

CLI Users

Managed devices include a default **admin** account for CLI access. This chapter discusses how to create custom user accounts.

- CLI users, on page 1
- Guidelines for CLI users, on page 3
- Add an internal user at the CLI, on page 3
- Troubleshoot LDAP authentication connections, on page 6

CLI users

A CLI user is a user account that

- can be added as internal users or as external users on a LDAP or RADIUS server,
- is maintained separately on each managed device, and
- requires separate configuration for access to the Cloud-Delivered Firewall Management Center versus managed devices.

User account separation

When you add a user to the Cloud-Delivered Firewall Management Center, that user only has access to the Cloud-Delivered Firewall Management Center; you cannot then use that username to log directly into a managed device. You must separately add a user on the managed device.

Internal and external users

Managed devices support two categories of users—internal and external—for user access control.

Internal users authenticate through a local database on the device. External users authenticate through LDAP or RADIUS servers when they are not found in the local database. Both user types enable secure access management for network devices.

User authentication types

Managed devices support these user authentication methods:

- Internal user: The device checks a local database for user authentication.

- External user: If the user is not present in the local database, the system queries an external LDAP or RADIUS authentication server.

CLI access

CLI access is a command-line interface capability that

- provides a Firepower CLI running on top of Linux for device administration
- enables creation of internal users on devices using the CLI, and
- supports establishment of external users on Firewall Threat Defense devices using the Cloud-Delivered Firewall Management Center.

Security considerations for CLI access

Users with CLI Config level access can access the Linux shell using the **expert** command, and obtain `sudoers` privileges in the Linux shell, which can present a security risk.



Caution Users with CLI Config level access can access the Linux shell using the **expert** command, and obtain `sudoers` privileges in the Linux shell, which can present a security risk. For system security reasons, we strongly recommend:

- Only use the Linux shell under TAC supervision or when explicitly instructed by Firepower user documentation.
- Make sure that you restrict the list of users with CLI access appropriately.
- When granting CLI access privileges, restrict the list of users with Config level access.
- Do not add users directly in the Linux shell; only use the procedures in this chapter.
- Do not access Firepower devices using CLI expert mode unless directed by Cisco TAC or by explicit instructions in the Firepower user documentation.

CLI user roles

A CLI user role is an access control mechanism that

- determines the level of command access a user has on managed devices,
- controls whether users can execute configuration or non-configuration commands, and
- provides security by restricting unauthorized access to critical system functions.

Role types and permissions

On managed devices, user access to commands in the CLI depends on the role you assign.

- **None**: The user cannot log into the device on the command line.

- **Config**: The user can access all commands, including configuration commands. Exercise caution in assigning this level of access to users.
- **Basic**: The user can access non-configuration commands only. Allowed commands are dig, ping, and traceroute. Only internal users and Firewall Threat Defense external RADIUS users support the Basic role.

Guidelines for CLI users

This reference provides essential guidelines for CLI users, including username requirements, default account information, and user account limits for different device models.

Usernames

- You cannot add the same username for both internal and external users. If the external server uses a duplicate username, the deployment to the device fails.
- The username must be Linux-valid:
  - Maximum 32 alphanumeric characters, plus hyphen (-) and underscore (_)
  - All lowercase
  - Cannot start with hyphen (-); cannot be all numbers; cannot include a period (.), at sign (@), or slash (/)

Defaults

All devices include an **admin** user as a local user account; you cannot delete the **admin** user. The default initial password is **Admin123**; the system forces you to change this during the initialization process. The agent stores the modified credentials in the Windows registry and uses the same encryption as the Cloud-Delivered Firewall Management Center, which is Cipher Block Chaining (CBC). See the getting started guide for your model for more information about system initialization.

Number of user accounts

You can create a maximum of 43 user accounts for the Firepower 1000.

Add an internal user at the CLI

Create internal users to provide authorized access to the Firewall Threat Defense system with appropriate privilege levels.

Use the CLI to create internal users on the Firewall Threat Defense.

Procedure

Step 1 Log into the device CLI using an account with Config privileges.

The **admin** user account has the required privileges, but any account with Config privileges will work. You can use an SSH session or the Console port.

For certain Firewall Threat Defense models, the Console port puts you into the FXOS CLI. Use the **connect ftd** command to get to the Firewall Threat Defense CLI.

Step 2 Create the user account.

configure user add *username* {**basic** | **config**}

- *username*—Sets the username. The username must be Linux-valid:
  - Maximum 32 alphanumeric characters, plus hyphen (-) and underscore (_)
  - All lowercase
  - Cannot start with hyphen (-); cannot be all numbers; cannot include a period (.), at sign (@), or slash (/)
- **basic**—Gives the user basic access. This role does not allow the user to enter configuration commands. Allowed commands are dig, ping, and traceroute.
- **config**—Gives the user configuration access. This role gives the user full administrator rights to all commands.

Example:

This example adds a user account named johncrichton with Config access rights. The password is not shown as you type it.

```
> configure user add johncrichton config
Enter new password for user johncrichton: newpassword
Confirm new password for user johncrichton: newpassword
> show user
Login              UID   Auth Access  Enabled Reset    Exp Warn  Str Lock Max
admin             1000  Local Config  Enabled    No  Never  N/A  Dis   No N/A
johncrichton      1001  Local Config  Enabled    No  Never  N/A  Dis   No   5
```

Note Tell users they can change their own passwords using the **configure password** command.

Step 3 (Optional) Adjust the characteristics of the account to meet your security requirements.

You can use these commands to change the default account behavior.

- **configure user aging** *username max_days warn_days*

  Sets an expiration date for the user's password. Specify the maximum number of days for the password to be valid followed by the number of days before expiration the user will be warned about the upcoming expiration. Both values are 1 to 9999, but the warning days must be less than the maximum days. When you create the account, there is no expiration date for the password.

- **configure user forcereset** *username*

  Forces the user to change the password on the next login.

- **configure user maxfailedlogins** *username number*

Sets the maximum number of consecutive failed logins you will allow before locking the account, from 1 to 9999. Use the **configure user unlock** command to unlock accounts. The default for new accounts is 5 consecutive failed logins. The admin account is exempt from being locked out after a maximum number of failed logins unless you enable security certification compliance.

- **configure user minpasswdlen** *username number*

  Sets a minimum password length, which can be from 1 to 127.

- **configure user strengthcheck** *username* {**enable** | **disable**}

  Enables or disables password strength checking, which requires a user to meet specific password criteria when changing their password. When a user's password expires or if the **configure user forcereset** command is used, this requirement is automatically enabled the next time the user logs in.

Step 4 Manage user accounts as necessary.

Users can get locked out of their accounts, or you might need to remove accounts or fix other issues. Use these commands to manage the user accounts on the system.

- **configure user access** *username* {**basic** | **config**}

  Changes the privileges for a user account.

- **configure user delete** *username*

  Deletes the specified account.

- **configure user disable** *username*

  Disables the specified account without deleting it. The user cannot log in until you enable the account.

- **configure user enable** *username*

  Enables the specified account.

- **configure user password** *username*

  Changes the password for the specified user. Users should normally change their own password using the **configure password** command.

Caution Do not use the Linux **passwd** command in Expert Mode to change the admin user password. This command can cause file system corruption. Only use the regular Firewall Threat Defense CLI **configure user password admin** command (if you are not admin) or **configure password** command (if you are admin). If you do not know the password and can't log in at all, refer to the password recovery procedure.

- **configure user unlock** *username*

  Unlocks a user account that was locked due to exceeding the maximum number of consecutive failed login attempts.

Note Firewall Threat Defense uses the Pluggable Authentication Module (PAM) framework for user authentication. Within this framework, the pam_unix.so module handles password hashing and verification using the SHA-512 algorithm

Passwords are hashed using the SHA-512 algorithm.

Troubleshoot LDAP authentication connections

This reference provides troubleshooting information for resolving LDAP authentication connection problems in your system configuration.

If you create an LDAP authentication object and it either does not succeed in connecting to the server you select or does not retrieve the list of users you want, you can tune the settings in the object.

If the connection fails when you test it, try the following suggestions to troubleshoot your configuration:

- Use the messages displayed at the top of the web interface screen and in the test output to determine which areas of the object are causing the issue.
- Check that the user name and password you used for the object are valid:
  - Check that you have the rights to browse to the directory indicated in your base-distinguished name by connecting to the LDAP server using a third-party LDAP browser.
  - Check that the user name is unique to the directory information tree for the LDAP server.
  - If you see an LDAP bind error 49 in the test output, the user binding for the user failed. Try authenticating to the server through a third-party application to see if the binding fails through that connection as well.
  - When configuring LDAP authentication, avoid using a backslash ("\") in the binding password. The presence of a backslash in the password causes the LDAP binding process to fail, resulting in unsuccessful external authentication.
- Check that you have correctly identified the server:
  - Check that the server IP address or host name is correct.
  - Check that you have TCP/IP access from your local appliance to the authentication server where you want to connect.
  - Check that access to the server is not blocked by a firewall and that the port you have configured in the object is open.
  - If you are using a certificate to connect via TLS or SSL, the host name in the certificate must match the host name used for the server.
  - Check that you have not used an IPv6 address for the server connection if you are authenticating CLI access.

    - If you used server type defaults, check that you have the correct server type and click **Set Defaults** again to reset the default values.

- If you typed in your base-distinguished name, click **Fetch DNs** to retrieve all the available base distinguished names on the server, and select the name from the list.
- If you are using any filters, access attributes, or advanced settings, check that each is valid and typed correctly.
- If you are using any filters, access attributes, or advanced settings, try removing each setting and testing the object without it.
- If you are using a base filter or a CLI access filter, make sure that the filter is enclosed in parentheses and that you are using a valid comparison operator (maximum 450 characters, including the enclosing parentheses).
- To test a more restricted base filter, try setting it to the base distinguished name for the user to retrieve just that user.
- If you are using an encrypted connection:
    - Check that the name of the LDAP server in the certificate matches the host name that you use to connect.
    - Check that you have not used an IPv6 address with an encrypted server connection.
- If you are using a test user, make sure that the user name and password are typed correctly.
- If you are using a test user, remove the user credentials and test the object.
- Test the query that you are using by connecting to the LDAP server and using this syntax:

```
ldapsearch -x -b 'base_distinguished_name'
-h LDAPserver_ip_address -p port -v -D
'user_distinguished_name' -W 'base_filter'
```

  For example, if you are trying to connect to the security domain on `myrtle.example.com` using the `domainadmin@myrtle.example.com` user and a base filter of (`cn=*`), you could test the connection using this statement:

```
ldapsearch -x -b 'CN=security,DC=myrtle,DC=example,DC=com'
-h myrtle.example.com -p 389 -v -D
'domainadmin@myrtle.example.com' -W '(cn=*)'
```

If you can test your connection successfully but authentication does not work after you deploy a platform settings policy, check that authentication and the object you want to use are both enabled in the platform settings policy that is applied to the device.

If you connect successfully but want to adjust the list of users retrieved by your connection, you can add or change a base filter or CLI access filter or use a more restrictive or less restrictive base DN.

While authenticating a connection to Active Directory (AD) server, rarely the connection event log indicates blocked LDAP traffic although the connection to AD server is successful. This incorrect connection log occurs when the AD server sends a duplicate reset packet. The Firewall Threat Defense device identifies the second reset packet as part of a new connection request and logs the connection with Block action.