Cisco UCS Manager Administration Management Using the CLI, Release 6.0

First Published: 2025-09-02

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

CONTENTS

Preface

Audience

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

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

Conventions

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

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

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

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

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

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

Warning IMPORTANT SAFETY INSTRUCTIONS

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

SAVE THESE INSTRUCTIONS

Related Cisco UCS Documentation

Documentation Roadmaps

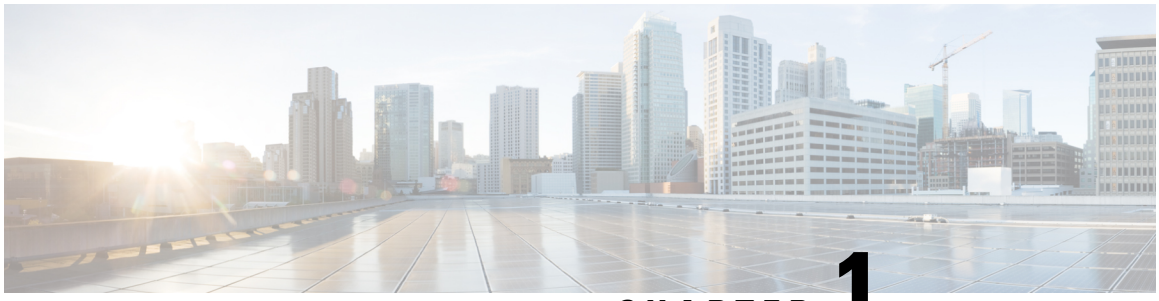
For a complete list of all B-Series documentation, see the *Cisco UCS B-Series Servers Documentation Roadmap* available at the following URL: https://www.cisco.com/c/en/us/td/docs/unified_computing/ucs/overview/guide/UCS_roadmap.html

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

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

Documentation Feedback

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



CHAPTER 1

New and Changed Information for This Release

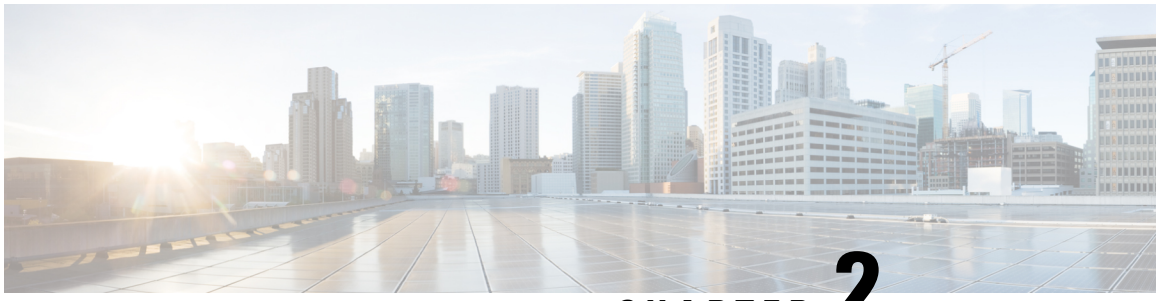
New and Changed Information

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

Table 1: New Features and Changed Behavior in Cisco UCS Manager, Release 6.0(1b)

Feature	Description	Where Documented
Enhanced Login Profile settings	Cisco UCS Manager support for enhanced Login Profile settings.	Configuring Login Profile, on page 61
Support for Cisco UCS 6600 Series Fabric Interconnect	Cisco UCS Manager supports UCS 6664 Fabric Interconnect	• Recovering the Admin Account Password in a Non-Cluster Configuration for Cisco UCS 6664 Fabric Interconnect, on page 18 • Recovering the Admin Account Password in a Standalone Configuration for Cisco UCS 6664 Fabric Interconnect, on page 24 • Recovering the Admin Account Password in a Cluster Configuration for Cisco UCS 6664 Fabric Interconnect, on page 30
Support for secure data deletion on the Fabric Interconnects	Cisco UCS Manager now supports secure data deletion on Fabric Interconnect Series 6400, 6500, 6600, and X-Direct Fabric Interconnects.	Fabric Interconnect Secure Erase (FI Secure Erase), on page 186

Feature	Description	Where Documented
Deprecated support for Cisco UCS 6300 series Fabric Interconnect.	Cisco UCS Manager support for Cisco UCS 6300 Series Fabric Interconnect is deprecated.	-



CHAPTER 2

Administration Management Overview

This chapter includes the following sections:

Administration Management Overview

Cisco UCS Manager provides a comprehensive set of administration features to effectively manage user access and system configurations in your environment.

You can configure the following basic administration configurations to manage user access in your environment:

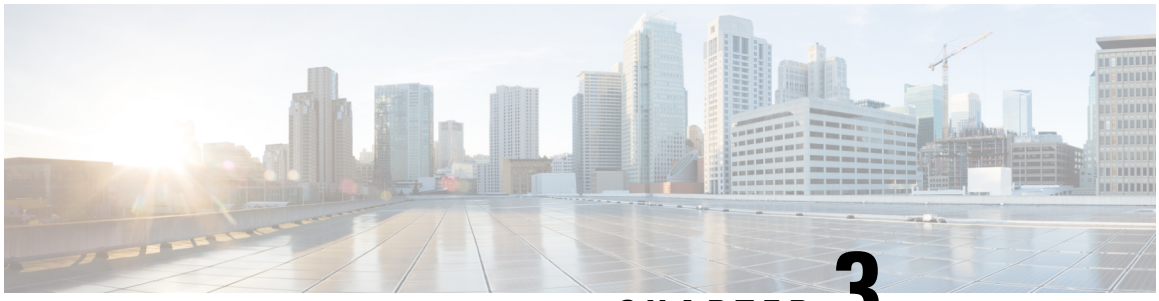
- **Passwords**—Choose a password during the initial setup for the default admin user account, and create a unique username and password for each user account to access the system.
- **RBAC**—Delegate and control user access privileges according to the role and restrict user access within an organization boundary defined for the tenant, such as multi-tenancy.
- **Authentication**—Create UCS Manager local user accounts, and remote user accounts using the LDAP, RADIUS, and TACACS+ protocols.
- **Communication Services**—Configure CIM XML, HTTP, HTTPS, SMASH CLP, SNMP, SSH, and Telnet to interface third-party applications with Cisco UCS.
- **Organizations**—Create organizations for policies, pools, and service profiles. You can create multiple sub-organizations under the default Root organization, and nest sub-organization under a different sub-organization.
- **CIMC**—Close the KVM, vMedia, and SOL sessions of any user. When UCS Manager receives an event from CIMC, it updates its session table and displays the information to all users.
- **Backup and Restore** —Take a snapshot of all or part of the system configuration and export the file to a location on your network. You can configure a full state, all configuration, system configuration, and logical configuration backup.
- **Call Home**—Configure e-mail alert notifications for UCS errors and faults. You can configure the e-mail notifications for Cisco TAC (predefined) or any other recipient.
- **Deferred Deployments**—Configure deployments for a service profile to deploy immediately or during a specified maintenance window. Use this to control when disruptive configuration changes to a service profile or a service profile template are implemented.

- **Scheduling**—Schedule a one time occurrence for a schedule, a recurring occurrence for a schedule, and delete schedules.
- **Fault Suppression**—Enable fault suppression to suppress SNMP trap and Call Home notifications during a planned maintenance time.

Cisco UCS Manager User CLI Documentation

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

Guide	Description
Cisco UCS Manager Getting Started Guide	Discusses Cisco UCS architecture and Day 0 operations, including Cisco UCS Manager initial configuration, and configuration best practices.
Cisco UCS Manager Administration Guide	Discusses password management, role-based access configuration, remote authentication, communication services, CIMC session management, organizations, backup and restore, scheduling options, BIOS tokens and deferred deployments.
Cisco UCS Manager Infrastructure Management Guide	Discusses physical and virtual infrastructure components used and managed by Cisco UCS Manager.
Cisco UCS Manager Firmware Management Guide	Discusses downloading and managing firmware, upgrading through Auto Install, upgrading through service profiles, directly upgrading at endpoints using firmware auto sync, managing the capability catalog, deployment scenarios, and troubleshooting.
Cisco UCS Manager Server Management Guide	Discusses the new licenses, registering Cisco UCS domains with Cisco UCS Central, power capping, server boot, server profiles and server-related policies.
Cisco UCS Manager Storage Management Guide	Discusses all aspects of storage management such as SAN and VSAN in Cisco UCS Manager.
Cisco UCS Manager Network Management Guide	Discusses all aspects of network management such as LAN and VLAN connectivity in Cisco UCS Manager.
Cisco UCS Manager System Monitoring Guide	Discusses all aspects of system and health monitoring including system statistics in Cisco UCS Manager.
Cisco UCS S3260 Server Integration with Cisco UCS Manager	Discusses all aspects of management of UCS S-Series servers that are managed through Cisco UCS Manager.



CHAPTER 3

Password Management

Guidelines for Cisco UCS Passwords

Each locally authenticated user account requires a password. A user with admin or aaa privileges can configure Cisco UCS Manager to perform a password strength check on user passwords. Listed in Table 2: ASCII Table of Allowed Characters for UCS Passwords, on page 5 are the allowed ASCII characters for UCS passwords.

Table 2: ASCII Table of Allowed Characters for UCS Passwords

ASCII Printable Characters	Description
A-Z	uppercase letters A to Z
a-z	lowercase letters a to z
0-9	digits 0 to 9
!	exclamation mark
"	quotation mark
#	hash or pound sign
%	percent sign
&	ampersand

ASCII Printable Characters	Description
'	apostrophe
(	left parenthesis
)	right parenthesis
*	asterisk
+	plus sign
,	comma
-	hyphen
.	period
/	slash
:	colon
;	semicolon
<	less-than
>	greater-than
@	at sign
[	left square bracket
\	backslash
]	right square bracket
^	caret
_	underscore
`	grave accent
{	left curly brace
|	vertical bar
}	right curly brace
~	tilde

Cisco recommends using a strong password; otherwise, the password strength check for locally authenticated users, Cisco UCS Manager rejects any password that does not meet the following requirements:

- If the **Password Strength Check** option is checked, passwords must be between 8 to 127 characters.

- If the **Password Strength Check** option is unchecked, administrators can create user accounts without a password as a placeholder, but a password containing 1 to 127 characters is required for successful authentication.
- Must contain at least three of the following:
  - Lower case letters
  - Upper case letters
  - Digits
  - Special characters
- Must not contain a character that is repeated more than three times consecutively, such as aaabbb.
- Must not be identical to the username or the reverse of the username.
- Must pass a password dictionary check. For example, the password must not be based on a standard dictionary word.
- Must not contain the following symbols: $ (dollar sign), ? (question mark), and = (equals sign).
- Should not be blank for local user and admin accounts.

Guidelines for Cisco UCS Usernames

The username is also used as the login ID for Cisco UCS Manager. When you assign login IDs to Cisco UCS user accounts, consider the following guidelines and restrictions:

- The login ID can contain between 1 and 32 characters, including the following:
  - Any alphabetic character
  - Any digit
  - _ (underscore)
  - - (dash)
  - . (dot)
- The login ID must be unique within Cisco UCS Manager.
- The login ID must start with an alphabetic character. It cannot start with a number or a special character, such as an underscore.
- The login ID is case-sensitive.
- You cannot create an all-numeric login ID.
- After you create a user account, you cannot change the login ID. You must delete the user account and create a new one.

Configuring the Maximum Number of Password Changes for a Change Interval

You must have admin or aaa privileges to change the password profile properties. Except for password history, these properties do not apply to users with admin or aaa privileges.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **scope password-profile**	Enters password profile security mode.
Step 3	UCS-A /security/password-profile # **set change-during-interval enable**	Restricts the number of password changes a locally authenticated user can make within a given number of hours.
Step 4	UCS-A /security/password-profile # **set change-count** *pass-change-num*	Specifies the maximum number of times a locally authenticated user can change his or her password during the Change Interval. This value can be anywhere from 0 to 10.
Step 5	UCS-A /security/password-profile # **set change-interval** *num-of-hours*	Specifies the maximum number of hours over which the number of password changes specified in the **Change Count** field are enforced. This value can be anywhere from 1 to 745 hours. For example, if this field is set to 48 and the **Change Count** field is set to 2, a locally authenticated user can make no more than 2 password changes within a 48 hour period.
Step 6	UCS-A /security/password-profile # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example enables the change during interval option, sets the change count to 5, sets the change interval to 72 hours, and commits the transaction:

```
UCS-A # scope security
UCS-A /security # scope password-profile
UCS-A /security/password-profile # set change-during-interval enable
UCS-A /security/password-profile* # set change-count 5
UCS-A /security/password-profile* # set change-interval 72
```

```
UCS-A /security/password-profile* # commit-buffer
UCS-A /security/password-profile #
```

Configuring a No Change Interval for Passwords

You must have admin or aaa privileges to change the password profile properties. Except for password history, these properties do not apply to users with admin or aaa privileges.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **scope password-profile**	Enters password profile security mode.
Step 3	UCS-A /security/password-profile # **set change-during-interval disable**	Disables the change during interval feature.
Step 4	UCS-A /security/password-profile # **set no-change-interval** *min-num-hours*	Specifies the minimum number of hours that a locally authenticated user must wait before changing a newly created password. This value can be anywhere from 1 to 745 hours. This interval is ignored if the **Change During Interval** property is set to **Disable**.
Step 5	UCS-A /security/password-profile # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example disables the change during interval option, sets the no change interval to 72 hours, and commits the transaction:

```
UCS-A # scope security
UCS-A /security # scope password-profile
UCS-A /security/password-profile # set change-during-interval disable
UCS-A /security/password-profile* # set no-change-interval 72
UCS-A /security/password-profile* # commit-buffer
UCS-A /security/password-profile #
```

Configuring the Password Expiration

The password expiration feature enables the admin or aaa privileged user to enforce the password reset for all the locally authenticated users at a defined time interval. The password reset interval is calculated based on the last password changed date and the password expiry duration.

The following tables provide different scenarios of password expiration for a new and an existing locally authenticated user.

The following table explains how the password expiry date is calculated when the password expiry option is enabled for the first time. For instance, the password expiry is enabled on 1st Dec, the password expiry duration is set as 9 days, and the password warning notification is 4 days.

User Scenarios	Effective Last Password change date	Notification	Effective Password Expiration date
New user is created on the same day that password expiry is enabled	1st Dec	5th Dec	10th Dec
Existing user's first login happens on the same day that password expiry is enabled	1st Dec	5th Dec	10th Dec
New user is created four days after password expiry is enabled	5th Dec	10th Dec	15th Dec
Existing user's first login happens four days after password expiry is enabled	5th Dec	10th Dec	15th Dec
New user changes the password on 2nd Dec	2nd Dec	7th Dec	12th Dec
Existing user changes the password on 2nd Dec	2nd Dec	7th Dec	12th Dec

The following table explains how the password expiry date is calculated when the password expiry option is disabled. For instance, the password expiry option is disabled on 1st Dec.

User Scenarios	Effective Last Password change date	Notification	Effective Password Expiration date
New user is created on the same day that password expiry is disabled	1st Dec	NA	NA
Existing user logs in on the same day that password expiry is disabled	1st Dec	NA	NA
New user is created four days after password expiry is disabled	5th Dec	NA	NA

User Scenarios	Effective Last Password change date	Notification	Effective Password Expiration date
Existing user's first login happens four days after password expiry is disabled	5th Dec	NA	NA
New user changes the password on 2nd Dec	2nd Dec	NA	NA
Existing user changes the password on 2nd Dec	2nd Dec	NA	NA

The following table explains how the password expiry date is calculated when the password expiry option is re-enabled. For instance, the password expiry option is re-enabled on 10th Dec, the password expiry duration is set as 9 days, and the password warning notification is 4 days.

User Scenarios	Effective Last Password change date	Notification	Effective Password Expiration date	Password Status
New user is created on the same day that password expiry is re-enabled	10th Dec	15th Dec	20th Dec	Active
Existing user logs in on the same day that password expiry is re-enabled	1st Dec	5th Dec	10th Dec	Expired
New user is created four days after password expiry is re-enabled	15th Dec	20th Dec	25th Dec	Active
Existing user logs in for the first time four days after password expiry is re-enabled	15th Dec	20th Dec	25th Dec	Active
Existing user logs in for the second time four days after password expiry is re-enabled	5th Dec	10th Dec	15th Dec	Active

The following table explains how the password expiry date is calculated when the system date is modified by the admin. For instance, the actual system date when the last password was changed is 10th Dec, the password expiry duration is set as 9 days, and the password warning notification is 4 days.

System modified date	Effective Password Expiration date	Password Status
System date modified backward		
8th Dec 2020	17th Dec 2020	Active
2nd Dec 2020	11th Dec 2020	Warning
Any date prior to 1st Dec 2020 (the date that makes the password expiry duration from actual date as zero)		Expired
System date modified forward		
14th Dec 2020	19th Dec 2020	Active
19th Dec 2020	19th Dec 2020	Warning
Any date after 19th Dec 2020		Expired

After the password expiry, the user must reset the password using the **Reset Password** link in the Cisco UCS Manager Login page.

Procedure

	Command or Action	Purpose
Step 1	UCS-A /security # **scope security**	Enters security mode.
Step 2	UCS-A /security # **scope password-profile**	Enters password profile security mode.
Step 3	UCS-A /security/password-profile # **set enable-passwd-expiry** *yes*	Enables the password expiration feature for the locally authenticated user.
Step 4	UCS-A /security/password-profile # **set passwd-expiry-duration** *60*	Specifies the number of days after which the password expires for the locally authenticated user. This value can be anywhere from 1 to 180. By default, the password is set to expire in 90 days. For example, if this field is set to 60 days. The locally authenticated user's password will expire after 60 days from the last password changed date.
Step 5	UCS-A /security/password-profile # **set passwd-expiry-warn-interval** *8*	Specifies the number of days by when the locally authenticated user must start to receive password expiry notification. This value can be anywhere from 1 to 30 days. By default, the warning is set to 15 days. For example, if this field is set to eight, the locally authenticated user will receive a warning notification eight days before the password expiry.

	Command or Action	Purpose
		Note The **Password Expiration Period** field value must be always greater than the value in the **Password Expiration Warning Time** field.
Step 6	UCS-A /security/password-profile # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example enables the password expiry option, sets the password expiry duration to 60 days, sets the warning interval to eight days, and commits the transaction:

```
UCS-A # scope security
UCS-A /security # scope password-profile
UCS-A /security/password-profile # set enable-passwd-expiry yes
UCS-A /security/password-profile* # set passwd-expiry-duration 60
UCS-A /security/password-profile* # set passwd-expiry-warn-interval 8
UCS-A /security/password-profile* # commit-buffer
UCS-A /security/password-profile #
```

Configuring the Password History Count

You must have admin or aaa privileges to change the password profile properties.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **scope password-profile**	Enters password profile security mode.
Step 3	UCS-A /security/password-profile # **set history-count** *num-of-passwords*	Specifies the number of unique passwords that a locally authenticated user must create before that user can reuse a previously used password This value can be anywhere from 0 to 15. By default, the **History Count** field is set to 0, which disables the history count and allows users to reuse previously used passwords at any time.
Step 4	UCS-A /security/password-profile # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example configures the password history count and commits the transaction:

```
UCS-A # scope security
UCS-A /security # scope password-profile
UCS-A /security/password-profile # set history-count 5
UCS-A /security/password-profile* # commit-buffer
UCS-A /security/password-profile #
```

Password Profile for Locally Authenticated Users

The password profile contains the password history and the password change interval properties for all locally authenticated users of Cisco UCS Manager. You cannot specify a different password profile for locally authenticated users.

Note You must have admin or aaa privileges to change the password profile properties. Except for password history, these properties do not apply to users with admin or aaa privileges.

Password History Count

The password history count prevents locally authenticated users from reusing the same password. When you configure the password history count, Cisco UCS Manager stores up to a maximum of 15 previously used passwords. The password history count stores the passwords in reverse chronological order with the most recent password first. This ensures that the user can only reuse the oldest password when the history count reaches its threshold.

A user can create and use the number of passwords configured in the password history count before reusing a password. For example, if you set the password history count to 8, a user cannot reuse the first password until the ninth password expires.

By default, the password history is set to 0. This value disables the history count and allows users to reuse previously used passwords at any time.

You can clear the password history count for a locally authenticated user and enable reuse of previous passwords.

Password Change Interval

The password change interval restricts the number of password changes that a locally authenticated user can make within a specific number of hours. The following table describes the two interval configuration options for the password change interval.

Interval Configuration	Description	Example
No password change allowed	Does not allow changing passwords for locally authenticated user within a specified number of hours after a password change. You can specify a no change interval between 1 and 745 hours. By default, the no change interval is 24 hours.	To prevent the user from changing passwords within 48 hours after a password change: • Set **Change during interval** to disable • Set **No change interval** to 48
Password changes allowed within change interval	Specifies the maximum number of times that a locally authenticated user password change can occur within a pre-defined interval. You can specify a change interval between 1 and 745 hours and a maximum number of password changes between 0 and 10. By default, a locally authenticated user is permitted a maximum of two password changes within a 48-hour interval.	To allow a password change for a maximum of one time within 24 hours after a password change: • Set **Change during interval** to enable • Set **Change count** to 1 • Set **Change interval** to 24

Clearing the Password History for a Locally Authenticated User

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **scope local-user** *user-name*	Enters local user security mode for the specified user account.
Step 3	UCS-A /security/local-user # **set clear password-history yes**	Clears the password history for the specified user account.
Step 4	UCS-A /security/local-user # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example configures the password history count and commits the transaction:

```
UCS-A # scope security
UCS-A /security # scope local-user admin
UCS-A /security/local-user # set clear password-history yes
UCS-A /security/local-user* # commit-buffer
UCS-A /security/local-user #
```

Password Encryption Key for Backup Configuration Files

Password Encryption Key for Backup Configuration Files

Beginning with release 4.2(3d), Cisco UCS Manager introduces **Password Encryption Key** to enhance security for backup configuration files.

You must set **Password Encryption Key** in order to create backup configuration files and also to import the backup files. Cisco UCS Manager release 4.2(3d) and later do not allow you to create backup configuration files or import backup configuration files without setting the **Password Encryption Key**. If the **Password Encryption Key** is not set, following error is displayed while creating a backup configuration file:

```
Backup/Export operation requires Password Encryption Key to be set, please
refer to Cisco UCS Manager Administration Guide to set the Password
Encryption key.
```

You cannot import a backup configuration file created from Cisco UCS Manager release 4.2(3d) and later into an earlier release. But, you can import a backup configuration file created from an earlier release to Cisco UCS Manager release 4.2(3d) and later with or without a **Password Encryption Key**.

Creating Password Encryption Key

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **set password-encryption-key**	Displays a user prompt to enter the password encryption key: **`Enter the password encryption key:`**
Step 3	Enter the password encryption key: *password-encryption-key*	You may not see the characters while entering the **Password Encryption Key**. Once you set the **Password Encryption Key**, you can only edit the key but cannot delete it.
Step 4	Confirm the password encryption key:	Retype the **Password Encryption Key** to confirm.
Step 5	UCS-A /security # **commit-buffer**	Saves the changes to the system.

Example

The following example shows how to create a password encryption key:

```
UCS-A# scope security
UCS-A /security # set password-encryption-key
Enter the password encryption key: password_encryption_key
```

```
Confirm the password encryption key:password_encryption_key
UCS-A /security/locale* # commit-buffer
```

Recovering a Lost Password

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. The procedure to recover the password for the admin account requires you to power cycle all fabric interconnects and will lead to a temporary data transmission outage.

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 accounts through Cisco UCS Manager. However, you must log in to Cisco UCS Manager with an account that includes aaa or admin privileges.

Caution For other Cisco UCS configurations, this procedure requires you to power down all fabric interconnects. As a result, all data transmission in the Cisco UCS domain is stopped until you restart the fabric interconnects.

Note Cisco UCS Fabric Interconnects does not have seperate kernel and system images. It has a single unified image.

Determining the Leadership Role of a Fabric Interconnect

Note To determine the role of the fabric interconnects in a cluster when the admin password is lost, open the Cisco UCS Manager GUI from the IP addresses of both fabric interconnects. The subordinate fabric interconnect fails with the following message:

```
UCSM GUI is not available on secondary node.
```

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.

Example

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 B 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 Non-Cluster Configuration for Cisco UCS 6664 Fabric Interconnect

This procedure will help you to recover the password that you set for the admin account when you performed an initial system setup on the fabric interconnect. The admin account is the system administrator or superuser account.

Before you begin

1. Physically connect the console port on the fabric interconnect to a computer terminal or console server.
2. Determine the running versions of the Cisco UCS 6664 Fabric Interconnect image.

Note The Cisco UCS 6664 Fabric Interconnect does not have separate kernel and system images. It has a single unified image.

Tip To find this information, you can log in with any user account on the Cisco UCS Domain.

Procedure

Step 1 Connect to the console port.

Step 2 UCS-A(local-mgmt)# **reboot**

This reboots the fabric interconnect.

You can also power cycle the fabric interconnect.

Step 3 In the console, press **Ctrl+c** key combinations as it boots to get the `loader` prompt:

Ctrl+c

You may need to press the selected key combination multiple times before your screen displays the `loader` prompt.

Step 4 At the loader prompt, run the following command:

```
loader > cmdline recoverymode=1
```

Step 5 Boot the Fabric Interconnect image on the fabric interconnect.

```
loader > boot /installables/switch/Cisco UCS 6600 Series FI Image
```

Example:

```
loader > boot /installables/switch/ucs-6600-k9-system.7.0.3.N2.3.40.000.gbin
```

Step 6 Enter the config terminal mode.

```
switch(boot)# config terminal
```

Step 7 Reset the admin password.

```
switch(boot)(config)# admin-password New_password
```

Choose a strong password that includes at least one capital letter and one number. The password cannot be blank.

The new password displays in clear text mode.

Step 8 Exit the config terminal mode to reboot the FI.

```
switch(boot)(config)# exit
switch(boot)# exit
```

Step 9 Wait for the login prompt and use the new password to login.

```
Cisco UCS 6600 Series Fabric Interconnect
login: admin
Password:New_password
```

Step 10 Sync the new password with .

```
UCS-A # scope security
UCS-A/security  # set password
Enter new password: New_password
Confirm new password: New_password
UCS-A/security* # commit-buffer
```

Recovering the Admin Account Password in a Non-Cluster Configuration for Cisco UCS Fabric Interconnects 9108 100G

This procedure will help you to recover the password that you set for the admin account when you performed an initial system setup on the fabric interconnect. The admin account is the system administrator or superuser account.

Before you begin

1. Physically connect the console port on the fabric interconnect to a computer terminal or console server.
2. Determine the running versions of the Cisco UCS Fabric Interconnects 9108 100G (Cisco UCS X-Series Direct) image.

Note The Cisco UCS X-Series Direct does not have separate kernel and system images. It has a single unified image.

Tip To find this information, you can log in with any user account on the Cisco UCS Domain.

Procedure

Step 1 Connect to the console port.

Step 2 UCS-A(local-mgmt)# **reboot**

This reboots the fabric interconnect.

You can also power cycle the fabric interconnect.

Step 3 In the console, press **Ctrl+c** key combinations as it boots to get the `loader` prompt:

Ctrl+c

You may need to press the selected key combination multiple times before your screen displays the `loader` prompt.

Step 4 At the loader prompt, run the following command:

`loader >` **cmdline recoverymode=1**

Step 5 Boot the Fabric Interconnect image on the fabric interconnect.

`loader >` **boot /installables/switch/***Cisco UCS X-Direct FI Image*

Example:

```
loader > boot
/installables/switch/ucs-x-direct-k9-system.7.0.3.N2.3.40.000.gbin
```

Step 6 Enter the config terminal mode.

`switch(boot)#` **config terminal**

Step 7 Reset the admin password.

`switch(boot)(config)#` **admin-password** *New_password*

Choose a strong password that includes at least one capital letter and one number. The password cannot be blank.

The new password displays in clear text mode.

Step 8 Exit the config terminal mode to reboot the FI.

```
switch(boot)(config)# exit
switch(boot)# exit
```

Step 9 Wait for the login prompt and use the new password to login.

```
Cisco UCS X-Direct Fabric Interconnect
login: admin
Password:New_password
```

Step 10 Sync the new password with .

```
UCS-A # scope security
UCS-A/security  # set password
Enter new password: New_password
Confirm new password: New_password
UCS-A/security* # commit-buffer
```

Recovering the Admin Account Password in a Non-Cluster Configuration for Cisco UCS 6500 Series Fabric Interconnect

This procedure will help you to recover the password that you set for the admin account when you performed an initial system setup on the fabric interconnect. The admin account is the system administrator or superuser account.

Before you begin

1. Physically connect the console port on the fabric interconnect to a computer terminal or console server.
2. Determine the running versions of the Cisco UCS 6500 Series Fabric Interconnect image.

Note Cisco UCS 6500 Series Fabric Interconnect does not have separate kernel and system images. It has a single unified image.

Tip To find this information, you can log in with any user account on the Cisco UCS domain.

Procedure

Step 1 Connect to the console port.

Step 2 UCS-A(local-mgmt)# **reboot**

This reboots the fabric interconnect.

You can also power cycle the fabric interconnect.

Step 3 In the console, press **Ctrl+c** key combinations as it boots to get the `loader` prompt:

Ctrl+c

You may need to press the selected key combination multiple times before your screen displays the `loader` prompt.

Step 4 At the loader prompt, run the following command:

```
loader > cmdline recoverymode=1
```

Step 5 Boot the Cisco UCS 6500 Series Fabric Interconnect image on the fabric interconnect.

```
loader > boot /installables/switch/Cisco UCS 6500 FI Image
```

Example:

```
loader > boot /installables/switch/ucs-6500-k9-system.7.0.3.N2.3.40.173.gbin
```

Step 6 Enter the config terminal mode.

```
switch(boot)# config terminal
```

Step 7 Reset the admin password.

```
switch(boot)(config)# admin-password New_password
```

Choose a strong password that includes at least one capital letter and one number. The password cannot be blank.

The new password displays in clear text mode.

Step 8 Exit the config terminal mode to reboot the FI.

```
switch(boot)(config)# exit
switch(boot)# exit
```

Step 9 Wait for the login prompt and use the new password to login.

```
Cisco UCS 6500 Series Fabric Interconnect
login: admin
Password:New_password
```

Step 10 Sync the new password with Cisco UCS Manager.

```
UCS-A # scope security
UCS-A/security  # set password
Enter new password: New_password
Confirm new password: New_password
UCS-A/security* # commit-buffer
```

Recovering the Admin Account Password in a Non-Cluster Configuration for Cisco UCS 6400 Series Fabric Interconnect

This procedure will help you to recover the password that you set for the admin account when you performed an initial system setup on the fabric interconnect. The admin account is the system administrator or superuser account.

Before you begin

1. Physically connect the console port on the fabric interconnect to a computer terminal or console server.
2. Determine the running versions of the Cisco UCS 6400 Series Fabric Interconnect image.

Note Cisco UCS 6400 Series Fabric Interconnect does not have separate kernel and system images. It has a single unified image.

Tip To find this information, you can log in with any user account on the Cisco UCS domain.

Procedure

Step 1 Connect to the console port.

Step 2 UCS-A(local-mgmt)# **reboot**

This reboots the fabric interconnect.

You can also power cycle the fabric interconnect.

Step 3 In the console, press **Ctrl+c** key combinations as it boots to get the `loader` prompt:

Ctrl+c

You may need to press the selected key combination multiple times before your screen displays the `loader` prompt.

Step 4 At the loader prompt, run the following command:

```
loader > cmdline recoverymode=1
```

Step 5 Boot the Cisco UCS 6400 Series Fabric Interconnect image on the fabric interconnect.

```
loader > boot /installables/switch/Cisco UCS 6400 FI Image
```

Example:

```
loader > boot /installables/switch/ucs-6400-k9-system.7.0.3.N2.3.40.173.gbin
```

Step 6 Enter the config terminal mode.

```
switch(boot)# config terminal
```

Step 7 Reset the admin password.

```
switch(boot)(config)# admin-password New_password
```

Choose a strong password that includes at least one capital letter and one number. The password cannot be blank.

The new password displays in clear text mode.

Step 8 Exit the config terminal mode to reboot the FI.

```
switch(boot)(config)# exit
switch(boot)# exit
```

Step 9 Wait for the login prompt and use the new password to login.

```
Cisco UCS 6400 Series Fabric Interconnect
login: admin
Password:New_password
```

Step 10 Sync the new password with Cisco UCS Manager.

```
UCS-A # scope security
UCS-A/security  # set password
Enter new password: New_password
Confirm new password: New_password
UCS-A/security* # commit-buffer
```

Recovering the Admin Account Password in a Standalone Configuration for Cisco UCS 6664 Fabric Interconnect

This procedure will help you to recover the password that you set for the admin account when you performed an initial system setup on the fabric interconnect. The admin account is the system administrator or superuser account.

Before you begin

1. Physically connect the console port on the fabric interconnect to a computer terminal or console server.
2. Determine the running versions of the () image.

Note Cisco UCS 6664 Fabric Interconnect does not have separate kernel and system images. It has a single unified image.

Tip To find this information, you can log in with any user account on the Cisco UCS domain.

Procedure

Step 1 Connect to the console port.

Step 2 UCS-A(local-mgmt)# **reboot**

This reboots the fabric interconnect.

You can also power cycle the fabric interconnect.

Step 3 In the console, press **Ctrl+c** key combinations as it boots to get the `loader` prompt:

Ctrl+c

You may need to press the selected key combination multiple times before your screen displays the `loader` prompt.

Step 4 At the loader prompt, run the following command:

```
loader > cmdline recoverymode=1
```

Step 5 Boot the Cisco UCS 6664 Fabric Interconnect image on the fabric interconnect.

```
loader > boot /installables/switch/Cisco UCS 6600 FI
```

Example:

```
loader > boot /installables/switch/ucs-6600-k9-system.7.0.3.N2.3.40.173.gbin
```

Step 6 Enter the config terminal mode.

```
switch(boot)# config terminal
```

Step 7 Reset the admin password.

```
switch(boot)(config)# admin-password New_password
```

Choose a strong password that includes at least one capital letter and one number. The password cannot be blank.

The new password displays in clear text mode.

Step 8 Exit the config terminal mode to reboot the FI.

```
switch(boot)(config)# exit
switch(boot)# exit
```

Step 9 Wait for the login prompt and use the new password to login.

```
Cisco UCS 6600 Series Fabric Interconnect
login: admin
Password:New_password
```

Step 10 Sync the new password with Cisco UCS Manager.

```
UCS-A # scope security
UCS-A/security  # set password
Enter new password: New_password
Confirm new password: New_password
UCS-A/security* # commit-buffer
```

Recovering the Admin Account Password in a Standalone Configuration for Cisco UCS Fabric Interconnects 9108 100G

This procedure will help you to recover the password that you set for the admin account when you performed an initial system setup on the fabric interconnect. The admin account is the system administrator or superuser account.

Before you begin

1. Physically connect the console port on the fabric interconnect to a computer terminal or console server.
2. Determine the running versions of the Cisco UCS Fabric Interconnects 9108 100G (Cisco UCS X-Series Direct) image.

Note Cisco UCS Fabric Interconnects 9108 100G does not have separate kernel and system images. It has a single unified image.

Tip To find this information, you can log in with any user account on the Cisco UCS domain.

Procedure

Step 1 Connect to the console port.

Step 2 UCS-A(local-mgmt)# **reboot**

This reboots the fabric interconnect.

You can also power cycle the fabric interconnect.

Step 3 In the console, press **Ctrl+c** key combinations as it boots to get the `loader` prompt:

Ctrl+c

You may need to press the selected key combination multiple times before your screen displays the `loader` prompt.

Step 4 At the loader prompt, run the following command:

```
loader > cmdline recoverymode=1
```

Step 5 Boot the Cisco UCS Fabric Interconnects 9108 100G image on the fabric interconnect.

```
loader > boot /installables/switch/Cisco UCS X-Direct FI
```

Example:

```
loader > boot
/installables/switch/ucs-x-direct-k9-system.7.0.3.N2.3.40.173.gbin
```

Step 6 Enter the config terminal mode.

```
switch(boot)# config terminal
```

Step 7 Reset the admin password.

```
switch(boot)(config)# admin-password New_password
```

Choose a strong password that includes at least one capital letter and one number. The password cannot be blank.

The new password displays in clear text mode.

Step 8 Exit the config terminal mode to reboot the FI.

```
switch(boot)(config)# exit
switch(boot)# exit
```

Step 9 Wait for the login prompt and use the new password to login.

```
Cisco UCS X-Series Direct
login: admin
Password:New_password
```

Step 10 Sync the new password with Cisco UCS Manager.

```
UCS-A # scope security
UCS-A/security  # set password
Enter new password: New_password
Confirm new password: New_password
UCS-A/security* # commit-buffer
```

Recovering the Admin Account Password in a Standalone Configuration for Cisco UCS 6500 Series Fabric Interconnect

This procedure will help you to recover the password that you set for the admin account when you performed an initial system setup on the fabric interconnect. The admin account is the system administrator or superuser account.

Before you begin

1. Physically connect the console port on the fabric interconnect to a computer terminal or console server.
2. Determine the running versions of the Cisco UCS 6500 Series Fabric Interconnect image.

Note Cisco UCS 6500 Series Fabric Interconnect does not have separate kernel and system images. It has a single unified image.

Tip To find this information, you can log in with any user account on the Cisco UCS domain.

Procedure

Step 1 Connect to the console port.

Step 2 UCS-A(local-mgmt)# **reboot**

This reboots the fabric interconnect.

You can also power cycle the fabric interconnect.

Step 3 In the console, press **Ctrl+c** key combinations as it boots to get the `loader` prompt:

Ctrl+c

You may need to press the selected key combination multiple times before your screen displays the `loader` prompt.

Step 4 At the loader prompt, run the following command:

```
loader > cmdline recoverymode=1
```

Step 5 Boot the Cisco UCS 6500 Series Fabric Interconnect image on the fabric interconnect.

```
loader > boot /installables/switch/Cisco UCS 6500 FI Image
```

Example:

```
loader > boot /installables/switch/ucs-6500-k9-system.7.0.3.N2.3.40.173.gbin
```

Step 6 Enter the config terminal mode.

```
switch(boot)# config terminal
```

Step 7 Reset the admin password.

```
switch(boot)(config)# admin-password New_password
```

Choose a strong password that includes at least one capital letter and one number. The password cannot be blank.

The new password displays in clear text mode.

Step 8 Exit the config terminal mode to reboot the FI.

```
switch(boot)(config)# exit
switch(boot)# exit
```

Step 9 Wait for the login prompt and use the new password to login.

```
Cisco UCS 65
00 Series Fabric Interconnect
login: admin
Password:New_password
```

Step 10 Sync the new password with Cisco UCS Manager.

```
UCS-A # scope security
UCS-A/security  # set password
Enter new password: New_password
Confirm new password: New_password
UCS-A/security* # commit-buffer
```

Recovering the Admin Account Password in a Standalone Configuration for Cisco UCS 6400 Series Fabric Interconnect

This procedure will help you to recover the password that you set for the admin account when you performed an initial system setup on the fabric interconnect. The admin account is the system administrator or superuser account.

Before you begin

1. Physically connect the console port on the fabric interconnect to a computer terminal or console server.
2. Determine the running versions of the Cisco UCS 6400 Series Fabric Interconnect image.

Note Cisco UCS 6400 Series Fabric Interconnect does not have separate kernel and system images. It has a single unified image.

Tip To find this information, you can log in with any user account on the Cisco UCS domain.

Procedure

Step 1 Connect to the console port.

Step 2 UCS-A(local-mgmt)# **reboot**

This reboots the fabric interconnect.

You can also power cycle the fabric interconnect.

Step 3 In the console, press **Ctrl+c** key combinations as it boots to get the `loader` prompt:

Ctrl+c

You may need to press the selected key combination multiple times before your screen displays the `loader` prompt.

Step 4 At the loader prompt, run the following command:

```
loader > cmdline recoverymode=1
```

Step 5 Boot the Cisco UCS 6400 Series Fabric Interconnect image on the fabric interconnect.

```
loader > boot /installables/switch/Cisco UCS 6400 FI Image
```

Example:

```
loader > boot /installables/switch/ucs-6400-k9-system.7.0.3.N2.3.40.173.gbin
```

Step 6 Enter the config terminal mode.

```
switch(boot)# config terminal
```

Step 7 Reset the admin password.

```
switch(boot)(config)# admin-password New_password
```

Choose a strong password that includes at least one capital letter and one number. The password cannot be blank.

The new password displays in clear text mode.

Step 8 Exit the config terminal mode to reboot the FI.

```
switch(boot)(config)# exit
switch(boot)# exit
```

Step 9 Wait for the login prompt and use the new password to login.

```
Cisco UCS 6400 Series Fabric Interconnect
login: admin
Password:New_password
```

Step 10 Sync the new password with Cisco UCS Manager.

```
UCS-A # scope security
UCS-A/security  # set password
Enter new password: New_password
Confirm new password: New_password
UCS-A/security* # commit-buffer
```

Recovering the Admin Account Password in a Cluster Configuration for Cisco UCS 6664 Fabric Interconnect

This procedure will help you to recover the password that you set for the admin account when you performed an initial system setup on the fabric interconnects. The admin account is the system administrator or superuser account.

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 Cisco UCS 6664 Fabric Interconnect image.

Note Cisco UCS 6664 Fabric Interconnect does not have separate kernel and system images. It has a single unified image.

   - Which fabric interconnect has the primary leadership role and which is the subordinate.

Tip To find this information, you can log in with any user account on the Cisco UCS domain.

Procedure

Step 1 Connect to the console port of the subordinate fabric interconnect.

Step 2 UCS-B(local-mgmt)# **reboot**

This reboots the subordinate fabric interconnect.

You can also power cycle the subordinate fabric interconnect.

Step 3 In the console, press **Ctrl+c** key combinations as it boots to get the `loader` prompt:

Ctrl+c

You may need to press the selected key combination multiple times before your screen displays the `loader` prompt.

Step 4 At the loader prompt, run the following command:

```
loader > cmdline recoverymode=1
```

Step 5 Boot the Cisco UCS 6664 Fabric Interconnect image on the fabric interconnect.

```
loader > boot /installables/switch/Cisco UCS 6600 FI Image
```

Example:

```
loader > boot /installables/switch/ucs-6600-k9-system.7.0.3.N2.3.40.173.gbin
```

Step 6 Enter the config terminal mode.

```
switch(boot)# config terminal
```

Step 7 Reset the admin password.

```
switch(boot)(config)# admin-password New_password
```

Choose a strong password that includes at least one capital letter and one number. The password cannot be blank.

The new password displays in clear text mode.

Step 8 Exit the config terminal mode to reboot the FI.

```
switch(boot)(config)# exit
switch(boot)# exit
```

Step 9 Wait for the login prompt and use the new password to login.

```
Cisco UCS 6600 Series Fabric Interconnect
login: admin
Password:New_password
```

Step 10 Sync the new password with Cisco UCS Manager and other FI.

```
UCS-B # scope security
UCS-B/security  # set password
Enter new password: New_password
Confirm new password: New_password
UCS-B/security* # commit-buffer
```

Recovering the Admin Account Password in a Cluster Configuration for Cisco UCS Fabric Interconnects 9108 100G

This procedure will help you to recover the password that you set for the admin account when you performed an initial system setup on the fabric interconnects. The admin account is the system administrator or superuser account.

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 Cisco UCS Fabric Interconnects 9108 100G (Cisco UCS X-Series Direct) image.

Note Cisco UCS X-Series Direct does not have separate kernel and system images. It has a single unified image.

   - Which fabric interconnect has the primary leadership role and which is the subordinate.

Tip To find this information, you can log in with any user account on the Cisco UCS domain.

Procedure

Step 1 Connect to the console port of the subordinate fabric interconnect.

Step 2 UCS-B(local-mgmt)# **reboot**

This reboots the subordinate fabric interconnect.

You can also power cycle the subordinate fabric interconnect.

Step 3 In the console, press **Ctrl+c** key combinations as it boots to get the `loader` prompt:

Ctrl+c

You may need to press the selected key combination multiple times before your screen displays the `loader` prompt.

Step 4 At the loader prompt, run the following command:

`loader >` **cmdline recoverymode=1**

Step 5 Boot the Cisco UCS Fabric Interconnects 9108 100G image on the fabric interconnect.

`loader >` **boot /installables/switch/***Cisco UCS X-Direct FI Image*

Example:

```
loader > boot
/installables/switch/ucs-x-direct-k9-system.7.0.3.N2.3.40.173.gbin
```

Step 6 Enter the config terminal mode.

`switch(boot)#` **config terminal**

Step 7 Reset the admin password.

`switch(boot)(config)#` **admin-password** *New_password*

Choose a strong password that includes at least one capital letter and one number. The password cannot be blank.

The new password displays in clear text mode.

Step 8 Exit the config terminal mode to reboot the FI.

```
switch(boot)(config)# exit
switch(boot)# exit
```

Step 9 Wait for the login prompt and use the new password to login.

```
Cisco UCS X-Direct Fabric Interconnect
login: admin
Password:New_password
```

Step 10 Sync the new password with Cisco UCS Manager and other FI.

```
UCS-B # scope security
UCS-B/security  # set password
Enter new password: New_password
Confirm new password: New_password
UCS-B/security* # commit-buffer
```

Recovering the Admin Account Password in a Cluster Configuration for Cisco UCS 6500 Series Fabric Interconnect

This procedure will help you to recover the password that you set for the admin account when you performed an initial system setup on the fabric interconnects. The admin account is the system administrator or superuser account.

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 Cisco UCS 6500 Series Fabric Interconnect image

Note Cisco UCS 6500 Series Fabric Interconnect does not have separate kernel and system images. It has a single unified image.

   - Which fabric interconnect has the primary leadership role and which is the subordinate

Tip To find this information, you can log in with any user account on the Cisco UCS domain.

Procedure

Step 1 Connect to the console port of the subordinate fabric interconnect.

Step 2 UCS-B(local-mgmt)# **reboot**

This reboots the subordinate fabric interconnect.

You can also power cycle the subordinate fabric interconnect.

Step 3 In the console, press **Ctrl+c** key combinations as it boots to get the `loader` prompt:

Ctrl+c

You may need to press the selected key combination multiple times before your screen displays the `loader` prompt.

Step 4 At the loader prompt, run the following command:

`loader >` **cmdline recoverymode=1**

Step 5 Boot the Cisco UCS 6500 Series Fabric Interconnect image on the fabric interconnect.

`loader >` **boot /installables/switch/***Cisco UCS 6500 Series FI Image*

Example:

`loader >` **boot /installables/switch/ucs-6500-k9-system.7.0.3.N2.3.40.173.gbin**

Step 6 Enter the config terminal mode.

```
switch(boot)# config terminal
```

Step 7 Reset the admin password.

`switch(boot)(config)#` **admin-password** *New_password*

Choose a strong password that includes at least one capital letter and one number. The password cannot be blank.

The new password displays in clear text mode.

Step 8 Exit the config terminal mode to reboot the FI.

```
switch(boot)(config)# exit
switch(boot)# exit
```

Step 9 Wait for the login prompt and use the new password to login.

```
Cisco UCS 6500 Series Fabric Interconnect
login: admin
Password:New_password
```

Step 10 Sync the new password with Cisco UCS Manager and other FI.

```
UCS-B # scope security
UCS-B/security  # set password
Enter new password: New_password
```

```
Confirm new password: New_password
UCS-B/security* # commit-buffer
```

Recovering the Admin Account Password in a Cluster Configuration for Cisco UCS 6400 Series Fabric Interconnect

This procedure will help you to recover the password that you set for the admin account when you performed an initial system setup on the fabric interconnects. The admin account is the system administrator or superuser account.

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 Cisco UCS 6400 Series Fabric Interconnect image

Note Cisco UCS 6400 Series Fabric Interconnect does not have separate kernel and system images. It has a single unified image.

   - Which fabric interconnect has the primary leadership role and which is the subordinate

Tip To find this information, you can log in with any user account on the Cisco UCS domain.

Procedure

Step 1 Connect to the console port of the subordinate fabric interconnect.

Step 2 UCS-B(local-mgmt)# **reboot**

This reboots the subordinate fabric interconnect.

You can also power cycle the subordinate fabric interconnect.

Step 3 In the console, press **Ctrl+c** key combinations as it boots to get the `loader` prompt:

Ctrl+c

You may need to press the selected key combination multiple times before your screen displays the `loader` prompt.

Step 4 At the loader prompt, run the following command:

`loader >` **cmdline recoverymode=1**

Step 5 Boot the Cisco UCS 6400 Series Fabric Interconnect image on the fabric interconnect.

```
loader > boot /installables/switch/Cisco UCS 6400 Series FI Image
```

Example:

```
loader > boot /installables/switch/ucs-6400-k9-system.7.0.3.N2.3.40.173.gbin
```

Step 6 Enter the config terminal mode.

```
switch(boot)# config terminal
```

Step 7 Reset the admin password.

```
switch(boot)(config)# admin-password New_password
```

Choose a strong password that includes at least one capital letter and one number. The password cannot be blank.

The new password displays in clear text mode.

Step 8 Exit the config terminal mode to reboot the FI.

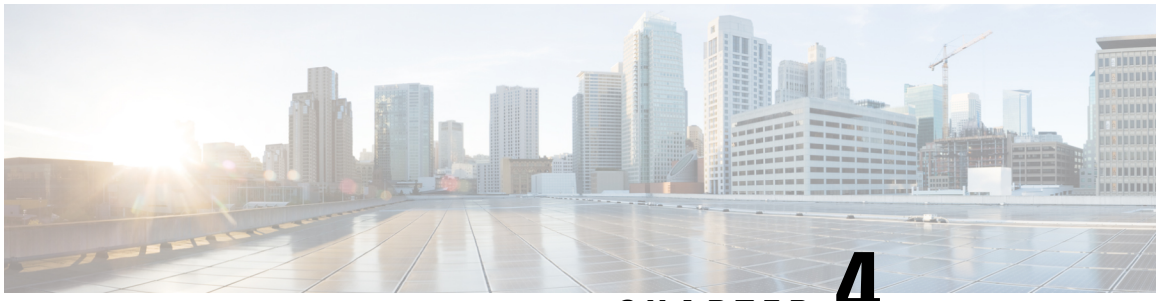
```
switch(boot)(config)# exit
switch(boot)# exit
```

Step 9 Wait for the login prompt and use the new password to login.

```
Cisco UCS 6400 Series Fabric Interconnect
login: admin
Password:New_password
```

Step 10 Sync the new password with Cisco UCS Manager and other FI.

```
UCS-B # scope security
UCS-B/security  # set password
Enter new password: New_password
Confirm new password: New_password
UCS-B/security* # commit-buffer
```



CHAPTER 4

Security Management

Security Management

The Cisco UCS Manager 4.3(5a) release introduces the **Security Management** tab in the **Admin** section. This section aims to offer multiple security management options to protect sensitive data and ensure network integrity. The tab currently includes Encryption Management and assists administrators in effectively managing security settings.

Encryption Management

Complementing the Security Management enhancements, Cisco introduces **Encryption Management**. This feature ensures that the management sessions are encrypted to prevent unauthorized access.

AES Encryption Management

The Cisco UCS Manager 4.3(5a) release introduces the AES Encryption Master Key option for Cisco UCS 6536, 6454, and 64108 Fabric Interconnects. With the Cisco UCS Manager 6.0(1b) release, this support is extended to Cisco UCS 6664 and X-Series Direct Fabric Interconnects. This feature provides encryption capabilities to protect sensitive data, enabling administrators to manage encryption settings effectively and ensure data security and compliance with encryption standards.

Creating AES Encryption

Advanced Encryption Standard (AES) is a widely used encryption standard designed to secure data. AES is considered more secure encryption algorithm and supports 128 bits or 256 bits.

Note You can use AES Encryption (Type 6) to secure key strings for authenticating MACsec sessions. For more information, see *Configuring a MACsec > Creating a MACsec Key* section of Network Management Guide.

To create AES Encryption, do the following:

Procedure

Step 1 In the Navigation pane, click **Admin**.

Step 2 Navigate to **Security Management > Encryption Management > AES Encryption**.

Step 3 In the **Actions** area, click **Create AES Encryption**.

Step 4 In the **Create AES Encryption** dialog box, complete the following fields:

Name	Description
Master Key	Enter the primary key for AES encryption. **Note** • The master key length must be between 16 to 64 characters. • The master key cannot have a combination of double quote ("), single quote ('), and space ( ). • The first and second characters of the master key cannot be a combination of single quote (') and double quote (").
Confirm Master Key	Re-enter the primary key to confirm it matches the Master Key.

Step 5 Click **OK**.

Updating the Master Key

The modification of the master key in AES encryption involves updating the primary key.

Procedure

Step 1 In the Navigation pane, click **Admin**.

Step 2 Navigate to **Security Management > Encryption Management > AES Encryption**.

Step 3 In the **Properties** area, update the existing entries in the following field:

Name	Description

Master Key	Modify the primary key used for AES encryption. **Note** • The master key length must be between 16 to 64 characters. • The master key cannot have a combination of double quote ("), single quote ('), and space ( ). • The first and second characters of the master key cannot be a combination of single quote (') and double quote (").
Master Key Set	Displays **Yes** once the Master Key field is set, indicating that the primary key is configured.
Confirm Master Key	Re-enter the primary key to confirm it matches the Master Key.

Step 4 Click **Save Changes** to confirm the master key update.

Delete AES Encryption

Deleting the AES encryption involves the removal of encryption keys and disabling the encryption mechanism that uses the Advanced Encryption Standard (AES) to secure data.

To delete AES Encryption, do the following:

Procedure

Step 1 In the Navigation pane, click **Admin**.

Step 2 Navigate to **Security Management > Encryption Management > AES Encryption**.

Step 3 In the **Actions** area, click the **Delete AES Encryption** link.

Step 4 Click **Save Changes** to confirm deletion.

Managing AES Master Key for Type-6 Encryption

To ensure the security of Type-6 keys, it is crucial that these keys are not included in backups. This prevents the possibility of restoring the keys to another system, which could compromise security. Cisco NX-OS is designed with this in mind, as it does not export the AES encryption key in the running configuration export. Therefore, even if the NX-OS running configuration is exported to another device, the Type-6 keys will not pose a security risk if the AES encryption key is not pre-configured on that device.

- **Secure Configuration Exports**: Type-6 AES encryption keys remain secure and are not inadvertently exposed during configuration exports and imports.
- **AES Master Key Export**: The AES master key is not included when exporting configurations in Cisco UCS Manager.

- **Importing Configurations**: The deployment FSM in UCSM will fail and raise a critical fault if AES encryption is not configured. A message will prompt the user to configure AES encryption.
- **Post-Configuration**: Once AES encryption is configured, the deployment FSM will successfully configure the Type-6 keys.

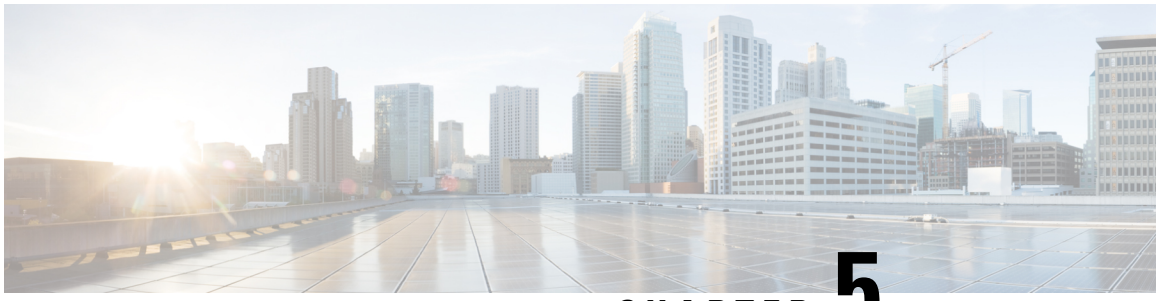
Updating AES Encryption Key:

When updating the AES Master Key, the corresponding Type-6 MACsec Keys also need to be updated. The new Type-6 MACsec key must be derived out of the new AES Master Key.

1. Configure a fallback key. The fallback key can be of Type-0, Type-7, or Type-6 key, based on the user preference.

Note If a Type-6 key is used for fallback, ensure the Type-6 key is derived out of the new master key.

2. Update the master key.
3. Delete the Type-6 MACsec key that was encrypted using the old master key.
4. Create a new Type-6 MACsec key, encrypted using the new master key, with the same Key ID.



CHAPTER 5

Role-Based Access Configuration

Role-Based Access Control Overview

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 a locale defines the organizations (domains) that a user is allowed access. Because users are not directly assigned privileges, you can manage individual user privileges by assigning the appropriate roles and locales.

A user is granted write access to the required 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 can update server configurations in the Engineering organization. They cannot, however, update server configurations in the Finance organization, unless the locales assigned to the user include the Finance organization.

User Accounts for Cisco UCS

User accounts access the system. You can configure up to 48 local user accounts in each Cisco UCS Manager domain. Each user account requires a unique username and password.

You can set user accounts with an SSH public key. The public key can be set in either of the two formats: OpenSSH or SECSH.

Admin Account

An admin account comes with each Cisco UCS domain. The admin account is a default user account and cannot be modified or deleted. This account is the system administrator or superuser account s full privileges. There is no default password assigned to the admin account; you must choose the password during the initial system setup.

The admin account is always active and does not expire. You cannot configure the admin account as inactive.

Locally Authenticated User Accounts

A locally authenticated user account is authenticated directly through the fabric interconnect and can be enabled or disabled by anyone with admin or aaa privileges. After a local user account is disabled, the user cannot log in. The database does not delete the configuration details for disabled local user accounts. If you re-enable a disabled local user account, the account becomes active with the existing configuration, including the username and password.

Remotely Authenticated User Accounts

A remotely authenticated user account is any user account that is authenticated through LDAP, RADIUS, or TACACS+.

If a user maintains a local user account and a remote user account simultaneously, the roles defined in the local user account override those maintained in the remote user account.

Expiration of User Accounts

You can configure user accounts to expire at a predefined time. When the expiration time is reached, the user account is disabled.

By default, user accounts do not expire.

Note After you configure a user account with an expiration date, you cannot reconfigure the account to not expire. However, you can configure the account to use the latest expiration date available.

Reserved Words: Locally Authenticated User Accounts

You cannot use the following words when creating a local user account in Cisco UCS.

- root
- bin
- daemon
- adm
- lp
- sync
- shutdown
- halt

- news
- uucp
- operator
- games
- gopher
- nobody
- nscd
- mailnull
- mail
- rpcuser
- rpc
- mtsuser
- ftpuser
- ftp
- man
- sys
- samdme
- debug

Web Session Limits for User Accounts

Cisco UCS Manager uses web session limits to restrict the number of web sessions (both GUI and XML) that a given user account is permitted to access at any one time.

Each Cisco UCS Manager domain supports a maximum of 32 concurrent web sessions per user and 256 total user sessions. By default, the number of concurrent web sessions allowed by Cisco UCS Manager is set to 32 per user, but you can configure this value up to the system maximum of 256.

CLI Session Limits for Cisco UCS Fabric Interconnects 9108 100G (Cisco UCS X-Series Direct): The default maximum number of CLI sessions for the Cisco UCS X-Series Direct is 16 to avoid excessive memory usage. You can increase this limit if needed, but it is recommended not to exceed 16 sessions to ensure optimal system performance and stability.

User Roles

User roles contain one or more privileges that define the operations that are allowed for a user. You can assign one or more roles to each user. Users with multiple roles have the combined privileges of all assigned roles. For example, if Role1 has storage-related privileges, and Role 2 has server-related privileges, users with Role1 and Role 2 have both storage-related and server-related privileges.

A Cisco UCS domain can contain up to 48 user roles, including the default user roles. Any user roles configured after the first 48 are accepted, but they are inactive with faults raised.

All roles include read access to all configuration settings in the Cisco UCS domain. Users with read-only roles cannot modify the system state.

You can create, modify or remove existing privileges, and delete roles. When you modify a role, the new privileges apply to all users with that role. Privilege assignment is not restricted to the privileges defined for the default roles. Meaning, you can use a custom set of privileges to create a unique role. For example, the default Server Administrator and Storage Administrator roles have a different set of privileges. However, you can create a Server and Storage Administrator role that combines the privileges of both roles.

Note If you delete a role after it was assigned to users, it is also deleted from those user accounts.

Modify the user profiles on AAA servers (RADIUS or TACACS+) to add the roles corresponding to the privileges granted to that user. The attribute stores the role information. The AAA servers return this attribute with the request and parse it to obtain the roles. LDAP servers return the roles in the user profile attributes.

Note If a local and a remote user account have the same username, Cisco UCS Manager overrides any roles assigned to the remote user with those assigned to the local user.

Default User Roles

The system contains the following default user roles:

AAA Administrator

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

Administrator

Complete read-and-write access to the entire system. Assigns this role to the default administrator account by default. You cannot change it.

Facility Manager

Read-and-write access to power management operations through the power management privilege. Read access to the remaining system.

Network Administrator

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

Operations

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

Read-Only

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

Server Compute

Read and write access to most aspects of service profiles. However, the user cannot create, modify or delete vNICs or vHBAs.

Server Equipment Administrator

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

Server Profile Administrator

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

Server Security Administrator

Read-and-write access to server security-related operations. Read access to the remaining system.

Storage Administrator

Read-and-write access to storage operations. Read access to the remaining system.

Reserved Words: User Roles

You cannot use the following words when creating custom roles in Cisco UCS.

- network-admin
- network-operator
- vdc-admin
- vdc-operator
- server-admin

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.

Tip Detailed information about these privileges and the tasks that they enable users to perform is available in *Privileges in Cisco UCS* available at the following URL: http://www.cisco.com/en/US/products/ps10281/prod_technical_reference_list.html.

Table 3: User Privileges

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

Privilege	Description	Default Role Assignment
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
org-management	Organization management	Operations
pod-config	Pod configuration	Network Administrator
pod-policy	Pod policy	Network Administrator
pod-qos	Pod QoS	Network Administrator
pod-security	Pod security	Network Administrator
power-mgmt	Read-and-write access to power management operations	Facility Manager
read-only	Read-only access Read-only cannot be selected as a privilege; it is assigned to every user role.	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-compute	Service profile compute	Server Compute 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 endpoint access	Server Profile Administrator
service-profile-network	Service profile network	Network Administrator
service-profile-network-policy	Service profile network policy	Network Administrator

Privilege	Description	Default Role Assignment
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 Profile Administrator
service-profile-server-oper	Service profile consumer	Server Profile 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

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.

Example

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.

Example

The following example shows how to add the server security and server policy privileges to the service-profile-security-admin role and commit 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 #
```

Replacing Privileges for 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.

	Command or Action	Purpose
Step 3	UCS-A /security/role # **set privilege** *privilege-name*	Replaces the existing privileges of the user role. **Note** You can specify more than one *privilege-name* on the same command line to replace the existing privilege with multiple privileges. After replacing the privileges, you can add privileges to the same role using the **add privilege** command.
Step 4	UCS-A /security/role # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example shows how to replace the existing privileges for the service-profile-security-admin role with the server security and server policy privileges and commit the transaction:

```
UCS-A# scope security
UCS-A /security # scope role service-profile-security-admin
UCS-A /security/role # set 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.

Example

The following example removes the server security and server policy privileges from 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 # 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.

Example

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 #
```

Locales

User Locales

You can assign a user to one or more locales. Each locale defines one or more organizations (domains) to which a user can access. Access is usually limited to the organizations specified in the locale. An exception is a locale without any organizations. It provides unrestricted access to system resources in all organizations.

A Cisco UCS domain can contain up to 48 user locales. Any user locales configured after the first 48 are accepted, but are inactive with faults raised.

Users with admin or aaa privileges 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, a user assigned to that locale can only assign the Engineering organization to other users.

Note You cannot assign a locale to users with one or more of the following privileges:

- aaa
- admin
- fault
- operations

You can hierarchically manage organizations. A user who is assigned to a top-level organization has automatic access to all organizations below 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.

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 org-root/org-ref-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.

Example

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 org-root/org-finance
UCS-A /security/locale* # commit-buffer
UCS-A /security/locale #
```

Assigning 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** *org-root/org-ref-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.

Example

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 org-root/org-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.

Example

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.

Example

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 #
```

Locally Authenticated User Accounts

Creating a User Account

At a minimum, Cisco recommends that you create the following users:

- Server administrator account
- Network administrator account
- Storage administrator

Before you begin

Perform the following tasks, if the system includes any of the following:

- Remote authentication services—Ensures that the users exist in the remote authentication server with the appropriate roles and privileges.
- Multitenancy with organizations—Creates one or more locales. If you do not have any locales, all users are created in root and are assigned roles and privileges in all organizations.
- SSH authentication—Obtains 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 a user account for the specified local user and enters security local user mode.
Step 3	UCS-A /security/local-user # **set account-status** {**active**| **inactive**}	Specifies whether the local user account is enabled or disabled. If the account status for a local user account is set to inactive, the user is prevented from logging into the system using their existing credentials.
Step 4	UCS-A /security/local-user # **set password** *password*	Sets the password for the user account
Step 5	(Optional) UCS-A /security/local-user # **set firstname** *first-name*	Specifies the first name of the user.
Step 6	(Optional) UCS-A /security/local-user # **set lastname** *last-name*	Specifies the last name of the user.
Step 7	(Optional) UCS-A /security/local-user # **set expiration** *month day-of-month year*	Specifies the date that the user account expires. The *month* argument is the first three letters of the month name. **Note** After you configure a user account with an expiration date, you cannot reconfigure the account to not expire. However, you can configure the account to use the latest expiration date available.
Step 8	(Optional) UCS-A /security/local-user # **set email** *email-addr*	Specifies the user e-mail address.
Step 9	(Optional) UCS-A /security/local-user # **set phone** *phone-num*	Specifies the user phone number.
Step 10	(Optional) UCS-A /security/local-user # **set sshkey** *ssh-key*	Specifies the SSH key used for passwordless access. **Note**

	Command or Action	Purpose
		By default, Cisco UCS Manager works in the Federal Information Processing Standards (FIPS) mode. In the FIPS mode, the DSA key is not supported.
Step 11	UCS-A security/local-user # **commit-buffer**	Commits the transaction.

Example

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

```
UCS-A# scope security
UCS-A /security # create local-user kikipopo
UCS-A /security/local-user* # set account-status active
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, enables the user account, 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 account-status active
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, enables the user account, 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 account-status active
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 #
```

Enabling the Password Strength Check for Locally Authenticated Users

You must have admin or aaa privileges to enable the password strength check. If enabled, Cisco UCS Manager does not permit a user to choose a password that does not meet the guidelines for a strong password.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **enforce-strong-password** {**yes** | **no**}	Specifies whether the password strength check is enabled or disabled.

Example

The following example enables the password strength check:

```
UCS-A# scope security
UCS-A /security # set enforce-strong-password yes
UCS-A /security #
```

Setting Web Session Limits for User Accounts

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 # **scope web-session-limits**	Enters system services web session limits mode.
Step 4	UCS-A /system/services/web-session-limits # **set peruser** *num-of-logins-per-user*	Sets the maximum number of concurrent HTTP and HTTPS sessions allowed for each user. Enter an integer between 1 and 256. By default, this value is set to 32.
Step 5	UCS-A /system/services/web-session-limits # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example sets the maximum number of HTTP and HTTPS sessions allowed by each user account to 60 and commits the transaction:

```
UCS-A# scope system
UCS-A /system # scope services
```

```
UCS-A /system/services # scope web-session-limits
UCS-A /system/services/web-session-limits* # set peruser 60
UCS-A /system/services/web-session-limits* # commit-buffer
UCS-A /system/services/web-session-limits #
```

Assigning a Role to a User Account

Changes in user roles and privileges do not take effect until the next time the user logs in. If a user is logged in when you assign a new role to or remove an existing role from a user account, the active session continues with the previous roles and privileges.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **scope local-user** *local-user-name*	Enters security local user mode for the specified local user account.
Step 3	UCS-A /security/local-user # **create role** *role-name*	Assigns the specified role to the user account . **Note** The **create role** command can be entered multiple times to assign more than one role to a user account.
Step 4	UCS-A security/local-user # **commit-buffer**	Commits the transaction.

Example

The following example assigns the operations role to the kikipopo local user account and commits the transaction:

```
UCS-A# scope security
UCS-A /security # scope local-user kikipopo
UCS-A /security/local-user # create role operations
UCS-A /security/local-user* # commit-buffer
UCS-A /security/local-user #
```

Assigning a Locale to a User Account

Note Do not assign locales to users with an admin or aaa role.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **scope local-user** *local-user-name*	Enters security local user mode for the specified local user account.
Step 3	UCS-A /security/local-user # **create locale** *locale-name*	Assigns the specified locale to the user account. **Note** The **create locale** command can be entered multiple times to assign more than one locale to a user account.
Step 4	UCS-A security/local-user # **commit-buffer**	Commits the transaction.

Example

The following example assigns the western locale to the kikipopo local user account and commits the transaction:

```
UCS-A# scope security
UCS-A /security # scope local-user kikipopo
UCS-A /security/local-user # create locale western
UCS-A /security/local-user* # commit-buffer
UCS-A /security/local-user #
```

Removing a Role from a User Account

Changes in user roles and privileges do not take effect until the next time the user logs in. If a user is logged in when you assign a new role to or remove an existing role from a user account, the active session continues with the previous roles and privileges.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **scope local-user** *local-user-name*	Enters security local user mode for the specified local user account.
Step 3	UCS-A /security/local-user # **delete role** *role-name*	Removes the specified role from the user account . **Note**

	Command or Action	Purpose
		The **delete role** command can be entered multiple times to remove more than one role from a user account.
Step 4	UCS-A security/local-user # **commit-buffer**	Commits the transaction.

Example

The following example removes the operations role from the kikipopo local user account and commits the transaction:

```
UCS-A# scope security
UCS-A /security # scope local-user kikipopo
UCS-A /security/local-user # delete role operations
UCS-A /security/local-user* # commit-buffer
UCS-A /security/local-user #
```

Removing a Locale from a User Account

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **scope local-user** *local-user-name*	Enters security local user mode for the specified local user account.
Step 3	UCS-A /security/local-user # **delete locale** *locale-name*	Removes the specified locale from the user account. **Note** The **delete locale** command can be entered multiple times to remove more than one locale from a user account.
Step 4	UCS-A security/local-user # **commit-buffer**	Commits the transaction.

Example

The following example removes the western locale from the kikipopo local user account and commits the transaction:

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

```
UCS-A /security/local-user #
```

Enabling or Disabling a User Account

You must have admin or aaa privileges to enable or disable a local user account.

Before you begin

Create a local user account.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **scope local-user**	Enters local-user security mode.
Step 3	UCS-A /security/local-user # **set account-status** {**active** | **inactive**}	Specifies whether the local user account is enabled or disabled. The admin user account is always set to active. It cannot be modified. **Note** If you set the account status to inactive, the configuration is not deleted from the database.

Example

The following example enables a local user account called accounting:

```
UCS-A# scope security
UCS-A /security # scope local-user accounting
UCS-A /security/local-user # set account-status active
```

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 local-user account.
Step 3	UCS-A /security # **commit-buffer**	Commits the transaction to the system configuration.

Example

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 #
```

Login Profile

The **Login Profile** feature in Cisco UCS Manager enhances security and manageability by allowing administrators to define specific login parameters and behaviors. This feature enables blocking user login attempts for a defined period after repeated failures, preventing unauthorized access and meeting security standards. It also offers customizable options to modify login controls, ensuring effective user management.

Configuring Login Profile

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**.	Enters security mode.
Step 2	UCS-A /security # **scope login-profile**	Enters the login-profile configuration mode.
Step 3	UCS-A /security # **set adminstate enable** or **set adminstate disable**	Enable or disable the admin state using **set adminstate enable** or **set adminstate disable**.
Step 4	UCS-A /security # **set attempted-within <seconds>**	Specify the duration in seconds within which a specific number of failed attempts can be made before login requests are blocked.
Step 5	UCS-A /security # **set block-login-for <seconds>**	Specify the duration in seconds for which login requests will be blocked after the specified number of failed attempts.
Step 6	UCS-A /security # **set failed-attempts <number>**	Set the number of failed attempts allowed. **Note** Replace **<seconds>** and **<number>** with the desired values within the supported range (1-65535).
Step 7	UCS-A /security /login-profile # **set user-blocking-level all-users** or **set user-blocking-level per-user**	Sets the user blocking level to either **all-users** or **per-user** based on the configuration preference. • **all-users**: Blocks login requests for all users simultaneously, including local and remote users. When this setting is active,

	Command or Action	Purpose
		no user will be able to log in until the block is removed. Additionally, existing sessions for all users may be affected or terminated once the blocking criteria are met. • **per-user**: Restricts login for a single user who exceeds the allowed number of failed login attempts within the specified time frame, ensuring that other users' sessions remain unaffected. Note that different domain users with the same username are treated as the same remote user under this configuration.
Step 8	UCS-A /security /login-profile # **commit-buffer**	Commits the changes to the system configuration.
Step 9	UCS-A /security /login-profile # **show detail**	Displays the updated login profile settings for verification.

Example

The following example demonstrates how to configure a login profile for **Per User** and commit the changes:

```
UCS-A# scope security
UCS-A/security# scope login-profile
UCS-A/security/login-profile# set adminstate enable
UCS-A/security/login-profile# set attempted-within 75
UCS-A/security/login-profile# set block-login-for 179
UCS-A/security/login-profile# set failed-attempts 4
UCS-A/security/login-profile# set user-blocking-level per-user
UCS-A/security/login-profile# commit-buffer
UCS-A/security/login-profile# show detail
Login profile:
Admin State: Enable
User blocking level: Per User
Block login for seconds: 179
Within failed login attempts: 4
Failed login attempts within seconds: 75
UCS-A/security/login-profile #
```

Monitoring User Sessions from the CLI

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 logged in to the system. An asterisk (*) next to the session ID denotes the current login session.

Example

The following example lists all local users logged in to the system. The asterisk indicates which session is the current login session.

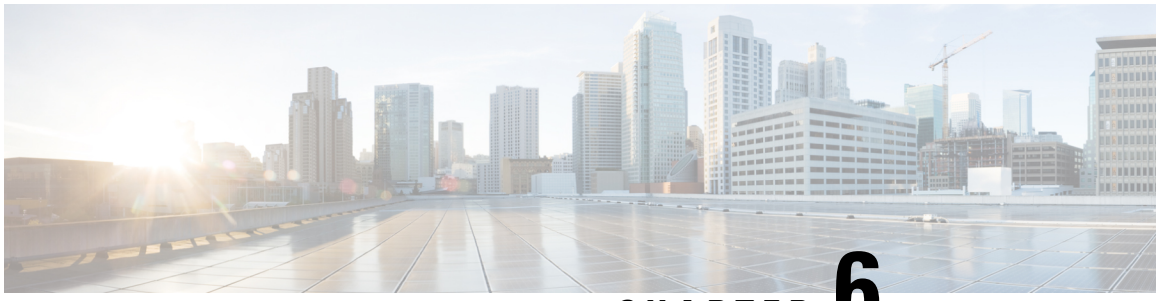
```
UCS-A# scope security
UCS-A /security # show user-session local
Session Id      User            Host                  Login Time
--------------- --------------- --------------------- ----------
pts_25_1_31264*  steve           192.168.100.111      2009-05-09T14:06:59
ttyS0_1_3532    jeff            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 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 6

Remote Authentication

Authentication Services

Cisco UCS supports the following two methods to authenticate user logins:

- Local user authentication - uses user accounts that exist locally in the Cisco UCS Manager
- Remote user authentication - uses one of the following protocols:
  - LDAP
  - RADIUS
  - TACACS+

Guidelines and Recommendations for 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 the system. The following guidelines impact user authorization:

User Accounts in Remote Authentication Services

User accounts can exist locally in Cisco UCS Manager or in the remote authentication server.

You can view the temporary sessions for users who log in through remote authentication services from the Cisco UCS Manager GUI and from the Cisco UCS Manager CLI.

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. Based on the role policy, a user might not be allowed to log in, or is granted only read-only privileges.

User Attributes in Remote Authentication Providers

For RADIUS and TACACS+ configurations, you must configure a user attribute for Cisco UCS in each remote authentication provider through which users log in to Cisco UCS Manager. This user attribute holds the roles and locales assigned to each user.

Note This step is not required for LDAP configurations that use the LDAP Group Mapping to assign roles and locales.

When a user logs in, Cisco UCS Manager does the following:

1. Queries the remote authentication service.
2. Validates the user.
3. If the user is validated, checks for the roles and locales assigned to that user.

The following table contains a comparison of the user attribute requirements for the remote authentication providers supported by Cisco UCS.

Table 4: Comparison of User Attributes by Remote Authentication Provider

Authentication Provider	Custom Attribute	Schema Extension	Attribute ID Requirements
LDAP	Not required if group mapping is used Optional if group mapping is not used	Optional. You can choose to do one of the following: • Do not extend the LDAP schema and configure an existing, unused attribute that meets the requirements. • Extend the LDAP schema and create a custom attribute with a unique name, such as CiscoAVPair.	The Cisco LDAP implementation requires a unicode type attribute. If you choose to create the CiscoAVPair custom attribute, use the following attribute ID: 1.3.6.1.4.1.9.287247.1 A sample OID is provided in the following section.

Authentication Provider	Custom Attribute	Schema Extension	Attribute ID Requirements
RADIUS	Optional	Optional. You can choose to do one of the following: • Do not extend the RADIUS schema and use an existing unused attribute that meets the requirements. • Extend the RADIUS schema and create a custom attribute with a unique name, such as cisco-avpair.	The vendor ID for the Cisco RADIUS implementation is 009 and the vendor ID for the attribute is 001. The following syntax example shows how to specify multiples user roles and locales if you choose to create the cisco-avpair attribute: `shell:roles="admin,aaa" shell:locales="L1,abc"`. Use a comma "," as the delimiter to separate multiple values.
TACACS+	Required	Required. You must extend the schema and create a custom attribute with the name cisco-av-pair.	The cisco-av-pair name is the string that provides the attribute ID for the TACACS+ provider. The following syntax example shows how to specify multiples user roles and locales when you create the cisco-av-pair attribute: `cisco-av-pair=shell:roles="admin aaa" shell:locales*"L1 abc"`. Using an asterisk (*) in the cisco-av-pair attribute syntax flags the locale as optional, preventing authentication failures for other Cisco devices that use the same authorization profile. Use a space as the delimiter to separate multiple values.

Sample OID for LDAP User Attribute

The following is a sample OID for a custom CiscoAVPair attribute:

```
CN=CiscoAVPair,CN=Schema,
CN=Configuration,CN=X
objectClass: top
objectClass: attributeSchema
cn: CiscoAVPair
distinguishedName: CN=CiscoAVPair,CN=Schema,CN=Configuration,CN=X
instanceType: 0x4
uSNCreated: 26318654
attributeID: 1.3.6.1.4.1.9.287247.1
attributeSyntax: 2.5.5.12
isSingleValued: TRUE
showInAdvancedViewOnly: TRUE
adminDisplayName: CiscoAVPair
adminDescription: UCS User Authorization Field
oMSyntax: 64
```

```
lDAPDisplayName: CiscoAVPair
name: CiscoAVPair
objectCategory: CN=Attribute-Schema,CN=Schema,CN=Configuration,CN=X
```

Two-Factor Authentication

Cisco UCS Manager uses two-factor authentication for remote user logins, which adds a level of security to account logins. Two-factor authentication login requires a username, a token, and a password combination in the password field. You can provide a PIN, a certificate, or a token.

Two-factor authentication uses authentication applications that maintain token servers to generate one-time tokens for users during the login process and store passwords in the AAA server. Requests are sent to the token server to retrieve a vendor-specific attribute. Cisco UCS Manager expects the token server to integrate with the AAA server, therefore it forwards the request to the AAA server. The password and token are validated at the same time by the AAA server. Users must enter the token and password sequence in the same order as it is configured in the AAA server.

Two-factor authentication is supported by associating RADIUS or TACACS+ provider groups with designated authentication domains and enabling two-factor authentication for those domains. Two-factor authentication does not support IPM and is not supported when the authentication realm is set to LDAP, local, or none.

Web Session Refresh and Web Session Timeout Period

The **Web Session Refresh Period** is the maximum amount of time allowed between refresh requests for a Cisco UCS Manager GUI web session. The **Web Session Timeout** is the maximum amount of time that can elapse after the last cookie/token refresh request has failed before a Cisco UCS Manager GUI web session becomes inactive.

You can increase the **Web Session Refresh Period** to a value greater than 60 seconds up 172800 seconds to avoid frequent session timeouts that requires regenerating and re-entering a token and password multiple times. The default value is 7200 seconds when two-factor authentication is enabled, and is 600 seconds when two-factor authentication is not enabled.

You can specify a value between 300 and 172800 for the **Web Session Timeout Period**. The default is 8000 seconds when two-factor authentication is enabled, and 7200 seconds when two-factor authentication is not enabled.

LDAP Providers and Groups

Nested LDAP Groups

You can add an LDAP group as a member of another group and nest groups to consolidate member accounts and to reduce the replication of traffic. Cisco UCS Manager release 2.1(2) and higher enables you to search LDAP groups that are nested within another group defined in an LDAP group map.

Note Nested LDAP search support is supported only for Microsoft Active Directory servers. The supported versions are Microsoft Windows 2003 SP3, Microsoft Windows 2008 R2, and Microsoft Windows 2012.

By default, user rights are inherited when you nest an LDAP group within another group. For example, if you make Group_1 a member of Group_2, the users in Group_1 have the same permissions as the members of Group_2. You can then search users that are members of Group_1 by choosing only Group_2 in the LDAP group map, instead of having to search Group_1 and Group_2 separately.

You do not always need to create subgroups in a group map in Cisco UCS Manager.

LDAP Group Rule

The LDAP group rule determines whether Cisco UCS should use LDAP groups when assigning user roles and locales to a remote user.

Configuring Properties for LDAP Providers

The properties that you configure in this task are the default settings for all provider connections of this type defined in Cisco UCS Manager. If an individual provider includes a setting for any of these properties, Cisco UCS uses that setting and ignores the default setting.

If you are using Active Directory as your LDAP server, create a user account in the Active Directory server to bind with Cisco UCS. Give this account a non-expiring password.

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	(Optional) UCS-A /security/ldap # **set timeout** *seconds*	Sets the time interval the system waits for a response from the LDAP server before noting the server as down.
Step 7	UCS-A /security/ldap # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example sets the LDAP attribute to CiscoAvPair, the base distinguished name to "DC=cisco-ucsm-aaa3,DC=qalab,DC=com", the filter to sAMAccountName=$userid, and the timeout interval to 5 seconds, and commits the transaction:

```
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* # commit-buffer
UCS-A /security/ldap #
```

Note User login will fail if the userdn for an LDAP user exceeds 255 characters.

What to do next

Create an LDAP provider.

Creating an LDAP Provider

Cisco UCS Manager supports a maximum of 16 LDAP providers.

Before you begin

If you are using Active Directory as your LDAP server, create a user account in the Active Directory server to bind with Cisco UCS. Give this account a non-expiring password.

- In the LDAP server, perform one of the following configurations:
  - Configure LDAP groups. LDAP groups contain user role and locale information.
  - Configure users with the attribute that holds the user role and locale information for Cisco UCS Manager. You can choose whether to extend the LDAP schema for this attribute. If you do not want to extend the schema, use an existing LDAP attribute to hold the Cisco UCS user roles and locales. If you prefer to extend the schema, create a custom attribute, such as the CiscoAVPair attribute.

    The Cisco LDAP implementation requires a unicode type attribute.

    If you choose to create the CiscoAVPair custom attribute, use the following attribute ID: 1.3.6.1.4.1.9.287247.1
  - For a cluster configuration, add the management port IPv4 or IPv6 addresses for both fabric interconnects. This configuration ensures that remote users can continue to log in if the first fabric interconnect fails and the system fails over to the second fabric interconnect. All login requests are sourced from these IP addresses, not the virtual IPv4 or IPv6 address used by Cisco UCS Manager.
- If you want to use secure communications, create a trusted point containing the certificate of the root certificate authority (CA) of the LDAP server in Cisco UCS Manager.
- Ensure the *Assign Default Role* option is set in the Role Policy for remote users. This setting enables successful remote authentication when the authentication server does not provide a role. If this setting is not enabled, login fails even when the correct credentials are entered.
- If you need to change the LDAP providers or add or delete them, you need to change the authentication realm for the domain to local, make the changes to the providers, and then change the domain authentication realm back to LDAP.

- If you want to use the special characters listed in the following table for defining the attributes of an Active Directory bind distinguished name, you must replace the special character with an escape, by using a backslash (\) followed by the corresponding hexadecimal value of the character.

Special Character	Description	Hexadecimal Value
,	comma	0x2C
+	plus sign	0x2B
"	double quote	0x22
\	backslash	0x5C
<	left angle bracket	0x3C
>	right angle bracket	0x3E
;	semicolon	0x3B
LF	line feed	0x0A
CR	carriage return	0x0D
=	equals sign	0x3D
/	forwards slash	0x2F

https://msdn.microsoft.com/en-us/library/aa366101 provides more details on replacing special characters with its escape and hexadecimal equivalent.

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 # **create server** *server-name*	Creates an LDAP server instance and enters security LDAP server mode. If SSL is enabled, the *server-name* , typically an IP address or FQDN, must exactly match a Common Name (CN) in the LDAP server's security certificate. Unless an IP address is specified, a DNS server must be configured in Cisco UCS Manager.
Step 4	(Optional) UCS-A /security/ldap/server # **set attribute** *attr-name*	An LDAP attribute that stores the values for the user roles and locales. This property is always a name-value pair. The system queries the user record for the value that matches this attribute name.

	Command or Action	Purpose
		If you do not want to extend your LDAP schema, you can configure an existing, unused LDAP attribute with the Cisco UCS roles and locales. Alternatively, you can create an attribute named CiscoAVPair in the remote authentication service with the following attribute ID: `1.3.6.1.4.1.9.287247.1` This value is required unless a default attribute has been set on the LDAP **General** tab.
Step 5	(Optional) UCS-A /security/ldap/server # **set basedn** *basedn-name*	The specific distinguished name in the LDAP hierarchy where the server begins a search when a remote user logs in and the system attempts to obtain the user's DN based on their username. You can set the length of the base DN to a maximum of 255 characters minus the length of CN=username, where username identifies the remote user attempting to access Cisco UCS Manager using LDAP authentication. This value is required unless a default base DN has been set on the LDAP **General** tab.
Step 6	(Optional) UCS-A /security/ldap/server # **set binddn** *binddn-name*	The distinguished name (DN) for an LDAP database account that has read and search permissions for all objects under the base DN. The maximum supported string length is 255 ASCII characters.
Step 7	(Optional) UCS-A /security/ldap/server # **set filter** *filter-value*	The LDAP search is restricted to those user names that match the defined filter. This value is required unless a default filter has been set on the LDAP **General** tab.
Step 8	Required: UCS-A /security/ldap/server # **set password**	The password for the LDAP database account specified in the **Bind DN** field. You can enter any standard ASCII characters except for space, § (section sign), ? (question mark), or = (equal sign). To set the password, press **Enter** after typing the **set password** command and enter the key value at the prompt.
Step 9	(Optional) UCS-A /security/ldap/server # **set order** *order-num*	The order that the Cisco UCS uses this provider to authenticate users.

	Command or Action	Purpose
Step 10	(Optional) UCS-A /security/ldap/server # **set port** *port-num*	The port through which Cisco UCS communicates with the LDAP database. The standard port number is 389.
Step 11	UCS-A /security/ldap/server # **set ssl** {**yes** **no**}	Enables or disables the use of encryption when communicating with the LDAP server. The options are as follows: • **yes** —Encryption is required. If encryption cannot be negotiated, the connection fails. If enabled, do not change the port to 636, leave it as 389. Cisco UCS negotiates a TLS session on port 636 for SSL, but initial connection starts unencrypted on 389. • **no** —Encryption is disabled. Authentication information is sent as clear text. LDAP uses STARTTLS. This allows encrypted communication using port 389.
Step 12	UCS-A /security/ldap/server # **set timeout** *timeout-num*	The length of time in seconds the system spends trying to contact the LDAP database before it times out. Enter an integer from 1 to 60 seconds, or enter 0 (zero) to use the global timeout value specified on the LDAP **General** tab. The default is 30 seconds.
Step 13	UCS-A /security/ldap/server # **set vendor** *{ms-ad | openldap}*	Enables or disables the use of the nested LDAP group search capability on the LDAP server. The options are as follows: • **ms-ad**—Nested LDAP group searches are supported with this option. If you set the vendor to *ms-ad* (Microsoft Active Directory), and enable and set the *ldap-group-rule* to recursive, Cisco UCS Manager can search through any nested LDAP groups. • **openldap**—Nested LDAP group searches are not supported with this option. If you set the vendor to *openldap*, and enable and set the *ldap-group-rule* to recursive, Cisco UCS Managerwill not search through any nested LDAP groups. If you choose this option, you must create each

	Command or Action	Purpose
		LDAP subgroup as an LDAP group map in Cisco UCS Manager, even if the parent group is already set up in a group map. **Note** When you upgrade Cisco UCS Manager from an earlier version to release 2.1(2), the LDAP provider's vendor attribute is set to **openldap** by default, and LDAP authentication continues to operate successfully.
Step 14	UCS-A /security/ldap/server # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example creates an LDAP server instance named 10.193.169.246, configures the binddn, password, order, port, SSL settings, vendor attribute, and commits the transaction:

```
UCS-A# scope security
UCS-A /security # scope ldap
UCS-A /security/ldap* # create server 10.193.169.246
UCS-A /security/ldap/server* # set binddn
"cn=Administrator,cn=Users,DC=cisco-ucsm-aaa3,DC=qalab,DC=com"
UCS-A /security/ldap/server* # set password
Enter the password:
Confirm the password:
UCS-A /security/ldap/server* # set order 2
UCS-A /security/ldap/server* # set port 389
UCS-A /security/ldap/server* # set ssl yes
UCS-A /security/ldap/server* # set timeout 30
UCS-A /security/ldap/server* # set vendor ms-ad
UCS-A /security/ldap/server* # commit-buffer
UCS-A /security/ldap/server #
```

The following example creates an LDAP server instance named 12:31:71:1231:45b1:0011:011:900, configures the binddn, password, order, port, SSL settings, vendor attribute, and commits the transaction:

```
UCS-A# scope security
UCS-A /security # scope ldap
UCS-A /security/ldap* # create server 12:31:71:1231:45b1:0011:011:900
UCS-A /security/ldap/server* # set binddn
"cn=Administrator,cn=Users,DC=cisco-ucsm-aaa3,DC=qalab,DC=com"
UCS-A /security/ldap/server* # set password
Enter the password:
Confirm the password:
UCS-A /security/ldap/server* # set order 1
UCS-A /security/ldap/server* # set port 389
UCS-A /security/ldap/server* # set ssl yes
UCS-A /security/ldap/server* # set timeout 45
UCS-A /security/ldap/server* # set vendor ms-ad
UCS-A /security/ldap/server* # commit-buffer
UCS-A /security/ldap/server #
```

What to do next

For implementations involving a single LDAP database, select LDAP as the authentication service.

For implementations involving multiple LDAP databases, configure an LDAP provider group.

Changing the LDAP Group Rule for 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 # **scope server** *ldap-provider*	Enters security LDAP provider mode.
Step 4	UCS-A /security/ldap/server # **scope ldap-group-rule**	Enters LDAP group rule mode.
Step 5	UCS-A /security/ldap/server/ldap-group-rule # **set authorization** {**enable** | **disable**}	Specifies whether Cisco UCS searches LDAP groups when assigning user roles and locales to a remote user. • **disable**—Cisco UCS does not access any LDAP groups. • **enable**—Cisco UCS searches the LDAP provider groups mapped in this Cisco UCS domain. If the remote user is found, Cisco UCS assigns the user roles and locales defined for that LDAP group in the associated LDAP group map. **Note** Role and locale assignment is cumulative. If a user is included in multiple groups, or has a role or locale specified in the LDAP attribute, Cisco UCS assigns that user all the roles and locales mapped to any of those groups or attributes.
Step 6	UCS-A /security/ldap/server/ldap-group-rule # **set member-of-attribute** *attr-name*	The attribute Cisco UCS uses to determine group membership in the LDAP database. The supported string length is 63 characters. The default string is **`memberOf`**.
Step 7	UCS-A /security/ldap/server/ldap-group-rule # **set traversal** {**non-recursive** | **recursive**}	Specifies whether Cisco UCS takes the settings for a group member's parent group, if necessary. This can be:

	Command or Action	Purpose
		• **non-recursive**—Cisco UCS only searches those groups that the user belongs to. • **recursive**—Cisco UCS searches all the ancestor groups belonging to the user.
Step 8	UCS-A /security/ldap/server/ldap-group-rule # **set use-primary-group** {**yes** | **no**}	Configures the primary group as an LDAP group map in Cisco UCS domain for membership validation. You can enable Cisco UCS Manager to download and verify the user primary group membership.
Step 9	UCS-A /security/ldap/server/ldap-group-rule # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example sets the LDAP group rule to enable authorization, sets the member of attribute to memberOf, sets the traversal to non-recursive, and commits the transaction:

```
UCS-A# scope security
UCS-A /security # scope ldap
UCS-A /security/ldap # scope server ldapprovider
UCS-A /security/ldap/server # scope ldap-group-rule
UCS-A /security/ldap/server/ldap-group-rule # set authorization enable
UCS-A /security/ldap/server/ldap-group-rule* # set member-of-attribute memberOf
UCS-A /security/ldap/server/ldap-group-rule* # set traversal non-recursive
UCS-A /security/ldap/server/ldap-group-rule* # set use-primary-group yes
UCS-A /security/ldap/server/ldap-group-rule* # commit-buffer
UCS-A /security/ldap/server/ldap-group-rule #
```

Deleting 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 # **delete server** *serv-name*	Deletes the specified server.
Step 4	UCS-A /security/ldap # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes the LDAP server called ldap1 and commits the transaction:

```
UCS-A# scope security
UCS-A /security # scope ldap
UCS-A /security/ldap # delete server ldap1
UCS-A /security/ldap* # commit-buffer
UCS-A /security/ldap #
```

LDAP Group Mapping

LDAP group mapping eliminates having to define role or locale information in the LDAP user object. UCSM can use group membership information to assign a role or locale to an LDAP user during login for organizations using LDAP groups to restrict access to LDAP databases.

When a user logs in to Cisco UCS Manager, the LDAP group map pulls information about the user's role and locale. If the role and locale criteria match the information in the policy, access is granted. Cisco UCS Manager supports a maximum of 28, 128, or 160 LDAP group maps depending on the release version.

Note Cisco UCS Manager Release 3.1(1) supports a maximum of 128 LDAP group maps, and Release 3.1(2) and later releases support a maximum of 160 LDAP group maps.

The role and locale definitions that you configure locally in the Cisco UCS Manager do not update automatically based on changes to an LDAP directory. When deleting or renaming LDAP groups in an LDAP directory, you must also update the Cisco UCS Manager with the change.

You can configure an LDAP group map to include any of the following combinations of roles and locales:

- Roles only
- Locales only
- Both roles and locales

For example, consider an LDAP group representing a group of server administrators at a specific location. The LDAP group map might include user roles such as server profile and server equipment. To restrict access to server administrators at a specific location, you can set the locale to a particular site name.

Note Cisco UCS Manager includes out-of-the-box user roles, but does not include any locales. Mapping an LDAP provider group to a locale requires that you create a custom locale.

Creating an LDAP Group Map

Before you begin

- Create an LDAP group in the LDAP server.

- Configure the distinguished name for the LDAP group in the LDAP server.
- Create locales in Cisco UCS Manager (optional).
- Create custom roles in Cisco UCS Manager (optional).

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 # **create ldap-group** *group-dn*	Creates an LDAP group map for the specified DN. The maximum number of characters for group-dn is 240. **Note** If you plan to enter a special character for this command, you need to prefix the special character with an escape character \\ (double back slash).
Step 4	UCS-A /security/ldap/ldap-group # **create locale** *locale-name*	Maps the LDAP group to the specified locale.
Step 5	UCS-A /security/ldap/ldap-group # **create role** *role-name*	Maps the LDAP group to the specified role.
Step 6	UCS-A /security/ldap/ldap-group # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example maps the LDAP group mapped to a DN, sets the locale to pacific, sets the role to admin, and commits the transaction:

```
UCS-A# scope security
UCS-A /security # scope ldap
UCS-A /security/ldap # create ldap-group cn=security,cn=users,dc=lab,dc=com
UCS-A /security/ldap/ldap-group* # create locale pacific
UCS-A /security/ldap/ldap-group* # create role admin
UCS-A /security/ldap/ldap-group* # commit-buffer
UCS-A /security/ldap/ldap-group #
```

What to do next

Set the LDAP group rule.

Deleting an LDAP Group Map

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 # **delete ldap-group** *group-dn*	Deletes the LDAP group map for the specified DN.
Step 4	UCS-A /security/ldap # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes an LDAP group map and commits the transaction:

```
UCS-A# scope security
UCS-A /security # scope ldap
UCS-A /security/ldap # delete ldap-group cn=security,cn=users,dc=lab,dc=com
UCS-A /security/ldap* # commit-buffer
UCS-A /security/ldap #
```

RADIUS Providers

Configuring Properties for RADIUS Providers

The properties that you configure in this task are the default settings for all provider connections of this type defined in Cisco UCS Manager. If an individual provider includes a setting for any of these properties, Cisco UCS uses that setting and ignores the default setting.

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	(Optional) UCS-A /security/radius # **set retries** *retry-num*	Sets the number of times to retry communicating with the RADIUS server before noting the server as down.
Step 4	(Optional) UCS-A /security/radius # **set timeout** *seconds*	Sets the time interval that the system waits for a response from the RADIUS server before noting the server as down.

	Command or Action	Purpose
Step 5	UCS-A /security/radius # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example sets the RADIUS retries to 4, sets the timeout interval to 30 seconds, and commits the transaction:

```
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* # commit-buffer
UCS-A /security/radius #
```

What to do next

Create a RADIUS provider.

Creating a RADIUS Provider

Cisco UCS Manager supports a maximum of 16 RADIUS providers.

Before you begin

Perform the following configuration in the RADIUS server:

- Configure users with the attribute that holds the user role and locale information for Cisco UCS Manager. You can choose whether to extend the RADIUS schema for this attribute. If you do not want to extend the schema, use an existing RADIUS attribute to hold the Cisco UCS user roles and locales. If you prefer to extend the schema, create a custom attribute, such as the cisco-avpair attribute.

  The vendor ID for the Cisco RADIUS implementation is 009 and the vendor ID for the attribute is 001.

  The following syntax example shows how to specify multiples user roles and locales if you choose to create the cisco-avpair attribute: `shell:roles="admin,aaa" shell:locales="L1,abc"`. Use a comma "," as the delimiter to separate multiple values.

- For a cluster configuration, add the management port IPv4 or IPv6 addresses for both fabric interconnects. This configuration ensures that remote users can continue to log in if the first fabric interconnect fails and the system fails over to the second fabric interconnect. All login requests are sourced from these IP addresses, not the virtual IP address used by Cisco UCS Manager.

- Ensure the *Assign Default Role* option is set in the Role Policy for remote users. This setting enables successful remote authentication when the authentication server does not provide a role. If this setting is not enabled, login fails even when the correct credentials are entered.

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 # **create server** *server-name*	Creates a RADIUS server instance and enters security RADIUS server mode
Step 4	(Optional) UCS-A /security/radius/server # **set authport** *authport-num*	Specifies the port used to communicate with the RADIUS server.
Step 5	UCS-A /security/radius/server # **set key**	Sets the RADIUS server key. To set the key value, press **Enter** after typing the **set key** command and enter the key value at the prompt.
Step 6	(Optional) UCS-A /security/radius/server # **set order** *order-num*	Specifies when in the order this server will be tried.
Step 7	(Optional) UCS-A /security/radius/server # **set retries** *retry-num*	Sets the number of times to retry communicating with the RADIUS server before noting the server as down.
Step 8	(Optional) UCS-A /security/radius/server # **set timeout** *seconds*	Sets the time interval that the system waits for a response from the RADIUS server before noting the server as down. **Tip** It is recommended that you configure a higher **Timeout** value if you select two-factor authentication for RADIUS providers.
Step 9	UCS-A /security/radius/server # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example creates a server instance named radiusserv7, sets the authentication port to 5858, sets the key to radiuskey321, sets the order to 2, sets the retries to 4, sets the timeout to 30, enables two-factor authentication, and commits the transaction:

```
UCS-A# scope security
UCS-A /security # scope radius
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
UCS-A /security/radius/server* # set order 2
UCS-A /security/radius/server* # set retries 4
UCS-A /security/radius/server* # set timeout 30
```

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

What to do next

For implementations involving a single RADIUS database, select RADIUS as the primary authentication service.

For implementations involving multiple RADIUS databases, configure a RADIUS provider group.

Deleting 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 # **delete server** *serv-name*	Deletes the specified server.
Step 4	UCS-A /security/radius # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes the RADIUS server called radius1 and commits the transaction:

```
UCS-A# scope security
UCS-A /security # scope radius
UCS-A /security/radius # delete server radius1
UCS-A /security/radius* # commit-buffer
UCS-A /security/radius #
```

TACACS+ Providers

Configuring Properties for TACACS+ Providers

The properties that you configure in this task are the default settings for all provider connections of this type defined in Cisco UCS Manager. If an individual provider includes a setting for any of these properties, Cisco UCS uses that setting and ignores the default setting.

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	(Optional) UCS-A /security/tacacs # **set timeout** *seconds*	Sets the time interval that the system waits for a response from the TACACS+ server before noting the server as down.
Step 4	UCS-A /security/tacacs # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example sets the TACACS+ timeout interval to 45 seconds and commits the transaction:

```
UCS-A# scope security
UCS-A /security # scope tacacs
UCS-A /security/tacacs # set timeout 45
UCS-A /security/tacacs* # commit-buffer
UCS-A /security/tacacs #
```

What to do next

Create a TACACS+ provider.

Creating a TACACS+ Provider

Cisco UCS Manager supports a maximum of 16 TACACS+ providers.

Before you begin

Perform the following configuration in the TACACS+ server:

- Create the cisco-av-pair attribute. You cannot use an existing TACACS+ attribute.

  The cisco-av-pair name is the string that provides the attribute ID for the TACACS+ provider.

  The following syntax example shows how to specify multiples user roles and locales when you create the cisco-av-pair attribute: `cisco-av-pair=shell:roles="admin aaa" shell:locales*"L1 abc"`. Using an asterisk (*) in the cisco-av-pair attribute syntax flags the locale as optional, preventing authentication failures for other Cisco devices that use the same authorization profile. Use a space as the delimiter to separate multiple values.

- For a cluster configuration, add the management port IPv4 or IPv6 addresses for both fabric interconnects. This configuration ensures that remote users can continue to log in if the first fabric interconnect fails and the system fails over to the second fabric interconnect. All login requests are sourced from these IP addresses, not the virtual IP address used by Cisco UCS Manager.

- Ensure the *Assign Default Role* option is set in the Role Policy for remote users. This setting enables successful remote authentication when the authentication server does not provide a role. If this setting is not enabled, login fails even when the correct credentials are entered.

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 # **create server** *server-name*	Creates an TACACS+ server instance and enters security TACACS+ server mode
Step 4	(Optional) UCS-A /security/tacacs/server # **set key**	Sets the TACACS+ server key. To set the key value, press **Enter** after typing the **set key** command and enter the key value at the prompt.
Step 5	(Optional) UCS-A /security/tacacs/server # **set order** *order-num*	Specifies when in the order this server will be tried.
Step 6	(Optional) UCS-A /security/tacacs/server # **set timeout***seconds*	Sets the time interval that the system waits for a response from the TACACS+ server before noting the server as down. **Tip** It is recommended that you configure a higher timeout value if you select two-factor authentication for TACACS+ providers.
Step 7	UCS-A /security/tacacs/server # **set port** *port-num*	Specifies the port used to communicate with the TACACS+ server.
Step 8	UCS-A /security/tacacs/server # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example creates a server instance named tacacsserv680, sets the key to tacacskey321 and confirms the key, sets the order to 4, sets the authentication port to 5859, and commits the transaction:

```
UCS-A# scope security
UCS-A /security # scope tacacs
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 order 4
UCS-A /security/tacacs/server* # set port 5859
UCS-A /security/tacacs/server* # commit-buffer
UCS-A /security/tacacs/server #
```

What to do next

For implementations involving a single TACACS+ database, select TACACS+ as the primary authentication service.

For implementations involving multiple TACACS+ databases, configure a TACACS+ provider group.

Deleting 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 # **delete server** *serv-name*	Deletes the specified server.
Step 4	UCS-A /security/tacacs # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes the TACACS server called tacacs1 and commits the transaction:

```
UCS-A# scope security
UCS-A /security # scope tacacs
UCS-A /security/tacacs # delete server TACACS1
UCS-A /security/tacacs* # commit-buffer
UCS-A /security/tacacs #
```

Multiple Authentication Systems

Multiple Authentication Services

You can configure Cisco UCS to use multiple authentication services by configuring the following features:

- Provider groups
- Authentication domains

After provider groups and authentication domains are configured in Cisco UCS Manager, you can use the following syntax to log in to the system using Cisco UCS Manager CLI: **ucs:** *auth-domain* \ *user-name* .

When multiple authentication domains and native authentication are configured with a remote authentication service, use one of the following syntax examples to log in with SSH, Telnet or Putty.

Note SSH log in is case-sensitive.

From a Linux terminal using SSH:

- **ssh ucs-***auth-domain**username*@{*UCSM-ip-address* | *UCMS-ipv6-address*}

```
ssh ucs-example\\jsmith@192.0.20.11
ssh ucs-example\\jsmith@2001::1
```

- **ssh -l ucs-***auth-domain**username* {*UCSM-ip-address*| *UCSM-ipv6-address*| *UCSM-host-name*}

```
ssh -l ucs-example\\jsmith 192.0.20.11
ssh -l ucs-example\\jsmith 2001::1
```

- **ssh** {*UCSM-ip-address* | *UCSM-ipv6-address* | *UCSM-host-name*} **-l ucs-***auth-domain**username*

```
ssh 192.0.20.11 -l ucs-example\\jsmith
ssh 2001::1 -l ucs-example\\jsmith
```

- **ssh ucs-***auth-domain**username*@{*UCSM-ip-address* | *UCSM-ipv6-address*}

```
ssh ucs-ldap23\\jsmith@192.0.20.11
ssh ucs-ldap23\\jsmith@2001::1
```

From a Linux terminal using Telnet:

- **telnet ucs-***UCSM-host-name* **ucs-***auth-domain**username*

```
telnet ucs-qa-10
login: ucs-ldap23\blradmin
```

- **telnet ucs-**{*UCSM-ip-address* | *UCSM-ipv6-address*}*ucs-auth-domain**username*

```
telnet 10.106.19.12 2052
ucs-qa-10-A login: ucs-ldap23\blradmin
```

From a Putty client:

- Login as: **ucs-***auth-domain**username*

```
Login as: ucs-example\jsmith
```

Note If the default authentication is set to local, and the console authentication is set to LDAP, you can log in to the fabric interconnect from a Putty client using **ucs-local\admin**, where admin is the name of the local account.

Configuring Multiple Authentication Systems

Provider Groups

A provider group is a set of providers that the Cisco UCS accesses during the authentication process. All of the providers within a provider group are accessed in the order that the Cisco UCS provider uses to authenticate

users. If all of the configured servers are unavailable or unreachable, Cisco UCS Manager automatically falls back to the local authentication method using the local username and password.

Cisco UCS Manager allows you to create a maximum of 16 provider groups, with a maximum of eight providers allowed per group.

Creating an LDAP Provider Group

Creating an LDAP provider group allows you to authenticate using multiple LDAP databases.

Before you begin

Create one or more LDAP providers.

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 # **create auth-server-group** *auth-server-group-name*	Creates an LDAP provider group and enters authentication server group security LDAP mode.
Step 4	UCS-A /security/ldap/auth-server-group # **create server-ref** *ldap-provider-name*	Adds the specified LDAP provider to the LDAP provider group and enters server reference authentication server group security LDAP mode.
Step 5	UCS-A /security/ldap/auth-server-group/server-ref # **set order** *order-num*	Specifies the order in which Cisco UCS uses this provider to authenticate users. Valid values include no-value and 0-16, with the lowest value indicating the highest priority. Setting the order to no-value is equivalent to giving that server reference the highest priority.
Step 6	UCS-A /security/ldap/auth-server-group/server-ref # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example creates an LDAP provider group called ldapgroup, adds two previously configured providers called ldap1 and ldap2 to the provider group, sets the order, and commits the transaction:

```
UCS-A# scope security
UCS-A /security # scope ldap
UCS-A /security/ldap # create auth-server-group ldapgroup
UCS-A /security/ldap/auth-server-group* # create server-ref ldap1
UCS-A /security/ldap/auth-server-group/server-ref* # set order 1
```

```
UCS-A /security/ldap/auth-server-group/server-ref* # up
UCS-A /security/ldap/auth-server-group* # create server-ref ldap2
UCS-A /security/ldap/auth-server-group/server-ref* # set order 2
UCS-A /security/ldap/auth-server-group/server-ref* # commit-buffer
UCS-A /security/ldap/auth-server-group/server-ref #
```

What to do next

Configure an authentication domain or select a default authentication service.

Deleting an LDAP Provider Group

Before you begin

Remove the provider group from an authentication configuration.

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 # **delete auth-server-group** *auth-server-group-name*	Deletes the LDAP provider group.
Step 4	UCS-A /security/ldap # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes an LDAP provider group called ldapgroup and commits the transaction:

```
UCS-A# scope security
UCS-A /security # scope ldap
UCS-A /security/ldap # delete auth-server-group ldapgroup
UCS-A /security/ldap* # commit-buffer
UCS-A /security/ldap #
```

Creating a RADIUS Provider Group

Creating a RADIUS provider group allows you to authenticate using multiple RADIUS databases.

Before you begin

Create one or more RADIUS providers.

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 # **create auth-server-group** *auth-server-group-name*	Creates a RADIUS provider group and enters authentication server group security RADIUS mode.
Step 4	UCS-A /security/RADIUS/auth-server-group # **create server-ref** *radius-provider-name*	Adds the specified RADIUS provider to the RADIUS provider group and enters server reference authentication server group security RADIUS mode.
Step 5	UCS-A /security/radius/auth-server-group/server-ref # **set order** *order-num*	Specifies the order in which Cisco UCS uses this provider to authenticate users. Valid values include no-value and 0-16, with the lowest value indicating the highest priority. Setting the order to no-value is equivalent to giving that server reference the highest priority.
Step 6	UCS-A /security/radius/auth-server-group/server-ref # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example creates a RADIUS provider group called radiusgroup, adds two previously configured providers called radius1 and radius2 to the provider group, sets the order, and commits the transaction:

```
UCS-A# scope security
UCS-A /security # scope radius
UCS-A /security/radius # create auth-server-group radiusgroup
UCS-A /security/radius/auth-server-group* # create server-ref radius1
UCS-A /security/radius/auth-server-group/server-ref* # set order 1
UCS-A /security/radius/auth-server-group/server-ref* # up
UCS-A /security/radius/auth-server-group* # create server-ref radius2
UCS-A /security/radius/auth-server-group/server-ref* # set order 2
UCS-A /security/radius/auth-server-group/server-ref* # commit-buffer
UCS-A /security/radius/auth-server-group/server-ref #
```

What to do next

Configure an authentication domain or select a default authentication service.

Deleting a RADIUS Provider Group

Remove the provider group from an authentication configuration.

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 # **delete auth-server-group** *auth-server-group-name*	Deletes the RADIUS provider group.
Step 4	UCS-A /security/radius # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes a RADIUS provider group called radiusgroup and commits the transaction:

```
UCS-A# scope security
UCS-A /security # scope radius
UCS-A /security/radius # delete auth-server-group radiusgroup
UCS-A /security/radius* # commit-buffer
UCS-A /security/radius #
```

Creating a TACACS Provider Group

Creating a TACACS+ provider group allows you to authenticate using multiple TACACS+ databases.

Before you begin

Create 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 # **create auth-server-group** *auth-server-group-name*	Creates a TACACS provider group and enters authentication server group security TACACS mode.
Step 4	UCS-A /security/tacacs/auth-server-group # **create server-ref** *tacacs-provider-name*	Adds the specified TACACS provider to the TACACS provider group and enters server reference authentication server group security TACACS mode.

	Command or Action	Purpose
Step 5	UCS-A /security/tacacs/auth-server-group/server-ref # **set order** *order-num*	Specifies the order in which Cisco UCS uses this provider to authenticate users. Valid values include no-value and 0-16, with the lowest value indicating the highest priority. Setting the order to no-value is equivalent to giving that server reference the highest priority.
Step 6	UCS-A /security/tacacs/auth-server-group/server-ref # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example creates a TACACS provider group called tacacsgroup, adds two previously configured providers called tacacs1 and tacacs2 to the provider group, sets the order, and commits the transaction:

```
UCS-A# scope security
UCS-A /security # scope tacacs
UCS-A /security/tacacs # create auth-server-group tacacsgroup
UCS-A /security/tacacs/auth-server-group* # create server-ref tacacs1
UCS-A /security/tacacs/auth-server-group/server-ref* # set order 1
UCS-A /security/tacacs/auth-server-group/server-ref* # up
UCS-A /security/tacacs/auth-server-group* # create server-ref tacacs2
UCS-A /security/tacacs/auth-server-group/server-ref* # set order 2
UCS-A /security/tacacs/auth-server-group/server-ref* # commit-buffer
UCS-A /security/tacacs/auth-server-group/server-ref #
```

What to do next

Configure an authentication domain or select a default authentication service.

Deleting a TACACS Provider Group

Remove the provider group from an authentication configuration.

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 # **delete auth-server-group** *auth-server-group-name*	Deletes the TACACS provider group.
Step 4	UCS-A /security/tacacs # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes a TACACS provider group called tacacsgroup and commits the transaction:

```
UCS-A# scope security
UCS-A /security # scope tacacs
UCS-A /security/tacacs # delete auth-server-group tacacsgroup
UCS-A /security/tacacs* # commit-buffer
UCS-A /security/tacacs #
```

Authentication Domains

The Cisco UCS Manager uses Authentication Domains to leverage multiple authentication systems. You can specify and configure each authentication domain during login; otherwise, Cisco UCS Manager uses the default authentication service configuration.

You can create up to eight authentication domains. Each authentication domain is associated with a provider group and a realm in the Cisco UCS Manager. The Cisco UCS Manager uses all servers within the realm if you do not specify a provider group.

Creating an Authentication Domain

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **create auth-domain** *domain-name*	Creates an authentication domain and enters authentication domain mode. **Note** For systems using the remote authentication protocol, the authentication domain name is considered part of the username and counts toward the 32-character limit for locally created usernames. Because Cisco UCS inserts 5 characters for formatting, authentication fails if the domain name and username combined characters total exceeds 27.
Step 3	(Optional) UCS-A /security/auth-domain # **set refresh-period** *seconds*	When a web client connects to Cisco UCS Manager, the client must send refresh requests to Cisco UCS Manager to keep the web session active. This option specifies the maximum amount of time allowed between refresh requests for a user in this domain. If this time limit is exceeded, Cisco UCS Manager considers the web session inactive, but it does not terminate the session.

	Command or Action	Purpose
		Specify an integer between 60 and 172800. The default is 600 seconds when Two-Factor Authentication is not enabled and 7200 seconds when it is enabled. **Note** The number of seconds set for the **Web Session Refresh Period** must be less than the number of seconds set for the **Web Session Timeout**. Do not set the **Web Session Refresh Period** to the same value as the **Web Session Timeout**.
Step 4	(Optional) UCS-A /security/auth-domain # **set session-timeout** *seconds*	The maximum amount of time that can elapse after the last cookie/token refresh request has failed before Cisco UCS Manager considers a web session as inactive. If this time limit is exceeded, Cisco UCS Manager automatically terminates the web session. Specify an integer between 300 and 172800. The default is 7200 seconds when Two-Factor Authentication is not enabled and 8000 seconds when it is enabled. **Note** If you set two-factor authentication for a RADIUS or TACACS+ realm, consider increasing the **session-refresh** and **session-timeout** periods so that remote users will not have to re-authenticate too frequently.
Step 5	(Optional) UCS-A /security/auth-domain # **create default-auth**	Creates a default authentication for the authentication domain.
Step 6	(Optional) UCS-A /security/auth-domain/default-auth # **set auth-server-group** *auth-serv-group-name*	Sets the provider group for the authentication domain.
Step 7	UCS-A /security/auth-domain/default-auth # **set realm** {**ldap** | **local** | **radius** | **tacacs**}	Sets the realm for the authentication domain.
Step 8	(Optional) UCS-A /security/auth-domain/default-auth # **set use-2-factor yes**	Sets the authentication method to two-factor authentication for the realm. **Note** Two-factor authentication applies only to the RADIUS and TACACS+ realms.
Step 9	UCS-A /security/auth-domain/default-auth # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example creates an authentication domain called domain1 with a web refresh period of 3600 seconds (1 hour) and a session timeout period of 14400 seconds (4 hours). It then configures domain1 to use the providers in radius1, sets the realm type to radius, enables two-factor authentication, and commits the transaction:

```
UCS-A# scope security
UCS-A /security # create auth-domain domain1
UCS-A /security/auth-domain* # set refresh-period 3600
UCS-A /security/auth-domain* # set session-timeout 14400
UCS-A /security/auth-domain* # create default-auth
UCS-A /security/auth-domain/auth-domain* # set auth-server-group radius1
UCS-A /security/auth-domain/auth-domain* # set realm radius
UCS-A /security/auth-domain/auth-domain* # set user-2-factor yes
UCS-A /security/auth-domain/auth-domain* # commit-buffer
UCS-A /security/auth-domain/auth-domain #
```

Primary Authentication Service

Selecting the Console Authentication Service

Before you begin

If the system uses a remote authentication service, create a provider for that authentication service. If the system uses only local authentication through Cisco UCS, you do not need to create a provider first.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **scope console-auth**	Enters console authorization security mode.
Step 3	UCS-A /security/console-auth # **set realm** *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 • **none** —Allows local users to log on without specifying a password • **radius** —Specifies RADIUS authentication • **tacacs** —Specifies TACACS+ authentication

	Command or Action	Purpose
Step 4	(Optional) UCS-A /security/console-auth # **set auth-server-group** *auth-serv-group-name*	The associated provider group, if any.
Step 5	(Optional) UCS-A /security/default-auth # **set use-2-factor yes**	Sets the authentication method to two-factor authentication for the realm. **Note** Two-factor authentication applies only to the RADIUS and TACACS+ realms.
Step 6	UCS-A /security/console-auth # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example sets the authentication realm to TACACS+, sets the console authentication provider group to provider1, enables two-factor authentication, and commits the transaction:

```
UCS-A# scope security
UCS-A /security # scope console-auth
UCS-A /security/console-auth # set realm tacacs
UCS-A /security/console-auth # set auth-server-group provider1
UCS-A /security/console-auth* # set use-2-factor yes
UCS-A /security/console-auth* # commit-buffer
UCS-A /security/console-auth #
```

Selecting the Default Authentication Service

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **scope default-auth**	Enters default authorization security mode.
Step 3	UCS-A /security/default-auth # **set realm** *auth-type*	Specifies the default authentication, where *auth-type* is one of the following keywords: • **ldap**—Specifies LDAP authentication • **local**—Specifies local authentication • **none**—Allows local users to log on without specifying a password • **radius**—Specifies RADIUS authentication • **tacacs**—Specifies TACACS+ authentication

	Command or Action	Purpose
Step 4	(Optional) UCS-A /security/default-auth # **set auth-server-group** *auth-serv-group-name*	The associated provider group, if any.
Step 5	(Optional) UCS-A /security/default-auth # **set refresh-period** *seconds*	When a web client connects to Cisco UCS Manager, the client must send refresh requests to Cisco UCS Manager to keep the web session active. This option specifies the maximum amount of time allowed between refresh requests for a user in this domain. If this time limit is exceeded, Cisco UCS Manager considers the web session inactive, but it does not terminate the session. Specify an integer between 60 and 172800. The default is 600 seconds when Two-Factor Authentication is not enabled and 7200 seconds when it is enabled.
Step 6	(Optional) UCS-A /security/default-auth # **set session-timeout** *seconds*	The maximum amount of time that can elapse after the last cookie/token refresh request has failed before Cisco UCS Manager considers a web session as inactive. If this time limit is exceeded, Cisco UCS Manager automatically terminates the web session. Specify an integer between 300 and 172800. The default is 7200 seconds when Two-Factor Authentication is not enabled and 8000 seconds when it is enabled. **Note** If you set two-factor authentication for a RADIUS or TACACS+ realm, consider increasing the **session-refresh** and **session-timeout** periods so that remote users will not have to re-authenticate too frequently.
Step 7	(Optional) UCS-A /security/default-auth # **set use-2-factor yes**	Sets the authentication method to two-factor authentication for the realm. **Note** Two-factor authentication applies only to the RADIUS and TACACS+ realms.
Step 8	UCS-A /security/default-auth # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example sets the default authentication to RADIUS, the default authentication provider group to provider1, enables two-factor authentications, sets the refresh period to 7200 seconds (2 hours), the session timeout period to 28800 seconds (8 hours), and enables two-factor authentication. It then commits the transaction.

```
UCS-A# scope security
UCS-A /security # scope default-auth
UCS-A /security/default-auth # set realm radius
UCS-A /security/default-auth* # set auth-server-group provider1
UCS-A /security/default-auth* # set use-2-factor yes
UCS-A /security/default-auth* # set refresh-period 7200
UCS-A /security/default-auth* # set session-timeout 28800
UCS-A /security/default-auth* # commit-buffer
UCS-A /security/default-auth #
```

Role Policy for Remote Users

By default, if user roles are not configured in Cisco UCS Manager read-only access is granted to all users logging in to Cisco UCS Manager from a remote server using the LDAP, RADIUS, or TACACS protocols. For security reasons, it might be desirable to restrict access to those users matching an established user role in Cisco UCS Manager.

You can configure the role policy for remote users in the following ways:

assign-default-role

Does not restrict user access to Cisco UCS Manager based on user roles. Read-only access is granted to all users unless other user roles have been defined in Cisco UCS Manager.

no-login

Restricts user access to Cisco UCS Manager based on user roles. If user roles have not been assigned for the remote authentication system, access is denied.

Configuring the Role Policy for Remote Users

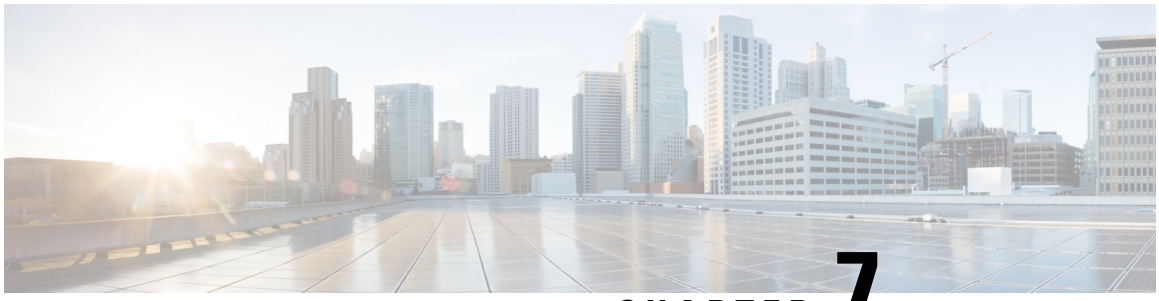
Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **set remote-user default-role** {**assign-default-role** | **no-login**}	Specifies whether user access to Cisco UCS Manager is restricted based on user roles.
Step 3	UCS-A /security # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example sets the role policy for remote users and commits the transaction:

```
UCS-A# scope security
UCS-A /security # set remote-user default-role assign-default-role
UCS-A /security* # commit-buffer
UCS-A /security #
```



CHAPTER 7

How to Enable and Disable the Call Home Feature

Call Home in UCS Overview

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

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

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

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

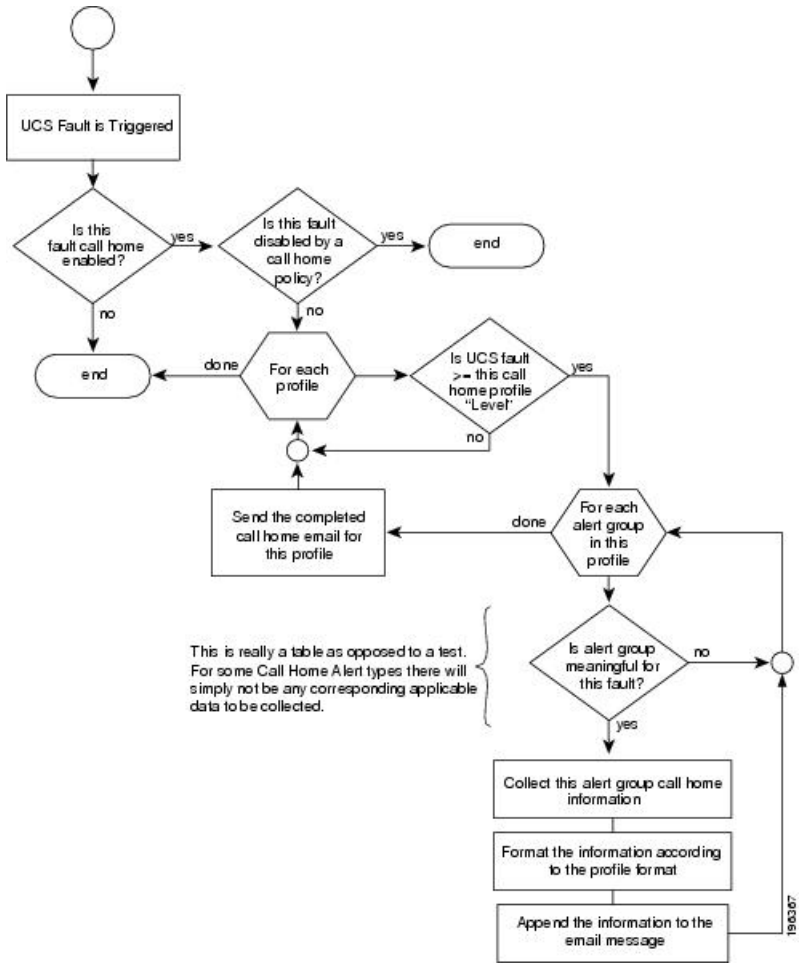
Cisco UCS delivers Call Home messages in the following formats:

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

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

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

Figure 1: Flow of Events after a Fault is Triggered



SMTP Authentication

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

You can toggle **SMTP Authentication** between

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

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

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

Enabling Call Home

Procedure

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

Example

The following example enables Call Home and commits the transaction:

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

What to do next

For more information on the Call Home feature, see the *Cisco UCS System Monitoring Guide*.

Disabling Call Home

Procedure

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

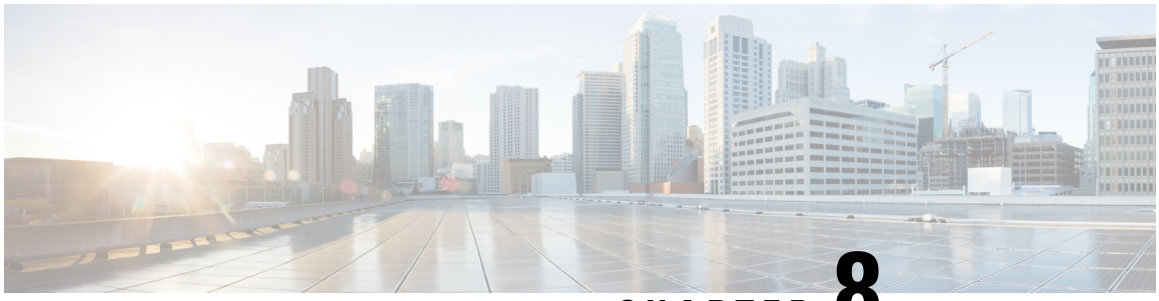
Example

The following example disables Call Home and commits the transaction:

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

What to do next

For more information on the Call Home feature, see the *Cisco UCS System Monitoring Guide*.



CHAPTER 8

UCS Manager Communication Services

Communication Services

You can use the communication services defined below to interface third-party applications with Cisco UCS.

Cisco UCS Manager supports IPv4 and IPv6 address access for the following services:

- CIM XML
- HTTP
- HTTPS
- SNMP
- SSH
- Telnet

Cisco UCS Manager supports out-of-band IPv4 address access to the **Cisco UCS KVM Direct** launch page from a web browser. To provide this access, you must enable the following service:

- CIMC Web Service

Communication Service	Description
CIM XML	The Common Information Model (CIM) XML) service is disabled by default and is only available in read-only mode. The default port is 5988. The CIM XML is a standards-based protocol for exchanging CIM information that the Distributed Management Task Force defines.

Communication Service	Description
CIMC Web Service	This service is disabled by default. When this service is enabled, users can directly access a server CIMC using one of the out-of-band management IP addresses assigned directly to the server, or associated with the server through a service profile. **Note** CIMC Web Service can only be enabled or disabled globally. You cannot configure KVM direct access for individual CIMC IP addresses.
HTTP	By default, HTTP is enabled on port 80. You can run the Cisco UCS Manager GUI in an HTTP or HTTPS browser. If you select HTTP, all data is exchanged in clear text mode. For a secure browser session, we recommend that you enable HTTPS and disable HTTP. By default, Cisco UCS implements a browser redirects to an HTTPS equivalent and recommends that you do not change this behavior. **Note** If you are upgrading to Cisco UCS, version 1.4(1), the browser redirect to a secure browser does not occur by default. To redirect the HTTP browser to an HTTPS equivalent, enable the **Redirect HTTP to HTTPS** in Cisco UCS Manager.
HTTPS	By default, HTTPS is enabled on port. With HTTPS, all data is exchanged in encrypted mode through a secure server. For a secure browser session, We recommend that you only use HTTPS and either disable or redirect HTTP communications.
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 that the Distributed Management Task Force defines.
SNMP	By default, this service is disabled. If enabled, the default port is 161. You must configure the community and at least one SNMP trap. Enable this service only if your system includes integration with an SNMP server.
SSH	This service is enabled on port 22. You cannot disable it, and you cannot change the default port. This service provides access to the Cisco UCS Manager CLI.
Telnet	By default, this service is disabled. This service provides access to the Cisco UCS Manager CLI.

NonSecure Communication Services

Setting Web Session Limits

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system** .	Enters system mode.
Step 2	UCS-A /system # **scope services** .	Enters services mode.
Step 3	UCS-A /system/services # **scope web-session-limits** .	Enters webs session limits mode.
Step 4	UCS-A /system/services/web-session-limits # **set** {**maximum-event-interval** |**per-user** | **total**}*number* .	Enables you to set the following web session limits:
Step 5	UCS-A /system/services/web-session-limits # **commit-buffer** .	Commits the transaction to the system configuration.

Name	Description
Maximum Sessions Per User	The maximum number of concurrent HTTP and HTTPS sessions allowed for each user. Enter an integer between 1 and 256.
Maximum Sessions	The maximum number of concurrent HTTP and HTTPS sessions allowed for all users within the system. Enter an integer between 1 and 256.
Maximum Event Interval (in seconds)	The maximum time interval between two events. Tracks various types of event change notifications, such as responses to any user requests from the UI. If the interval expires, the UI session is terminated. Enter and integer between 120-3600

Example

The following example shows how to set the maximum-event-interval:

```
UCS-A# scope system
UCS-A /system # scope services
UCS-A /system/services # scope web-session-limits
UCS-A /system/services/web-session-limits # set maximum-event-interval 300
UCS-A /system/services/web-session-limits # commit buffer
```

Viewing Web Session Limits

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system**	Enters system mode.
Step 2	UCS-A /system # **scope services**	Enters services mode.
Step 3	/system/services # **show web-session-limits**	Shows the web session settings.

Example

The following example shows how to view web session limits:

```
UCS-A# scope system
UCS-A /system # scope services
UCS-A /system/services # show web-session-limits
Web Sessions:
    Maximum logins for single user Maximum Sessions Maximum Event Interval (sec)
    ------------------------------ ---------------- ----------------------------
    32                             256              600
UCS-A /system/services #
```

Setting Shell Session Limits

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system** .	Enters system mode.
Step 2	UCS-A /system # **scope services** .	Enters services mode.
Step 3	UCS-A /system/services # **scope shell-session-limits**	
Step 4	UCS-A /system/services/shell-session-limits # **set** {**per-user** |**total**}*number*.	Enables you to set the following shell session limits:

	Command or Action	Purpose	
		Name	**Description**
		Maximum Sessions Per User	The maximum number of concurrent shell sessions allowed per user. Enter an integer between 1-32.
		Maximum Sessions	The maximum number of concurrent shell sessions allowed for all users within the system. Enter an integer between 1-32.
Step 5	UCS-A /system/services/shell-session-limits # **commit-buffer** .	Commits the transaction to the system configuration.	

Example

The following example shows how to set the maximum sessions:

```
UCS-A# scope system
UCS-A /system # scope services
UCS-A /system/services # scope shell-session-limits
UCS-A /system/services/shell-session-limits # set maximum-sessions 20
UCS-A /system/services/shell-session-limits # commit buffer
```

Viewing Shell Session Limits

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system**	Enters system mode.
Step 2	UCS-A /system # **scope services**	Enters services mode.
Step 3	/system/services # **show shell-session-limits**	Shows the shell session settings.

Example

The following example shows how to view shell session limits:

```
UCS-A# scope system
UCS-A /system # scope services
UCS-A /system/services # show shell-session-limits
Shell Sessions:
    Maximum logins for single user Maximum Sessions
    ------------------------------ ----------------
    32                             32
UCS-A /system/services #
```

Configuring CIM XML

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 for the CIM XML connection.
Step 5	UCS-A /system/services # **commit-buffer**	Commits the transaction to the system configuration.

Example

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

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.

	Command or Action	Purpose
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.

Example

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
Warning: When committed, this closes all the web sessions.
UCS-A /system/services* # commit-buffer
UCS-A /system/services #
```

Unconfiguring HTTP

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 # **disable http**	Disables the HTTP service.
Step 4	UCS-A /system/services # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example disables HTTP and commits the transaction:

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

Secure Communication Services

Configuring HTTPS

Caution After you complete the HTTPS configuration, including changing the port and key ring for the HTTPS to use, all current HTTP and HTTPS sessions are closed without warning as soon as you save or commit the transaction.

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.
Step 4	(Optional) UCS-A /system/services # **set https port** *port-num*	Specifies the port to be used for the HTTPS connection.
Step 5	(Optional) UCS-A /system/services # **set https keyring** *keyring-name*	Specifies the name of the key ring you created for HTTPS.
Step 6	(Optional) UCS-A /system/services # **set https cipher-suite-mode** *cipher-suite-mode*	The level of Cipher Suite security used by the Cisco UCS domain. *cipher-suite-mode* can be one of the following keywords: • **high-strength** • **medium-strength** • **low-strength** • **custom**—Allows you to specify a user-defined Cipher Suite specification string.
Step 7	(Optional) UCS-A /system/services # **set https cipher-suite** *cipher-suite-spec-string*	Specifies a custom level of Cipher Suite security for this Cisco UCS domain if **cipher-suite-mode** is set to **custom**. *cipher-suite-spec-string* can contain up to 256 characters and must conform to the OpenSSL Cipher Suite specifications. You cannot use any spaces or special characters except ! (exclamation point), + (plus sign), - (hyphen), and : (colon). For details, see http://httpd.apache.org/docs/2.0/mod/mod_ssl.html#sslciphersuite.

	Command or Action	Purpose
		For example, the medium strength specification string Cisco UCS Manager uses as the default is: ALL:!ADH:!EXPORT56:!LOW:RC4+RSA:+HIGH:+MEDIUM:+EXP:+eNULL **Note** This option is ignored if **cipher-suite-mode** is set to anything other than **custom**.
Step 8	(Optional) UCS-A /system/services # **set https ssl-protocol** *default|tls1-2|tls1-3*	Enables you to choose which SSL protocols can be used. This can be one of the following: • **Default (Allow all except SSLv2 and SSLv3)** • **Only TLSv1.2** **Note** If you choose Only TLSv1.2, all web client connections trying to use less secure versions of TLS are blocked. • **Only TLSv1.3** **Note** If you choose Only TLSv1.3, all web client connections trying to use less secure versions of TLS are blocked.
Step 9	UCS-A /system/services # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example enables HTTPS, sets the port number to 443, sets the key ring name to kring7984, sets the Cipher Suite security level to high, configures the web server to accept only connections using TLSv1.2, 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
Warning: When committed, this closes all the web sessions.
UCS-A /system/services* # set https keyring kring7984
UCS-A /system/services* # set https cipher-suite-mode high
UCS-A /system/services* # set https ssl-protocol tls1-2
UCS-A /system/services* # commit-buffer
UCS-A /system/services #
```

Unconfiguring HTTPS

Before you begin

Disable HTTP to HTTPS redirection.

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 # **disable https**	Disables the HTTPS service.
Step 4	UCS-A /system/services # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example disables HTTPS and commits the transaction:

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

Certificates, Key Rings, and Trusted Points

HTTPS uses components of the Public Key Infrastructure (PKI) to establish secure communications between two devices, such as a client's browser and Cisco UCS Manager.

Encryption Keys and Key Rings

Each PKI device holds a pair of asymmetric Rivest-Shamir-Adleman (RSA) encryption keys, one kept private and one made public, stored in an internal key ring. A message encrypted with either key can be decrypted with the other key. To send an encrypted message, the sender encrypts the message with the receiver's public key, and the receiver decrypts the message using its own private key. A sender can also prove its ownership of a public key by encrypting (also called 'signing') a known message with its own private key. If a receiver can successfully decrypt the message using the public key in question, the sender's possession of the corresponding private key is proven. Encryption keys can vary in length, with typical lengths from 512 bits to 2048 bits. In general, a longer key is more secure than a shorter key. Cisco UCS Manager provides a default key ring with an initial 1024-bit key pair, and allows you to create additional key rings.

The default key ring certificate must be manually regenerated if the cluster name changes or the certificate expires.

This operation is only available in the UCS Manager CLI.

Certificates

To prepare for secure communications, two devices first exchange their digital certificates. A certificate is a file containing a device's public key along with signed information about the device's identity. To merely support encrypted communications, a device can generate its own key pair and its own self-signed certificate. When a remote user connects to a device that presents a self-signed certificate, the user has no easy method to verify the identity of the device, and the user's browser will initially display an authentication warning. By default, Cisco UCS Manager contains a built-in self-signed certificate containing the public key from the default key ring.

You can change the self-signed KVM certificate on CIMC for Cisco UCS servers to a user-generated public certificate. However, a password protected X.509 certificate private key is not supported. Changing the KVM Certificate, on page 121 Creating a KVM Certificate, on page 120 provides detailed information about this process.

Important The certificate must be in Base64 encoded X.509 (CER) format.

Trusted Points

To provide stronger authentication for Cisco UCS Manager, you can obtain and install a third-party certificate from a trusted source, or trusted point, that affirms the identity of your device. The third-party certificate is signed by the issuing trusted point, which can be a root certificate authority (CA) or an intermediate CA or trust anchor that is part of a trust chain that leads to a root CA. To obtain a new certificate, you must generate a certificate request through Cisco UCS Manager and submit the request to a trusted point.

Related Topics

Changing the KVM Certificate, on page 121

Creating an Untrusted CA-Signed Certificate

As an alternative to using a public Certificate Authority (CA) to generate and sign a certificate, you can operate your own CA and sign your own certificates. To generate the certificate-key pair, you must generate a 2048 bit RSA key and an x.509 PEM certificate. This section shows commands for creating a CA and generating a certificate using the OpenSSL certificate server running on Linux. For detailed information about OpenSSL, see http://www.openssl.org.

Note These commands are to be entered on a Linux server with the OpenSSL package.

Before you begin

Obtain and install a certificate server software package on a server within your organization.

Procedure

	Command or Action	Purpose
Step 1	**openssl genrsa -out** *CA_keyfilename keysize* **Example:** `# openssl genrsa -out cert.private 2048`	This command generates an RSA private key that will be used by the CA. The specified file name contains an RSA key of the specified key size.
Step 2	**openssl req -new -x509 -days** *numdays* **-key** *CA_keyfilename* **-out** *CA_certfilename* **Example:** `# openssl req -new -x509 -days 365 -key cert.private -out cert.pem`	This command generates a new self-signed certificate for the CA using the specified key. The certificate is valid for the specified period. The command prompts the user for additional certificate information. The certificate server is an active CA.
Step 3	(Optional) **openssl x509 -req -days** *numdays* **-in** *CSR_filename* **-CA** *CA_certfilename* **-set_serial 04 -CAkey** *CA_keyfilename* **-out** *server_certfilename* **-extfile openssl.conf** **Example:** `# openssl x509 -req -days 365 -in csr.txt -CA cert.pem -set_serial 04 -CAkey cert.private -out myserver05.crt -extfile openssl.conf`	This command directs the CA to use your CSR file to generate a server certificate. Your server certificate is contained in the output file.

Example

This example shows how to create a CA and to generate a server certificate signed by the new CA. These commands are entered on a Linux server running OpenSSL.

```
# /usr/bin/openssl genrsa -out cert.private 2048
Generating RSA private key, 2048 bit long modulus
.............++++++
.....++++++
e is 65537 (0x10001)
# /usr/bin/openssl req -new -x509 -days 365 -key cert.private -out cert.pem
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [GB]:US
State or Province Name (full name) [Berkshire]:California
Locality Name (eg, city) [Newbury]:San Jose
Organization Name (eg, company) [My Company Ltd]:Example Incorporated
Organizational Unit Name (eg, section) []:Unit A
Common Name (eg, your name or your server's hostname) []:example.com
Email Address []:admin@example.com
# /usr/bin/openssl x509 -req -days 365 -in csr.txt -CA cert.pem -set_serial 01
```

```
-CAkey cert.private -out server.crt -extfile openssl.conf
Signature ok
subject=/C=US/ST=California/L=San Jose/O=Example Inc./OU=Unit
A/CN=example.com/emailAddress=john@example.com
Getting CA Private Key
#
```

Creating a Key Ring

Cisco UCS Manager supports a maximum of 8 key rings, including the default key ring.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **create keyring** *keyring-name*	Creates and names the key ring.
Step 3	UCS-A /security/keyring # **set modulus** {**mod2048** | **mod2560** | **mod3072** | **mod3584** | **mod4096**}	Sets the SSL key length in bits.
Step 4	UCS-A /security/keyring # **commit-buffer**	Commits the transaction.

Example

The following example creates a keyring with a key size of 1024 bits:

```
UCS-A# scope security
UCS-A /security # create keyring kr220
UCS-A /security/keyring* # set modulus 'modvalue'
UCS-A /security/keyring* # commit-buffer
UCS-A /security/keyring #
```

What to do next

Create a certificate request for this key ring.

Regenerating the Default Key Ring

The default key ring certificate must be manually regenerated if the cluster name changes or the certificate expires.

Procedure

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

	Command or Action	Purpose
Step 2	UCS-A /security # **scope keyring default**	Enters key ring security mode for the default key ring.
Step 3	UCS-A /security/keyring # **set regenerate yes**	Regenerates the default key ring.
Step 4	UCS-A /security/keyring # **commit-buffer**	Commits the transaction.

Example

The following example regenerates the default key ring:

```
UCS-A# scope security
UCS-A /security # scope keyring default
UCS-A /security/keyring* # set regenerate yes
UCS-A /security/keyring* # commit-buffer
UCS-A /security/keyring #
```

Creating a Certificate Request for a Key Ring with Basic Options

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **scope keyring** *keyring-name*	Enters configuration mode for the key ring.
Step 3	UCS-A /security/keyring # **create certreq** {**ip** [*ipv4-addr* | *ipv6-v6*] |**subject-name** *name*}	Creates a certificate request using the IPv4 or IPv6 address specified, or the name of the fabric interconnect. You are prompted to enter a password for the certificate request.
Step 4	UCS-A /security/keyring/certreq # **commit-buffer**	Commits the transaction.
Step 5	UCS-A /security/keyring # **show certreq**	Displays the certificate request, which you can copy and send to a trust anchor or certificate authority.

Example

The following example creates and displays a certificate request with an IPv4 address for a key ring, with basic options:

```
UCS-A# scope security
UCS-A /security # scope keyring kr220
UCS-A /security/keyring # create certreq ip 192.168.200.123 subject-name sjc04
Certificate request password:
Confirm certificate request password:
```

```
UCS-A /security/keyring* # commit-buffer
UCS-A /security/keyring # show certreq
Certificate request subject name: sjc04
Certificate request ip address: 192.168.200.123
Certificate request e-mail name:
Certificate request country name:
State, province or county (full name):
Locality (eg, city):
Organization name (eg, company):
Organization Unit name (eg, section):
Request:
-----BEGIN CERTIFICATE REQUEST-----
MIIBfTCB5wIBADARMQ8wDQYDVQQDEwZzYW1jMDQwgZ8wDQYJKoZIhvcNAQEBBQAD
gY0AMIGJAoGBALpKn1t8qMZO4UGqILKFXQQc2c8b/vW2rnRF8OPhKbhghLA1YZ1F
JqcYEG5Yl1+vgohLBTd45s0GC8m4RTLJWHo4SwccAUXQ5Zngf45YtX1WsylwUWV4
0re/zgTk/WCd56RfOBvWR2Dtztu2pGA14sd761zLxt29K7R8mzj6CAUVAgMBAAGg
LTArBgkqhkiG9w0BCQ4xHjAcMBoGA1UdEQEB/wQQMA6CBnNhbWMwNIcECsEiXjAN
BgkqhkiG9w0BAQQFAAOBgQCsxN0qUHYGFoQw56RwQueLTNPnrndqUwuZHUO03Teg
nhsyu4satpyiPqVV9viKZ+spvc6x5PWIcTWgHhH8BimOb/0OKuG8kwfIGGsEDlAv
TTYvUP+BZ9OFiPbRIA718S+V8ndXr1HejiQGxlDNqoN+odCXPc5kjoXD0lZTL09H
BA==
-----END CERTIFICATE REQUEST-----

UCS-A /security/keyring #
```

Creating a Certificate Request for a Key Ring with Advanced Options

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **scope keyring** *keyring-name*	Enters configuration mode for the key ring.
Step 3	UCS-A /security/keyring # **create certreq**	Creates a certificate request.
Step 4	UCS-A /security/keyring/certreq* # **set country** *country name*	Specifies the country code of the country in which the company resides.
Step 5	UCS-A /security/keyring/certreq* # **set dns** *DNS Name*	Specifies the Domain Name Server (DNS) address associated with the request. A maximum of three comma separated domain names can be entered in this field. For example, you can enter www.example1.com,www.example2.com,www.example3.com
Step 6	UCS-A /security/keyring/certreq* # **set e-mail** *E-mail name*	Specifies the email address associated with the certificate request.
Step 7	UCS-A /security/keyring/certreq* # **set ip** *certificate request ip-address* | **ipv6** *certificate request ipv6-address*	Specifies the IPv4 or IPv6 address of the Cisco UCS domain.

	Command or Action	Purpose
Step 8	UCS-A /security/keyring/certreq* # **set fi-a-ip** *certificate request FI A ip-address* | **fi-a-ipv6** *certificate request FI A ipv6-address*	The IPv4 or IPv6 address of fabric interconnect A.
Step 9	UCS-A /security/keyring/certreq* # **set fi-b-ip** *certificate request FI B ip-address* | **fi-b-ipv6** *certificate request FI B ipv6-address*	The IPv4 or IPv6 address of fabric interconnect B.
Step 10	UCS-A /security/keyring/certreq* # **set locality** *locality name (eg, city)*	Specifies the city or town in which the company requesting the certificate is headquartered.
Step 11	UCS-A /security/keyring/certreq* # **set org-name** *organization name*	Specifies the organization requesting the certificate.
Step 12	UCS-A /security/keyring/certreq* # **set org-unit-name** *organizational unit name*	Specifies the organizational unit.
Step 13	UCS-A /security/keyring/certreq* # **set password** *certificate request password*	Specifies an optional password for the certificate request.
Step 14	UCS-A /security/keyring/certreq* # **set state** *state, province or county*	Specifies the state or province in which the company requesting the certificate is headquartered.
Step 15	UCS-A /security/keyring/certreq* # **set subject-name** *certificate request name*	Specifies the fully qualified domain name of the Fabric Interconnect. **Note** Ensure the **subject-name** is not same as the **dns** name.
Step 16	UCS-A /security/keyring/certreq* # **commit-buffer**	Commits the transaction.
Step 17	UCS-A /security/keyring # **show certreq**	Displays the certificate request, which you can copy and send to a trust anchor or certificate authority.

Example

The following example creates and displays a certificate request with an IPv4 address for a key ring, with advanced options:

```
UCS-A# scope security
UCS-A /security # scope keyring kr220
UCS-A /security/keyring # create certreq
UCS-A /security/keyring/certreq* # set ip 192.168.200.123
UCS-A /security/keyring/certreq* # set fi-a-ip 192.168.200.124
UCS-A /security/keyring/certreq* # set fi-b-ip 192.168.200.125
UCS-A /security/keyring/certreq* # set subject-name sjc04
UCS-A /security/keyring/certreq* # set country US
UCS-A /security/keyring/certreq* # set dns bg1-samc-15A,bg2-samc-15A,bg3-samc-15A
```

```
UCS-A /security/keyring/certreq* # set e-mail test@cisco.com
UCS-A /security/keyring/certreq* # set locality new york city
UCS-A /security/keyring/certreq* # set org-name "Cisco Systems"
UCS-A /security/keyring/certreq* # set org-unit-name Testing
UCS-A /security/keyring/certreq* # set state new york
UCS-A /security/keyring/certreq* # commit-buffer
UCS-A /security/keyring/certreq # show certreq
Certificate request subject name: sjc04
Certificate request ip address: 192.168.200.123
Certificate request FI A ip address: 192.168.200.124
Certificate request FI B ip address: 192.168.200.125
Certificate request e-mail name: test@cisco.com
Certificate request ipv6 address: ::
Certificate request FI A ipv6 address: ::
Certificate request FI B ipv6 address: ::
Certificate request country name: US
State, province or county (full name): New York
Locality name (eg, city): new york city
Organization name (eg, company): Cisco
Organization Unit name (eg, section): Testing
Request:
-----BEGIN CERTIFICATE REQUEST-----
MIIBfTCB5wIBADARMQ8wDQYDVQQDEwZzYW1jMDQwgZ8wDQYJKoZIhvcNAQEBBQAD
gY0AMIGJAoGBALpKn1t8qMZO4UGqILKFXQQc2c8b/vW2rnRF8OPhKbhghLA1YZ1F
JqcYEG5Yl1+vgohLBTd45s0GC8m4RTLJWHo4SwccAUXQ5Zngf45YtX1WsylwUWV4
0re/zgTk/WCd56RfOBvWR2Dtztu2pGA14sd761zLxt29K7R8mzj6CAUVAgMBAAGg
LTArBgkqhkiG9w0BCQ4xHjAcMBoGA1UdEQEB/wQQMA6CBnNhbWMwNIcECsEiXjAN
BgkqhkiG9w0BAQQFAAOBgQCsxN0qUHYGFoQw56RwQueLTNPnrndqUwuZHUO03Teg
nhsyu4satpyiPqVV9viKZ+spvc6x5PWIcTWgHhH8BimOb/0OKuG8kwfIGGsEDlAv
TTYvUP+BZ9OFiPbRIA718S+V8ndXr1HejiQGxlDNqoN+odCXPc5kjoXD0lZTL09H
BA==
-----END CERTIFICATE REQUEST-----

-----BEGIN CERTIFICATE REQUEST-----
MIIBfTCB5wIBADARMQ8wDQYDVQQDEwZzYW1jMDQwgZ8wDQYJKoZIhvcNAQEBBQAD
gY0AMIGJAoGBALpKn1t8qMZO4UGqILKFXQQc2c8b/vW2rnRF8OPhKbhghLA1YZ1F
JqcYEG5Yl1+vgohLBTd45s0GC8m4RTLJWHo4SwccAUXQ5Zngf45YtX1WsylwUWV4
0re/zgTk/WCd56RfOBvWR2Dtztu2pGA14sd761zLxt29K7R8mzj6CAUVAgMBAAGg
LTArBgkqhkiG9w0BCQ4xHjAcMBoGA1UdEQEB/wQQMA6CBnNhbWMwNIcECsEiXjAN
BgkqhkiG9w0BAQQFAAOBgQCsxN0qUHYGFoQw56RwQueLTNPnrndqUwuZHUO03Teg
nhsyu4satpyiPqVV9viKZ+spvc6x5PWIcTWgHhH8BimOb/0OKuG8kwfIGGsEDlAv
TTYvUP+BZ9OFiPbRIA718S+V8ndXr1HejiQGxlDNqoN+odCXPc5kjoXD0lZTL09H
BA==
-----END CERTIFICATE REQUEST-----

-----BEGIN CERTIFICATE REQUEST-----
MIIBfTCB5wIBADARMQ8wDQYDVQQDEwZzYW1jMDQwgZ8wDQYJKoZIhvcNAQEBBQAD
gY0AMIGJAoGBALpKn1t8qMZO4UGqILKFXQQc2c8b/vW2rnRF8OPhKbhghLA1YZ1F
JqcYEG5Yl1+vgohLBTd45s0GC8m4RTLJWHo4SwccAUXQ5Zngf45YtX1WsylwUWV4
0re/zgTk/WCd56RfOBvWR2Dtztu2pGA14sd761zLxt29K7R8mzj6CAUVAgMBAAGg
LTArBgkqhkiG9w0BCQ4xHjAcMBoGA1UdEQEB/wQQMA6CBnNhbWMwNIcECsEiXjAN
BgkqhkiG9w0BAQQFAAOBgQCsxN0qUHYGFoQw56RwQueLTNPnrndqUwuZHUO03Teg
nhsyu4satpyiPqVV9viKZ+spvc6x5PWIcTWgHhH8BimOb/0OKuG8kwfIGGsEDlAv
TTYvUP+BZ9OFiPbRIA718S+V8ndXr1HejiQGxlDNqoN+odCXPc5kjoXD0lZTL09H
BA==
-----END CERTIFICATE REQUEST-----
UCS-A /security/keyring/certreq #
```

What to do next

- Copy the text of the certificate request, including the BEGIN and END lines, and save it in a file. Send the file with the certificate request to a trust anchor or certificate authority to obtain a certificate for the key ring.
- Create a trusted point and set the certificate chain for the certificate of trust received from the trust anchor.

Creating a KVM Certificate

You can use this procedure to create the KVM certificate. This operation will result in a reboot of the CIMC.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *chassis-id / blade-id*	Enters chassis server mode for the specified server.
Step 2	UCS-A /chassis/server # **scope cimc**	Enters chassis server CIMC mode.
Step 3	UCS-A /chassis/server/cimc # **create kvm-certificate**	Creates the KVM certificate.
Step 4	UCS-A /chassis/server/cimc/kvm-certificate* # set certificate	Sets the specified user-generated public certificate.
Step 5	UCS-A /chassis/server/cimc/kvm-certificate* # set key	Sets the corresponding user-generated private key. **Note** Password protected X.509 certificate private key is not supported.
Step 6	UCS-A /chassis/server/cimc/kvm-certificate* # **commit-buffer**	Commits the transaction to the system configuration. This operation will result in a reboot of the CIMC.

Example

The following example creates a KVM certificate and commits the transaction:

```
UCS-A# scope server 1/3
UCS-A /chassis/server # scope cimc
UCS-A /chassis/server/cimc # create kvm-certificate
UCS-A /chassis/server/cimc/kvm-certificate* # set certificate
Enter lines one at a time. Enter ENDOFBUF to finish. Press ^C to abort.
Prompt Certificate:
>
...

UCS-A /chassis/server/cimc/kvm-certificate* # set key
```

```
Enter lines one at a time. Enter ENDOFBUF to finish. Press ^C to abort.
Prompt Key:
>
...

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

Clearing a KVM Certificate

This operation will result in a reboot of the CIMC.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *chassis-id / blade-id*	Enters chassis server mode for the specified server.
Step 2	UCS-A /chassis/server # **scope cimc**	Enters chassis server CIMC mode.
Step 3	UCS-A /chassis/server/cimc # **clear kvm-certificate**	Clears the KVM certificate.
Step 4	UCS-A /chassis/server/cimc* # **commit-buffer**	Commits the transaction to the system configuration. This operation will result in a reboot of the CIMC.

Example

The following example clears a KVM certificate and commits the transaction:

```
UCS-A# scope server 1/3
UCS-A /chassis/server # scope cimc
UCS-A /chassis/server/cimc # clear kvm-certificate
Warning: When committed, this operation will result in CIMC reboot.
UCS-A /chassis/server/cimc* # commit-buffer
UCS-A /chassis/server/cimc #
```

Changing the KVM Certificate

You can use this procedure to change the KVM certificate to a user-generated public certificate.

Procedure

Step 1 In the **Navigation** pane, click **Equipment**.

Step 2 Expand **Equipment** > **Chassis** > ***Chassis Number*** > **Servers**.

Step 3 Click the server for which you want to change the KVM certificate.

Step 4 In the **Work** pane, click the **Inventory** tab.

Step 5 Click the **CIMC** subtab.

Step 6 In the **Actions** area, click **Change KVM Certificate**:

Step 7 In the **Change KVM Certificate** dialog box, complete the following fields:

Field	Description
Certificate field	A user-generated public certificate.
Key field	The corresponding user-generated private key. **Note** Password protected X.509 certificate private key is not supported.

Step 8 Click **OK**.

Step 9 If a confirmation dialog box appears, click **Yes**.

This operation will result in a reboot of the CIMC

Creating a Trusted Point

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **create trustpoint** *name*	Creates and names a trusted point.
Step 3	UCS-A /security/trustpoint # **set certchain** [*certchain*]	Specifies certificate information for this trusted point. If you do not specify certificate information in the command, you are prompted to enter a certificate or a list of trustpoints defining a certification path to the root certificate authority (CA). On the next line following your input, type ENDOFBUF to finish. **Important** The certificate must be in Base64 encoded X.509 (CER) format.
Step 4	UCS-A /security/trustpoint # **commit-buffer**	Commits the transaction.

Example

The following example creates a trusted point and provides a certificate for the trusted point:

```
UCS-A# scope security
UCS-A /security # create trustpoint tPoint10
UCS-A /security/trustpoint* # set certchain
Enter lines one at a time. Enter ENDOFBUF to finish. Press ^C to abort.
Trustpoint Certificate Chain:
> -----BEGIN CERTIFICATE-----
> MIIDMDCCApmgAwIBAgIBADANBgkqhkiG9w0BAQQFADB0MQswCQYDVQQGEwJVUzEL
> BxMMU2FuIEpvc2UsIENBMRUwEwYDVQQKEwxFeGFtcGxlIEluYy4xEzARBgNVBAsT
> ClRlc3QgR3JvdXAxGTAXBgNVBAMTEHRlc3QuZXhhbXBsZS5jb20xHzAdBgkqhkiG
> 9w0BCQEWEHVzZXJAZXhhbXBsZS5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJ
> AoGBAMZw4nTepNIDhVzb0j7Z2Je4xAG56zmSHRMQeOGHemdh66u2/XAoLx7YCcYU
> ZgAMivyCsKgb/6CjQtsofvtrmC/eAehuK3/SINv7wd6Vv2pBt6ZpXgD4VBNKOND1
> GMbkPayVlQjbG4MD2dx2+H8EH3LMtdZrgKvPxPTE+bF5wZVNAgMBAAGgJTAjBgkq
> hkiG9w0BCQcxFhMUQSBjaGFsbGVuZ2UgcGFzc3dvcmQwDQYJKoZIhvcNAQEFBQAD
> gYEAG61CaJoJaVMhzCl903O6Mg51zq1zXcz75+VFj2I6rH9asckCld3mkOVx5gJU
> Ptt5CVQpNgNLdvbDPSsXretysOhqHmp9+CLv8FDuy1CDYfuaLtvlWvfhevskV0j6
> jtcEMyZ+f7+3yh421ido3nO4MIGeBgNVHSMEgZYwgZOAFLlNjtcEMyZ+f7+3yh42
> 1ido3nO4oXikdjB0MQswCQYDVQQGEwJVUzELMAkGA1UECBMCQ0ExFDASBgNVBAcT
> C1NhbnRhIENsYXJhMRswGQYDVQQKExJOdW92YSBTeXN0ZW1zIEluYy4xFDASBgNV
> BAsTC0VuZ2luZWVyaW5nMQ8wDQYDVQQDEwZ0ZXN0Q0GCAQAwDAYDVR0TBAUwAwEB
> /zANBgkqhkiG9w0BAQQFAAOBgQAhWaRwXNR6B4g6Lsnr+fptHv+WVhB5fKqGQqXc
> wR4pYiO4z42/j9Ijenh75tCKMhW51az8copP1EBmOcyuhf5C6vasrenn1ddkkYt4
> PR0vxGc40whuiozBolesmsmjBbedUCwQgdFDWhDIZJwK5+N3x/kfa2EHU6id1avt
> 4YL5Jg==
> -----END CERTIFICATE-----
> ENDOFBUF
UCS-A /security/trustpoint* # commit-buffer
UCS-A /security/trustpoint #
```

What to do next

Obtain a key ring certificate from the trust anchor or certificate authority and import it into the key ring.

Importing a Certificate into a Key Ring

Before you begin

- Configure a trusted point that contains the certificate chain for the key ring certificate.
- Obtain a key ring certificate from a trust anchor or certificate authority.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **scope keyring** *keyring-name*	Enters configuration mode for the key ring that will receive the certificate.

	Command or Action	Purpose
Step 3	UCS-A /security/keyring # **set trustpoint** *name*	Specifies the trusted point for the trust anchor or certificate authority from which the key ring certificate was obtained.
Step 4	UCS-A /security/keyring # **set cert**	Launches a dialog for entering and uploading the key ring certificate. At the prompt, paste the certificate text that you received from the trust anchor or certificate authority. On the next line following the certificate, type ENDOFBUF to complete the certificate input. **Important** The certificate must be in Base64 encoded X.509 (CER) format.
Step 5	UCS-A /security/keyring # **commit-buffer**	Commits the transaction.

Example

The following example specifies the trust point and imports a certificate into a key ring:

```
UCS-A# scope security
UCS-A /security # scope keyring kr220
UCS-A /security/keyring # set trustpoint tPoint10
UCS-A /security/keyring* # set cert
Enter lines one at a time. Enter ENDOFBUF to finish. Press ^C to abort.
Keyring certificate:
> -----BEGIN CERTIFICATE-----
> MIIB/zCCAWgCAQAwgZkxCzAJBgNVBAYTAlVTMQswCQYDVQQIEwJDQTEVMBMGA1UE
> BxMMU2FuIEpvc2UsIENBMRUwEwYDVQQKEwxFeGFtcGxlIEluYy4xEzARBgNVBAsT
> ClRlc3QgR3JvdXAxGTAXBgNVBAMTEHRlc3QuZXhhbXBsZS5jb20xHzAdBgkqhkiG
> 9w0BCQEWEHVzZXJAZXhhbXBsZS5jb20wgZ8wDQYJKoZIhvcNAQEBBQADgY0AMIGJ
> AoGBAMZw4nTepNIDhVzb0j7Z2Je4xAG56zmSHRMQeOGHemdh66u2/XAoLx7YCcYU
> ZgAMivyCsKgb/6CjQtsofvtrmC/eAehuK3/SINv7wd6Vv2pBt6ZpXgD4VBNKONDl
> GMbkPayVlQjbG4MD2dx2+H8EH3LMtdZrgKvPxPTE+bF5wZVNAgMBAAGgJTAjBgkq
> hkiG9w0BCQcxFhMUQSBjaGFsbGVuZ2UgcGFzc3dvcmQwDQYJKoZIhvcNAQEFBQAD
> gYEAG61CaJoJaVMhzCl903O6Mg51zq1zXcz75+VFj2I6rH9asckCld3mkOVx5gJU
> Ptt5CVQpNgNLdvbDPSsXretysOhqHmp9+CLv8FDuy1CDYfuaLtvlWvfhevskV0j6
> mK3Ku+YiORnv6DhxrOoqau8r/hyI/L43l7IPN1HhOi3oha4=
> -----END CERTIFICATE-----
> ENDOFBUF
UCS-A /security/keyring* # commit-buffer
UCS-A /security/keyring #
```

What to do next

Configure your HTTPS service with the key ring.

Deleting a Key Ring

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **delete keyring** *name*	Deletes the named key ring.
Step 3	UCS-A /security # **commit-buffer**	Commits the transaction.

Example

The following example deletes a key ring:

```
UCS-A# scope security
UCS-A /security # delete keyring key10
UCS-A /security* # commit-buffer
UCS-A /security #
```

Deleting a Trusted Point

Before you begin

Ensure that the trusted point is not used by a key ring.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope security**	Enters security mode.
Step 2	UCS-A /security # **delete trustpoint** *name*	Deletes the named trusted point.
Step 3	UCS-A /security # **commit-buffer**	Commits the transaction.

Example

The following example deletes a trusted point:

```
UCS-A# scope security
UCS-A /security # delete trustpoint tPoint10
UCS-A /security* # commit-buffer
UCS-A /security #
```

Enabling HTTP Redirection to HTTPS

Before you begin

Enable both HTTP and HTTPS.

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-redirect**	Enables the HTTP redirect service. If enabled, all attempts to communicate via HTTP are redirected to the equivalent HTTPS address. This option effectively disables HTTP access to this Cisco UCS domain.
Step 4	UCS-A /system/services # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example enables HTTP to HTTPS redirection and commits the transaction:

```
UCS-A# scope system
UCS-A /system # scope services
UCS-A /system/services # enable http-redirect
Warning: When committed, this closes all the web sessions.
UCS-A /system/services* # commit-buffer
UCS-A /system/services #
```

Network-Related Services

SNMP

SNMP Functional Overview

The SNMP framework consists of three parts:

- An SNMP manager—The system used to control and monitor the activities of network devices using SNMP.
- An SNMP agent—The software component within Cisco UCS, the managed device that maintains the data for Cisco UCS, and reports the data as needed to the SNMP manager. Cisco UCS includes the agent

and a collection of MIBs. To enable the SNMP agent and create the relationship between the manager and agent, enable and configure SNMP in Cisco UCS Manager.

- A managed information base (MIB)—The collection of managed objects on the SNMP agent. Cisco UCS release 1.4(1) and higher supports a larger number of MIBs than earlier releases.

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

- RFC 3410 (http://tools.ietf.org/html/rfc3410)
- RFC 3411 (http://tools.ietf.org/html/rfc3411)
- RFC 3412 (http://tools.ietf.org/html/rfc3412)
- RFC 3413 (http://tools.ietf.org/html/rfc3413)
- RFC 3414 (http://tools.ietf.org/html/rfc3414)
- RFC 3415 (http://tools.ietf.org/html/rfc3415)
- RFC 3416 (http://tools.ietf.org/html/rfc3416)
- RFC 3417 (http://tools.ietf.org/html/rfc3417)
- RFC 3418 (http://tools.ietf.org/html/rfc3418)
- RFC 3584 (http://tools.ietf.org/html/rfc3584)

SNMP Notifications

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

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

SNMP Security Levels and Privileges

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

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

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

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

Supported Combinations of SNMP Security Models and Levels

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

Table 5: SNMP Security Models and Levels

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

SNMPv3 Security Features

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

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

SNMP Support in Cisco UCS

Cisco UCS provides the following support for SNMP:

Support for MIBs

Cisco UCS supports read-only access to MIBs.

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

Authentication Protocols for SNMPv3 Users

Cisco UCS supports the following authentication protocols for SNMPv3 users:

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

Cisco UCS Manager Release 3.2(3) and later releases do not support MD5 authentication if SNMPv3 is in Federal Information Processing Standards (FIPS) mode. Hence, any existing or newly created SNMPv3 users with MD5 authentication will not be deployed with these releases and the following fault message will appear:

```
Major     F1036    2018-02-01T14:36:32.995     99095 SNMP User testuser can't be
deployed. Error: MD5 auth is not supported
```

To deploy such a user, modify the authentication type to **SHA**.

AES Privacy Protocol for SNMPv3 Users

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

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

Cisco UCS Manager Release 3.2(3) and later releases do not support SNMPv3 users without AES encryption. Hence, any existing or newly created SNMPv3 users without AES encryption will not be deployed with these releases, and the following fault message will appear:

```
Major     F1036    2018-02-01T14:36:32.995     99095 SNMP User testuser can't be
deployed. Error: AES is not enabled
```

To deploy such a user, enable **AES-128** encryption.

Enabling SNMP and Configuring SNMP Properties

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

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope monitoring**	Enters monitoring mode.
Step 2	UCS-A /monitoring # **enable snmp**	Enables SNMP.
Step 3	UCS-A /monitoring # **set snmp community**	Enters snmp community mode.

	Command or Action	Purpose
Step 4	UCS-A /monitoring # **Enter a snmp community:** *community-name*	Specifies SNMP community. Use the community name as a password. The community name can be any alphanumeric string up to 32 characters.
Step 5	UCS-A /monitoring # **set snmp syscontact** *system-contact-name*	Specifies the system contact person responsible for the SNMP. The system contact name can be any alphanumeric string up to 255 characters, such as an email address or name and telephone number.
Step 6	UCS-A /monitoring # **set snmp syslocation** *system-location-name*	Specifies the location of the host on which the SNMP agent (server) runs. The system location name can be any alphanumeric string up to 512 characters.
Step 7	UCS-A /monitoring # **commit-buffer**	Commits the transaction to the system configuration.

Example

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

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

What to do next

Create SNMP traps and users.

Creating an SNMP Trap

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope monitoring**	Enters monitoring mode.
Step 2	UCS-A /monitoring # **enable snmp**	Enables SNMP.
Step 3	UCS-A /monitoring # **create snmp-trap** {*hostname* | *ip-addr* | *ip6-addr*}	Creates an SNMP trap host with the specified host name, IPv4 address, or IPv6 address.

	Command or Action	Purpose
		The host name can be a fully qualified domain name of an IPv4 address.
Step 4	UCS-A /monitoring/snmp-trap # **set community** *community-name*	Specifies the SNMP community name to be used for the SNMP trap.
Step 5	UCS-A /monitoring/snmp-trap # **set port** *port-num*	Specifies the port to be used for the SNMP trap.
Step 6	UCS-A /monitoring/snmp-trap # **set version** {**v1** | **v2c** | **v3**}	Specifies the SNMP version and model used for the trap.
Step 7	(Optional) UCS-A /monitoring/snmp-trap # **set notificationtype** {**traps** | **informs**}	The type of trap to send. If you select v2c or v3 for the version, this can be: • **traps**—SNMP trap notifications • **informs**—SNMP inform notifications
Step 8	(Optional) UCS-A /monitoring/snmp-trap # **set v3 privilege** {**auth** | **noauth** | **priv**}	If you select v3 for the version, the privilege associated with the trap. This can be: • **auth**—Authentication but no encryption • **noauth**—No authentication or encryption • **priv**—Authentication and encryption
Step 9	UCS-A /monitoring/snmp-trap # **commit-buffer**	Commits the transaction to the system configuration.

Example

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

```
UCS-A# scope monitoring
UCS-A /monitoring # enable snmp
UCS-A /monitoring* # create snmp-trap 192.168.100.112
UCS-A /monitoring/snmp-trap* # set community SnmpCommSystem2
UCS-A /monitoring/snmp-trap* # set port 2
UCS-A /monitoring/snmp-trap* # set version v3
UCS-A /monitoring/snmp-trap* # set notificationtype traps
UCS-A /monitoring/snmp-trap* # set v3 privilege priv
UCS-A /monitoring/snmp-trap* # commit-buffer
UCS-A /monitoring/snmp-trap #
```

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

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

Deleting an SNMP Trap

Procedure

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

Example

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

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

Creating an SNMPv3 User

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope monitoring**	Enters monitoring mode.
Step 2	UCS-A /monitoring # **enable snmp**	Enables SNMP.
Step 3	UCS-A /monitoring # **create snmp-user** *user-name*	Creates the specified SNMPv3 user. An SNMP username cannot be the same as a local username. Choose an SNMP username that does not match a local username.
Step 4	UCS-A /monitoring/snmp-user # **set aes-128** {**no** | **yes**}	Enables or disables the use of AES-128 encryption.

	Command or Action	Purpose
Step 5	UCS-A /monitoring/snmp-user # **set auth** {**md5** | **sha**}	Specifies the use of MD5 or DHA authentication.
Step 6	UCS-A /monitoring/snmp-user # **set password**	Specifies the user password. After you enter the **set password** command, you are prompted to enter and confirm the password.
Step 7	UCS-A /monitoring/snmp-user # **set priv-password**	Specifies the user privacy password. After you enter the **set priv-password** command, you are prompted to enter and confirm the privacy password.
Step 8	UCS-A /monitoring/snmp-user # **commit-buffer**	Commits the transaction to the system configuration.

Example

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

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

Deleting an SNMPv3 User

Procedure

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

Example

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

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

Enabling Telnet

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.

Example

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 #
```

Enabling the CIMC Web Service

To enable the CIMC Web Service:

- You must be logged in with admin privileges.
- The CIMC web service must be disabled, as it is enabled by default.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system** /	Enters the system mode.
Step 2	UCS-A /system #**scope services**/	Enters the services mode for the system.
Step 3	UCS-A/system/services #**enable cimcwebsvc**/	Enable the CIMC web service.

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

Example

The following example shows how to enable the CIMC web service and save the transaction:

```
UCS-A# scope system
UCS-A/system # scope services
UCS-A/system/services # enable cimcwebsvc
UCS-A/system/services *# commit-buffer
UCS-A/system/services # commit-buffer
UCS-A/system/services # show cimcwebsvc
Name: cimcwebservice
    Admin State: Enabled
```

Disabling the CIMC Web Service

To disable the CIMC Web Service:

- You must be logged in with admin privileges.
- The CIMC web service must be enabled.

Note The CIMC web service is enabled by default.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system** /	Enters the system mode.
Step 2	UCS-A /system #**scope services**/	Enters the services mode for the system.
Step 3	UCS-A/system/services #**disable cimcwebsvc**/	Disables the CIMC web service.
Step 4	UCS-A/system/services *# **commit-buffer**/	Commits the transaction to the system configuration.

Example

The following example shows how to disable the CIMC web service and save the transaction:

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

```
Name: cimcwebservice
    Admin State: Disabled
```

Disabling Communication Services

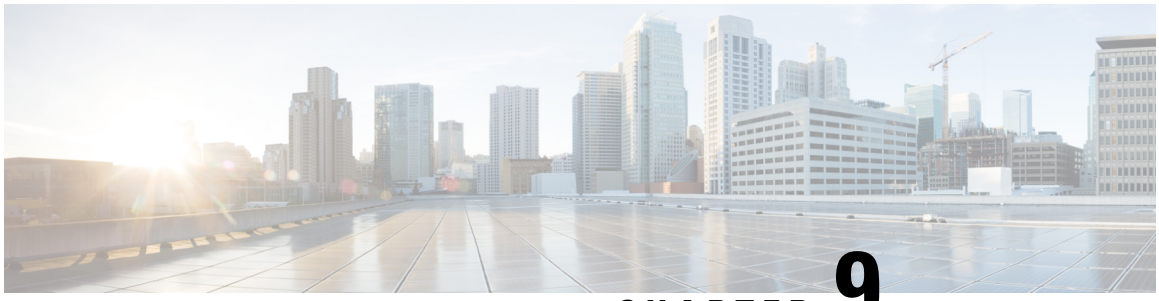
Procedure

	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 # **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.

Example

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 9

CIMC Session Management

CIMC Session Management

You can view and close any KVM, vMedia, and SOL sessions in Cisco UCS Manager. If you have administrator privileges, you can discontinue the KVM, vMedia, and SoL sessions of any user. . Cisco Integrated Management Controller (CIMC) provides session information to Cisco UCS Manager. When Cisco UCS Manager gets an event from CIMC, it updates its session table and displays the information to all users.

The session information consists of the following information:

- Name—The name of the user who launched the session.
- Session ID—The ID associated with the session. The format of the session ID for blades is [unique identifier] _ [chassis id] _ [Blade id]. The format of the session ID for racks is [unique identifier] _ 0 _ [Rack id].
- Type of session—KVM, vMedia, or SoL.
- Privilege level of the user—Read-Write, Read Only, or Granted.
- Administrative state—Active or Inactive. The value is active if the session is active. The value is inactive if the session terminate command has been issued but the session has not been terminated. This situation occurs when FSM of the server is in progress with another operation or when the connectivity to CIMC is lost.
- Source Address—The IP address of the computer from which the session was opened.
- Service Profile—The service profile associated with the session. The service profile attribute value for a CIMC session is displayed only if the session is opened on an IP address that is provided from the service profile.
- Server—The name of the server associated with the session.
- Login time—The date and time the session started.
- Last Update Time—The last time the session information was updated by CIMC.

A new session is generally added when a user connects to KVM, vMedia, or SOL. A Pnuos vMedia session will be displayed in the session table during the server discovery with the user name __vmediausr__.

The CIMC session data is available under the **CIMC Sessions** tab in Cisco UCS Manager GUI. Any CIMC session terminated by the user is audit logged with proper details.

Note To perform the GUI and CLI tasks that are described in this guide, a CIMC image version of 2.1(2a) or above is required for the session management support for the blade servers. The latest CIMC image version of 1.5(1l) and above is required for the rack-servers.

Viewing the CIMC Sessions Opened by the Local Users

Follow this task to view all the CIMC sessions opened by the local users or the CIMC sessions opened by a specific local user.

Note Viewing CIMC sessions of a specific server or a service-profile option is not present in CLI. It is available in GUI.

Procedure

	Command or Action	Purpose
Step 1	UCS-A # **scope security**	Enters security configuration mode.
Step 2	UCS-A /security # **show cimc-sessions local**	Displays all CIMC sessions opened by the local users.
Step 3	UCS-A /security # **show cimc-sessions local** *user-name*	Displays all CIMC sessions opened by a specific local user.

Example

The following examples show how to view:

- All CIMC sessions opened by local users
- CIMC session opened by a specific local user
- Details of the CIMC session opened by a specific local user.

```
All sessions opened by local users:
UCS-A # scope security
UCS-A /security # show cimc-sessions local

Session ID   Type     User      Source Addr     Admin State
----------   ------ --------- --------------- ------------
42_1_1       Kvm      admin     10.106.22.117     Active
4_1_5        Kvm      admin     10.106.22.117     Active
5_1_5        Vmedia   admin     10.106.22.117     Active

Session opened by a specific local user:
UCS-A /security # show cimc-sessions local admin
```

```
Session ID   Type     User      Source Addr     Admin State
----------   ------ --------- --------------- ------------
42_1_1       Kvm      admin     10.106.22.117     Active

Details of session opened by a specific local user:
UCS-A /security # show cimc-sessions local admin detail
Session ID 42_1_1
   Type: Kvm
   User: admin
   Source Addr: 10.106.22.117
   Login Time: 2013-06-28T06:09:53.000
   Last Updated Time: 2013-06-28T06:21:52.000
   Admin State: Active
   Priv: RW
   Server: sys/chassis-1/blade-1
   Service Profile:
```

Viewing the CIMC Sessions Opened by the Remote Users

Follow this task to view all the CIMC sessions opened by the remote users or the CIMC sessions opened by a specific remote user.

Procedure

	Command or Action	Purpose
Step 1	UCS-A # **scope security**	Enters security configuration mode.
Step 2	UCS-A /security # **show cimc-sessions remote**	Displays all CIMC sessions opened by the remote users.
Step 3	UCS-A /security # **show cimc-sessions remote** *user-name*	Displays all CIMC sessions opened by a specific remote user.

Example

The following examples show how to view:

- All CIMC sessions opened by remote users
- CIMC session opened by a specific remote user
- Details of the CIMC session opened by a specific remote user.

```
All sessions opened by remote users:
UCS-A # scope security
UCS-A /security # show cimc-sessions remote

Session ID   Type     User            Source Addr      Admin State
----------   ------ ---------        ---------------   ------------
43_1_1       Kvm     administrator     10.106.22.117     Active
6_1_5        Kvm     test-remote       10.106.22.117     Active
7_1_5        Vmedia  test-remote       10.106.22.117     Active

Session opened by a specific remote user:
```

```
UCS-A /security # show cimc-sessions remote administrator

Session ID   Type     User             Source Addr      Admin State
----------   ------ ---------        ---------------   ------------
43_1_1       Kvm     administrator     10.106.22.117     Active

Details of session opened by a specific remote user:
UCS-A /security # show cimc-sessions remote administrator detail
Session ID 43_1_1
   Type: Kvm
   User: administrator
   Source Addr: 10.106.22.117
   Login Time: 2013-06-28T06:09:53.000
   Last Updated Time: 2013-06-28T06:21:52.000
   Admin State: Active
   Priv: RW
   Server: sys/chassis-1/blade-1
   Service Profile:
```

Viewing the CIMC Sessions Opened by an IPMI User

To view the CIMC sessions opened by an IPMI user, complete the following steps:

Procedure

	Command or Action	Purpose
Step 1	UCS-A # **scope org** *org-name*	Enters the root organization mode.
Step 2	UCS-A /org # **scope ipmi-access-profile** *profile-name*	Enters the IPMI access profile name.
Step 3	UCS-A /org/ipmi-access-profile # **scope ipmi-user** *user-name*	Enters an IPMI user name.
Step 4	UCS-A /org/ipmi-access-profile/ipmi-user # **show cimc-sessions**	Displays all CIMC sessions opened by the specified IPMI User.

Example

The following example shows how to view all the CIMC sessions opened by an IPMI user:

```
UCS-A # scope org Finance
UCS-A /org* # scope ipmi-access-profile ReadOnly
UCS-A /org/ipmi-access-profile* # scope ipmi-user alice
UCS-A /org/ipmi-access-profile/ipmi-user # show cimc-sessions

Session ID    Type    User        Source Addr       Admin State
----------   ------  ---------   ---------------   --------------
45_1_1        sol     alice       10.106.22.117     Active
```

Clearing the CIMC Sessions of a Server

This task shows how to clear all CIMC sessions opened on a server. You can also clear the CIMC sessions on a server based on the session type and the user name.

Procedure

	Command or Action	Purpose
Step 1	UCS-A # **scope security**	Enters security configuration mode.
Step 2	UCS-A /security # **terminate cimc-sessions server** *chassis-id/blade-id*	Clears the CIMC sessions on a specific blade server of a chassis.
Step 3	UCS-A /security # **terminate cimc-sessions server** *Rack-server-id*	Clears the CIMC sessions on a specific rack server.
Step 4	UCS-A /security # **terminate cimc-sessions server** *server-id* **type** *session-type*	Clears the CIMC sessions of a specific type on a server.
Step 5	UCS-A /security # **terminate cimc-sessions server** *server-id* **user-name** *user-name*	Clears the CIMC sessions of a specific user on a server.

Example

The first example shows how to clear all CIMC sessions on a server. The second example shows how to clear the CIMC sessions of a specific type on a server. The third example shows how to clear the CIMC sessions of a specific user on a server:

```
UCS-A /security # scope security
UCS-A /security # terminate cimc-sessions server 2/1
This will close KVM sessions. Are you sure? (yes/no):yes
UCS-A /security

UCS-A # scope security
UCS-A /security # terminate cimc-sessions server 2/1 type kvm
This will close KVM sessions. Are you sure? (yes/no):yes

UCS-A # scope security
UCS-A /security # terminate cimc-sessions server 2/1 user-name test-user
This will close KVM sessions. Are you sure? (yes/no):yes
```

Clearing the CIMC Sessions of a Modular Server

This task shows how to clear all CIMC sessions opened on a server. You can also clear the CIMC sessions on a server based on the session type and the user name.

Procedure

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

	Command or Action	Purpose
Step 2	UCS-A /security # **terminate cimc-sessions server** *chassis-id* / *cartridge-id* / *server-id*	Clears the CIMC sessions on a specific modular server of a cartridge on a chassis.
Step 3	UCS-A /security # **terminate cimc-sessions server** *chassis-id* / *cartridge-id* / *server-id* **type** *session-type*	Clears the CIMC sessions of a specific type on a server.
Step 4	UCS-A /security # **terminate cimc-sessions server** *chassis-id* / *cartridge-id* / *server-id* **user-name** *user-name*	Clears the CIMC sessions of a specific user on a server.

Example

The first example shows how to clear all CIMC sessions on a server. The second example shows how to clear the CIMC sessions of a specific type on a server. The third example shows how to clear the CIMC sessions of a specific user on a server:

```
UCS-A /security # scope security
UCS-A /security # terminate cimc-sessions server 1/2/1
This will close cimc sessions. Are you sure? (yes/no):yes
UCS-A /security

UCS-A # scope security
UCS-A /security # terminate cimc-sessions server 1/2/1 type kvm
This will close KVM sessions. Are you sure? (yes/no):yes

UCS-A # scope security
UCS-A /security # terminate cimc-sessions server 1/2/1 user-name test-user
This will close cimc sessions. Are you sure? (yes/no):yes
```

Clearing All CIMC Sessions Opened by a Local User

This task shows how to clear the sessions opened by a local user.

Procedure

	Command or Action	Purpose
Step 1	UCS-A # **scope security**	Enters security configuration mode.
Step 2	UCS-A /security # **terminate cimc-sessions local-user** *user-name*	Clears all CIMC sessions opened by a local user.
Step 3	UCS-A /security # **terminate cimc-sessions local-user** *user-name* **type** {**kvm** | **vmedia sol** | **all**}	Clears all CIMC sessions of specific session type opened by a local user.

Example

The following example shows how to clear the CIMC sessions opened by a local user:

```
UCS-A /security# scope security
UCS-A /security# terminate cimc-sessions local-user testuser
This will close cimc sessions. Are you sure? (yes/no):yes
UCS-A /security#
```

Clearing All CIMC Sessions Opened by a Remote User

This task shows how to clear CIMC sessions opened by a remote user.

Procedure

	Command or Action	Purpose
Step 1	UCS-A # **scope security**	Enters security configuration mode.
Step 2	UCS-A /security # **terminate cimc-sessions remote-user** *user-name*	Clears all CIMC sessions opened by a remote user.
Step 3	UCS-A /security # **terminate cimc-sessions remote-user** *user-name* **type** {**kvm** | **vmedia** **sol** | **all**}	Clears all CIMC sessions of specific session type opened by a remote user.

Example

The following example shows how to clear all CIMC sessions opened by a remote user:

```
UCS-A /security# scope security
UCS-A /security# terminate cimc-sessions remote-user testuser
This will close cimc sessions. Are you sure? (yes/no):yes
UCS-A /security#
```

Clearing a Specific CIMC Session Opened by a Local User

This task shows how to clear a specific CIMC session opened by a local user.

Procedure

	Command or Action	Purpose
Step 1	UCS-A # **scope security**	Enters security configuration mode.
Step 2	UCS-A /security # **scope local-user** *user-name*	Enters local user mode.
Step 3	UCS-A /security/local user # **terminate cimc-session** *session-id*	Clears the chosen CIMC session.
Step 4	UCS-A /security/local user* # **commit-buffer**	Commits the transaction.

Example

The following example shows how to clear a specific CIMC session opened by a local user and commits the transaction:

```
UCS-A /security# scope security
UCS-A /security# scope local-user admin
UCS-A /security/local user # terminate cimc-session 6_1_2
UCS-A /security/local user*# commit-buffer
UCS-A /security/local user#
```

Clearing a Specific CIMC Session Opened by a Remote User

This task shows how to clear a specific CIMC session opened by a remote user.

Procedure

	Command or Action	Purpose
Step 1	UCS-A # **scope security**	Enters security configuration mode.
Step 2	UCS-A /security # **scope remote -user** *user-name*	Enters remote user mode.
Step 3	UCS-A /security/remote user # **terminate cimc-session** *session-id*	Clears the chosen CIMC session.
Step 4	UCS-A /security/remote user* # **commit-buffer**	Commits the transaction.

Example

The following example shows how to clear a specific CIMC session opened by a remote user and commits the transaction:

```
UCS-A /security# scope security
UCS-A /security# scope remote-user admin
UCS-A /security/remote user # terminate cimc-session 6_1_3
UCS-A /security/remote user*# commit-buffer
UCS-A /security/remote user#
```

Clearing a CIMC Session Opened by an IPMI User

To clear a CIMC session opened by an IPMI user, complete the following steps:

Procedure

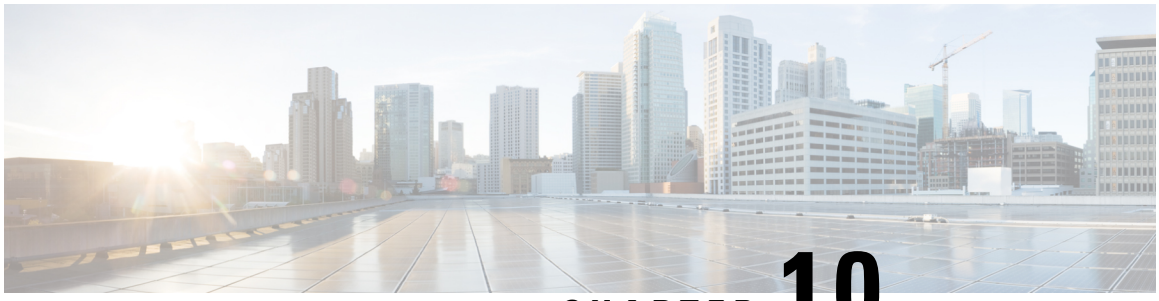
	Command or Action	Purpose
Step 1	UCS-A # **scope org** *org-name*	Enters the root organization mode.

	Command or Action	Purpose
Step 2	UCS-A /org # **scope ipmi-access-profile** *profile-name*	Enters the IPMI access profile name.
Step 3	UCS-A /org/ipmi-access-profile # **scope ipmi-user** *user-name*	Enters the IPMI user.
Step 4	UCS-A /org/ipmi-access-profile/ipmi-user # **terminate cimc-sessions** *session-id*	Terminates a specific CIMC session opened by an IPMI user.
Step 5	UCS-A /org/ipmi-access-profile/ipmi-user * **commit-buffer**	Commits the changes.

Example

The following example displays how to clear a specific CIMC session opened by an IPMI user and commits the changes:

```
UCS-A # scope org Finance
UCS-A /org* # scope ipmi-access-profile ReadOnly
UCS-A /org/ipmi-access-profile* # scope ipmi-user alice
UCS-A /org/ipmi-access-profile/ipmi-user # terminate cimc-sessions 5_1_2
UCS-A /org/ipmi-access-profile/ipmi-user* # commit-buffer
```



CHAPTER 10

Setting the Management IP Address

Management IP Address

Each server in a Cisco UCS domain must have a one or more management IP addresses assigned to its Cisco Integrated Management Controller (CIMC) or to the service profile associated with the server. Cisco UCS Manager uses these IP addresses for external access that terminates in the CIMC. This external access can be through one of the following services:

- KVM console
- Serial over LAN
- An IPMI tool

The management IP addresses used to access the CIMC on a server can be out-of-band (OOB) addresses, through which traffic traverses the fabric interconnect via the management port, or inband addresses, through which traffic traverses the fabric interconnect via the fabric uplink port. Up to six IP addresses can be configured to access the CIMC on a server, two out-of-band (OOB) and four inband.

You can configure the following management IP addresses:

- A static OOB IPv4 address assigned directly to the server
- An OOB IPv4 address assigned to the server from a global ext-mgmt pool
- An inband IPv4 address derived from a service profile associated with the server
- An inband IPv4 address drawn from a management IP pool and assigned to a service profile or service profile template
- An static inband IPv6 address assigned directly to the server
- An inband IPv6 address derived from a service profile associated with the server

You can assign multiple management IP addresses to each CIMC on the server and to the service profile associated with the server. If you do so, you must use different IP addresses for each of them.

A management IP address that is assigned to a service profile moves with that service profile. If KVM or SoL sessions are active when you migrate the service profile to another server, Cisco UCS Manager terminates the sessions and does not restart them after the migration is completed. You configure the IP address when you create or modify a service profile.

Note You cannot assign a static IP address to a server or service profile if that IP address has already been assigned to a server or service profile in the Cisco UCS domain. If you attempt to do so, Cisco UCS Manager warns you that the IP address is already in use and rejects the configuration.

A unicast Internet Control Message Protocol (ICMP) request will be sent to the gateway IP address every second from each server that is configured with an inband IP address. This request is to check if connectivity for the inband traffic through the current Fabric Interconnect (FI) is up, and to initiate a failover to the other FI if it is down. The path selected for inband and the failover operations are completely independent of the server data traffic. The default polling interval is 1 second and the polling interval is configurable to a maximum of 5 seconds. After three failed polls, the CIMC will failover to the other FI. During failover, the CIMC will issue a Gratuitous Address Resolution Protocol (GARP) on the newly selected uplinks to notify the network that the MAC has been moved to a new location.

Configuring the Management IP Address on a Modular Server

Configuring a Modular Server to Use a Static IP Address

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *chassis-id / cartridge-id / server-id*	Enters server mode for the specified server.
Step 2	UCS-A /chassis/cartridge/server # **scope cimc**	Enters server CIMC mode.
Step 3	UCS-A /chassis/cartridge/server/cimc # **create ext-static-ip**	Creates a static management IP address for the specified server.
Step 4	UCS-A /chassis/cartridge/server/cimc/ext-static-ip # **set addr** *ip-addr*	Specifies the static IPv4 address to be assigned to the server.
Step 5	UCS-A /chassis/cartridge/server/cimc/ext-static-ip # **set default-gw** *ip-addr*	Specifies the default gateway that the IP address should use.
Step 6	UCS-A /chassis/cartridge/server/cimc/ext-static-ip # **set subnet** *ip-addr*	Specifies the subnet mask for the IP address.

	Command or Action	Purpose
Step 7	UCS-A /chassis/cartridge/server/cimc/ext-static-ip # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example configures a static management IP address for chassis 1 cartridge 1 server 1, sets the static IPv4 address, sets the default gateway, sets the subnet mask, and commits the transaction:

```
UCS-A# scope server 1/1/1
UCS-A /chassis/cartridge/server # scope cimc
UCS-A /chassis/cartridge/server/cimc # create ext-static-ip
UCS-A /chassis/cartridge/server/cimc/ext-static-ip* # set addr 192.168.10.10
UCS-A /chassis/cartridge/server/cimc/ext-static-ip* # set default-gw 192.168.10.1
UCS-A /chassis/cartridge/server/cimc/ext-static-ip* # set subnet 255.255.255.0
UCS-A /chassis/cartridge/server/cimc/ext-static-ip* # commit-buffer
UCS-A /chassis/cartridge/server/cimc/ext-static-ip #
```

Configuring a Modular Server to Use a Static IPv6 Address

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *chassis-id / cartridge-id / server-id*	Enters server mode for the specified server.
Step 2	UCS-A /chassis/cartridge/server # **scope cimc**	Enters server CIMC mode.
Step 3	UCS-A /chassis/cartridge/server/cimc # **create ext-static-ip6**	Creates a static management IPv6 address for the specified server.
Step 4	UCS-A /chassis/cartridge/server/cimc/ext-static-ip6 # **set addr ipv6-addr**	Specifies the static IPv6 address to be assigned to the server.
Step 5	UCS-A /chassis/cartridge/server/cimc/ext-static-ip6 # **set default-gw ip6-addr**	Specifies the default gateway that the IPv6 address should use.
Step 6	UCS-A /chassis/cartridge/server/cimc/ext-static-ip6 # **set prefix ip6-addr**	Specifies the network prefix for an IPv6 address.
Step 7	UCS-A /chassis/cartridge/server/cimc/ext-static-ip6 # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example configures a static management IPv6 address for chassis 1 cartridge 1 server 1, sets a static IPv6 address, sets the default gateway, sets the network prefix, and commits the transaction:

```
UCS-A# scope server 1/1/1
UCS-A /chassis/cartridge/server # scope cimc
UCS-A /chassis/cartridge/server/cimc # create ext-static-ip6
UCS-A /chassis/cartridge/server/cimc/ext-static-ip* # set addr 2001:888::10
UCS-A /chassis/cartridge/server/cimc/ext-static-ip* # set default-gw 2001:888::100
UCS-A /chassis/cartridge/server/cimc/ext-static-ip* # set prefix 64
UCS-A /chassis/cartridge/server/cimc/ext-static-ip* # commit-buffer
UCS-A /chassis/cartridge/server/cimc/ext-static-ip #
```

Configuring a Server to Use the Management IP Pool

Deleting the static management IP address returns the specified server to the management IP pool.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope server** *chassis-id* / *cartridge-id* / *server-id*	Enters server mode for the specified server.
Step 2	UCS-A /chassis/cartridge/server # **scope cimc**	Enters server CIMC mode.
Step 3	UCS-A /chassis/cartridge/server/cimc # **delete** {*ext-static-ip* | *ext-static-ip6*}	Deletes the external static IPv4 or IPv6 address and returns the server to the management IP pool.
Step 4	UCS-A /chassis/cartridge/server/cimc/ # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes the static management IP address for chassis 1 cartridge 1 server 1 and commits the transaction:

```
UCS-A# scope server 1/1/1
UCS-A /chassis/cartridge/server # scope cimc
UCS-A /chassis/cartridge/server/cimc # delete ext-static-ip
UCS-A /chassis/cartridge/server/cimc* # commit-buffer
UCS-A /chassis/cartridge/server/cimc/ #
```

The following example deletes the static management IPv6 address for chassis 1 cartridge 1 server 1 and commits the transaction:

```
UCS-A# scope server 1/1/1
UCS-A /chassis/cartridge/server # scope cimc
UCS-A /chassis/cartridge/server/cimc # delete ext-static-ip6
UCS-A /chassis/cartridge/server/cimc* # commit-buffer
UCS-A /chassis/cartridge/server/cimc/ #
```

Setting the Management IP Address on a Service Profile or 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 # **scope service-profile** *profile-name*	Enters organization service profile mode for the specified service.
Step 3	UCS-A /org/service-profile # **set ext-mgmt-ip-state** {*none* | *ext-pooled-ip* | *ext-pooled-ip6* | *ext-static-ip* | *ext-static-ip6*}	Specifies how the management IPv4 or IPv6 address will be assigned to the service profile. You can set the management IP address policy using the following options: • None--The service profile is not assigned an IP address. • Pooled--The service profile is assigned an IP address from the management IPv4 or IPv6 pool. • Static--The service profile is assigned the configured static IPv4 or IPv6 address. **Note** Setting the ext-management-ip-state to static for a service profile template is not supported and will result in an error.
Step 4	UCS-A /org/service-profile # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example sets the management address policy for a service profile called accounting to static IPv4 and then commits the transaction:

```
UCS-A# scope org /
UCS-A /org # scope service-profile accounting
UCS-A /org/service-profile # set ext-mgmt-ip-state ext-static-ip
UCS-A /org/service-profile* # commit-buffer
UCS-A /org/service-profile #
```

What to do next

If you have set the management IP address to static, configure a server to use a static IP address.

Configuring the Management IP Pool

Management IP Pools

The default management IP pool, **IP Pool ext-mgmt** is a collection of external IPv4 and IPv6 addresses. Cisco UCS Manager reserves each block of IP addresses in the management IP pool for external access that terminates in the CIMC on a server.

By default, the **IP Pool ext-mgmt** is used to configure the CIMC outbound management IP address. You cannot change this IP pool if already a static IP address is assigned to the server from this pool. If you want to configure the outbound management IP address for CIMC from a static IP address, then you can delete the IP addresses from the default management IP pool.

You can configure separate out-of-band IPv4 address pools, and in-band IPv4 or IPv6 address pools. You can configure in-band pools that contain both IPv4 and IPv6 address blocks.

Tip To avoid assigning an IP pool that contains only IPv4 addresses as the in-band IPv6 policy, or assigning an IP pool that contains only IPv6 addresses as the in-band IPv4 policy to a server CIMC, it is suggested that you configure separate in-band address pools, each with only IPv4 or IPv6 addresses.

You can configure service profiles and service profile templates to use IP addresses from the management IP pools. You cannot configure servers to use the management IP pool.

All IP addresses in the management IP pool must be in the same IPv4 subnet, or have the same IPv6 network prefix as the IP address of the fabric interconnect.

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

Configuring IP Address Blocks for the Management IP Pool

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

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org /**	Enters root organization mode.
Step 2	UCS-A /org # **scope ip-pool ext-mgmt**	Enters organization IP pool mode. **Note**

	Command or Action	Purpose
		You cannot create (or delete) a management IP pool. You can only enter (scope to) the existing default pool.
Step 3	(Optional) UCS-A /org/ip-pool # **set descr** *description*	Provides a description for the management IP pool. This description applies to all address blocks in the management IP pool. **Note** If your description includes spaces, special characters, or punctuation, you must begin and end your description with quotation marks. The quotation marks will not appear in the description field of any **show** command output.
Step 4	UCS-A /org/ip-pool # **set assignmentorder** {**default** | **sequential**}	This can be one of the following: • **default**—Cisco UCS Manager selects a random identity from the pool. • **sequential**—Cisco UCS Manager selects the lowest available identity from the pool.
Step 5	UCS-A /org/ip-pool # **create block** *first-ip-addr last-ip-addr gateway-ip-addr subnet-mask*	Creates a block (range) of IP addresses, and enters organization IP pool block mode. You must specify the first and last IP addresses in the address range, the gateway IP address, and subnet mask. **Note** An IP pool can contain more than one IP block. To create multiple blocks, enter multiple **create block** commands from organization IP pool mode.
Step 6	UCS-A /org/ip-pool/block # **set primary-dns** *ip-addrress* |**secondary-dns** *ip-address*	Specifies the primary DNS and secondary DNS IP addresses.
Step 7	UCS-A /org/ip-pool/ ipv6-block # **commit-buffer**	Commits the transaction to the system configuration.
Step 8	UCS-A /org/ip-pool/block # **exit**	Exits IPv4 block configuration mode.
Step 9	UCS-A /org/ip-pool # **create ipv6-block** *first-ip6-addr last-ip6-addr gateway-ip6-addr prefix*	Creates a block (range) of IPv6 addresses, and enters organization IP pool IPv6 block mode. You must specify the first and last IPv6 addresses in the address range, the gateway IPv6 address, and network prefix. **Note**

	Command or Action	Purpose
		An IP pool can contain more than one IPv6 block. To create multiple IPv6 blocks, enter multiple **create ipv6-block** commands from organization IP pool mode.
Step 10	UCS-A /org/ip-pool/ipv6-block # **set primary-dns** *ip6-address* **secondary-dns** *ip6-address*	Specifies the primary DNS and secondary DNS IPv6 addresses.
Step 11	UCS-A /org/ip-pool/ipv6-block # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example configures an IPv4 address block for the management IP pool, specifies the primary and secondary IPv4 addresses, creates an IPv6 block, specifies the primary and secondary IPv6 addresses and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # scope ip-pool ext-mgmt-ip
UCS-A /org/ip-pool* # set descr "This is a management ip pool example."
UCS-A /org/ip-pool* # create block 192.168.100.1 192.168.100.200 192.168.100.10 255.255.255.0
UCS-A /org/ip-pool/block* # set primary-dns 192.168.100.1 secondary-dns 192.168.100.20
UCS-A /org/ip-pool/block* commit-buffer
UCS-A /org/ip-pool/block exit
UCS-A /org/ip-pool* # create ipv6-block 2001:888::10 2001:888::100 2001:888::1 64
UCS-A /org/ip-pool/ipv6- block* set primary-dns 2001:888::11 secondary-dns 2001:888::12
UCS-A /org/ip-pool/ipv6- block* commit-buffer
UCS-A /org/ip-pool/ipv6- block #UCS-A /org/ip-pool/block* # commit-buffer
UCS-A /org/ip-pool/block #
```

The following example configures an IPv6 address block for the management IP pool and commits the transaction:

```
UCS-A# scope org /
UCS-A /org #scope ip-pool ext-mgmt-ip
UCS-A /org/ip-pool* # set descr "This is a management IPv6 pool example."
UCS-A /org/ip-pool* # create ipv6-block 2001:888::10 2001:888::100 2001:888::1 64
UCS-A /org/ip-pool/ipv6-block* # commit-buffer
UCS-A /org/ip-pool/ipv6-block* #
```

What to do next

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

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

Example

The following example deletes an IP address block from the management IP pool and commits the transaction:

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

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

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

Changing the System Name

Procedure

	Command or Action	Purpose
Step 1	UCS-A # **scope system**	Enters system mode.
Step 2	UCS-A /system # **set name** *name*	Sets the system name.

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

The name is updated on both fabric interconnects within about 30 seconds after the transaction is committed.

Example

The following example changes the system name and commits the transaction:

```
UCS-A# scope system
UCS-A /system* # set name SanJose5
UCS-A /system* # commit-buffer
UCS-A /system #
```

Changing the Management Subnet of a Cluster

When changing the IPv4 management subnet in a cluster configuration, you must change the following three IPv4 addresses simultaneously and you must configure all three in the same subnet:

- Management port IP address for fabric interconnect A
- Management port IP address for fabric interconnect B
- Cluster IP (virtual IP) address

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope fabric-interconnect a**	Enters fabric interconnect mode for fabric A.
Step 2	UCS-A /fabric-interconnect # **set out-of-band ip** *ip-address* **netmask** *netmask* **gw** *gateway-ip-address*	Sets the IP address, network mask, and gateway IP address of the fabric interconnect.
Step 3	UCS-A /fabric-interconnect # **scope fabric-interconnect b**	Enters fabric interconnect mode for fabric B.
Step 4	UCS-A /fabric-interconnect # **set out-of-band ip** *ip-address* **netmask** *netmask* **gw** *gateway-ip-address*	Sets the IP address, netmask, and gateway IP address of the fabric interconnect.
Step 5	UCS-A /fabric-interconnect # **scope system**	Enters system mode.
Step 6	UCS-A /system # **set virtual-ip** *vip-address*	Sets the virtual IP address for the cluster.
Step 7	UCS-A /system # **commit-buffer**	Commits the transaction to the system configuration.

When you commit the transaction, you are disconnected from the management session. Reconnect at the new management IP address.

Example

This example changes both fabric-interconnect IP addresses, changes the virtual IP address, and commits the transaction, disconnecting the session:

```
UCS-A# scope fabric-interconnect a
UCS-A /fabric-interconnect # set out-of-band ip 192.0.2.111 netmask 255.255.255.0 gw 192.0.2.1
UCS-A /fabric-interconnect* # scope fabric-interconnect b
UCS-A /fabric-interconnect* # set out-of-band ip 192.0.2.112 netmask 255.255.255.0 gw
192.0.2.1
UCS-A /fabric-interconnect* # scope system
UCS-A /system* # set virtual-ip 192.0.2.113
UCS-A /system* # commit-buffer
```

Changing the Management Prefix of a Cluster

When changing the IPv6 management prefix in a cluster configuration, you must change the following three IPv6 addresses simultaneously and you must configure all three with the same network prefix:

- Management port IPv6 address for fabric interconnect A
- Management port IPv6 address for fabric interconnect B
- Cluster IPv6 (virtual IPv6) address

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope fabric-interconnect a**	Enters fabric interconnect mode for fabric A.
Step 2	UCS-A fabric-interconnect # **scope ipv6-config**	Enters IPv6 configuration mode for fabric A.
Step 3	UCS-A fabric-interconnect/ ipv6-config # **set out-of-band ipv6** *ipv6-addr* **ipv6-gw** *ipv6-gw-addr* **ipv6-prefix** *prefix*	Sets the management IPv6 address, gateway IPv6 address, and network prefix for fabric A.
Step 4	UCS-A fabric-interconnect/ipv6-config # **scope fabric-interconnect b**	Enter fabric interconnect mode for fabric B.
Step 5	UCS-A fabric-interconnect/ # **scope ipv6-config**	Enter IPv6 configuration mode for fabric B
Step 6	UCS-A/fabric-interconnect/ipv6-config # **set out-of-band ipv6** *ipv6-addr* **ipv6-gw** *ipv6-gw-addr* **ipv6-prefix** *prefix*	Sets the management IPv6 address, gateway IPv6 address, and network prefix for fabric B.
Step 7	UCS-A/fabric-interconnect/ipv6-config # **scope system**	Enters system mode.

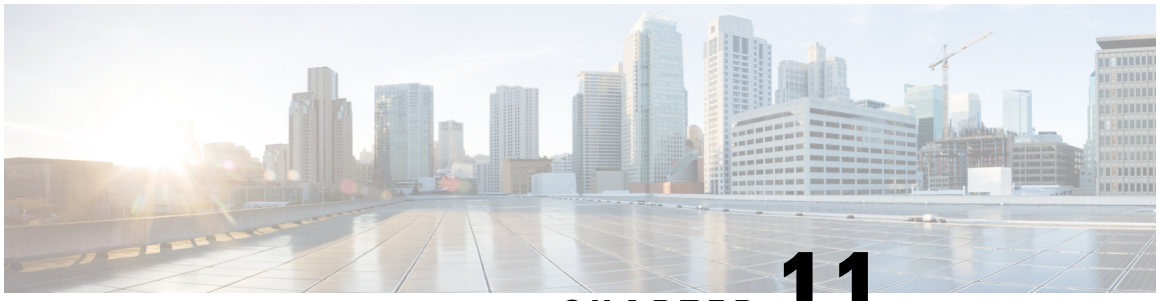
	Command or Action	Purpose
Step 8	UCS-A/system # **set virtual-ip ipv6** *virtual-ip6-addr*	Sets the virtual IPv6 address for the cluster.
Step 9	UCS-A/system # **commit-buffer**	Commits the transaction to the system configuration.

When you commit the transaction, you are disconnected from the management session. Reconnect at the new management IPv6 address.

Example

This example changes both management IPv6 addresses, changes the virtual IPv6 address, and commits the transaction:

```
UCS-A# scope fabric-interconnect a
UCS-A /fabric-interconnect # scope ipv6-config
UCS-A /fabric-interconnect/ipv6-config # set out-of-band ipv6 2001:10::157
UCS-A /fabric-interconnect/ipv6-config* # set out-of-band ipv6-gw 2001:10::1
UCS-A /fabric-interconnect/ipv6-config* # set out-of-band ipv6-prefix 64
UCS-A /fabric-interconnect/ipv6-config* # scope fabric-interconnect b
UCS-A /fabric-interconnect* # scope ipv6-config
UCS-A /fabric-interconnect/ipv6-config* # set out-of-band ipv6 2001:10::158
UCS-A /fabric-interconnect/ipv6-config* # set out-of-band ipv6-gw 2001:10::1
UCS-A /fabric-interconnect/ipv6-config* # set out-of-band ipv6-prefix 64
UCS-A /fabric-interconnect/ipv6-config* # scope system
UCS-A /system* # set virtual-ip ipv6 2001:10::156
UCS-A /system* # commit-buffer
UCS-A /system #
```



CHAPTER 11

Organizations in UCS Manager

Organizations in a Multitenancy Environment

Multi-tenancy allows you to divide the large physical infrastructure of an Cisco UCS domain 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 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 in 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.
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.

Example

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

```
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.

	Command or Action	Purpose
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.

Example

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

```
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

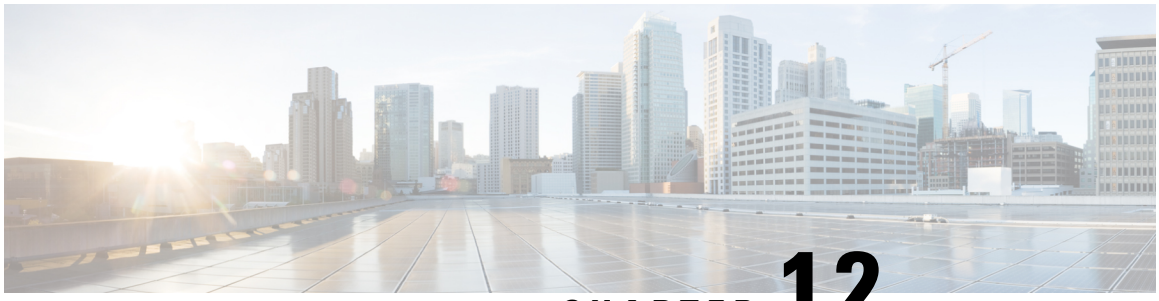
Procedure

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

Example

The following example deletes the organization under the root organization named Finance and commits the transaction:

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



CHAPTER 12

Backup and Restore

Backup and Restore Operations

Backup Operations in UCS

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.

Considerations and Recommendations for Backup Operations

Before you create a backup operation, consider the following:

Backup Locations

The backup location is the destination or folder on the network where you want Cisco UCS Manager to export the backup file. You can maintain only one backup operation for each location where you plan to save a backup file.

Potential to Overwrite Backup Files

If you rerun a backup operation without changing the filename, Cisco UCS Manager overwrites the existing file on the server. To avoid overwriting existing backup files, change the filename in the backup operation or copy the existing file to another location.

Multiple Types of Backups

You can run and export more than one type of backup to the same location. Change the backup type before you rerun the backup operation. We recommend that you change the filename for easier identification and to avoid overwriting the existing backup file.

Scheduled Backups

You can 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, save, or export the configuration file until you set the admin state of the backup operation to enabled.

Incremental Backups

You cannot perform incremental backups.

Encryption of Full State Backups

Full state backups are encrypted so that passwords and other sensitive information are not exported as clear text.

Backups from Cisco UCS Manager

Port configurations that include global VLANs and VSANs are not restored when you do an all-config backup in Cisco UCS Manager. Reconfigure the ports from Cisco UCS Central.

FSM Tasks for Backup Policy and Configuration Export Policy

When configuring both **Backup Policy** and **Config Export Policy** on the **Policy Backup & Export** tab and using the same hostname for both policies, Cisco UCS Manager will create only one **Backup Operation** in the **Backup Configuration** page to run both tasks. Each policy run will not have a separate FSM task.

To see a separate FSM task for each policy, you can create a hostname alias in your DNS server to point to the same FTP/TFTP/SCP/SFTP server. Then you can use one hostname for the **Backup Policy** and another hostname for the **Config Export Policy**.

Password Encryption Key for Backup Configuration Files

Beginning with release 4.2(3d), Cisco UCS Manager introduces **Password Encryption Key** to enhance security for backup configuration files.

You must set **Password Encryption Key** in order to create backup configuration files and also to import the backup files. Cisco UCS Manager release 4.2(3d) and later do not allow you to create backup configuration files or import backup configuration files without setting the **Password Encryption Key**. If the **Password Encryption Key** is not set, following error is displayed while creating a backup configuration file:

```
Backup/Export operation requires Password Encryption Key to be set, please
refer to Cisco UCS Manager Administration Guide to set the Password
Encryption key.
```

You cannot import a backup configuration file created from Cisco UCS Manager release 4.2(3d) and later into an earlier release. But, you can import a backup configuration file created from an earlier release to Cisco UCS Manager release 4.2(3d) and later with or without a **Password Encryption Key**.

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.

Creating a Backup Operation

Before you begin

Obtain the backup server IPv4 or IPv6 address and authentication credentials.

Beginning with release 4.2(3d), Cisco UCS Manager introduces **Password Encryption Key** to enhance security for backup configuration files.

You must set **Password Encryption Key** in order to create backup configuration files and also to import the backup files. Cisco UCS Manager release 4.2(3d) and later do not allow you to create backup configuration files or import backup configuration files without setting the **Password Encryption Key**. If the **Password Encryption Key** is not set, following error is displayed while creating a backup configuration file:

```
Backup/Export operation requires Password Encryption Key to be set, please
refer to Cisco UCS Manager Administration Guide to set the Password
Encryption key.
```

You cannot import a backup configuration file created from Cisco UCS Manager release 4.2(3d) and later into an earlier release. But, you can import a backup configuration file created from an earlier release to Cisco UCS Manager release 4.2(3d) and later with or without a Password Encryption Key.

For more information on how to set **Password Encryption Key**, see Creating Password Encryption Key, on page 16.

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:**// *username@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:

	Command or Action	Purpose
		• **all-configuration** —Backs up the server-, fabric-, and system-related configuration • **logical-configuration** —Backs up the fabric- and service profile-related configuration • **system-configuration** —Backs up the system-related configuration • **full-state** —Backs up the full state for disaster recovery **Note** • Full-state backup files cannot be imported using an import operation. They are used only to restore the configuration for a fabric interconnect. • You can only use a full state backup file to restore a system that is running the same version as the system from which the backup file was exported. You can save multiple backup operations, but only one operation for each hostname is saved. If you use the **enable** keyword, the backup operation automatically runs as soon as you enter the **commit-buffer** command. If you use the **disable** keyword, the backup operation will not 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.

Example

The following example shows how to create a disabled all-configuration backup operation for hostname host35 and commit the transaction:

```
UCS-A# scope system
UCS-A /system* # create backup scp://user@host35/backups/all-config9.bak all-configuration
 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. **Note** For backup operations using FTP, SCP, SFTP, you are prompted for the password. Enter the password before committing the transaction.
Step 4	UCS-A /system/backup # **commit-buffer**	Commits the transaction.

Example

The following example enables a backup operation named host35, enters the password for the SCP protocol, and commits the transaction:

```
UCS-A# scope system
UCS-A /system # scope backup host35
UCS-A /system/backup # enable
Password:
UCS-A /system/backup* # commit-buffer
UCS-A /system/backup #
```

Modifying a Backup Operation

You can modify a backup operation to save a file of another backup type to that location or to change the filename and avoid overwriting previous backup files.

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	(Optional) UCS-A /system/backup # **disable**	Disables an enabled backup operation so that it does not automatically run when the transaction is committed.

	Command or Action	Purpose
Step 4	(Optional) UCS-A /system/backup # **enable**	Automatically runs the backup operation as soon as you commit the transaction.
Step 5	(Optional) UCS-A /system/backup # **set descr** *description*	Provides a description for the backup operation. **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	(Optional) UCS-A /system/backup # **set protocol** {**ftp** | **scp** | **sftp** | **tftp**}	Specifies the protocol to use when communicating with the remote server.
Step 7	(Optional) UCS-A /system/backup # **set remote-file** *filename*	Specifies the name of the configuration file that is being backed up.
Step 8	(Optional) UCS-A /system/backup # **set type** *backup-type*	Specifies the type of backup file to be made. The *backup-type* argument can be one of the following values: • **all-configuration** —Backs up the server, fabric, and system related configuration • **logical-configuration** —Backs up the fabric and service profile related configuration • **system-configuration** —Backs up the system related configuration • **full-state** —Backs up the full state for disaster recovery **Note** • Full-state backup files cannot be imported using an import operation. They are used only to restore the configuration for a fabric interconnect. • You can only use a full state backup file to restore a system that is running the same version as the system from which the backup file was exported.

	Command or Action	Purpose
Step 9	(Optional) UCS-A /system/backup # **set preserve-pooled-values** {**no** | **yes**}	Specifies whether pool-derived identity values, such as vHBA WWPN, vNIC MAC, WWNN, and UUID, will be saved with the backup.
Step 10	(Optional) UCS-A /system/backup # **set user** *username*	Specifies the username the system should use to log in to the remote server. This step does not apply if the TFTP protocol is used.
Step 11	(Optional) UCS-A /system/backup # **set password**	After you press **Enter**, you are prompted to enter the password. Specifies the password for the remote server username. This step does not apply if the TFTP protocol is used.
Step 12	UCS-A /system/backup # **commit-buffer**	Commits the transaction.

Example

The following example adds a description and changes the protocol, username, and password for the host35 backup operation and commits the transaction:

```
UCS-A# scope system
UCS-A /system # scope backup host35
UCS-A /system/backup # set descr "This is a backup operation for host35."
UCS-A /system/backup* # set protocol sftp
UCS-A /system/backup* # set user UserName32
UCS-A /system/backup* # set password
Password:
UCS-A /system/backup* # set preserve-pooled-values no
UCS-A /system/backup* # commit-buffer
UCS-A /system #
```

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.

Example

The following example deletes a backup operation for the host35 hostname and commits the transaction:

```
UCS-A# scope system
UCS-A /system # delete backup host35
UCS-A /system* # commit-buffer
UCS-A /system #
```

Scheduled Backups

You can configure policies in Cisco UCS to schedule the following types of backups:

- Full state
- All configuration

You cannot schedule any other type of backup.

Backup Types

You can perform one or more of the following types of backups in Cisco UCS Manager and Cisco UCS Central:

- **Full state**—A binary file that includes a snapshot of the entire system. You can use the file generated from this backup to restore the system during disaster recovery. This file can restore or rebuild the configuration on the original fabric interconnect, or recreate the configuration on a different fabric interconnect. You cannot use this file for an import.

Note You can only use a full state backup file to restore a system that is running the same version as the system from which the backup file was exported. It is also important that the bundles from which the backup was taken remain present in the Cisco UCS Manager and should not be deleted.

- **All configuration**—An XML file that includes all system and logical configuration settings. You can use the file generated from this backup to import these configuration settings to the original fabric interconnect or to a different fabric interconnect. You cannot use this file for a system restore. This file does not include passwords for locally authenticated users.
- **System configuration**—An XML file that includes all system configuration settings such as usernames, roles, and locales. You can use the file generated from this backup to import these configuration settings to the original fabric interconnect or to a different fabric interconnect. You cannot use this file for a system restore.
- **Logical configuration**—An XML file that includes all logical configuration settings such as service profiles, VLANs, VSANs, pools, and policies. You can use the file generated from this backup to import these configuration settings to the original fabric interconnect or to a different fabric interconnect. You cannot use this file for a system restore.

Full State Backup Policy

The full state backup policy allows you to schedule regular full state backups of a snapshot of the entire system. You can choose whether to configure the full state backup to occur on a daily, weekly, or biweekly basis.

Cisco UCS Manager maintains a maximum number of backup files on the remote server. The maxfiles parameter is used when Cisco UCS Manager is registered with Cisco UCS Central. The maxfiles parameter is user configurable on Cisco UCS Central and controls the number of backup files stored on Cisco UCS Central.

If Cisco UCS Manager is not registered with Cisco UCS Central, and the user is storing backup files on a remote backup server, the backup files are not managed by Cisco UCS Manager. The remote machine server administrator must monitor the disk usage and rotate the backup files to create space for new backup files.

Configuring the Full State Backup Policy

Before you begin

Obtain the backup server IPv4 or IPv6 address and authentication credentials.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter **/** as the *org-name*.
Step 2	UCS-A /org # **scope backup-policy default**	Enters the all configuration export policy mode.
Step 3	UCS-A /org/backup-policy # **set hostname** {*hostname* | *ip-addr* | *ip6-addr*}	Specifies the hostname, IPv4 or IPv6 address of the location where the backup policy is stored. This can be a server, storage array, local drive, or any read/write media that the fabric interconnect can access through the network. **Note** If you use a hostname rather than an IPv4 or IPv6 address, you must configure a DNS server. If the Cisco UCS domain is not registered with Cisco UCS Central or DNS management is set to **local**, configure a DNS server in Cisco UCS Manager. If the Cisco UCS domain is registered with Cisco UCS Central and DNS management is set to **global**, configure a DNS server in Cisco UCS Central.
Step 4	UCS-A /org/backup-policy # **set protocol** {**ftp** | **scp** | **sftp** | **tftp**}	Specifies the protocol to use when communicating with the remote server.

	Command or Action	Purpose
Step 5	UCS-A /org/backup-policy # **set user** *username*	Specifies the username the system should use to log in to the remote server. This step does not apply if the TFTP protocol is used.
Step 6	UCS-A /system/backup-policy # **set password**	After you press **Enter**, you are prompted to enter the password. Specifies the password for the remote server username. This step does not apply if the TFTP protocol is used.
Step 7	UCS-A /system/backup-policy # **set remote-file** *filename*	Specifies the full path to the backup file. This field can contain the filename as well as the path. If you omit the filename, the backup procedure assigns a name to the file.
Step 8	UCS-A /system/backup-policy # **set adminstate** {**disabled** | **enabled**}	Specifies the admin state for the policy. This can be one of the following: • **enabled**—Cisco UCS Manager exports the backup file using the schedule specified in the **Schedule** field. • **disabled**—Cisco UCS Manager does not export the file.
Step 9	UCS-A /system/backup-policy # **set schedule** {**daily** | **weekly** | **bi-weekly**}	Specifies the frequency with which Cisco UCS Manager exports the backup file.
Step 10	UCS-A /system/backup-policy # **set descr** *description*	Specifies a description for the backup policy. Enter up to 256 characters. You can use any characters or spaces except ` (accent mark), \ (backslash), ^ (carat), " (double quote), = (equal sign), > (greater than), < (less than), or ' (single quote).
Step 11	UCS-A /backup-policy # **commit-buffer**	Commits the transaction.

Example

The following example shows how to configure the full state backup policy for a weekly backup and commit the transaction:

```
UCS-A# scope org /
UCS-A /org # scope backup-policy default
UCS-A /org/backup-policy # set hostname host35
UCS-A /org/backup-policy* # set protocol scp
UCS-A /org/backup-policy* # set user UserName32
UCS-A /backup-policy* # set password
Password:
UCS-A /backup-policy* # set remote-file /backups/full-state1.bak
UCS-A /backup-policy* # set adminstate enabled
```

```
UCS-A /backup-policy* # set schedule weekly
UCS-A /backup-policy* # set descr "This is a full state weekly backup."
UCS-A /backup-policy* # commit-buffer
UCS-A /backup-policy #
```

Configuring the All Configuration Export Policy

Before you begin

Obtain the backup server IPv4 or IPv6 address and authentication credentials.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter **/** as the *org-name*.
Step 2	UCS-A /org # **scope cfg-export-policy default**	Enters the all configuration export policy mode.
Step 3	UCS-A /org/cfg-export-policy # **set hostname** {*hostname* | *ip-addr* | *ip6-addr*}	Specifies the hostname, IPv4 or IPv6 address of the location where the configuration file is stored. This can be a server, storage array, local drive, or any read/write media that the fabric interconnect can access through the network. **Note** If you use a hostname rather than an IPv4 or IPv6 address, you must configure a DNS server. If the Cisco UCS domain is not registered with Cisco UCS Central or DNS management is set to **local**, configure a DNS server in Cisco UCS Manager. If the Cisco UCS domain is registered with Cisco UCS Central and DNS management is set to **global**, configure a DNS server in Cisco UCS Central.
Step 4	UCS-A /org/cfg-export-policy # **set protocol** {**ftp** | **scp** | **sftp** | **tftp**}	Specifies the protocol to use when communicating with the remote server.
Step 5	UCS-A /org/cfg-export-policy # **set user** *username*	Specifies the username the system should use to log in to the remote server. This step does not apply if the TFTP protocol is used.
Step 6	UCS-A /system/cfg-export-policy # **set password**	After you press **Enter**, you are prompted to enter the password. Specifies the password for the remote server username. This step does not apply if the TFTP protocol is used.

	Command or Action	Purpose
Step 7	UCS-A /system/cfg-export-policy # **set remote-file** *filename*	Specifies the full path to the exported configuration file. This field can contain the filename as well as the path. If you omit the filename, the backup procedure assigns a name to the file.
Step 8	UCS-A /system/cfg-export-policy # **set adminstate** {**disabled** | **enabled**}	Specifies the admin state for the policy. This can be one of the following: • **enabled**—Cisco UCS Manager exports the configuration information using the schedule specified in the **Schedule** field. • **disabled**—Cisco UCS Manager does not export the information.
Step 9	UCS-A /system/cfg-export-policy # **set schedule** {**daily** | **weekly** | **bi-weekly**}	Specifies the frequency with which Cisco UCS Manager exports the configuration information.
Step 10	UCS-A /system/cfg-export-policy # **set descr** *description*	Specifies a description for the configuration export policy. Enter up to 256 characters. You can use any characters or spaces except ` (accent mark), \ (backslash), ^ (carat), " (double quote), = (equal sign), > (greater than), < (less than), or ' (single quote).
Step 11	UCS-A /cfg-export-policy # **commit-buffer**	Commits the transaction.

Example

The following example shows how to configure the all configuration export policy for a weekly backup and commit the transaction:

```
UCS-A# scope org /
UCS-A /org # scope cfg-export-policy default
UCS-A /org/cfg-export-policy # set hostname host35
UCS-A /org/cfg-export-policy* # set protocol scp
UCS-A /org/cfg-export-policy* # set user UserName32
UCS-A /cfg-export-policy* # set password
Password:
UCS-A /cfg-export-policy* # set remote-file /backups/all-config9.bak
UCS-A /cfg-export-policy* # set adminstate enabled
UCS-A /cfg-export-policy* # set schedule weekly
UCS-A /cfg-export-policy* # set descr "This is an all configuration backup."
UCS-A /cfg-export-policy* # commit-buffer
UCS-A /cfg-export-policy #
```

All Configuration Export Policy

The all configuration backup policy allows you to schedule a regular backup and export of all system and logical configuration settings. This backup does not include passwords for locally authenticated users. You can choose whether to configure the all configuration backup to occur on a daily, weekly, or bi-weekly basis.

Cisco UCS maintains a maximum number of backup files on the remote server. When that number is exceeded, Cisco UCS overwrites the oldest backup file.

Configuring Backup/Export Configuration Reminders

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters the organization mode for the specified organization. To enter the root organization mode, enter **/** as the *org-name*.
Step 2	UCS-A /org # **scope backup-exp-policy**	Enters the backup/export configuration policy mode.
Step 3	UCS-A /org/backup-exp-policy # **show**	Displays the existing backup/export configuration policy.
Step 4	UCS-A /org/backup-exp-policy # **set adminstate** {**disable** | **enable**}	Specifies the admin state for the policy. This can be one of the following: • **enable**—Cisco UCS Manager raises a fault if a backup is not taken during the specified time period. • **disable**—Cisco UCS Manager does not raise a fault if a backup is not taken during the specified time period.
Step 5	UCS-A /org/backup-exp-policy # **set frequency** *Number_of_Days*	Specifies the number of days before you are reminded to take a backup. Enter an integer between 1 and 365. The default value is 30 days.
Step 6	UCS-A /org/backup-exp-policy # **commit-buffer**	Commits the transaction.

Example

The following example shows how to view the current backup/export config policy, change the frequency of the reminders, and commit the transaction:

```
UCS-A# scope org /
UCS-A /org # scope backup-exp-policy
UCS-A /org/backup-exp-policy # set frequency 5
```

```
UCS-A /org/backup-exp-policy* # commit-buffer
UCS-A /org/backup-exp-policy #
```

Import Operations

Import Methods

You can use one of the following methods to import and update a system configuration through Cisco UCS:

- **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 domain 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.

Import Configuration

You can import any configuration file that was exported from Cisco UCS. The file does not need to have been exported from the same Cisco UCS.

Note You cannot import configuration from a higher release to a lower release.

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 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.

Creating an Import Operation

You cannot import a Full State backup file. You can import any of the following configuration files:

- All configuration
- System configuration
- Logical configuration

Before you begin

Collect the following information to import a configuration file:

- Backup server IP address and authentication credentials
- Fully-qualified name of a backup file

Beginning with release 4.2(3d), Cisco UCS Manager introduces **Password Encryption Key** to enhance security for backup configuration files.

You must set **Password Encryption Key** in order to create backup configuration files and also to import the backup files. Cisco UCS Manager release 4.2(3d) and later do not allow you to create backup configuration files or import backup configuration files without setting the **Password Encryption Key**. If the **Password Encryption Key** is not set, following error is displayed while creating a backup configuration file:

```
Backup/Export operation requires Password Encryption Key to be set, please
refer to Cisco UCS Manager Administration Guide to set the Password
Encryption key.
```

You cannot import a backup configuration file created from Cisco UCS Manager release 4.2(3d) and later into an earlier release. But, you can import a backup configuration file created from an earlier release to Cisco UCS Manager release 4.2(3d) and later with or without a Password Encryption Key.

For more information on how to set **Password Encryption Key**, see Creating Password Encryption Key, on page 16.

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:**// *username***@***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 for each hostname is saved. If you use the **enable** keyword, the import operation automatically runs as soon as you enter the **commit-buffer** command. If you use the **disable** keyword, the import operation will not run until it is enabled. When enabling an import operation, you must specify the hostname you used when creating the import operation. If you use the **merge** keyword, the configuration information is merged with the existing information. If there are conflicts, the

	Command or Action	Purpose
		system replaces the information on the current system with the information in the import configuration file. If you use the **replace** keyword, the system takes each object in the import configuration file and overwrites the corresponding object in the current configuration.
Step 3	(Optional) UCS-A /system/import-config# **set descr** *description*	Provides a description for the import operation. **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 /system/import-config # **commit-buffer**	Commits the transaction.

Example

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/all-config9.bak 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 backup file. You can import any of the following configuration files:

- All configuration
- System configuration
- Logical configuration

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.

	Command or Action	Purpose
Step 3	UCS-A /system/import-config # **enable**	Enables the import operation.
Step 4	UCS-A /system/import-config # **commit-buffer**	Commits the transaction.

Example

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 host35
UCS-A /system/import-config # enable
UCS-A /system/import-config* # commit-buffer
UCS-A /system/import-config #
```

Modifying an Import Operation

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 import configuration mode for the specified hostname.
Step 3	(Optional) UCS-A /system/import-config # **disable**	Disables an enabled import operation so that it does not automatically run when the transaction is committed.
Step 4	(Optional) UCS-A /system/import-config # **enable**	Automatically runs the import operation as soon as you commit the transaction.
Step 5	(Optional) UCS-A /system/import-config # **set action** {**merge** | **replace**}	Specifies one of the following action types to use for the import operation: • **Merge** —The configuration information is merged with the existing information. If there are conflicts, the system replaces the information on the current system with the information in the import configuration file. • **Replace** —The system takes each object in the import configuration file and overwrites the corresponding object in the current configuration.

	Command or Action	Purpose
Step 6	(Optional) UCS-A /system/import-config # **set descr** *description*	Provides a description for the import operation. **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 7	(Optional) UCS-A /system/import-config # **set password**	After you press **Enter**, you are prompted to enter the password. Specifies the password for the remote server username. This step does not apply if the TFTP protocol is used. **Note** Cisco UCS Manager does not store this password. Therefore, you do not need to enter this password unless you intend to enable and run the import operation immediately.
Step 8	(Optional) UCS-A /system/import-config # **set protocol** {**ftp** | **scp** | **sftp** | **tftp**}	Specifies the protocol to use when communicating with the remote server.
Step 9	(Optional) UCS-A /system/import-config # **set remote-file** *filename*	Specifies the name of the configuration file that is being imported.
Step 10	(Optional) UCS-A /system/import-config # **set user** *username*	Specifies the username the system should use to log in to the remote server. This step does not apply if the TFTP protocol is used.
Step 11	UCS-A /system/import-config # **commit-buffer**	Commits the transaction.

Example

The following example adds a description, changes the password, protocol and username for the host35 import operation, and commits the transaction:

```
UCS-A# scope system
UCS-A /system # scope import-config host35
UCS-A /system/import-config # set descr "This is an import operation for host35."
UCS-A /system/import-config* # set password
Password:
UCS-A /system/import-config* # set protocol sftp
UCS-A /system/import-config* # set user jforlenz32
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.

Example

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 #
```

System Restore

You can use the restore function for disaster recovery.

You can restore a system configuration from any full state backup file that was exported from Cisco UCS. The file does not need to have been exported from Cisco UCS on the system that you are restoring. When restoring using a backup file that was exported from a different system, we recommend that you use a system with the same or similar system configuration and hardware, including fabric interconnects, servers, adapters, and I/O module or FEX connectivity. Mismatched hardware and system configuration can lead to the restored system not fully functioning. If there is a mismatch between the I/O module links or servers on the two systems, acknowledge the chassis and servers after the restore operation.

- Chassis Discovery Policy and Chassis Connectivity Policy are in non port channel mode
- Virtual Machine Management is enabled - VMware, Linux KVM, or Microsoft Hypervisor

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. For more information, see the appropriate *Cisco UCS Central Installation and Upgrade Guide*.

Note You can only use a full state backup file to restore a system that is running the same version as the system from which the backup file was exported. It is also important that the bundles from which the backup was taken remain present in the Cisco UCS Manager and should not be deleted.

Restoring the Configuration for a Fabric Interconnect

It is recommended to use a full state backup file to restore a system running the same version as the source system from which the backup was exported. You can use a full state backup to restore a system within the same release train. For example, you can use a backup from a system running on Cisco UCS Manager 4.1(3b) release version to restore a system on release 4.1(3m). It is also important to ensure that the bundles from which the backup was taken remain present in Cisco UCS Manager and are not deleted.

To avoid issues with VSAN or VLAN configuration, a backup should be restored on the fabric interconnect that was the primary fabric interconnect at the time of backup.

Beginning with release 4.2(3d), Cisco UCS Manager introduces **Password Decryption Key** to enhance security for backup configuration files.

Password Decryption Key should be same as mentioned in **Password Encryption Key** while creating the backup configuration file. Same key is set as **Password Encryption Key** after successful restore.

Note For release 4.2(3d) and later, you can perform this procedure only with a backup configuration file created from release 4.2(3d) or later.

Before you begin

Collect the following information to restore the system configuration:

- Fabric interconnect management port IPv4 address and subnet mask, or IPv6 address and prefix
- Default gateway IPv4 or IPv6 address
- Backup server IPv4 or IPv6 address and authentication credentials
- Fully-qualified name of a Full State backup file

Note You must have access to a Full State configuration file to perform a system restore. You cannot perform a system restore with any other type of configuration or backup file.

Procedure

Step 1 Connect to the console port.

Step 2 If the fabric interconnect is off, power on the fabric interconnect.

You will see the power on self-test message as the fabric interconnect boots.

Step 3 At the installation method prompt, enter **console** .

Step 4 Select **UCSM** as management mode.

Step 5 Enter **restore** to restore the configuration from a full-state backup.

Step 6 Enter **y** to confirm that you want to restore from a full-state backup.

Step 7 Enter the IP address for the management port on the fabric interconnect.

Step 8 Enter the subnet mask for the management port on the fabric interconnect.

Step 9 Enter the IP address for the default gateway.

Step 10 Enter one of the following protocols to use when retrieving the backup configuration file:

- **scp**
- **ftp**
- **tftp**
- **sftp**

Step 11 Enter the IP address of the backup server.

Step 12 Enter the key for decrypting the backup file.

Step 13 Enter the full path and filename of the Full State backup file.

Note
You can only use a full state backup file to restore a system that is running the same version as the system from which the backup file was exported. It is also important that the bundles from which the backup was taken remain present in the Cisco UCS Manager and should not be deleted.

Step 14 Enter the username and password to access the backup server.

The fabric interconnect logs in to the backup server, retrieves a copy of the specified Full State backup file, and restores the system configuration. For a cluster configuration, you do not need to restore the secondary fabric interconnect. As soon as the secondary fabric interconnect reboots, Cisco UCS synchronizes the configuration with the primary fabric interconnect.

Example

The following example restores a system configuration from the Backup.bak file, which was retrieved from the 20.10.20.10 backup server using FTP:

```
  Enter the configuration method. (console/gui) ? console
Enter the management mode. (ucsm/intersight)? ucsm

  Enter the setup mode; setup newly or restore from backup. (setup/restore) ? restore

  NOTE:
    To configure Fabric interconnect using a backup file on a remote server,
    you will need to setup management interface.
    The management interface will be re-configured (if necessary),
    based on information stored in the backup file.


  Continue to restore this Fabric interconnect from a backup file (yes/no) ? yes

  Physical Switch Mgmt0 IPv4 address : 192.168.10.10

  Physical Switch Mgmt0 IPv4 netmask : 255.255.255.0

  IPv4 address of the default gateway : 192.168.10.1
```

```
  Enter the protocol to get backup file (scp/ftp/tftp/sftp) ?  scp
  Enter the IP address of backup server: 20.10.20.10
  Enter the key for decrypting the backup file: File Decryption Key
  Enter fully qualified backup file name: Backup.bak
  Enter user ID: user
  Enter password:
    Retrieved backup configuration file.
  Configuration file - Ok


Cisco UCS 6100 Series Fabric Interconnect
UCS-A login:
```

Erasing the Configuration

Caution You should erase the configuration only when it is necessary. Erasing the configuration completely removes the configuration and reboots the system in an unconfigured state. You must then either restore the configuration from a backup file or perform an initial system setup.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **connect local-mgmt**	Enters the local management CLI.
Step 2	UCS-A(local-mgmt)# **erase configuration**	Erases the configuration. You are prompted to confirm that you want to erase the configuration. Entering **yes** erases the configuration and reboots the system in an unconfigured state.

Example

The following example erases the configuration:

```
UCS-A# connect local-mgmt
UCS-A(local-mgmt)# erase configuration
All UCS configurations will be erased and system will reboot. Are you sure? (yes/no): yes
```

Fabric Interconnect Secure Erase (FI Secure Erase)

Secure erase is a comprehensive operation that erases all persistent storage on the Fabric Interconnect, including configuration, log data, and the full contents of flash and SSDs.

Before you begin

Caution Erasing the fabric interconnect data using secure erase command is a comprehensive operation that allows administrators to securely and permanently delete all data and configurations on supported Fabric Interconnects, ensuring customer data privacy and eliminating the possibility of data recovery. Before performing secure erase, ensure all necessary data backups are completed and you have verified the need for this action. This operation is irreversible; all data and configuration will be permanently deleted and cannot be recovered.

The secure erase is supported on Cisco UCS 6400, 6500, 6600 Series, and X-Series Direct Fabric Interconnects.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **connect local-mgmt**	Enters the local management CLI.
Step 2	UCS-A(local-mgmt)# **fi-secure-erase**	Securely erases the fabric interconnect data and configuration. You are prompted to confirm that you want to erase the configuration. Entering **yes** erases the configuration and reboots the fabric interconnect in an unconfigured state. `The Secure Erase operation will erase ALL persistent storage on the fabric interconnect. This includes configuration, all log data, and the full contents of flash and SSDs. Special steps are taken in an effort to render data non-recoverable. Please, proceed with caution and understanding that this operation cannot be undone and will leave the system in a fresh-from-factory state. (yes/no):yes ... [System reboots, and all config is gone]`

Step 3	*(Optional)*UCS-A(local-mgmt)# **fi-secure-erase preserve-Image**	This is an optional step that securely erases the fabric interconnect data and configuration while preserving the system image.
		You are prompted to confirm that you want to erase the configuration with preserve image. Entering **yes** erases the configuration and reboots the fabric interconnect in an unconfigured state.
		`!!!! WARNING !!!!`
		`The Secure Erase operation will cause a reboot & erase ALL persistent storage on the Fabric interconnect. This includes configuration, all`

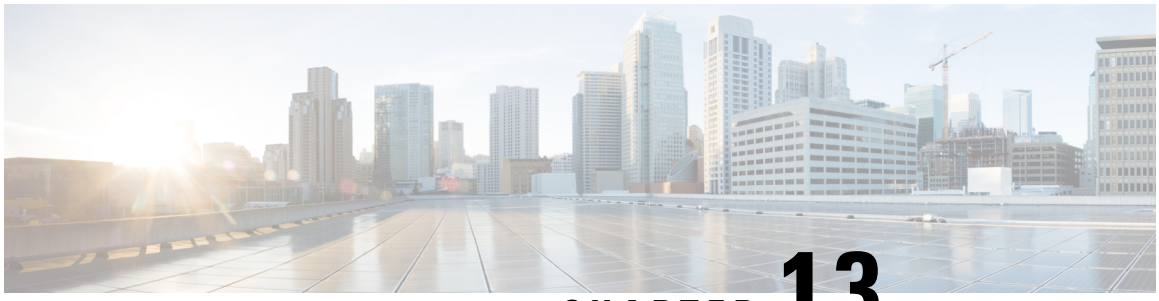
	Command or Action	Purpose
		log data, and the full contents of flash and SSDs. Special steps are taken in an effort to render data non-recoverable. Please, proceed with caution and understanding that this operation cannot be undone and will leave the system in a fresh-from-factory state. (yes/no):yes ... [System reboots, UCSM and NX-OS images remain on bootflash] !!!! WARNING !!!! ...The 'preserveImage' flag is "selected" hence system images will be intact... Are you sure? Enter 'yes' to continue [no]: yes
Step 4	Confirm the operation when prompted. A warning message will appear, emphasizing that the data cannot be recovered once erased.	
Step 5	Wait for the process to complete.	The Fabric Interconnect will automatically reboot after the operation.

Example

The following example erases the fabric interconnect configuration:

```
UCS-A# connect local-mgmt
UCS-A(local-mgmt)# fi-secure-erase
The Secure Erase operation will cause a reboot & erase ALL persistent storage on the Fabric interconnect.
This includes configuration, all log data, and the full contents of flash and SSDs.
Special steps are taken in an effort to render data non-recoverable. Please, proceed
with caution and understanding that this operation cannot be undone and will leave the
system ina fresh-from-factory state.

(yes/no): yes
```



CHAPTER 13

Scheduling Options

Deployment Scheduling Options

Creating a Schedule

Procedure

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

Example

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

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

What to do next

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

Creating a One Time Occurrence for a Schedule

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system**	Enters system mode.
Step 2	UCS-A /system # **scope schedule** *sched-name*	Enters scheduler system mode.
Step 3	UCS-A /system/scheduler # **create occurrence one-time** *occurrence-name*	Creates a one-time occurrence.
Step 4	UCS-A /system/scheduler/one-time # **set date** *month day-of-month year hour minute*	Sets the date and time this occurrence should run.
Step 5	(Optional) UCS-A /system/scheduler/one-time # **set concur-tasks** {**unlimited** | *max-num-concur-tasks*	Sets the maximum number of tasks that can run concurrently during this occurrence. If the maximum number of tasks is reached, the scheduler waits for the amount of time set in the minimum interval property before scheduling new tasks.
Step 6	(Optional) UCS-A /system/scheduler/one-time # **set max-duration** {**none** | *num-of-days num-of-hours num-of-minutes num-of-seconds*}	Sets the maximum length of time that this schedule occurrence can run. Cisco UCS completes as many scheduled tasks as possible within the specified time.
Step 7	(Optional) UCS-A /system/scheduler/one-time # **set min-interval** {**none** | *num-of-days num-of-hours num-of-minutes num-of-seconds*}	Sets the minimum length of time that the system should wait before starting a new task.
Step 8	(Optional) UCS-A /system/scheduler/one-time # **set proc-cap** {**unlimited** | *max-num-of-tasks*}	Sets the maximum number of scheduled tasks that can be run during this occurrence.
Step 9	UCS-A /system/scheduler/one-time # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example creates a one time occurrence named onetimemaint for a scheduler named maintsched, sets the maximum number of concurrent tasks to 5, sets the start date to April 1, 2011 at 11:00, and commits the transaction:

```
UCS-A# scope system
UCS-A /system # scope scheduler maintsched
UCS-A /system/scheduler # create occurrence one-time onetimemaint
UCS-A /system/scheduler/one-time* # set date apr 1 2011 11 00
UCS-A /system/scheduler/one-time* # set concur-tasks 5
```

```
UCS-A /system/scheduler/one-time* # commit-buffer
UCS-A /system/scheduler/one-time #
```

Creating a Recurring Occurrence for a Schedule

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system**	Enters system mode.
Step 2	UCS-A /system # **scope schedule** *sched-name*	Enters scheduler system mode.
Step 3	UCS-A /system/scheduler # **create occurrence recurring** *occurrence-name*	Creates a recurring occurrence.
Step 4	(Optional) UCS-A /system/scheduler/recurring # **set day** {**even-day** | **every-day** | **friday** | **monday** | **never** | **odd-day** | **saturday** | **sunday** | **thursday** | **tuesday** | **wednesday**}	Specifies the day on which Cisco UCS runs an occurrence of this schedule. By default, this property is set to never.
Step 5	(Optional) UCS-A /system/scheduler/recurring # **set hour** *hour*	Specifies the hour at which this occurrence starts. **Note** Cisco UCS ends all recurring occurrences on the same day in which they start, even if the maximum duration has not been reached. For example, if you specify a start time of 11 p.m. and a maximum duration of 3 hours, Cisco UCS starts the occurrence at 11 p.m. but ends it at 11:59 p.m. after only 59 minutes.
Step 6	(Optional) UCS-A /system/scheduler/recurring # **set minute** *minute*	Specifies the minute at which this occurrence starts.
Step 7	(Optional) UCS-A /system/scheduler/recurring # **set concur-tasks** {**unlimited** | *max-num-concur-tasks*	Sets the maximum number of tasks that can run concurrently during this occurrence. If the maximum number of tasks is reached, the scheduler waits for the amount of time set in the minimum interval property before scheduling new tasks.
Step 8	(Optional) UCS-A /system/scheduler/recurring # **set max-duration** {**none** | *num-of-hours num-of-minutes num-of-seconds*}	Sets the maximum length of time that this schedule occurrence can run. Cisco UCS completes as many scheduled tasks as possible within the specified time.

	Command or Action	Purpose
Step 9	(Optional) UCS-A /system/scheduler/recurring # **set min-interval** {**none** | *num-of-days num-of-hours num-of-minutes num-of-seconds*}	Sets the minimum length of time that the system should wait before starting a new task.
Step 10	(Optional) UCS-A /system/scheduler/recurring # **set proc-cap** {**unlimited** | *max-num-of-tasks*}	Sets the maximum number of scheduled tasks that can be run during this occurrence.
Step 11	UCS-A /system/scheduler/recurring # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example creates a recurring occurrence called recurringmaint for a scheduler called maintsched, sets the maximum number of concurrent tasks to 5, sets the day this occurrence will run to even days, sets the time it will start to 11:05, and commits the transaction:

```
UCS-A# scope system
UCS-A /system # scope scheduler maintsched
UCS-A /system/scheduler # create occurrence recurring recurringmaint
UCS-A /system/scheduler/recurring* # set day even-day
UCS-A /system/scheduler/recurring* # set hour 11
UCS-A /system/scheduler/recurring* # set minute 5
UCS-A /system/scheduler/recurring* # set concur-tasks 5
UCS-A /system/scheduler/recurring* # commit-buffer
UCS-A /system/scheduler/recurring #
```

Deleting a One Time Occurrence from a Schedule

If this is the only occurrence in a schedule, that schedule is reconfigured with no occurrences. If the schedule is included in a maintenance policy and that policy is assigned to a service profile, any pending activities related to the server associated with the service profile cannot be deployed. You must add a one time occurrence or a recurring occurrence to the schedule to deploy the pending activity.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system**	Enters system mode.
Step 2	UCS-A /system # **scope scheduler** *sched-name*	Enters scheduler system mode.
Step 3	UCS-A /system/scheduler # **delete occurrence one-time** *occurrence-name*	Deletes the specified one-time occurrence.
Step 4	UCS-A /system/scheduler # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes a one time occurrence called onetimemaint from scheduler maintsched and commits the transaction:

```
UCS-A# scope system
UCS-A /system # scope scheduler maintsched
UCS-A /system/scheduler # delete occurrence one-time onetimemaint
UCS-A /system/scheduler* # commit-buffer
UCS-A /system/scheduler #
```

Deleting a Recurring Occurrence from a Schedule

If this is the only occurrence in a schedule, that schedule is reconfigured with no occurrences. If the schedule is included in a maintenance policy and that policy is assigned to a service profile, any pending activities related to the server associated with the service profile cannot be deployed. You must add a one time occurrence or a recurring occurrence to the schedule to deploy the pending activity.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope system**	Enters system mode.
Step 2	UCS-A /system # **scope scheduler** *sched-name*	Enters scheduler system mode.
Step 3	UCS-A /system/scheduler # **delete occurrence recurring** *occurrence-name*	Deletes the specified recurring occurrence.
Step 4	UCS-A /system/scheduler # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example deletes a recurring occurrence named onetimemaint from scheduler maintsched and commits the transaction:

```
UCS-A# scope system
UCS-A /system # scope scheduler maintsched
UCS-A /system/scheduler # delete occurrence recurring onetimemaint
UCS-A /system/scheduler* # commit-buffer
UCS-A /system/scheduler #
```

Deleting a Schedule

If this schedule is included in a maintenance policy, the policy is reconfigured with no schedule. If that policy is assigned to a service profile, any pending activities related to the server associated with the service profile cannot be deployed. You must add a schedule to the maintenance policy to deploy the pending activity.

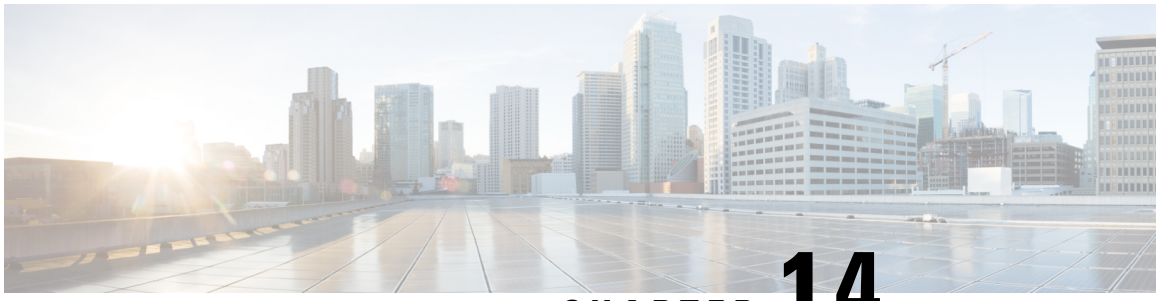
Procedure

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

Example

The following example deletes a scheduler named maintenancesched and commits the transaction:

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



CHAPTER 14

Deferred Deployments of Service Profile Updates

Service Profile Deferred Deployments

Some modifications to a service profile or to an updating service profile template can be disruptive and require a reboot of the server. You can, however, configure deferred deployment to control when those disruptive configuration changes are implemented. For example, you can choose to deploy the service profile changes immediately or have them deployed during a specified maintenance window. You can also choose whether or not a service profile deployment requires explicit user acknowledgment.

Deferred deployment is available for all configuration changes that occur through the association of a service profile with a server. These configuration changes can be prompted by a change to a service profile, to a policy that is included in a service profile, or to an updating service profile template. For example, you can defer the upgrade and activation of firmware through host firmware packages and management firmware packages, such as server BIOS, RAID controller, host HBA, and network adapters. However, you cannot defer the direct deployment of firmware images for components that do not use either of the firmware packages, such as Cisco UCS Manager, fabric interconnects, I/O modules, and FI-IO modules..

Deferred deployment is not available for the following actions which require the reboot of a server:

- Initial association of a service profile with a server
- Final disassociation of a service profile from a server, without associating the service profile with a different server
- Decommissioning a server
- Re-acknowledging a server
- Resetting a server

If you want to defer the deployment of service profile changes, you must configure one or more maintenance policies and configure each service profile with a maintenance policy. If you want to define the time period when the deployment should occur, you also need to create at least one schedule with one or more recurring occurrences or one time occurrences, and include that schedule in a maintenance policy.

Note The Cisco UCS X-Series Direct does not support I/O Module operations; it utilizes the FI-I/O Module instead.

Schedules for Deferred Deployments

A schedule contains a set of occurrences. These occurrences can be one time only or can recur at a specified time and day each week. The options defined in the occurrence, such as the duration of the occurrence or the maximum number of tasks to be run, determine whether a service profile change is deployed. For example, if a change cannot be deployed during a given maintenance window because the maximum duration or number of tasks was reached, that deployment is carried over to the next maintenance window.

Each schedule checks periodically to see whether the Cisco UCS domain entered one or more maintenance windows. If so, the schedule executes the deployments that are eligible according to the constraints specified in the maintenance policy.

A schedule contains one or more occurrences, which determine the maintenance windows associated with that schedule. An occurrence can be one of the following:

One Time Occurrence

One time occurrences define a single maintenance window. These windows continue until the maximum duration of the window or the maximum number of tasks that can be run in the window is reached.

Recurring Occurrence

Recurring occurrences define a series of maintenance windows. These windows continue until the maximum number of tasks or the end of the day specified in the occurrence was reached.

Pending Activities for Deferred Deployments

If you configure a deferred deployment in a Cisco UCS domain, Cisco UCS Manager enables you to view all pending activities. You can see activities that are waiting for user acknowledgement and those that are scheduled.

If a Cisco UCS domain has pending activities, Cisco UCS Manager GUI notifies users with admin privileges when they log in.

Cisco UCS Manager displays information about all pending activities, including the following:

- Name of the service profile to deploy and associate with a server
- Server affected by the deployment
- Disruption caused by the deployment
- Change performed by the deployment

Note You cannot specify the maintenance window in which a specific pending activity is applied to the server. The maintenance window depends upon how many activities are pending and which maintenance policy is assigned to the service profile. However, any user with admin privileges can manually initiate a pending activity and reboot the server immediately, whether it is waiting for user acknowledgment or for a maintenance window.

Guidelines and Limitations for Deferred Deployments

Service Profile Association Changes and Maintenance Policy Options

When changing service profile association, the following maintenance policy options can affect how the changes are applied:

- If the **On Next Boot** and **User Ack** options are enabled in a maintenance policy, the service profile association change displays a warning that an acknowledgement is required. However, association will happen immediately.
- If the **On Next Boot** and **User Ack** options are not enabled in a maintenance policy, the service profile association change displays a warning that an acknowledgement is required, and will remain pending until acknowledged.

Cannot Undo All Changes to Service Profiles or Service Profile Templates

If you cancel a pending change, Cisco UCS Manager attempts to roll back the change without rebooting the server. However, for complex changes, Cisco UCS Manager may have to reboot the server a second time to roll back the change. For example, if you delete a vNIC, Cisco UCS Manager reboots the server according to the maintenance policy included in the service profile. You cannot cancel this reboot and change, even if you restore the original vNIC in the service profile. Instead, Cisco UCS Manager schedules a second deployment and reboot of the server.

Association of Service Profile Can Exceed Boundaries of Maintenance Window

After Cisco UCS Manager begins the association of the service profile, the scheduler and maintenance policy do not have any control over the procedure. If the service profile association does not complete within the allotted maintenance window, the process continues until it is completed. For example, this can occur if the association does not complete in time because of retried stages or other issues.

Cannot Specify Order of Pending Activities

Scheduled deployments run in parallel and independently. You cannot specify the order in which the deployments occur. You also cannot make the deployment of one service profile change dependent upon the completion of another.

Cannot Perform Partial Deployment of Pending Activity

Cisco UCS Manager applies all changes made to a service profile in the scheduled maintenance window. You cannot make several changes to a service profile at the same time and then have those changes be spread across several maintenance windows. When Cisco UCS Manager deploys the service profile changes, it updates the service profile to match the most recent configuration in the database.

Maintenance Policy Configuration

Maintenance Policy

The maintenance policy specifies how deploys the service profile changes. The deployment can occur in one of the following ways:

- Immediately
- When acknowledged by a user with administrator privileges
- Automatically at the time specified in a schedule
- On the next reboot or shutdown without waiting for the user acknowledgment or the timer scheduling option

A UCSM and CIMC version on blade or rack server must be running firmware from 3.1.x bundle, for **On Next Boot** to work.

If the **On Next Boot** option is enabled in a maintenance policy, and you downgrade from Cisco UCS Manager Release 3.1(1) or later releases to any release earlier than Cisco UCS Manager Release 2.2(8), firmware downgrade will fail. Disable **On Next Boot** from the maintenance policy to continue with the downgrade.

You can use the soft shutdown timer in the maintenance policy to configure the wait time for performing a hard shutdown. The soft shutdown timer is applicable when you reboot the server for the following:

- Reset the server using the **Gracefully Restart OS** option.
- Shut down the server with the **In case of graceful shutdown failure, a hard shutdown will be issued after X seconds** option.
- Modify a service profile that requires a server reboot.

If the maintenance policy is configured to deploy the change during a scheduled maintenance window, the policy must include a valid schedule. The schedule deploys the changes in the first available maintenance window.

Note A maintenance policy only prevents an immediate server reboot when a configuration change is made to an associated service profile. However, a maintenance policy does not prevent the following actions from taking place right away:

- Deleting an associated service profile from the system
- Disassociating a server profile from a server
- Directly installing a firmware upgrade without using a service policy
- Resetting the server

Creating a Maintenance Policy

Before you begin

If you plan to configure this maintenance policy for deferred deployment, create a schedule.

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 maint-policy** *policy-name*	Creates the specified maintenance policy and enters maintenance policy mode.
Step 3	UCS-A /org/maint-policy # **set reboot-policy** {**immediate** | **timer-automatic** | **user-ack**} | **on next boot**	When a service profile is associated with a server, the server needs to be rebooted to complete the association. Specifying the reboot-policy command determines when the reboot occurs for all service profiles that include this maintenance policy. Possible values include: • **immediate**--The server reboots as soon as the change is made to the service profile. • **timer-automatic** --You select the schedule that specifies when maintenance operations can be applied to the server using the **`set scheduler`** command. Cisco UCS reboots the server and completes the service profile changes at the scheduled time. • **user-ack** --The user must explicitly acknowledge the changes by using the **apply pending-changes** command before changes are applied.
Step 4	UCS-A /org/maint-policy # {**enable** | **disable**} **on-next-boot**	Enabling this option reboots the server automatically during the next reboot or shutdown after the service profile association is complete, without having to wait for the User Ack or the Timer Automatic maintenance window Schedule option. **Note** De-selecting the **On Next Boot** option disables the Maintenance Policy on the BMC. If the **reboot-policy** is set to **timer-automatic** or **user-ack**, enabling this option means the changes will be applied whenever the server reboots, even if that reboot occurs outside a scheduled maintenance window or without user acknowledgment.

	Command or Action	Purpose
Step 5	(Optional) UCS-A /org/maint-policy # **set scheduler** *scheduler-name*	If the reboot-policy property is set to timer-automatic, you must select the schedule that specifies when maintenance operations can be applied to the server. Cisco UCS reboots the server and completes the service profile changes at the scheduled time.
Step 6	UCS-A /org/maint-policy # **commit-buffer**	Commits the transaction to the system configuration.

Example

The following example creates a maintenance policy called maintenance, sets the system to reboot immediately when a service profile is associated with a server, and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # create maint-policy maintenance
UCS-A /org/maint-policy* # set reboot-policy immediate
UCS-A /org/maint-policy* # commit-buffer
UCS-A /org/maint-policy #
```

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

Example

The following example deletes a maintenance policy called maintenance and commits the transaction:

```
UCS-A# scope org /
UCS-A /org # delete maint-policy maintenance
UCS-A /org/maint-policy* # commit-buffer
UCS-A /org/maint-policy #
```

Pending Activities

Pending Activities for Deferred Deployments

If you configure a deferred deployment in a Cisco UCS domain, Cisco UCS Manager enables you to view all pending activities. You can see activities that are waiting for user acknowledgement and those that are scheduled.

If a Cisco UCS domain has pending activities, Cisco UCS Manager GUI notifies users with admin privileges when they log in.

Cisco UCS Manager displays information about all pending activities, including the following:

- Name of the service profile to deploy and associate with a server
- Server affected by the deployment
- Disruption caused by the deployment
- Change performed by the deployment

Note You cannot specify the maintenance window in which a specific pending activity is applied to the server. The maintenance window depends upon how many activities are pending and which maintenance policy is assigned to the service profile. However, any user with admin privileges can manually initiate a pending activity and reboot the server immediately, whether it is waiting for user acknowledgment or for a maintenance window.

Viewing Pending Activities

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters organization mode. 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 # **show pending-changes** [**detail** | **expand**]	Displays details about pending-changes.

Example

The following example shows how to display pending changes for a service profile called accounting:

```
UCS-A# scope org /
UCS-A /org # scope service-profile accounting
UCS-A /org/service-profile # show pending-changes detail

Pending Changes:
    Scheduler:
    Changed by: admin
    Acked by:
    Mod. date: 2010-09-20T20:36:09.254
    State: Untriggered
    Admin State: Untriggered
    Pend. Changes: 0
    Pend. Disr.: 0
UCS-A /org/service-profile #
```

Deploying a Service Profile Change Waiting for User Acknowledgement

Cisco UCS Manager CLI cannot deploy all pending service profile changes (for multiple service profiles) waiting for user acknowledgement. To simultaneously deploy all pending service profile changes for multiple service profiles, use Cisco UCS Manager GUI.

Important You cannot stop Cisco UCS Manager from rebooting the affected server after you acknowledge a pending activity.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters organization mode. 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 # **apply pending-changes immediate**	Applies the pending changes immediately. Cisco UCS Manager immediately reboots the server affected by the pending activity.

Example

The following example shows how to apply pending changes for a service profile named accounting:

```
UCS-A# scope org /
UCS-A /org # scope service-profile accounting
UCS-A /org/service-profile # apply pending-changes immediate
UCS-A /org/service-profile #
```

Deploying a Scheduled Service Profile Change Immediately

Cisco UCS Manager CLI cannot deploy all scheduled service profile changes (for multiple service profiles) at the same time. To simultaneously deploy all scheduled service profile changes for multiple service profiles, use Cisco UCS Manager GUI.

Important You cannot stop Cisco UCS Manager from rebooting the affected server after you acknowledge a pending activity.

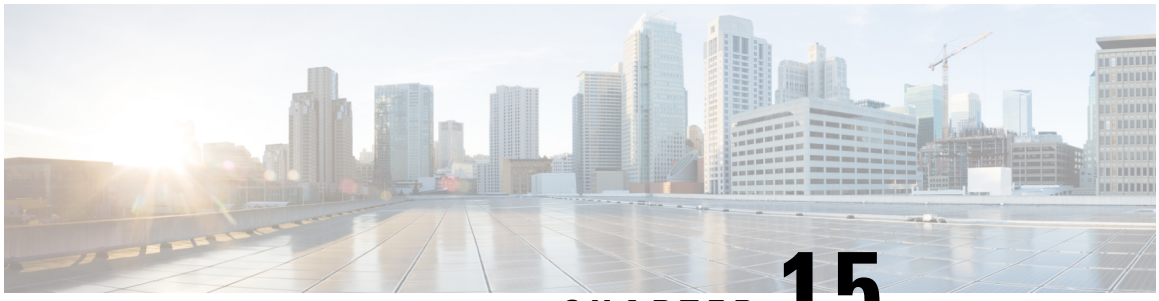
Procedure

	Command or Action	Purpose
Step 1	UCS-A# **scope org** *org-name*	Enters organization mode. 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 # **apply pending-changes immediate**	Applies the pending changes immediately. Cisco UCS Manager immediately reboots the server affected by the pending activity.

Example

The following example shows how to apply pending changes for a service profile called accounting:

```
UCS-A# scope org /
UCS-A /org # scope service-profile accounting
UCS-A /org/service-profile # apply pending-changes immediate
UCS-A /org/service-profile #
```



CHAPTER 15

UCS Fault Suppression

Fault Suppression for System Maintenance

Global Fault Policy

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

A fault in Cisco UCS has the following lifecycle:

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

Configuring the Fault Collection Policy

Procedure

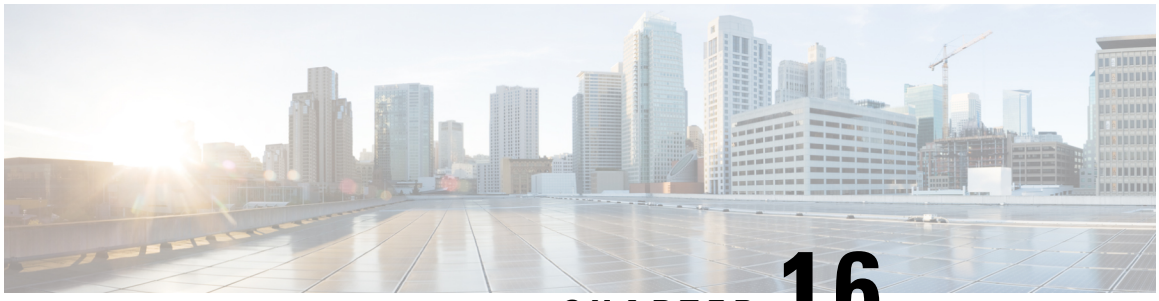
	Command or Action	Purpose
Step 1	UCS-A# **scope monitoring**	Enters monitoring mode.

	Command or Action	Purpose
Step 2	UCS-A /monitoring # **scope fault policy**	Enters monitoring fault policy mode.
Step 3	UCS-A /monitoring/fault-policy # **set clear-action** {**delete** | **retain**}	Specifies whether to retain or delete all cleared messages. If the **retain** option is specified, then the length of time that the messages are retained is determined by the **set retention-interval** command.
Step 4	UCS-A /monitoring/fault-policy # **set flap-interval** *seconds*	Specifies the time interval (in seconds) the system waits before changing a fault state. Flapping occurs when a fault is raised and cleared several times in rapid succession. To prevent this, the system does not allow a fault to change state until the flapping interval has elapsed after the last state change. If the fault is raised again during the flapping interval, it returns to the active state, otherwise, the fault is cleared.
Step 5	UCS-A /monitoring/fault-policy # **set retention-interval** {*days hours minutes seconds* | **forever**}	Specifies the time interval the system retains all cleared fault messages before deleting them. The system can retain cleared fault messages forever, or for the specified number of days, hours, minutes, and seconds.
Step 6	UCS-A /monitoring/fault-policy # **commit-buffer**	Commits the transaction.

Example

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

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



CHAPTER 16

Cisco Intersight Management

Intersight Management Mode

Cisco Intersight™ is a management platform delivered as a service with embedded analytics for your Cisco and 3rd party IT infrastructure. Intersight Managed Mode (IMM) is a new architecture that manages the UCS Fabric Interconnected systems through a Redfish-based standard model. Intersight Managed Mode unifies the capabilities of the UCS Systems and the cloud-based flexibility of Intersight, thus unifying the management experience for the standalone and Fabric Interconnect attached systems. Intersight Management Model standardizes policy and operation management for Cisco UCS 6600 Series Fabric Interconnect, UCSX-S9108-100G, UCS-FI-6454, UCS-FI-64108, UCS-FI-6536 and Cisco UCS B-Series (M5, M6), Cisco UCS C-Series (M5, M6, M7,M8), and Cisco UCS X-Series (M6 ,M7 ,M8) servers.

You can choose between the native UCSM Managed Mode (UMM) or Intersight Managed Mode (IMM) for the Fabric attached UCS Systems during initial setup of the Fabric Interconnects. If you choose to switch back between UMM and IMM, you must erase the present configuration and start from initial setup. Before erasing the configuration, you must ensure to unclaim the device from Intersight and decommission all rack servers.

Note For more information, see https://intersight.com/help/resources#intersight_managed_mode.

Cisco Intersight Managed Mode (IMM) transition tool helps bootstrap new IMM deployments by replicating the configuration attributes of the existing Cisco UCS Manager (UCSM) infrastructure and by converting the existing Service Profile Templates to IMM Server Profile Templates to accelerate deployment of new servers in IMM.

Note For more information, see the latest *Cisco Intersight Managed Mode Transition Tool User Guide*: https://www.cisco.com/c/en/us/td/docs/unified_computing/Intersight/b_cisco_intersight_managed_mode_transition_tool_user_guide.pdf

Device Connector

Device connector connects Cisco UCS Manager to Cisco Intersight, the cloud-hosted server management system. It enables Cisco UCS Manager to be managed and monitored through Cisco Intersight.

To register a device with Cisco Intersight in the cloud, you must do the following:

1. Connect Cisco UCS Manager with Cisco Intersight by configuring the device connector proxy settings, if they are required.

Note For Cisco UCS X-Series M7/M8 and/or Cisco UCS C-Series M7/M8 servers, Cisco UCS Manager requires Cisco Intersight connection and Intersight Infrastructure Service Licenses. When you **Unclaim** or **Disable** the device connector, ignoring the warning, a major fault is triggered.

2. Use the device serial number and security code to validate your access to the device from Cisco Intersight and claim the device.

Updating Device Connector

When you upgrade Cisco UCS Manager, the device connector is automatically updated to the image integrated with the Cisco UCS Manager version. The device connector does not get downgraded when you downgrade the Cisco UCS Manager version.

You can update the device connector through the Cisco Intersight GUI. You can also update the device connector through the local management shell in Cisco UCS Manager CLI.

Procedure

	Command or Action	Purpose
Step 1	UCS-A# **connect local-mgmt**	Enters local management mode.
Step 2	UCS-A(local-mgmt)# **copy** [*from-filesystem*:] [*from-path*] *filename to-path* [*dest-filename*]	Copies the device connector image file from a remote server to a local destination by using the specified file transfer protocol. You need to copy the file to one fabric interconnect only. • *from-filesystem*—The remote file system containing the file to be copied. This file system can be specified by using one of the following options: • **ftp:** [// [*username@*] *server*] • **scp:** [// [*username@*] *server*] • **sftp:** [// [*username@*] *server*] • **tftp:** [//*server* [:*port*]]

	Command or Action	Purpose
		If the file system is not specified, the current working file system is assumed. If a remote protocol is specified with no server name, you are prompted to enter the server name. • *from-path*—Absolute or relative path to the file to be copied. If no path is specified, the current working directory is assumed. • *filename*—The name of the source file to be copied. • *to-path*—Absolute or relative path to the copied file. If no path is specified, the current working directory is assumed. The path includes the local file system to contain the copied file. This file system can be specified from one of the following options: • **volatile:** • **workspace:** • *dest-filename*—The new name for the copied file. If a dest-filename is specified, the copied file is renamed at the destination location. **Note** You cannot download the device connector image file through Cisco UCS Manager GUI.
Step 3	UCS-A(local-mgmt)# **update-device-connector workspace:** | **volatile:**/*filename* [**skip-upgrade-on-peer**]	Updates the device connector image on the peer fabric interconnect and then the local fabric interconnect. Using the **skip-upgrade-on-peer** option skips update on the peer fabric interconnect.

Example

The following example updates the device connector on both fabric interconnects:

```
UCS-A# connect local-mgmt
UCS-A(local-mgmt)# copy scp://username@10.100.100.100/filepath/filename.bin workspace:/
UCS-A(local-mgmt)# update-device-connector workspace:/filename.bin
Update Started
Updating Device Connector on peer Fabric interconnect
Successfully updated device connector on peer Fabric interconnect
```

```
Updating Device Connector on local Fabric interconnect
Successfully updated device connector on local Fabric interconnect
UCS-A(local-mgmt)#
```

The following example updates the device connector on the local fabric interconnect only:

```
UCS-A# connect local-mgmt
UCS-A(local-mgmt)# copy scp://username@10.100.100.100/filepath/filename.bin workspace:/
UCS-A(local-mgmt)# update-device-connector workspace:/filename.bin skip-upgrade-on-peer
Update Started
Updating Device Connector on local Fabric interconnect
Successfully updated device connector on local Fabric interconnect
UCS-A(local-mgmt)#
```

Local Management

traceroute

To view the route to a network host, use the **traceroute** command in local management command mode.

traceroute *host-name* [**source** *source*]

Syntax Description

host-name	The host name or IP address of the destination network host.
source *source*	(Optional) Specifies the IP address to be used as the source address in outgoing probe packets.

Command Default

None

Command Modes

Local management (local-mgmt)

Command History

Release	Modification
1.0(1)	This command was introduced.

Usage Guidelines

Use this command to trace the route of IP packets to a network host.

You can use the optional **source** keyword to force the source address of the probe packets to be another IP address of the sending host.

Examples

This example shows how to trace the route to a network host:

```
switch-A(local-mgmt)# traceroute 10.64.58.50
traceroute to 10.64.58.50, 30 hops max, 60 byte packets
 1  10.197.123.1   (10.197.123.1)   0.284 ms 0.317 ms  0.351 ms
 2  10.127.103.165 (10.127.103.165) 0.277 ms 0.292 ms  0.388 ms
 3  10.127.42.101  (10.127.42.101)  0.721 ms 0.731 ms  0.761 ms
 4  10.127.42.101  (10.127.42.101)  0.803 ms 0.810 ms  0.813 ms
 5  10.127.188.30  (10.127.188.30)  0.813 ms 0.816 ms  0.829 ms
```

```
 6  * * *
 7  10.225.71.226 (10.225.71.226)   0.883 ms  0.979 ms  0.566 ms
 8  10.127.43.165 (10.127.43.165)   0.774 ms  0.750 ms  1.964 ms
 9  10.127.42.30  (10.127.42.30)    0.770 ms  0.732 ms  0.984 ms
10  72.163.187.110 (72.163.187.110) 1.005 ms  0.962 ms  0.972 ms
11  72.163.171.142 (72.163.171.142) 1.836 ms  1.827 ms  1.902 ms
12  10.64.58.50 (10.64.58.50)       1.620 ms  1.688 ms  1.727 ms

switch-A(local-mgmt)#
```