Managing User Accounts

Configuring Local Users

Before you begin

You must log in as a user with admin privileges to configure or modify local user accounts.

Procedure

	Command or Action	Purpose
Step 1	Server# **scope user** *usernumber*	Enters user command mode for the user number.
Step 2	Server /user # **set enabled** {**yes** | **no**}	Enables or disables the user account on the CIMC.
Step 3	Server /user # **set name** *username*	Specifies the username for the user.
Step 4	Server /user # **set password**	Specifies the password for the user. You are prompted to enter the password twice.
Step 5	Server /user # **set role** {**readonly** | **user** | **admin**}	Specifies the role assigned to the user. The role can be one of the following: • readonly—This user can view information but cannot make any changes. • user—This user can do the following: • View all information • Manage the power control options such as power on, power cycle, and power off

	Command or Action	Purpose
		• Launch the KVM console and virtual media • Clear all logs • Toggle the locator LED • admin—This user can perform all actions available through the GUI, CLI, and IPMI.
Step 6	Server /user # **commit**	Commits the transaction to the system configuration.

Example

This example configures user 5 as an admin:

```
Server# scope user 5
Server /user # set enabled yes
Server /user *# set name john
Server /user *# set password
Please enter password:
Please confirm password:
Server /user *# set role readonly
Server /user *# commit
Server /user #  show
User   Name             Role      Enabled   SSH Key Count
------ --------------- -------- ---------  --------------
5      user             readonly  yes       (n/a)
```

LDAP Servers (Active Directory)

CIMC supports directory services that organize information in a directory and manage access to this information. CIMC supports Lightweight Directory Access Protocol (LDAP), which stores and maintains directory information in a network. In addition, CIMC supports Microsoft Active Directory (AD). Active Directory is a technology that provides a variety of network services including LDAP-like directory services, Kerberos-based authentication, and DNS-based naming. The CIMC utilizes the Kerberos-based authentication service of LDAP.

When LDAP is enabled in the CIMC, user authentication and role authorization is performed by the LDAP server for user accounts not found in the local user database. The LDAP user authentication format is username@domain.com.

By checking the Enable Encryption check box in the **LDAP Settings** area, you can require the server to encrypt data sent to the LDAP server.

Configuring the LDAP Server

The CIMC can be configured to use LDAP for user authentication and authorization. To use LDAP, configure users with an attribute that holds the user role and locale information for the CIMC. You can use an existing LDAP attribute that is mapped to the CIMC user roles and locales, or you can modify the LDAP schema to add a new custom attribute, such as the Cisco AV Pair attribute, which has an attribute ID of 1.3.6.1.4.1.9.287247.1.

Important For more information about altering the schema, see the article at http://technet.microsoft.com/en-us/library/bb727064.aspx.

Note This example creates a custom attribute named Cisco AV Pair, but you can also use an existing LDAP attribute that is mapped to the CIMC user roles and locales.

The following steps must be performed on the LDAP server:

Step 1 Ensure that the LDAP schema snap-in is installed.

Step 2 Using the schema snap-in, add a new attribute with the following properties:

Properties	Value
Common Name	`CiscoAVPair`
LDAP Display Name	`CiscoAVPair`
Unique X500 Object ID	`1.3.6.1.4.1.9.287247.1`
Description	`CiscoAVPair`
Syntax	`Case Sensitive String`

Step 3 Add the CiscoAVPair attribute to the user class using the snap-in:

a. Expand the **Classes** node in the left pane and type `U` to select the user class.

b. Click the **Attributes** tab and click **Add**.

c. Type `C` to select the CiscoAVPair attribute.

d. Click **OK**.

Step 4 Add the following user role values to the CiscoAVPair attribute, for the users that you want to have access to CIMC:

Role	Cisco AVPair Attribute Value
admin	`shell:roles="admin"`
user	`shell:roles="user"`
read-only	`shell:roles="read-only"`

Note For more information about adding values to attributes, see the article at http://technet.microsoft.com/en-us/library/bb727064.aspx.

What to do next

Use the CIMC to configure the LDAP server.

Configuring LDAP in CIMC

Configure LDAP in CIMC when you want to use an LDAP server for local user authentication and authorization.

Before you begin

You must log in as a user with admin privileges to perform this task.

Procedure

	Command or Action	Purpose
Step 1	Server # **scope ldap**	Enters LDAP command mode.
Step 2	Server /ldap # **set enabled** {**yes** | **no**}	Enables or disables LDAP security. When enabled, user authentication and role authorization is performed by LDAP for user accounts not found in the local user database.
Step 3	Server /ldap # **set domain** *LDAP domain name*	Specifies an LDAP domain name.
Step 4	Server /ldap # **set timeout** *seconds*	Specifies the number of seconds the CIMC waits until the LDAP search operation times out. The value must be between 0 and 1800 seconds.
Step 5	Server /ldap # **set encrypted** {**yes** | **no**}	If encryption is enabled, the server encrypts all information sent to AD.
Step 6	Server /ldap # **set base-dn** *domain-name*	Specifies the Base DN that is searched on the LDAP server.
Step 7	Server /ldap # **set attribute** *name*	Specifies an LDAP attribute that contains the role and locale information for the user. This property is always a name-value pair. The system queries the user record for the value that matches this attribute name. You can use an existing LDAP attribute that is mapped to the CIMC user roles and locales or you can create a custom attribute, such as the CiscoAVPair attribute, which has the following attribute ID: `1.3.6.1.4.1.9.287247.1` **Note** If you do not specify this property, user access is denied.

	Command or Action	Purpose
Step 8	Server /ldap # **set filter-attribute**	Specifies the account name attribute. If Active Directory is used, then specify **sAMAccountName** for this field.
Step 9	Server /ldap # **commit**	Commits the transaction to the system configuration.
Step 10	Server /ldap # **show** [**detail**]	(Optional) Displays the LDAP configuration.

Example

This example configures LDAP using the CiscoAVPair attribute:

```
Server# scope ldap
Server /ldap # set enabled yes
Server /ldap *# set domain sample-domain
Server /ldap *# set timeout 60
Server /ldap *# set base-dn example.com
Server /ldap *# set attribute CiscoAVPair
Server /ldap *# set filter-attribute sAMAccountName
Server /ldap *# commit
Server /ldap # show detail
LDAP Settings:
    Enabled: yes
    Domain: sample-domain
    BaseDN: example.com
    Timeout (for each server): 60
    Filter-Attribute: sAMAccountName
    Attribute: CiscoAvPair
Server /ldap #
```

What to do next

To use LDAP groups for group authorization, see section Configuring LDAP Groups in CIMC.

Configuring LDAP Groups in CIMC

Note When Active Directory (AD) group authorization is enabled and configured, user authentication is also done on the group level for users that are not found in the local user database or who are not individually authorized to use CIMC in the Active Directory.

Before you begin

- You must log in as a user with admin privileges to perform this task.
- Active Directory (or LDAP) must be enabled and configured.

Procedure

	Command or Action	Purpose
Step 1	Server # **scope ldap**	Enters the LDAP command mode for AD configuration.
Step 2	Server /ldap# **scope ldap-group-rule**	Enters the LDAP group rules command mode for AD configuration.
Step 3	Server /ldap/ldap-group-rule # **set group-auth** {**yes** | **no**}	Enables or disables LDAP group authorization.
Step 4	Server /ldap # **scope role-group** *index*	Selects one of the available group profiles for configuration, where *index* is a number between 1 and 28.
Step 5	Server /ldap/role-group # **set name** *group-name*	Specifies the name of the group in the AD database that is authorized to access the server.
Step 6	Server /ldap/role-group # **set domain** *domain-name*	Specifies the AD domain the group must reside in.
Step 7	Server /ldap/role-group # **set role** {**admin** | **user** | **readonly**}	Specifies the permission level (role) assigned to all users in this AD group. This can be one of the following: • **admin**—The user can perform all actions available. • **user**—The user can perform the following tasks: • View all information • Manage the power control options such as power on, power cycle, and power off • Launch the KVM console and virtual media • Clear all logs • Toggle the locator LED • **readonly**—The user can view information but cannot make any changes.
Step 8	Server /ldap/role-group # **commit**	Commits the transaction to the system configuration.

Example

This example shows how to configure LDAP group authorization:

```
Server# scope ldap
Server /ldap # scope ldap-group-rule
Server /ldap/ldap-group-rule # set group-auth yes
Server /ldap *# scope role-group 5
Server /ldap/role-group # set name Training
Server /ldap/role-group* # set domain example.com
Server /ldap/role-group* # set role readonly
Server /ldap/role-group* # commit
ucs-c250-M2 /ldap # show role-group
Group  Group Name       Domain Name      Assigned Role
------ -----------     --------------   --------------
```

```
1       (n/a)            (n/a)            admin
2       (n/a)            (n/a)            user
3       (n/a)            (n/a)            readonly
4       (n/a)            (n/a)            (n/a)
5       Training         example.com      readonly

Server /ldap/role-group #
```

TACACS+ Server

TACACS+ is a security protocol that provides centralized validation of users attempting to gain access to a router or network access server. TACACS+ services are maintained in a database on a TACACS+ server. You must configure a TACACS+ server before you configure the TACACS+ features on your network access server and make them available.

On the TACACS+ server, ensure you configure Cisco attribute-value (AV) pair privilege level (priv-lvl) for Cisco Integrated Management Controller (CIMC) service for the minimum privilege level of administrators and operators.

Restrictions for TACACS+ Support for CIMC

- CIMC supports connection to up to 6 TACACS+ servers.
- Users must first successfully complete TACACS+ authentication before proceeding to TACACS+ authorization.
- TACACS+ and LDAP configurations are exclusive, only one configuration is enabled at a time.
- Default time out is five seconds.
- Default TCP port connection is 49.
- Default login is PAP login where the username and password arrive at the network access server in a PAP protocol packet instead of details entered by the user.
- Only IPv4 is supported.
- Pre-shared key (PSK) size is 32 characters.
- Supported special characters in shared secret key are: **! @% ^ * - _ .**

TACACS+ Operation

Before you begin

When a user attempts a simple ASCII login by authenticating to CIMC using TACACS+, the following options are provided:

CIMC eventually receives one of the following responses from the TACACS+ server:

- ACCEPT—The user is authenticated and service may begin. If CIMC is configured to require authorization, authorization begins at this time.

- REJECT—The user has failed to authenticate. The user may be denied further access, or will be prompted to retry the login sequence depending on the TACACS+ server.
- CONTINUE—The user is prompted for additional authentication information.

What to do next

After authentication, CIMC sends authorization request to the TACACS+ server. Based on authorization result, CIMC assigns the user's role.

Configure TACACS+ Server

Procedure

	Command or Action	Purpose
Step 1	Server # **scope tacacs+**	Enters TACACS+ configuration mode.
Step 2	Server /tacacs+ # **set enabled** [**yes** | **no**]	Enables or disables TACACS+ based authentication.
Step 3	Server /tacacs+ # **fallback-only-on-no-connectivity** [**yes** | **no**]	Enables or disables fallback to other authentication precedence.
Step 4	Server /tacacs+/tacacs-server # **scope tacacs-server 1**	Enters tacacs-server 1 configuration mode.
Step 5	Server /tacacs+/tacacs-server # **set tacacs-server** *ip-address*	Sets the TACACS server IP address.
Step 6	Server / tacacs+/tacacs-server # **set tacacs-port** *port*	Sets the TACACS port.
Step 7	Server /tacacs+/tacacs-server # **set tacacs-key** *key-string*	Sets the pre-shared key to initiate authentication with the server. The maximum length of the key is 32 characters.
Step 8	Server /tacacs+/tacacs-server # **scope tacacs-server 1**	Enters tacacs-server 1 configuration mode.
Step 9	Server /tacacs+/tacacs-server # **set tacacs-server** *ip-address*	Sets the TACACS server IP address.
Step 10	Server /tacacs+/tacacs-server # **set tacacs-port** *port*	Sets the TACACS port.
Step 11	Server /tacacs+/tacacs-server # **set tacacs-key***key-string*	Sets the pre-shared key to initiate authentication with the server. The maximum length of the key is 32 characters.
Step 12	Server /tacacs # **commit**	Commits the transaction to the system configuration.
Step 13	Server /tacacs # **show** [**detail**]	(Optional) Displays the TACACS configuration.

Example

This example shows how to configure a TACACS server:

```
Server /# scope tacacs+
Server /tacacs+ #set enabled yes
Server /tacacs+ *#set fallback-only-on-no-connectivity no
Server /tacacs+ *#commit
Server /tacacs+ #scope tacacs-server 1
Server /tacacs+/tacacs-server #set tacacs-server 10.126.254.174
Server /tacacs+/tacacs-server *#set tacacs-port 49
Server /tacacs+/tacacs-server *#set tacacs-key
Please enter tacacs-key: _Abcded_abcde_123_abcd12_zxy123_
Please confirm tacacs-key: _Abcded_abcde_123_abcd12_zxy123_
Server /tacacs+/tacacs-server #commit
```

This example shows how to verify a TACACS+ server configuration:

```
Server /tacacs+/tacacs-server #show  detail
Server Id 1:
Server IP address/Hostname: 10.126.254.174
Server Key: ******
Server Port: 49
```

Viewing User Sessions

Procedure

	Command or Action	Purpose
Step 1	Server# **show user-session**	Displays information about current user sessions.

The command output displays the following information about current user sessions:

Name	Description
Session ID column	The unique identifier for the session.
Username column	The username for the user.
IP Address column	The IP address from which the user accessed the server.
Type column	The method by which the user accessed the server. For example, CLI, vKVM, and so on.
Action column	If your user account is assigned the **admin** user role, this column displays **Terminate** if you can force the associated user session to end. Otherwise it displays **N/A**. **Note** You cannot terminate your current session from this tab.

Example

This example displays information about current user sessions:

```
Server# show user-session
ID     Name             IP Address        Type         Killable
------ ---------------- ----------------- ------------ --------
15     admin            10.20.30.138      CLI          yes

Server /user #
```

Terminating a User Session

Before you begin

You must log in as a user with admin privileges to terminate a user session.

Procedure

	Command or Action	Purpose
Step 1	Server # **show user-session**	Displays information about current user sessions. The user session to be terminated must be eligible to be terminated (killable) and must not be your own session.
Step 2	Server /user-session # **scope user-session** *session-number*	Enters user session command mode for the numbered user session that you want to terminate.
Step 3	Server /user-session # **terminate**	Terminates the user session.

Example

This example shows how the admin at user session 10 terminates user session 15:

```
Server# show user-session
ID     Name             IP Address        Type         Killable
------ ---------------- ----------------- ------------ --------
10     admin            10.20.41.234      CLI          yes
15     admin            10.20.30.138      CLI          yes

Server# scope user-session 15
Server /user-session # terminate
User session 15 terminated.

Server /user-session #
```