UCS-A /org/server-qual/processor* # set minprocs 1
UCS-A /org/server-qual/processor* # set maxthreads 16
UCS-A /org/server-qual/processor* # set minthreads 8
UCS-A /org/server-qual/processor* # set stepping 5
UCS-A /org/server-qual/processor* # commit-buffer
UCS-A /org/server-qual/processor #
```

Deleting a Processor 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 processor**	Deletes the processor qualification.

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

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 processor
UCS-A /org/server-qual* # commit-buffer
UCS-A /org/server-qual #
```

Configuring 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* | **unspecified**}	Specifies the storage block size.
Step 5	UCS-A /org/server-qual/storage # **set maxcap** {*max-cap-num* | **unspecified**}	Specifies the maximum capacity of the storage array.
Step 6	UCS-A /org/server-qual/storage # **set mincap** {*min-cap-num* | **unspecified**}	Specifies the minimum capacity of the storage array.
Step 7	UCS-A /org/server-qual/storage # **set numberofblocks** {*block-num* | **unspecified**}	Specifies the number of blocks.
Step 8	UCS-A /org/server-qual/storage # **set perdiskcap** {*disk-cap-num* | **unspecified**}	Specifies the per-disk capacity.
Step 9	UCS-A /org/server-qual/storage # **set units** {*unit-num* | **unspecified**}	Specifies the number of storage units.
Step 10	UCS-A /org/server-qual/storage # **commit-buffer**	Commits the transaction to the system configuration.

The following example configures 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 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* # 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.

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 #
```

Server Inheritance Policy Configuration

Server Inheritance Policy

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

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	UCS-A /org/server-inherit-policy # **set descr** *description*	(Optional) 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/server-inherit-policy # **set destination org** *org-name*	(Optional) Specifies the organization for which the server is to be used.
Step 5	UCS-A /org/server-inherit-policy # **set qualifier** *server-qual-name*	(Optional) 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.

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.

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 #
```

SOL Policy Configuration

Serial over LAN Policy

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 SOL 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 (SOL) policy and enters organization SOL policy mode.
Step 3	UCS-A /org/sol-policy # **set descr** *description*	(Optional) 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	UCS-A /org/sol-policy # **set speed** {**115200** | **19200** | **38400** | **57600** | **9600**}	Specifies the serial baud rate.
Step 5	UCS-A /org/sol-policy # {**disable** | **enable**}	Disables or enables the SOL policy. By default, the SOL 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.

The following example creates a SOL policy named Sol9600, provides a description for the policy, sets the speed to 9,600 baud, enables the policy, and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # create sol-policy Sol9600
UCS-A /org/sol-policy* # set descr "Sets SOL policy to 9600 baud."
UCS-A /org/sol-policy* # set speed 9600
UCS-A /org/sol-policy* # enable
UCS-A /org/sol-policy* # commit-buffer
UCS-A /org/sol-policy #
```

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

The following example deletes the SOL policy named Sol9600 and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # delete sol-policy Sol9600
UCS-A /org* # commit-buffer
UCS-A /org #
```

CHAPTER 22

Configuring Service Profiles

This chapter includes:

Service Profiles that Inherit Server Identity

This type of service profile is the simplest to use and create. This profile mimics the management of a rack mounted server. It is tied to a specific server and cannot be moved to another server.

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

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

- MAC addresses for the two NICs
- For the Cisco UCS CNA M71KR adapters, the WWN addresses for the two HBAs
- BIOS versions

- Server UUID

Important The server identity and configuration information inherited through this service profile may not be the values burned into the server hardware at manufacture if those values have been subsequently changed before this profile is associated with the server.

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 and then associated 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 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:

- 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

Service Profile Templates

Service profile templates enable you to create a large number of similar service profiles. With a service profile template, you can quickly create several service profiles with the same basic parameters, such as the number of vNICs and vHBAs, and with identity information drawn from the same pools.

Tip If you need only one service profile with similar values to an existing service profile, you can clone a service profile in the Cisco UCS Manager GUI.

For example, if you need several service profiles with similar values to configure servers to host database software, you can create a service profile template, either manually or from an existing service profile. You then use the template to create the service profiles.

Cisco UCS supports the following types of service profile templates:

Initial template Service profiles created from an initial template inherit all of the properties of the template. However, after you create the profile, it is no longer connected to the template. If you need to make changes to one or more profiles created from this template, you must change each profile individually.

Updating template	Service profiles created from an updating template inherit all properties of the template and remain connected to the template. Any changes to the template automatically update the service profiles created from the template.

Configuring a Service Profile Template

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* {**initial-template** | **updating-template**}	Creates the specified service profile template and enters organization service profile mode.
Step 3	UCS-A /org/service-profile # **set agent-policy** *policy-name*	Associates the specified agent 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	UCS-A /org/service-profile # **set descr** *description*	(Optional) 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 host-fw-policy** *policy-name*	Associates the specified host firmware policy with the service profile.
Step 7	UCS-A /org/service-profile # **set identity** {**uuid-pool** *pool-name* | **wwnn-pool** *pool-name*}	Specifies the pool that the server uses to acquire a UUID or WWNN.
Step 8	UCS-A /org/service-profile # **set ipmi-access-profile** *profile-name*	Associates the specified IPMI access profile with the service profile.
Step 9	UCS-A /org/service-profile # **set local-disk-policy** *policy-name*	Associates the specified local disk policy with the service profile.
Step 10	UCS-A /org/service-profile # **set mgmt-fw-policy** *policy-name*	Associates the specified management firmware policy with the service profile.
Step 11	UCS-A /org/service-profile # **set scrub-policy** *policy-name*	Associates the specified scrub policy with the service profile.

	Command or Action	Purpose
Step 12	UCS-A /org/service-profile # **set sol-policy** *policy-name*	Associates the specified serial over LAN policy with the service profile.
Step 13	UCS-A /org/service-profile # **set stats-policy** *policy-name*	Associates the specified statistics policy with the service profile.
Step 14	UCS-A /org/service-profile # **commit-buffer**	Commits the transaction to the system configuration.

The following example creates a service profile template and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # create service-profile ServTemp2 updating-template
UCS-A /org/service-profile* # set agent-policy AgentPol933
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 host-fw-policy Epuser987
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 mgmt-fw-policy MgmtFwPol75
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* # 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.
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

	Command or Action	Purpose
		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.

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 or server pool.

Configuring a Service Profile Instance without Using a Template

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.
Step 3	UCS-A /org/service-profile # **set agent-policy** *policy-name*	Associates the specified agent 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	UCS-A /org/service-profile # **set descr** *description*	(Optional) 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 host-fw-policy** *epuser-name*	Associates the specified host forwarding policy with the service profile.
Step 7	UCS-A /org/service-profile # **set identity** {**dynamic-uuid** {*uuid* |	Specifies how the server acquires a UUID or WWNN. You can do one of the following:

	Command or Action	Purpose
	derived} | **dynamic-wwnn** {*wwnn* | **derived**} | **uuid-pool** *pool-name* | **wwnn-pool** *pool-name*}	• 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 8	UCS-A /org/service-profile # **set ipmi-access-profile** *profile-name*	Associates the specified IPMI access profile with the service profile.
Step 9	UCS-A /org/service-profile # **set local-disk-policy** *policy-name*	Associates the specified local disk policy with the service profile.
Step 10	UCS-A /org/service-profile # **set mgmt-fw-policy** *policy-name*	Associates the specified management forwarding policy with the service profile.
Step 11	UCS-A /org/service-profile # **set scrub-policy** *policy-name*	Associates the specified scrub policy with the service profile.
Step 12	UCS-A /org/service-profile # **set sol-policy** *policy-name*	Associates the specified serial over LAN policy with the service profile.
Step 13	UCS-A /org/service-profile # **set stats-policy** *policy-name*	Associates the specified statistics policy with the service profile.
Step 14	UCS-A /org/service-profile # **commit-buffer**	Commits the transaction to the system configuration.

The following example creates a service profile instance and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # create service-profile ServInst90 instance
UCS-A /org/service-profile* # set agent-policy AgentPol933
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 host-fw-policy Epuser987
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 mgmt-fw-policy MgmtFwPol75
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* # 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 to a server or server pool.

Configuring a vNIC 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 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 adaptor-profile** *policy-name*	Specifies the adapter policy to use for the vNIC.
Step 5	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 6	UCS-A /org/service-profile/vnic # **set order** {*order-num* | **unspecified**}	Specifies the PCI scan order for the vNIC.
Step 7	UCS-A /org/service-profile/vnic # **set pin-group** *group-name*	Specifies the pin group to use for the vNIC.
Step 8	UCS-A /org/service-profile/vnic # **set qos-policy** *policy-name*	Specifies the QoS policy to use for the vNIC.
Step 9	UCS-A /org/service-profile/vnic # **set stats-policy** *policy-name*	Specifies the stats policy to use for the vNIC.
Step 10	UCS-A /org/service-profile/vnic # **set template-name** *policy-name*	Specifies the dynamic vNIC connectivity policy to use for the vNIC.

	Command or Action	Purpose
Step 11	UCS-A /org/service-profile/vnic # **commit-buffer**	Commits the transaction to the system configuration.

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 vhba vhba3 fabric a
UCS-A /org/service-profile/vnic* # set adaptor-profile AdaptPol2
UCS-A /org/service-profile/vnic* # set identity mac-pool MacPool3
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* # 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 adaptor-profile** *policy-name*	Specifies the adapter policy to use for the vHBA.
Step 5	UCS-A /org/service-profile/vhba # **set identity** {**dynamic-wwpn** {*wwpn* | **derived**} | **wwpn-pool** *wwn-pool-name*}	Specifies the storage identity (world wide port name [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*. • Derive the WWPN from one burned into the hardware at manufacture. • Assign a WWPN from a WWN pool.
Step 6	UCS-A /org/service-profile/vhba # **set order** {*order-num* | **unspecified**}	Specifies the PCI scan order for the vHBA.

	Command or Action	Purpose
Step 7	UCS-A /org/service-profile/vhba # **set pers-bind** {**disabled** | **enabled**}	Disables or enables persistent binding to fibre channel targets.
Step 8	UCS-A /org/service-profile/vhba # **set pin-group** *group-name*	Specifies the pin group to use for the vHBA.
Step 9	UCS-A /org/service-profile/vhba # **set qos-policy** *policy-name*	Specifies the QoS policy to use for the vHBA.
Step 10	UCS-A /org/service-profile/vhba # **set stats-policy** *policy-name*	Specifies the stats policy to use for the vHBA.
Step 11	UCS-A /org/service-profile/vhba # **set template-name** *policy-name*	Specifies the vHBA SAN connectivity policy to use for the vHBA.
Step 12	UCS-A /org/service-profile/vhba # **commit-buffer**	Commits the transaction to the system configuration.

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 adaptor-profile AdaptPol2
UCS-A /org/service-profile/vhba* # set identity wwpn-pool SanPool7
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	UCS-A /org/service-profile/local-disk-config # **set descr** *description*	(Optional) Provides a description for the local disk configuration.

	Command or Action	Purpose
Step 5	UCS-A /org/service-profile/local-disk-config # **set mode** {**no-local-storage** | **raid-mirrored** | **raid-striped** | **unspecified**}	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	UCS-A /org/service-profile/local-disk-config/partition # **set descr** *description*	(Optional) 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.

The following example configures a local disk for a service profile:

```
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-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 SOL 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 (SOL) configuration for the service profile and enters organization service profile SOL configuration mode.

	Command or Action	Purpose
Step 4	UCS-A /org/service-profile/sol-config # **set admin-state** {**disable** | **enable**}	Disables or enables the SOL administrative state.
Step 5	UCS-A /org/service-profile/sol-config # **set descr** *description*	(Optional) Provides a description for the SOL 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.

The following example configures SOL 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 sol-config Sol9600
UCS-A /org/service-profile/sol-config* # set admin-state enable
UCS-A /org/service-profile/sol-config* # set descr "Sets SOL 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 Boot Definition Configuration

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 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	UCS-A /org/service-profile/boot-definition # **set descr** *description*	(Optional) Provides a description for the boot definition.
Step 5	UCS-A /org/service-profile/boot-definition # **set reboot-on-update** {**no** | **yes**}	(Optional) 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.

	Command or Action	Purpose
Step 6	UCS-A /org/service-profile/boot-definition # **commit-buffer**	Commits the transaction to the system configuration.

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 , page 188."

- **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.

  Cisco recommends 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 will boot from the exact same operating system image. Therefore, the new server will appear to be the exact same server to the network.

  If you choose the Storage Boot option, continue to "Configuring a Storage Boot for a Service Profile Boot Definition , page 189."

- **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 , page 191."

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 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.

The following example configures a LAN boot for a service profile boot definition:

```
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*.

	Command or Action	Purpose
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, then 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.
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.

The following example configures a storage boot for a service profile boot definition:

```
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/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.

The following example configures a read-only virtual media boot for a service profile boot definition:

```
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 1
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*.

	Command or Action	Purpose
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.

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 #
```

Associating a Service Profile with a Server or Server Pool

Follow this procedure if you did not associate the service profile with a server or server pool when you created it, or to change the 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** *chassis-id* / *slot-id* | **server-pool** *pool-name qualifier*}	Associates the service profile with a single server, or to the specified server pool with the specified server pool policy qualifications.
Step 4	UCS-A /org/service-profile # **commit-buffer**	Commits the transaction to the system configuration.

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 #
```

Disassociating a Service Profile from 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 # **disassociate**	Disassociates the service profile from a server.
Step 4	UCS-A /org/service-profile # **commit-buffer**	Commits the transaction to the system configuration.

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 #
```

CHAPTER 23

Installing an OS on a Server

This chapter includes:

OS Installation Methods

Servers in the Cisco UCS support several operating systems, including Windows- and Linux-based operating systems. Regardless of the OS being installed, you can install it on a server using one of the following methods:

- PXE install server
- KVM dongle directly connected to the server
- KVM console in the UCS Manager GUI
- Third-party tool (not covered in this document)

PXE Install Server

A Preboot Execution Environment (PXE) install server allows clients (servers) to boot and install an OS over the network. To use this method, a PXE environment must be configured and available on a VLAN, typically a dedicated provisioning VLAN, and a client server must be set to boot from the network. When a client server boots, it sends a PXE request across the network, and the PXE install server acknowledges the request and starts a sequence of events that installs the OS on the client server.

PXE servers can use installation disks, disk images, and scripts to install the OS. Proprietary disk images can also be used install an OS and additional components or applications.

PXE installation is an efficient method for consistently installing an OS on a large number of servers. However, considering that this method requires configuring a PXE environment, if you do not already have an PXE install server set up, it might be easier to use one of the other installation methods if you are installing an OS on only one or two servers,

KVM Dongle

The KVM dongle plugs into the front of a server and allows you to directly connect a keyboard, video monitor, mouse, and USB CD/DVD or floppy drive to the server. This direct access to the server allows you to locally install an OS.

To install an OS from a CD/DVD or floppy drive connected to the USB port, you must ensure that the CD/DVD or floppy drive is set as the first boot device in the service profile.

KVM Console

The KVM console is an interface accessible from the Cisco UCS Manager GUI that emulates a direct KVM connection. Unlike the KVM dongle, which requires you to be physically connected to the server, the KVM console allows you to connect to the server from a remote location across the network.

Instead of using CD/DVD or floppy drives directly connected to the server, the KVM console uses virtual media, which are actual disk drives or disk image files that are mapped to virtual CD/DVD or floppy drives. You can map any of the following to virtual drives:

- CD/DVD or floppy drives on your computer
- Disk image files on your computer
- CD/DVD or floppy drives on the network
- Disk image files on the network

To install an OS from a virtual CD/DVD or floppy drive, you must ensure that the virtual CD/DVD or floppy drive is set as the first boot device in the service profile.

Installing an OS using the KVM console may be slower than using the KVM dongle because the installation files must be downloaded across the network to the server. If you map a disk drive or disk image file from a network share to a virtual drive, then the installation may be even slower because the installation files must be downloaded from the network to the KVM console (your computer) and then from the KVM console to the server. When using this installation method, we recommend that you have the installation media as close as possible to the system with the KVM console.

Installation Targets

The installation target is the location where you install the OS. The UCS server has two possible installation targets: a local hard drive or a SAN LUN. During the OS installation process, drivers for the local disk controller or HBA must be loaded so that the installer can find the drives. If the installer cannot find any drives, then the drivers were probably not loaded. Newer OS installation disks should have the drivers; however, older OS installation disks may not have them.

If your OS installation disk does not have the needed drivers, you must provide them during the installation process. For local drives, you need LSI controller drivers, and for HBAs you need Emulex or Qlogic drivers.

Installing an OS Using a PXE Installation Server

Before You Begin

- Verify that a PXE installation environment has been configured to install the appropriate OS, and that the client server can be reached over a VLAN.
- Verify that a service profile is associated with the server onto which the OS is being installed.

Procedure

Step 1 Depending on whether your service profile is associated with a boot policy, or contains a local boot definition, perform one of the following:

a) Set the boot order for the associated boot policy to boot from the LAN first.

```
scope org org-name
scope boot-policy policy-name
scope lan
set order 1
commit-buffer
```

For more information about configuring a boot policy, see "Configuring a Boot Policy, page 148."

b) Set the boot order for the local boot definition to boot from the LAN first.

```
scope org org-name
scope service-profile profile-name
scope boot-definition
scope lan
set order 1
commit-buffer
```

For more information about configuring a local boot definition for a service profile, see "Configuring a Boot Definition for a Service Profile, page 187."

Step 2 Reboot the server.

```
scope server chassis-num/slot-num
reset hard-reset-immediate
commit-buffer
```

If a PXE install server is available on a VLAN, the installation process begins when the server reboots. PXE installations are typically automated and require no additional user input. Refer to the installation guide for the OS being installed to guide you through the rest of the installation process.

What to Do Next

After the OS installation is complete, reset the LAN boot order to its original setting.

Installing an OS Using the KVM Dongle

Before You Begin

- Locate the following items:
  - USB keyboard and mouse
  - Video monitor
  - USB CD/DVD drive
  - USB floppy drive (optional)
  - OS installation disk or disk image file
- Verify that a service profile is associated with the server onto which the OS is being installed.

Procedure

Step 1 Connect the KVM dongle to the front of the server.

Step 2 Connect the keyboard, video monitor, mouse, USB CD/DVD drive, and optionally a USB floppy drive to the KVM console.

Note The USB dongle contains only two USB ports. To connect more than two USB devices to the dongle, first connect a USB hub to the dongle and then connect your USB devices to the hub.

Step 3 Load the OS installation disk into the USB CD/DVD drive connected to the dongle.

Step 4 Log in to the Cisco UCS Manager CLI.

Step 5 Depending on whether your service profile is associated with a boot policy, or contains a local boot definition, perform one of the following:

a) Set the boot order for the associated boot policy to boot from the virtual media first.

```
scope org org-name
scope boot-policy policy-name
scope virtual-media read-only
set order 1
commit-buffer
```

For more information about configuring a boot policy, see "Configuring a Boot Policy, page 148."

b) Set the boot order for the local boot definition to boot from the virtual media first.

```
scope org org-name
scope service-profile profile-name
scope boot-definition
scope virtual-media read-only
set order 1
commit-buffer
```

For more information about configuring a local boot definition for a service profile, see "Configuring a Boot Definition for a Service Profile, page 187."

Step 6 Reboot the server.

```
scope server chassis-num/slot-num
reset hard-reset-immediate
```

```
commit-buffer
```

When the server reboots, it begins the installation process from the CD/DVD drive. Refer to the installation guide for the OS being installed to guide you through the rest of the installation process.

What to Do Next

After the OS installation is complete, reset the virtual media boot order to its original setting.

Installing an OS Using the KVM Console

Before You Begin

- Locate the OS installation disk or disk image file.
- Verify that a service profile is associated with the server onto which the OS is being installed.

Procedure

Step 1 Load the OS installation disk into your CD/DVD drive, or copy the disk image files to your computer.

Step 2 Log in to the Cisco UCS Manager GUI.

Step 3 In the **Navigation** pane, click the **Servers** tab.

Step 4 In the **Servers** tab, expand **Service Profiles**.

Step 5 Expand the node for the organization that contains the service profile associated to the server on which the OS is being installed and click the service profile.
If the system does not include multi-tenancy, expand the root node and click the service profile.

Step 6 In the **Work** pane, click the **General** tab.

Step 7 In the **Actions** area of the **General** tab, click **KVM Console**.

Step 8 From the KVM console, choose **Tools ➤ Launch Virtual Media** to open the Virtual Media Session dialog box.

Step 9 In the Virtual Media Session dialog box, map the virtual media using either of the following methods:

- Check the **Mapped** checkbox for the CD/DVD drive containing the OS installation disk.
- Click **Add Image**, navigate to and select the OS installation disk image, click **Open** to mount the disk image, and then check the **Mapped** checkbox for the mounted disk image.

Note You must keep the Virtual Media Session dialog box open during the OS installation process; closing the dialog box unmaps all virtual media.

Step 10 Log in to the Cisco UCS Manager CLI.

Step 11 Depending on whether your service profile is associated with a boot policy, or contains a local boot definition, perform one of the following:

a) Set the boot order for the associated boot policy to boot from the virtual media first.

```
scope org org-name
scope boot-policy policy-name
scope virtual-media read-only
```

```
set order 1
commit-buffer
```

For more information about configuring a boot policy, see "Configuring a Boot Policy, page 148."

b) Set the boot order for the local boot definition to boot from the virtual media first.

```
scope org org-name
scope service-profile profile-name
scope boot-definition
scope virtual-media read-only
set order 1
commit-buffer
```

For more information about configuring a local boot definition for a service profile, see "Configuring a Boot Definition for a Service Profile, page 187."

Step 12 Reboot the server.

```
scope server chassis-num/slot-num
reset hard-reset-immediate
commit-buffer
```

When the server reboots, it begins the installation process from the virtual CD/DVD drive. Refer to the installation guide for the OS being installed to guide you through the rest of the installation process.

What to Do Next

After the OS installation is complete, reset the virtual media boot order to its original setting.

PART VI

System Management

CHAPTER 24

Managing Time Zones

This chapter includes:

Time Zones

Cisco UCS requires an instance-specific time zone setting and an NTP server to ensure the correct time display in Cisco UCS Manager. If you do not configure both of these settings in a Cisco UCS instance, the time does not display correctly.

Setting the Time Zone

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system**	Enters system mode.
Step 2	UCS-A /system # **scope services**	Enters system services mode.
Step 3	UCS-A /system/services # **set timezone**	
Step 4	At this point, you are prompted to enter a number corresponding to your continent, country, and time zone region. Enter the appropriate information at each prompt.	
Step 5	When you have finished specifying the location information, you are prompted to confirm that the correct time zone information is being set. Enter **1** (yes) to confirm, or **2** (no) to cancel the operation.	

The following example configures the timezone to the Americas continent, United States country, and Pacific time zone region:

```
UCS-A# scope system
UCS-A /system # scope services
UCS-A /system/services # set timezone
Please identify a location so that time zone rules can be set correctly.
Please select a continent or ocean.
1) Africa            4) Arctic Ocean     7) Australia       10) Pacific Ocean
2) Americas          5) Asia             8) Europe
3) Antarctica        6) Atlantic Ocean   9) Indian Ocean
#? Artic ocean
Please enter a number in range.
#? 2
Please select a country.
 1) Anguilla              18) Ecuador               35) Paraguay
 2) Antigua & Barbuda     19) El Salvador           36) Peru
 3) Argentina             20) French Guiana         37) Puerto Rico
 4) Aruba                 21) Greenland             38) St Kitts & Nevis
 5) Bahamas               22) Grenada               39) St Lucia
 6) Barbados              23) Guadeloupe            40) St Pierre & Miquelon
 7) Belize                24) Guatemala             41) St Vincent
 8) Bolivia               25) Guyana                42) Suriname
 9) Brazil                26) Haiti                 43) Trinidad & Tobago
10) Canada                27) Honduras              44) Turks & Caicos Is
11) Cayman Islands        28) Jamaica               45) United States
12) Chile                 29) Martinique            46) Uruguay
13) Colombia              30) Mexico                47) Venezuela
14) Costa Rica            31) Montserrat            48) Virgin Islands (UK)
15) Cuba                  32) Netherlands Antilles  49) Virgin Islands (US)
16) Dominica              33) Nicaragua
17) Dominican Republic    34) Panama
#? 45
Please select one of the following time zone regions.
 1) Eastern Time
 2) Eastern Time - Michigan - most locations
 3) Eastern Time - Kentucky - Louisville area
 4) Eastern Time - Kentucky - Wayne County
 5) Eastern Standard Time - Indiana - most locations
 6) Eastern Standard Time - Indiana - Crawford County
 7) Eastern Standard Time - Indiana - Starke County
 8) Eastern Standard Time - Indiana - Switzerland County
 9) Central Time
10) Central Time - Michigan - Wisconsin border
11) Central Time - North Dakota - Oliver County
12) Mountain Time
13) Mountain Time - south Idaho & east Oregon
14) Mountain Time - Navajo
15) Mountain Standard Time - Arizona
16) Pacific Time
17) Alaska Time
18) Alaska Time - Alaska panhandle
19) Alaska Time - Alaska panhandle neck
20) Alaska Time - west Alaska
21) Aleutian Islands
22) Hawaii
#? 16

The following information has been given:

        United States
        Pacific Time

Therefore timezone 'America/Los_Angeles' will be set.
Local time is now:      Fri May 15 07:39:25 PDT 2009.
Universal Time is now:  Fri May 15 14:39:25 UTC 2009.
Is the above information OK?
1) Yes
2) No
#? 1
UCS-A /system/services #
```

Adding an NTP Server

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system**	Enters system mode.
Step 2	UCS-A /system # **scope services**	Enters system services mode.
Step 3	UCS-A /system/services # **create ntp-server** {*hostname* | *ip-addr*}	Configures the system to use the NTP server with the specified hostname or IP address.
Step 4	UCS-A /system/services # **commit-buffer**	Commits the transaction to the system configuration.

The following example configures an NTP server with the IP address 192.168.200.101

```
UCS-A# scope system
UCS-A /system # scope services
UCS-A /system/services # create ntp-server 192.168.200.101
UCS-A /system/services* # commit-buffer
UCS-A /system/services #
```

Deleting an NTP Server

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system**	Enters system mode.
Step 2	UCS-A /system # **scope services**	Enters system services mode.
Step 3	UCS-A /system/services # **delete ntp-server** {*hostname* | *ip-addr*}	Deletes the specified NTP server.

The following example deletes the NTP server with the IP address 192.168.200.101

```
UCS-A# scope system
UCS-A /system # scope services
UCS-A /system/services # delete ntp-server 192.168.200.101
UCS-A /system/services* # commit-buffer
UCS-A /system/services #
```

CHAPTER 25

Managing the Chassis

This chapter includes:

Acknowledging a Chassis

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

Procedure

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

The following example acknowledges chassis 2 and commits the transaction:

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

Decommissioning a Chassis

This procedure removes the chassis from the configuration. As long as the chassis physically remains in the Cisco UCS instance, Cisco UCS Manager considers the chassis to be decommissioned and ignores it.

Procedure

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

The following example decommissions chassis 2 and commits the transaction:

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

Recommissioning a Chassis

This procedure returns the chassis to the configuration and applies the chassis discovery policy to the chassis.

Before You Begin

Collect the following information about the chassis to be recommissioned:

- Vendor name
- Model name
- Serial number

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **recommission chassis** *vendor-name model-name serial-num*	Recommissions the specified chassis.
Step 2	UCS-A# **commit-buffer**	Commits the transaction to the system configuration.

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

```
UCS-A# recommission chassis "Cisco Systems Inc" "Cisco UCS 5108" FOX1252GNNN
UCS-A* # commit-buffer
UCS-A #
```

Turning On the Locator LED for a Chassis

Procedure

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

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

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

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

Turning Off the Locator LED for a Chassis

Procedure

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

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

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

CHAPTER 26

Managing the Servers

This chapter includes:

Booting a Server

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.

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

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

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

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.

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 #
```

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# **commit-buffer**	Commits the transaction to the system configuration.

The following example immediately power cycles 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 Server

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

Procedure

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

The following example performs an immediate hard reset of 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 Server

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

Procedure

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

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

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

Removing a Server from a Chassis

Perform the following procedure when you remove a server from a chassis. Do not physically remove the server first.

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.

The following example removes 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 do not want to physically remove the server hardware, you must re-acknowledge the slot to have Cisco UCS Manager rediscover the server.

Decommissioning a Server

This procedure removes the server from the configuration. As long as the server physically remains in the Cisco UCS instance, Cisco UCS Manager considers the server to be decommissioned and ignores it.

Procedure

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

The following example decommissions server 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 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**	Turns on the server locator LED.
Step 3	UCS-A /chassis/server # **commit-buffer**	Commits the transaction to the system configuration.

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

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

Turning Off the Locator LED for a 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**	Turns off the server locator LED.
Step 3	UCS-A /chassis/server # **commit-buffer**	Commits the transaction to the system configuration.

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

Resetting the CMOS for a Server

On rare occasions, troubleshooting a server may require you to reset the CMOS. This procedure is not part of the normal maintenance of a server.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *chassis-num* / *server-num*	Enters chassis server mode for the specified chassis.
Step 2	UCS-A /chassis/server # **reset-cmos**	Resets the CMOS for the server.
Step 3	UCS-A /chassis/server # **commit-buffer**	Commits the transaction to the system configuration.

The following example resets the CMOS for server 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 BMC for a Server

On rare occasions, such as an issue with the current running firmware, troubleshooting a server may require you to reset the BMC. This procedure is not part of the normal maintenance of a server. After you reset the BMC, 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 # **scope bmc**	Enters chassis server BMC mode
Step 3	UCS-A /chassis/server/bmc # **reset**	Resets the BMC for the server.
Step 4	UCS-A /chassis/server/bmc # **commit-buffer**	Commits the transaction to the system configuration.

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

```
UCS-A# scope server 2/4
UCS-A /chassis/server* # scope bmc
UCS-A /chassis/server/bmc* # reset
```

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

CHAPTER 27

Managing the IO Modules

This chapter includes:

Resetting the IOM

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope chassis** *chassis-num*	Enters chassis mode for the specified chassis.
Step 2	UCS-A /chassis # **scope iom** {**a** **b**}	Enters chassis IOM mode for the specified IOM.
Step 3	UCS-A /chassis/iom # **reset**	Resets the IOM.
Step 4	UCS-A /chassis/iom # **commit-buffer**	Commits the transaction to the system configuration.

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

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

CHAPTER 28

Configuring Call Home

This chapter includes:

Call Home

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

Call Home provides email-based and web-based notification of critical system events. A versatile range of message formats are available for optimal compatibility with pager services or XML-based automated parsing applications. Common uses of this feature may include direct paging of a network support engineer, notification of a Network Operations Center, XML delivery to a support website, and utilization of Cisco Smart Call Home services for direct case generation with the Cisco Systems Technical Assistance Center (TAC).

The Call Home feature can deliver alert messages containing information about configuration, diagnostics, environmental conditions, inventory, and syslog events.

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

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

Cisco UCS delivers Call Home messages in the following formats:

- Short text format that is suitable for pagers or printed reports.
- XML-Matching readable format that uses Extensible Markup Language (XML) and Adaptive Messaging Language (AML) XML schema definition (XSD). The AML XSD is published on the Cisco.com website at http://www.cisco.com/. The XML format enables communication with the Cisco Systems Technical Assistance Center.

Call Home Considerations

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

- You must configure at least one destination profile. The destination profile or profiles that you use depend upon whether the receiving entity is a pager, email, or automated service such as Cisco Smart Call Home.
- If the destination profile uses email message delivery, you must specify a Simple Mail Transfer Protocol (SMTP) server when you configure Call Home.
- The contact email, phone, and street address information should be configured so that the receiver can determine the origin of messages received.
- The fabric interconnect must have IP connectivity to an email server or the destination HTTP server.
- If Cisco Smart Call Home is used, an active service contract must cover the device being configured.

Cisco Smart Call Home

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

Note Using Smart Call Home requires the following:

- A CCO ID associated with a corresponding Cisco Unified Computing Support Service or Cisco Unified Computing Mission Critical Support Service contract for your company.
- Cisco Unified Computing Support Service or Cisco Unified Computing Mission Critical Support Service for the device to be registered.

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

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

To configure Smart Call Home, you must do the following:

- Enable the Smart Call Home feature
- Configure the contact information
- Configure the email information
- Configure the SMTP server information
- Configure the default CiscoTAC-1 profile
- Send a Smart Call Home inventory message to start the registration process

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

Call Home Configuration

Configuring Call Home

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope monitoring**	Enters monitoring mode.
Step 2	UCS-A /monitoring # **scope callhome**	Enters monitoring call home mode.
Step 3	UCS-A /monitoring/callhome # **enable**	Enables Call Home.
Step 4	UCS-A /monitoring/callhome # **set contact** *name*	Specifies the name of the main Call Home contact person.
Step 5	UCS-A /monitoring/callhome # **set email** *email-addr*	Specifies the email address of the main Call Home contact person.
Step 6	UCS-A /monitoring/callhome # **set phone-contact** *phone-num*	Specifies the phone number of the main Call Home contact person. The phone number must be in international format, starting with a + (plus sign) and a country code.

	Command or Action	Purpose
Step 7	UCS-A /monitoring/callhome # **set street-address** *street-addr*	Specifies the street address of the main Call Home contact person.
Step 8	UCS-A /monitoring/callhome # **set customer-id** *id-num*	Specifies the customer identification number from the service agreement. The number can be up to 255 alphanumeric characters in free format.
Step 9	UCS-A /monitoring/callhome # **set contract-id** *id-num*	Specifies the contract identification number from the service agreement. The number can be up to 255 alphanumeric characters in free format.
Step 10	UCS-A /monitoring/callhome # **set site-id** *id-num*	Specifies the site identification number from the service agreement. The number can be up to 255 alphanumeric characters in free format.
Step 11	UCS-A /monitoring/callhome # **set from-email** *email-addr*	Specifies the email address to use for the From field in Call Home messages.
Step 12	UCS-A /monitoring/callhome # **set reply-to-email** *email-addr*	Specifies the email address to use for the Reply To field in Call Home messages.
Step 13	UCS-A /monitoring/callhome # **set hostname** {*hostname* | *ip-addr*}	Specifies the hostname or IP address of the SMTP server that Call Home uses to send email messages.
Step 14	UCS-A /monitoring/callhome # **set port** *port-num*	Specifies the SMTP server port that Call Home uses to send email messages. Valid port numbers are 1 to 65535.
Step 15	UCS-A /monitoring/callhome # **set throttling** {**off** | **on**}	Enables or disables Call Home throttling. When enabled, throttling prevents too many Call Home email messages from being sent for the same event. By default, throttling is enabled.
Step 16	UCS-A /monitoring/callhome # **set urgency** {**alerts** | **critical** | **debugging** | **emergencies** | **errors** | **information** | **notifications** | **warnings**}	Specifies the urgency level for Call Home email messages.
Step 17	UCS-A /monitoring/callhome # **commit-buffer**	Commits the transaction to the system configuration.

The following example configures Call Home:

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

```
UCS-A /monitoring/callhome* # set port 25
UCS-A /monitoring/callhome* # set throttling on
UCS-A /monitoring/callhome* # set urgency information
UCS-A /monitoring/callhome* # commit-buffer
UCS-A /monitoring/callhome #
```

Disabling Call Home

Procedure

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

The following example disables Call Home:

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

Enabling Call Home

Procedure

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

The following example enables Call Home:

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

System Inventory Message Configuration

Configuring System Inventory Messages

Procedure

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

The following example configures Call Home system inventory messages:

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

Sending a System Inventory Message

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

Procedure

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

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

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

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

Call Home Profile Configuration

Configuring a Call Home Profile

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

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope monitoring**	Enters monitoring mode.
Step 2	UCS-A /monitoring # **scope callhome**	Enters monitoring call home mode.
Step 3	UCS-A /monitoring/callhome # **create profile** *profile-name*	Enters monitoring call home profile mode.
Step 4	UCS-A /monitoring/callhome/profile # **set level** {**critical** | **debug** | **disaster** | **fatal** | **major** | **minor** | **normal** | **notification** | **warning**}	Specifies the event level for the profile. Each profile can have its own unique event level.
Step 5	UCS-A /monitoring/callhome/profile # **set alertgroups** *group-name* • **ciscotac** • **diagnostic** • **environmental** • **inventory** • **license** • **lifecycle** • **linecard**	Specifies one or more groups that are alerted based on the profile. The *group-name* argument can be one or more of the following keywords entered on the same command line:

	Command or Action	Purpose
	• **supervisor** • **syslogport** • **system** • **test**	
Step 6	UCS-A /monitoring/callhome/profile # **add alertgroups** *group-names*	(Optional) Adds one or more groups to the existing list of groups that are alerted based on the Call Home profile. **Note** You must use the **add alertgroups** command to add more alert groups to the existing alert group list. Using the **set alertgroups** command will replace any pre-existing alert groups with a new group list.
Step 7	UCS-A /monitoring/callhome/profile # **set format** {**shorttxt** | **xml**}	Specifies the formatting method to use for the e-mal messages.
Step 8	UCS-A /monitoring/callhome/profile # **set maxsize** *id-num*	Specifies the maximum size (in characters) of the email message.
Step 9	UCS-A /monitoring/callhome/profile # **create destination** *email-addr*	Specifies the email address to which Call Home alerts should be sent. Use multiple **create destination** commands in monitoring call home profile mode to specify multiple email recipients. Use the **delete destination** command in monitoring call home profile mode to delete a specified email recipient.
Step 10	UCS-A /monitoring/callhome/profile/destination # **commit-buffer**	Commits the transaction to the system configuration.

The following example configures a Call Home profile:

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

Deleting a Call Home Profile

Procedure

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

The following example deletes the Call Home profile named TestProfile:

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

Call Home Policy Configuration

Configuring a Call Home Policy

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

Procedure

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

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

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

Disabling a Call Home Policy

Procedure

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

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

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

Enabling a Call Home Policy

Procedure

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

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

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

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

Deleting a Call Home Policy

Procedure

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

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

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

Call Home for Smart Call Home Configuration

Configuring Smart Call Home

Procedure

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

	Command or Action	Purpose
Step 3	UCS-A /monitoring/callhome # **enable**	Enables Call Home.
Step 4	UCS-A /monitoring/callhome # **set contact** *name*	Specifies the name of the main Call Home contact person.
Step 5	UCS-A /monitoring/callhome # **set email** *email-addr*	Specifies the email address of the main Call Home contact person.
Step 6	UCS-A /monitoring/callhome # **set phone-contact** *phone-num*	Specifies the phone number of the main Call Home contact person. The phone number must be in international format, starting with a + (plus sign) and a country code.
Step 7	UCS-A /monitoring/callhome # **set street-address** *street-addr*	Specifies the street address of the main Call Home contact person.
Step 8	UCS-A /monitoring/callhome # **set customer-id** *id-num*	Specifies the customer identification number from the service agreement. The number can be up to 255 alphanumeric characters in free format.
Step 9	UCS-A /monitoring/callhome # **set contract-id** *id-num*	Specifies the contract identification number from the service agreement. The number can be up to 255 alphanumeric characters in free format.
Step 10	UCS-A /monitoring/callhome # **set site-id** *id-num*	Specifies the site identification number from the service agreement. The number can be up to 255 alphanumeric characters in free format.
Step 11	UCS-A /monitoring/callhome # **set from-email** *email-addr*	Specifies the email address to use for the From field in Call Home messages.
Step 12	UCS-A /monitoring/callhome # **set reply-to-email** *email-addr*	Specifies the email address to use for the Reply To field in Call Home messages.
Step 13	UCS-A /monitoring/callhome # **set hostname** {*hostname* | *ip-addr*}	Specifies the hostname or IP address of the SMTP server that Call Home uses to send email messages.
Step 14	UCS-A /monitoring/callhome # **set port** *port-num*	Specifies the SMTP server port that Call Home uses to send email messages. Valid port numbers are 1 to 65535.
Step 15	UCS-A /monitoring/callhome # **set throttling** {**off** | **on**}	Enables or disables Call Home throttling. When enabled, throttling prevents too many Call Home email messages from being sent for the same event. By default, throttling is enabled.
Step 16	UCS-A /monitoring/callhome # **set urgency** {**alerts** | **critical** | **debugging** | **emergencies** | **errors** | **information** | **notifications** | **warnings**}	Specifies the urgency level for Call Home email messages.

The following example configures Call Home:

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

What to Do Next

Continue to "Configuring the Default Cisco TAC-1 Profile, page 233" to configure a Call Home profile for use with Smart Call Home.

Configuring the Default Cisco TAC-1 Profile

The default settings of the CiscoTAC-1 profile are:

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

Before You Begin

Complete the "Configuring Smart Call Home, page 231" section.

Procedure

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

	Command or Action	Purpose
Step 6	UCS-A /monitoring/callhome/profile # **create destination callhome@cisco.com**	Specifies the email recipient to **callhome@cisco.com**.
Step 7	UCS-A /monitoring/callhome/profile/destination # **exit**	Exits to monitoring call home profile mode.
Step 8	UCS-A /monitoring/callhome/profile # **exit**	Exits to monitoring call home mode.

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

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

What to Do Next

Continue to "Configuring a System Inventory Message for Smart Call Home, page 234" to configure system inventory messages for use with Smart Call Home.

Configuring a System Inventory Message for Smart Call Home

Before You Begin

Complete the "Configuring the Default Cisco TAC-1 Profile, page 233" section.

Procedure

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

	Command or Action	Purpose
Step 6	UCS-A /monitoring/callhome/inventory # **commit-buffer**	Commits the transaction to the system configuration.

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

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

What to Do Next

Continue to "Registering Smart Call Home, page 235" to send an inventory message that starts the Smart Call Home registration process.

Registering Smart Call Home

Before You Begin

Complete the "Configuring a System Inventory Message for Smart Call Home, page 234" section.

Procedure

	Command or Action	Purpose
Step 1	UCS-A /monitoring/callhome/inventory # **send**	Sends the system inventory message to the Smart Call Home database. You will receive an email from Cisco that describes how to complete the registration process.

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

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

What to Do Next

Follow the link in the email message to complete the SmartCall Home registration.

CHAPTER 29

Backing Up and Restoring the Configuration

This chapter includes:

Backup and Export Configuration

When you perform a backup through Cisco UCS Manager, you take a snapshot of all or part of the system configuration and export the file to a location on your network. You cannot use Cisco UCS Manager to back up data on the servers.

You can perform a backup while the system is up and running. The backup operation only saves information from the management plane. It does not have any impact on the server or network traffic.

You cannot schedule a backup operation. You can, however, create a backup operation in advance and leave the admin state disabled until you are ready to run the backup. Cisco UCS Manager does not run the backup operation and save and exports the configuration file until you set the admin state to enabled.

You can only maintain one backup operation for each location where you plan to save a backup file. If you rerun a backup operation without changing the filename, Cisco UCS Manager overwrites the existing file on the server.

Backup Types

You can perform one of the following types of backups through Cisco UCS Manager:

- **Full state**—Includes a snapshot of the entire system. You can use the file generated from this backup for disaster recovery if you need to recreate every configuration on a fabric interconnect or to rebuild a fabric interconnect.
- **All configuration**—Includes all system and logical configuration settings
- **System configuration**—Includes all system configuration settings such as usernames, roles, and locales.
- **Logical configuration**—Includes all logical configuration settings such as service profiles, VLANs, VSANs, pools, and policies

Import Configuration

You can import any configuration file that was exported from Cisco UCS Manager. The file does not have to have been exported from the same Cisco UCS Manager.

The import function is available for all configuration, system configuration, and logical configuration files. You can perform an import while the system is up and running. An import operation modifies information on the management plane only. Some modifications caused by an import operation, such as a change to a vNIC assigned to a server, can cause a server reboot or other operations that disrupt traffic.

You cannot schedule an import operation. You can, however, create an import operation in advance and leave the admin state disabled until you are ready to run the import. Cisco UCS Manager will not run the import operation on the configuration file until you set the admin state to enabled.

You can maintain only one import operation for each location where you saved a configuration backup file.

Import Methods

You can use one of the following methods to import and update a system configuration through Cisco UCS Manager:

- **Merge**—The information in the imported configuration file is compared with the existing configuration information. If there are conflicts, the import operation overwrites the information on the Cisco UCS instance with the information in the import configuration file.
- **Replace**—The current configuration information is replaced with the information in the imported configuration file one object at a time.

System Restore

You can restore a system configuration from any full state backup file that was exported from Cisco UCS Manager. The file does not have to have been exported from the Cisco UCS Manager on the system that you are restoring.

The restore function is only available for a full state backup file. You cannot import a full state backup file. You perform a restore through the initial system setup.

You can use the restore function for disaster recovery.

Required User Role for Backup and Import Operations

You must have a user account that includes the admin role to create and run backup and import operations.

Backup Operations

Creating a Backup Operation

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system**	Enters system mode.
Step 2	UCS-A /system # **create backup** *URL backup-type* {**disabled** | **enabled**}	Creates a backup operation. Specify the *URL* for the backup file using one of the following syntax: • **ftp:**// *hostname* / *path* • **scp:**// *username@hostname* / *path* • **sftp:**// *username@hostname* / *path* • **tftp:**// *hostname* **:** *port-num* / *path* The *backup-type* argument can be one of the following values: • **all-configuration**—Backs up the server, fabric, and system related configuration • **full-state**—Backs up the full state for disaster recovery • **logical-configuration**—Backs up the fabric and service profile related configuration • **system-configuration**—Backs up the system related configuration You can save multiple backup operations, but only one operation per hostname is saved. If you use the **enable** keyword, the backup operation runs as soon as you enter the **commit-buffer** command. If you use the **disable** keyword, the backup operation will not be run until it is enabled. When enabling a backup operation, you must specify the hostname you used when creating the backup operation.
Step 3	UCS-A /system # **commit-buffer**	Commits the transaction.

The following example creates a disabled full-state backup operation for hostname host35 and commits the transaction:

```
UCS-A# scope system
UCS-A /system* # create backup scp://user@host35/backups full-state disabled
Password:
UCS-A /system* # commit-buffer
UCS-A /system #
```

Running a Backup Operation

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system**	Enters system mode.
Step 2	UCS-A /system # **scope backup** *hostname*	Enters system backup mode for the specified hostname.
Step 3	UCS-A /system/backup # **enable**	Enables the backup operation.
Step 4	UCS-A /system/backup # **commit-buffer**	Commits the transaction.

The following example enables a backup operation for the host35 hostname and commits the transaction:

```
UCS-A# scope system
UCS-A /system* # scope backup
UCS-A /system/backup* # enable
UCS-A /system/backup* # commit-buffer
UCS-A /system/backup #
```

Deleting a Backup Operation

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system**	Enters system mode.
Step 2	UCS-A /system # **delete backup** *hostname*	Deletes the backup operation for the specified hostname.
Step 3	UCS-A /system # **commit-buffer**	Commits the transaction.

The following example enables a backup operation for the host35 hostname and commits the transaction:

```
UCS-A# scope system
UCS-A /system* # delete backup
UCS-A /system* # commit-buffer
UCS-A /system #
```

Import Operations

Creating an Import Operation

You cannot import a full state configuration file. You must perform a system restore from a full state configuration file.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system**	Enters system mode.
Step 2	UCS-A /system # **create import-config** *URL* {**disabled** | **enabled**} {**merge** | **replace**}	Creates an import operation. Specify the *URL* for the file being imported using one of the following syntax: • **ftp://** *hostname* / *path* • **scp://** *username@hostname* / *path* • **sftp://** *username@hostname* / *path* • **tftp://** *hostname* **:** *port-num* / *path* You can save multiple import operations, but only one operation per hostname is saved. If you use the **enable** keyword, the import operation runs as soon as you enter the **commit-buffer** command. If you use the **disable** keyword, the import operation will not be run until it is enabled. When enabling an import operation, you must specify the hostname you used when creating the import operation.
Step 3	UCS-A /system/import-config # **commit-buffer**	Commits the transaction.

The following example creates a disabled import operation for hostname host35 that replaces the existing configuration and commits the transaction:

```
UCS-A# scope system
UCS-A /system* # create import-config scp://user@host35/backups disabled replace
Password:
UCS-A /system/import-config* # commit-buffer
UCS-A /system/import-config #
```

Running an Import Operation

You cannot import a full state configuration file. You must perform a system restore from a full state configuration file.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system**	Enters system mode.
Step 2	UCS-A /system # **scope import-config** *hostname*	Enters system backup mode for the specified hostname.
Step 3	UCS-A /system/import-config # **enable**	Enables the import operation.
Step 4	UCS-A /system/import-config # **commit-buffer**	Commits the transaction.

The following example enables an import operation for the host35 hostname and commits the transaction:

```
UCS-A# scope system
UCS-A /system* # scope import-config
UCS-A /system/import-config* # enable
UCS-A /system/import-config* # commit-buffer
UCS-A /system/import-config #
```

Deleting an Import Operation

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system**	Enters system mode.
Step 2	UCS-A /system # **delete import-config** *hostname*	Deletes the import operation for the specified hostname.
Step 3	UCS-A /system # **commit-buffer**	Commits the transaction.

The following example deletes the import operation for the host35 hostname and commits the transaction:

```
UCS-A# scope system
UCS-A /system* # delete import-config host35
UCS-A /system* # commit-buffer
UCS-A /system #
```

CHAPTER 30

Recovering a Lost Password

This chapter includes:

Password Recovery for the Admin Account

The admin account is the system administrator or superuser account. If an administrator loses the password to this account, you can have a serious security issue. As a result, the procedure to recover the password for the admin account requires you to power cycle all fabric interconnects in a Cisco UCS instance.

When you recover the password for the admin account, you actually change the password for that account. You cannot retrieve the original password for that account.

You can reset the password for all other local acounts through Cisco UCS Manager. However, you must log into Cisco UCS Manager with an account that includes aaa or admin privileges.

⚠ **Caution** This procedure requires you to power down all fabric interconnects in a Cisco UCS instance. As a result, all data transmission in the instance is stopped until you restart the fabric interconnects.

Determining the Leadership Role of a Fabric Interconnect

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **show cluster state**	Displays the operational state and leadership role for both fabric interconnects in a cluster.

The following example displays the leadership role for both fabric interconnects in a cluster, where fabric interconnect A has the primary role and fabric interconnect A has the subordinate role:

```
UCS-A# show cluster state
Cluster Id: 0x4432f72a371511de-0xb97c000de1b1ada4

A: UP, PRIMARY
B: UP, SUBORDINATE

HA READY
```

Recovering the Admin Account Password in a Standalone Configuration

Before You Begin

1 Physically connect the console port on the fabric interconnect to a computer terminal or console server

2 Obtain the following information:

- The firmware kernel version on the fabric interconnect
- The firmware system version

Procedure

Step 1 Connect to the console port.

Step 2 Power cycle the fabric interconnect:

a) Turn off the power to the fabric interconnect.
b) Turn on the power to the fabric interconnect.

Step 3 In the console, press one of the following key combinations as it boots to get the `loader` prompt:

- Ctrl+l
- Ctrl+Shift+r

You may need to press the selected key combination multiple times before your screen displays the `loader` prompt.

Step 4 Boot the kernel firmware version on the fabric interconnect.

```
loader > boot
/installables/fabric/kernel_firmware_version
```

Step 5 Enter config terminal mode.

```
Fabric(boot)# config terminal
```

Step 6 Reset the admin password.

```
Fabric(boot)(config)# admin-password
 password
```

The new password displays in clear text mode.

Step 7 Exit config terminal mode and return to the boot prompt.

Step 8 Boot the system firmware version on the fabric interconnect.

```
Fabric(boot)# load /installables/fabric/system_firmware_version
```

Step 9 After the system image loads, log in to Cisco UCS Manager.

Recovering the Admin Account Password in a Cluster Configuration

Before You Begin

1. Physically connect a console port on one of the fabric interconnects to a computer terminal or console server
2. Obtain the following information:
   - The firmware kernel version on the fabric interconnect
   - The firmware system version
   - Which fabric interconnect has the primary leadership role and which is the subordinate

Procedure

Step 1 Connect to the console port.

Step 2 For the subordinate fabric interconnect:

a) Turn off the power to the fabric interconnect.
b) Turn on the power to the fabric interconnect.
c) In the console, press one of the following key combinations as it boots to get the loader prompt:

- Ctrl+l
- Ctrl+Shift+r

You may need to press the selected key combination multiple times before your screen displays the `loader` prompt.

Step 3 Power cycle the primary fabric interconnect:

a) Turn off the power to the fabric interconnect.
b) Turn on the power to the fabric interconnect.

Step 4 In the console, press one of the following key combinations as it boots to get the `loader` prompt:

- Ctrl+l
- Ctrl+Shift+r

You may need to press the selected key combination multiple times before your screen displays the `loader` prompt.

Step 5 Boot the kernel firmware version on the primary fabric interconnect.

```
loader > boot
/installables/fabric/kernel_firmware_version
```

Step 6 Enter config terminal mode.

```
Fabric(boot)# config terminal
```

Step 7 Reset the admin password.

```
Fabric(boot)(config)# admin-password
 password
```

The new password displays in clear text mode.

Step 8 Exit config terminal mode and return to the boot prompt.

Step 9 Boot the system firmware version on the primary fabric interconnect.

```
Fabric(boot)# load /installables/fabric/system_firmware_version
```

Step 10 After the system image loads, log in to Cisco UCS Manager.

Step 11 In the console for the subordinate fabric interconnect, do the following to bring it up:

a) Boot the kernel firmware version on the subordinate fabric interconnect.

```
loader > boot
/installables/fabric/kernel_firmware_version
```

b) Boot the system firmware version on the subordinate fabric interconnect.

```
Fabric(boot)# load /installables/fabric/system_firmware_version
```

CHAPTER 31

Configuring Statistics-Related Policies

This chapter includes:

Statistics Collection Policies

Statistics Collection Policy

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

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

- Adapter—statistics related to the adapters in the fabric Interconnect
- Chassis—statistics related to the blade chassis
- Host—this policy is a placeholder for future support
- Port—statistics related to the ports, including server ports, uplink Ethernet ports, and uplink Fibre Channel ports
- Server—statistics related to servers

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

Configuring a Statistics Collection Policy

Procedure

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

The following example creates a statistics collection policy for ports, sets the collection interval to one minute, the reporting interval to 15 minutes, and commits the transaction:

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

Statistics Threshold Policies

Statistics Threshold Policy

A statistics threshold policy monitors statistics about certain aspects of the system and generates an event if the threshold is crossed. You can set both minimum and maximum thresholds. For example, you can configure the policy to raise an alarm if the CPU temperature exceeds a certain value, or if a server is overutilized or underutilized.

These threshold policies do not control the hardware or device-level thresholds enforced by endpoints, such as the BMC. Those thresholds are burned in to the hardware comonents at manufacture.

Cisco UCS enables you to configure statistics threshold policies for the following components:

- Servers and server components
- Uplink Ethernet ports
- Ethernet server ports, chassis, and Fabric Interconnects
- Fibre Channel port

Note You cannot create or delete a statistics threshold policy for Ethernet server ports, uplink Ethernet ports, or uplink Fibre Channel ports. You can only configure the existing default policy.

Server and Server Component Statistics Threshold Policy Configuration

Configuring a Server and Server Component Statistics Threshold 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 stats-threshold-policy** *policy-name*	Creates the specified statistics threshold policy and enters organization statistics threshold policy mode.
Step 3	UCS-A /org/stats-threshold-policy # **set descr** *description*	(Optional) 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/stats-threshold-policy # **commit-buffer**	Commits the transaction to the system configuration.

The following example creates the server and server component statistics threshold policy named ServStatsPolicy, provides a description for the policy, and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # create stats-threshold-policy ServStatsPolicy
UCS-A /org/stats-threshold-policy* # set descr "Server stats threshold policy."
UCS-A /org/stats-threshold-policy* # commit-buffer
UCS-A /org/stats-threshold-policy #
```

What to Do Next

Configure one or more policy classes for the statistics threshold policy. For more information, see "Configuring a Server and Server Component Statistics Threshold Policy Class, page 250."

Deleting a Server and Server Component Statistics Threshold 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 stats-threshold-policy** *policy-name*	Deletes the specified statistics threshold policy.
Step 3	UCS-A /org # **commit-buffer**	Commits the transaction to the system configuration.

The following example deletes the server and server component statistics threshold policy named ServStatsPolicy and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # delete stats-threshold-policy ServStatsPolicy
UCS-A /org* # commit-buffer
UCS-A /org #
```

Configuring a Server and Server Component Statistics Threshold Policy Class

Before You Begin

Configure or identify the server and server component statistics threshold policy that will contain the policy class. For more information, see "Configuring a Server and Server Component Statistics Threshold Policy, page 249."

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 stats-threshold-policy** *policy-name*	Enters organization statistics threshold policy mode.
Step 3	UCS-A /org/stats-threshold-policy # **create class** *class-name*	Creates the specified statistics threshold policy class and enters organization statistics threshold policy class mode. The *class-name* argument can be any of the class name keywords available for the particular statistics threshold policy being configured. To see a list of the available class name keywords, enter the **create class ?** command in organization statistics threshold policy mode.

	Command or Action	Purpose
		Note You can configure multiple classes for the statistics threshold policy.
Step 4	UCS-A /org/stats-threshold-policy/class # **create property** *property-name*	Creates the specified statistics threshold policy class property and enters organization statistics threshold policy class property mode. The *property-name* argument can be any of the property name keywords available for the particular policy class being configured. To see a list of the available property name keywords, enter the **create property ?** command in organization statistics threshold policy class mode. **Note** You can configure multiple properties for the policy class.
Step 5	UCS-A /org/stats-threshold-policy/class/property # **set normal-value** *value*	Specifies the normal value for the class property. The *value* format can vary depending on the class property being configured. To see the required format, enter the **set normal-value ?** command in organization statistics threshold policy class property mode.
Step 6	UCS-A /org/stats-threshold-policy/class/property # **create threshold-value** {**above-normal** | **below-normal**} {**cleared** | **condition** | **critical** | **info** | **major** | **minor** | **warning**}	Creates the specified threshold value for the class property and enters organization statistics threshold policy class property threshold value mode. **Note** You can configure multiple threshold values for the class property.
Step 7	UCS-A /org/stats-threshold-policy/class/property/threshold-value # **set** {**deescalating** | **escalating**} *value*	Specifies the de-escalating or escalating class property threshold value. The *value* format can vary depending on the class property threshold value being configured. To see the required format, enter the **set deescalating ?** or **set escalating ?** command in organization statistics threshold policy class property threshold value mode. **Note** You can specify both de-escalating and escalating class property threshold values.
Step 8	UCS-A /org/stats-threshold-policy/class/property/threshold-value # **commit-buffer**	Commits the transaction to the system configuration.

The following example creates the server and server component statistics threshold policy class for CPU statistics, creates a CPU temperature property, specifies that the normal CPU temperature is 48.5° C, creates an above normal warning threshold of 50° C, and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # scope stats-threshold-policy ServStatsPolicy
UCS-A /org/stats-threshold-policy* # create class cpu-stats
UCS-A /org/stats-threshold-policy/class* # create property cpu-temp
UCS-A /org/stats-threshold-policy/class/property* # set normal-value 48.5
UCS-A /org/stats-threshold-policy/class/property* # create threshold-value above-normal
warning
UCS-A /org/stats-threshold-policy/class/property/threshold-value* # set escalating 50.0
UCS-A /org/stats-threshold-policy/class/property/threshold-value* # commit-buffer
UCS-A /org/stats-threshold-policy/class/property/threshold-value #
```

Deleting a Server and Server Component Statistics Threshold Policy Class

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 stats-threshold-policy** *policy-name*	Enters the specified statistics threshold policy.
Step 3	UCS-A /org/stats-threshold-policy # **delete class** *class-name*	Deletes the specified statistics threshold policy class from the policy.
Step 4	UCS-A /org/stats-threshold-policy # **commit-buffer**	Commits the transaction to the system configuration.

The following example deletes the server and server component statistics threshold policy class for CPU statistics and commits the transaction:

```
UCS-A# scope org /
UCS-A /org* # scope stats-threshold-policy ServStatsPolicy
UCS-A /org/stats-threshold-policy* # delete class cpu-stats
UCS-A /org/stats-threshold-policy* # commit-buffer
UCS-A /org/stats-threshold-policy #
```

Uplink Ethernet Port Statistics Threshold Policy Configuration

Configuring an Uplink Ethernet Port Statistics Threshold Policy

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope eth-uplink**	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # **scope stats-threshold-policy default**	Enters Ethernet uplink statistics threshold policy mode.

	Command or Action	Purpose
		Note You cannot create (or delete) an uplink Ethernet port statistics threshold policy. You can only enter (scope to) the existing default policy.
Step 3	UCS-A /eth-uplink/stats-threshold-policy # **set descr** *description*	(Optional) 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 /eth-uplink/stats-threshold-policy # **commit-buffer**	Commits the transaction to the system configuration.

The following example enters the default uplink Ethernet port threshold policy, provides a description for the policy, and commits the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink* # scope stats-threshold-policy default
UCS-A /eth-uplink/stats-threshold-policy* # set descr "Uplink Ethernet port stats threshold
 policy."
UCS-A /eth-uplink/stats-threshold-policy* # commit-buffer
UCS-A /eth-uplink/stats-threshold-policy #
```

What to Do Next

Configure one or more policy classes for the statistics threshold policy. For more information, see "Configuring an Uplink Ethernet Port Statistics Threshold Policy Class, page 253."

Configuring an Uplink Ethernet Port Statistics Threshold Policy Class

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope eth-uplink**	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # **scope stats-threshold-policy default**	Enters Ethernet uplink statistics threshold policy mode.
Step 3	UCS-A /eth-uplink/stats-threshold-policy # **create class** *class-name*	Creates the specified statistics threshold policy class and enters Ethernet uplink statistics threshold policy class mode. The *class-name* argument can be any of the class name keywords available for the particular statistics threshold policy being configured. To see a list of the available class name keywords, enter the **create class ?** command in

	Command or Action	Purpose
		Ethernet uplink statistics threshold policy mode. **Note** You can configure multiple classes for the statistics threshold policy.
Step 4	UCS-A /eth-uplink/stats-threshold-policy/class # **create property** *property-name*	Creates the specified statistics threshold policy class property and enters Ethernet uplink statistics threshold policy class property mode. The *property-name* argument can be any of the property name keywords available for the particular policy class being configured. To see a list of the available property name keywords, enter the **create property ?** command in Ethernet uplink statistics threshold policy class mode. **Note** You can configure multiple properties for the policy class.
Step 5	UCS-A /eth-uplink/stats-threshold-policy/class/property # **set normal-value** *value*	Specifies the normal value for the class property. The *value* format can vary depending on the class property being configured. To see the required format, enter the **set normal-value ?** command in Ethernet uplink statistics threshold policy class property mode.
Step 6	UCS-A /eth-uplink/stats-threshold-policy/class/property # **create threshold-value** {**above-normal** | **below-normal**} {**cleared** | **condition** | **critical** | **info** | **major** | **minor** | **warning**}	Creates the specified threshold value for the class property and enters Ethernet uplink statistics threshold policy class property threshold value mode. **Note** You can configure multiple threshold values for the class property.
Step 7	UCS-A /eth-uplink/stats-threshold-policy/class/property/threshold-value # **set** {**deescalating** | **escalating**} *value*	Specifies the de-escalating or escalating class property threshold value. The *value* format can vary depending on the class property threshold value being configured. To see the required format, enter the **set deescalating ?** or **set escalating ?** command in Ethernet uplink statistics

	Command or Action	Purpose
		threshold policy class property threshold value mode. **Note** You can specify both de-escalating and escalating class property threshold values.
Step 8	UCS-A /eth-uplink/stats-threshold-policy/class/property/threshold-value # **commit-buffer**	Commits the transaction to the system configuration.

The following example creates the uplink Ethernet port statistics threshold policy class for Ethernet error statistics, creates a cyclic redundancy check (CRC) error count property, specifies that the normal CRC error count per polling interval is 1,000, creates an above normal warning threshold of 1,250, and commits the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink* # scope stats-threshold-policy default
UCS-A /eth-uplink/stats-threshold-policy* # create class ether-error-stats
UCS-A /eth-uplink/stats-threshold-policy/class* # create property crc-delta
UCS-A /eth-uplink/stats-threshold-policy/class/property* # set normal-value 1000
UCS-A /eth-uplink/stats-threshold-policy/class/property* # create threshold-value above-normal
 warning
UCS-A /eth-uplink/stats-threshold-policy/class/property/threshold-value* # set escalating
1250
UCS-A /eth-uplink/stats-threshold-policy/class/property/threshold-value* # commit-buffer
UCS-A /eth-uplink/stats-threshold-policy/class/property/threshold-value #
```

Deleting an Uplink Ethernet Port Statistics Threshold Policy Class

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope eth-uplink**	Enters Ethernet uplink mode.
Step 2	UCS-A /eth-uplink # **scope stats-threshold-policy default**	Enters Ethernet uplink statistics threshold policy mode.
Step 3	UCS-A /eth-uplink/stats-threshold-policy # **delete class** *class-name*	Deletes the specified statistics threshold policy class from the policy.
Step 4	UCS-A /eth-uplink/stats-threshold-policy # **commit-buffer**	Commits the transaction to the system configuration.

The following example deletes the uplink Ethernet port statistics threshold policy class for Ethernet error statistics and commits the transaction:

```
UCS-A# scope eth-uplink
UCS-A /eth-uplink* # scope stats-threshold-policy default
UCS-A /eth-uplink/stats-threshold-policy* # delete class ether-error-stats
UCS-A /eth-uplink/stats-threshold-policy* # commit-buffer
UCS-A /eth-uplink/stats-threshold-policy #
```

Server Port, Chassis, and Fabric Interconnect Statistics Threshold Policy Configuration

Configuring a Server Port, Chassis, and Fabric Interconnect Statistics Threshold Policy

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope eth-server**	Enters Ethernet server mode.
Step 2	UCS-A /eth-server # **scope stats-threshold-policy default**	Enters Ethernet server statistics threshold policy mode. **Note** You cannot create (or delete) a server port, chassis, and fabric interconnect statistics threshold policy. You can only enter (scope to) the existing default policy.
Step 3	UCS-A /eth-server/stats-threshold-policy # **set descr** *description*	(Optional) 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 /eth-server/stats-threshold-policy # **commit-buffer**	Commits the transaction to the system configuration.

The following example enters the default server port, chassis, and fabric interconnect statistics threshold policy, provides a description for the policy, and commits the transaction:

```
UCS-A# scope eth-server
UCS-A /eth-server* # scope stats-threshold-policy default
UCS-A /eth-server/stats-threshold-policy* # set descr "Server port, chassis, and fabric
interconnect stats threshold policy."
UCS-A /eth-server/stats-threshold-policy* # commit-buffer
UCS-A /eth-server/stats-threshold-policy #
```

What to Do Next

Configure one or more policy classes for the statistics threshold policy. For more information, see "Configuring a Server Port, Chassis, and Fabric Interconnect Statistics Threshold Policy Class, page 256."

Configuring a Server Port, Chassis, and Fabric Interconnect Statistics Threshold Policy Class

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope eth-server**	Enters Ethernet server mode.
Step 2	UCS-A /eth-server # **scope stats-threshold-policy default**	Enters Ethernet server statistics threshold policy mode.

	Command or Action	Purpose
Step 3	UCS-A /eth-server/stats-threshold-policy # **create class** *class-name*	Creates the specified statistics threshold policy class and enters Ethernet server statistics threshold policy class mode. The *class-name* argument can be any of the class name keywords available for the particular statistics threshold policy being configured. To see a list of the available class name keywords, enter the **create class ?** command in Ethernet server statistics threshold policy mode. **Note** You can configure multiple classes for the statistics threshold policy.
Step 4	UCS-A /eth-server/stats-threshold-policy/class # **create property** *property-name*	Creates the specified statistics threshold policy class property and enters Ethernet server statistics threshold policy class property mode. The *property-name* argument can be any of the property name keywords available for the particular policy class being configured. To see a list of the available property name keywords, enter the **create property ?** command in Ethernet server statistics threshold policy class mode. **Note** You can configure multiple properties for the policy class.
Step 5	UCS-A /eth-server/stats-threshold-policy/class/property # **set normal-value** *value*	Specifies the normal value for the class property. The *value* format can vary depending on the class property being configured. To see the required format, enter the **set normal-value ?** command in Ethernet server statistics threshold policy class property mode.
Step 6	UCS-A /eth-server/stats-threshold-policy/class/property # **create threshold-value** {**above-normal** | **below-normal**} {**cleared** | **condition** | **critical** | **info** | **major** | **minor** | **warning**}	Creates the specified threshold value for the class property and enters Ethernet server statistics threshold policy class property threshold value mode. **Note** You can configure multiple threshold values for the class property.

	Command or Action	Purpose
Step 7	UCS-A /eth-server/stats-threshold-policy/class/property/threshold-value # **set** {**deescalating** | **escalating**} *value*	Specifies the de-escalating or escalating class property threshold value. The *value* format can vary depending on the class property threshold value being configured. To see the required format, enter the **set deescalating ?** or **set escalating ?** command in Ethernet server statistics threshold policy class property threshold value mode. **Note** You can specify both de-escalating and escalating class property threshold values.
Step 8	UCS-A /eth-server/stats-threshold-policy/class/property/threshold-value # **commit-buffer**	Commits the transaction to the system configuration.

The following example creates the server port, chassis, and fabric interconnect statistics threshold policy class for chassis statistics, creates an input power (Watts) property, specifies that the normal power is 8kW, creates an above normal warning threshold of 11kW, and commits the transaction:

```
UCS-A# scope eth-server
UCS-A /eth-server* # scope stats-threshold-policy default
UCS-A /eth-server/stats-threshold-policy* # create class chassis-stats
UCS-A /eth-server/stats-threshold-policy/class* # create property input-power
UCS-A /eth-server/stats-threshold-policy/class/property* # set normal-value 8000.0
UCS-A /eth-server/stats-threshold-policy/class/property* # create threshold-value above-normal
 warning
UCS-A /eth-server/stats-threshold-policy/class/property/threshold-value* # set escalating
11000.0
UCS-A /eth-server/stats-threshold-policy/class/property/threshold-value* # commit-buffer
UCS-A /eth-server/stats-threshold-policy/class/property/threshold-value #
```

Deleting a Server Port, Chassis, and Fabric Interconnect Statistics Threshold Policy Class

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope eth-server**	Enters Ethernet server mode.
Step 2	UCS-A /eth-server # **scope stats-threshold-policy default**	Enters Ethernet server statistics threshold policy mode.
Step 3	UCS-A /eth-server/stats-threshold-policy # **delete class** *class-name*	Deletes the specified statistics threshold policy class from the policy.
Step 4	UCS-A /eth-server/stats-threshold-policy # **commit-buffer**	Commits the transaction to the system configuration.

The following example deletes the server port, chassis, and fabric interconnect statistics threshold policy class for chassis statistics and commits the transaction:

```
UCS-A# scope eth-server
UCS-A /eth-server* # scope stats-threshold-policy default
UCS-A /eth-server/stats-threshold-policy* # delete class chassis-stats
UCS-A /eth-server/stats-threshold-policy* # commit-buffer
UCS-A /eth-server/stats-threshold-policy #
```

Fibre Channel Port Statistics Threshold Policy Configuration

Configuring an Uplink Fibre Channel Port Statistics Threshold Policy

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope fc-uplink**	Enters Fibre Channel uplink mode.
Step 2	UCS-A /fc-uplink # **scope stats-threshold-policy default**	Enters Fibre Channel uplink statistics threshold policy mode. **Note** You cannot create (or delete) an uplink Fibre Channel port statistics threshold policy. You can only enter (scope to) the existing default policy.
Step 3	UCS-A /fc-uplink/stats-threshold-policy # **set descr** *description*	(Optional) 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 /fc-uplink/stats-threshold-policy # **commit-buffer**	Commits the transaction to the system configuration.

The following example enters the default uplink Fibre Channel port statistics threshold policy, provides a description for the policy, and commits the transaction:

```
UCS-A# scope fc-uplink
UCS-A /fc-uplink* # scope stats-threshold-policy default
UCS-A /fc-uplink/stats-threshold-policy* # set descr "Uplink Fibre Channel stats threshold
 policy."
UCS-A /fc-uplink/stats-threshold-policy* # commit-buffer
UCS-A /fc-uplink/stats-threshold-policy #
```

What to Do Next

Configure one or more policy classes for the statistics threshold policy. For more information, see "Configuring an Uplink Fibre Channel Port Statistics Threshold Policy Class, page 260."

Configuring an Uplink Fibre Channel Port Statistics Threshold Policy Class

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope fc-uplink**	Enters Fibre Channel uplink mode.
Step 2	UCS-A /fc-uplink # **scope stats-threshold-policy default**	Enters Fibre Channel uplink statistics threshold policy mode.
Step 3	UCS-A /fc-uplink/stats-threshold-policy # **create class** *class-name*	Creates the specified statistics threshold policy class and enters Fibre Channel uplink statistics threshold policy class mode. The *class-name* argument can be any of the class name keywords available for the particular statistics threshold policy being configured. To see a list of the available class name keywords, enter the **create class ?** command in Fibre Channel uplink statistics threshold policy mode. **Note** You can configure multiple classes for the statistics threshold policy.
Step 4	UCS-A /fc-uplink/stats-threshold-policy/class # **create property** *property-name*	Creates the specified statistics threshold policy class property and enters Fibre Channel uplink statistics threshold policy class property mode. The *property-name* argument can be any of the property name keywords available for the particular policy class being configured. To see a list of the available property name keywords, enter the **create property ?** command in Fibre Channel uplink statistics threshold policy class mode. **Note** You can configure multiple properties for the policy class.
Step 5	UCS-A /fc-uplink/stats-threshold-policy/class/property # **set normal-value** *value*	Specifies the normal value for the class property. The *value* format can vary depending on the class property being configured. To see the required format, enter the **set normal-value ?** command in Fibre Channel uplink statistics threshold policy class property mode.

	Command or Action	Purpose
Step 6	UCS-A /fc-uplink/stats-threshold-policy/class/property # **create threshold-value** {**above-normal** | **below-normal**} {**cleared** | **condition** | **critical** | **info** | **major** | **minor** | **warning**}	Creates the specified threshold value for the class property and enters Fibre Channel uplink statistics threshold policy class property threshold value mode. **Note** You can configure multiple threshold values for the class property.
Step 7	UCS-A /fc-uplink/stats-threshold-policy/class/property/threshold-value # **set** {**deescalating** | **escalating**} *value*	Specifies the de-escalating or escalating class property threshold value. The *value* format can vary depending on the class property threshold value being configured. To see the required format, enter the **set deescalating ?** or **set escalating ?** command in Fibre Channel uplink statistics threshold policy class property threshold value mode. **Note** You can specify both de-escalating and escalating class property threshold values.
Step 8	UCS-A /fc-uplink/stats-threshold-policy/class/property/threshold-value # **commit-buffer**	Commits the transaction to the system configuration.

The following example creates the uplink Fibre Channel port statistics threshold policy class for Fibre Channel statistics, creates an average bytes received property, specifies that the normal average number of bytes received per polling interval is 150MB, creates an above normal warning threshold of 200MB, and commits the transaction:

```
UCS-A# scope fc-uplink
UCS-A /fc-uplink* # scope stats-threshold-policy default
UCS-A /fc-uplink/stats-threshold-policy* # create class fc-stats
UCS-A /fc-uplink/stats-threshold-policy/class* # create property bytes-rx-avg
UCS-A /fc-uplink/stats-threshold-policy/class/property* # set normal-value 150000000
UCS-A /fc-uplink/stats-threshold-policy/class/property* # create threshold-value above-normal
 warning
UCS-A /fc-uplink/stats-threshold-policy/class/property/threshold-value* # set escalating
200000000
UCS-A /fc-uplink/stats-threshold-policy/class/property/threshold-value* # commit-buffer
UCS-A /fc-uplink/stats-threshold-policy/class/property/threshold-value #
```

Deleting an Uplink Fibre Channel Port Statistics Threshold Policy Class

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope fc-uplink**	Enters Fibre Channel uplink mode.

	Command or Action	Purpose
Step 2	UCS-A /fc-uplink # **scope stats-threshold-policy default**	Enters Fibre Channel uplink statistics threshold policy mode.
Step 3	UCS-A /fc-uplink/stats-threshold-policy # **delete class** *class-name*	Deletes the specified statistics threshold policy class from the policy.
Step 4	UCS-A /fc-uplink/stats-threshold-policy # **commit-buffer**	Commits the transaction to the system configuration.

The following example deletes the uplink Fibre Channel port statistics threshold policy class for Fibre Channel statistics and commits the transaction:

```
UCS-A# scope fc-uplink
UCS-A /fc-uplink* # scope stats-threshold-policy default
UCS-A /fc-uplink/stats-threshold-policy* # delete class fc-stats
UCS-A /fc-uplink/stats-threshold-policy* # commit-buffer
UCS-A /fc-uplink/stats-threshold-policy #
```

INDEX