Configure Authentication

This chapter describes the procedures to configure the authentication and multiple privilege levels. It describes the procedures to encrypt a password and change the static or line password. This chapter also explains to manage the RADIUS and TACACS server.

- Change a Static Enable Password, on page 1
- Change a Line Password, on page 2
- Encrypt Password, on page 2
- AAA Password Security Policies, on page 5
- Enabling Secret Encryption, on page 7
- Configure Secret for Users, on page 8
- Configure Secret Type 8 and Type 9, on page 9
- Configure Secret Type 10, on page 10

Change a Static Enable Password

Perform this task to change the Static Enable password.

Procedure

Step 1 **configure**

Step 2 **username** *name-of-the-user*

Example:

```
RP/0/RP0:hostname (config)# username user1
```

Enters the user name mode.

Step 3 **password** *text.*

Example:

```
RP/0/RP0:hostname (config-un)# password pwd1
```

Enters the password.

Step 4 **commit**

Change a Line Password

Perform this task to change the line password.

Procedure

Step 1 **configure**

Step 2 **username** *name-of-the-user*

Example:

```
RP/0/RP0:hostname (config)# username user1
```

Enters the user name mode.

Step 3 **password** *text*

Example:

```
RP/0/RP0:hostname (config-un)# password pwd1
```

Enters the password.

Step 4 **commit**

Encrypt Password

Perform this task to encrypt the password.

Procedure

Step 1 **configure**

Step 2 **username** *name-of-the-user*

Example:

```
RP/0/RP0:hostname (config)# username user1
```

Enters the user name mode.

Step 3 **encrypt password** *text*

Example:

```
RP/0/RP0:hostname (config-un)# password 7 pwd1
```

Encrypts password.

Step 4 **commit**

Configure Privilege Levels

Before you begin

Optics controller should be created before configuring the privilege levels.

Procedure

Step 1 **configure**

Step 2 **username** *name-of-the-user*

Example:

```
RP/0/RP0:hostname (config)# username user1
```

Enters the user name mode.

Step 3 **privilege level**

Example:

```
RP/0/RP0:hostname (config-un)user group 2
```

Configures the privilege level.

Step 4 **commit**

Manage RADIUS Server

Perform this task to manage the radius server.

Procedure

Step 1 **configure**

Step 2 **username** *name-of-the-user*

Example:

```
RP/0/RP0:hostname (config)# username user1
```

Enters the user name mode.

Step 3 **aaa new-model**

Example:

```
RP/0/RP0:hostname (config)# aaa new-model
```

Adds a new model.

Step 4 **radius-server host** *IP-address* **auth-port** *port-number* **acct-port** *port-number* **key** *name*

Example:

```
RP/0/RP0:hostname (config)# radius-server host 10.78.161.120 auth-port 1812 acct-port 1813
 key SECRET_KEY
```

Adds a radius server.

Step 5 **aaa authentication**

Example:

```
RP/0/RP0:hostname (config)# aaa authentication login default group radius local
```

Adds AAA authentication.

Step 6 **aaa authorization**

Example:

```
RP/0/RP0:hostname (config)# aaa authorization exec default group radius if-authenticated
```

Adds AAA authorization.

Step 7 **aaa accounting**

Example:

```
RP/0/RP0:hostname (config)# aaa accounting exec default start-stop group radius
```

Adds AAA accounting.

Step 8 **commit**

Manage TACACS Server

Perform this task to manage the TACACS server.

Procedure

Step 1 **configure**

Step 2 **username** *name-of-the-user*

Example:

```
RP/0/RP0:hostname (config)# username user1
```

Enters the user name mode.

Step 3 **aaa new-model**

Example:

```
RP/0/RP0:hostname (config)# aaa new-model
```

Adds a new model.

Step 4 **aaa authentication**

Example:

```
RP/0/RP0:hostname (config)# aaa authentication login default group tacacs+ local
```

Adds AAA authentication.

Step 5 **tacacs-server host** *IP-address*

Example:

```
RP/0/RP0:hostname (config)# tacacs-server host 10.78.161.120
```

Adds a TACACS server host.

Step 6 **tacacs-server key** *name*

Example:

```
RP/0/RP0:hostname (config)# tacacs-server key otntest
```

Adds a TACACS server key.

Step 7 **commit**

AAA Password Security Policies

Table 1: Feature History

Feature Name	Release Information	Description
AAA Password Security Policies	Cisco IOS XR Release 6.5.33	This feature introduces strong password security policies to strengthen the secret and password configuration of usernames. These policies also have the option of blocking a local user from accessing the router for a configurable amount of time if the maximum number of attempts to login to the device is reached. The feature thus enhances router security by enforcing strong user password policies. Commands added: • policy

The AAA password security policies enhance the secret configuration for the username. Currently, the password configuration in the username is supported. From the Cisco IOS-XR Release 6.5.33, the secret password policies are supported. This password policy is applicable only to local users.

AAA Password Securities have the following policies:

Lockout Policy

AAA provides a configuration option to restrict the users who try to authenticate using invalid login credentials. This option sets the maximum number of permissible authentication failure attempts for a user. The user who exceeds the maximum limit gets locked out until the configurable lockout timer is expired.

The following sample configuration specifies the maximum number of unsuccessful attempts before a user is locked out.

```
RP/0/RP1:tb6#sh run aaa password-policy pol44
aaa password-policy pol44
 lockout-time days 1
 authen-max-attempts 10
!

RP/0/RP1:tb6#
```

The following is a sample syslog when a user is locked out:

```
RP/0/RSP1/CPU0:Jun 21 09:21:28.226 : locald_DSC[308]: %SECURITY-LOCALD-5-USER_PASSWD_LOCKED
 : User 'user12’ is temporarily locked out for exceeding maximum unsuccessful logins.
This is a sample syslog when user is unlocked for authentication:
 RP/0/RSP1/CPU0:Jun 21 09:14:24.633 : locald_DSC[308]: %SECURITY-LOCALD-5-USER_PASSWD_UNLOCKED
 : User 'user12' is unlocked for authentications.
```

Lifetime Policy

The administrator can configure the maximum lifetime for the password and secret, and if this parameter isn’t set, then the password never expires.

For example, if a password has a lifetime of one month and the machine reboots on the 29th day, the password and secret is valid for one month after the reboot.

```
RP/0/RP0:R3#sh run aaa password-policy pol1
aaa password-policy pol1
 lifetime months 1
```

Reauthentication Policy

When a user attempts to log in and if the user secret credential has already expired, the user will be prompted to create a new secret.

When a user alters the secret after its lifespan expiration, the user will be authenticated against the new secret.

The following is an example showing the UI at login.

```
User Access Verification

Username: lab2
Password:

%Password has expired and must be changed.
(Requirements: Uppercase 1, Lowercase 0, Special 0,
Numeric 0, Min-length 2, Max-length 253, Min-difference 2).
Special characters restricted to !@#$%&*^()

New Password:
Confirm Password:

Password changed successfully. Please login with new password.

Username: lab2
Password:


RP/0/RP0/CPU0:ios#
```

Secret Complexity Policy

Security administrators can configure password policies to increase the complexity of the secret configuration the device. For example:

- Adding a policy to make the secret, a combination of upper and lowercase letters, numbers, and special characters.

  ```
  RP/0/RP0:R3#sh run aaa password-policy pol100
  aaa password-policy pol100
   numeric 3
   upper-case 2
   special-char 1
  !

  RP/0/RP0:R3#sh run username test_1
  username test_1
   policy pol100
   secret 5 $1$7tcr$mwCCVeDXHIy.nhzpDUSMl.
  ```

- Adding some more policies to strengthen the secret such as:
  - The maximum and minimum length of secret
  - The number of characters that must be changed in the new password compared to the old password

Enabling Secret Encryption

Table 2: Feature History

Feature Name	Release Information	Description
Stronger Secret Encryption	Cisco IOS XR Release 6.5.33	This feature introduces **secret** command that enables you to choose encryption types, such as Type 5, Type 8, Type 9, and Type 10, for encrypting the Secret. This feature employs hashing algorithms to build a more secure, strong, and robust secret to enhance the device security. Commands added: • secret

In configuring a user and group membership of that user, you can specify two types of passwords: encrypted or clear text.

The router supports both two-way and one-way (secret) encrypted user passwords. Secret is ideal for user login accounts because the original unencrypted password string cannot be deduced on the basis of the encrypted secret. Some applications (PPP, for example) require only two-way passwords because they must decrypt the stored password for their own function, such as sending the password in a packet. For a login user, both types of passwords may be configured, but a warning message is displayed if one type of password is configured while the other is already present.

If both secret and password are configured for a user, the secret takes precedence for all operations that do not require a password that can be decrypted, such as login. For applications such as PPP, the two-way encrypted password is used even if a secret is present.

Following are the different Cisco Password and Secret Types:

- **Type 5 —** Uses the Message-Digest (MD) hashing algorithms to create secret for a user.
- **Type 7 —**Uses the Vigenere cipher to create password for a user.
- **Type 8 —**Uses the Secure Hash Algorithm, 256-bits (SHA-256) to create secret for a user.
- **Type 9 —**Uses the scrypt hashing algorithm to create secret for a user.
- **Type 10 —**Uses the SHA512 algorithm to create secret for a user.

Configure Secret for Users

Each user is identified by a username that is unique across the administrative domain. Each user should be made a member of at least one user group. Deleting a user group may orphan the users associated with that group. The AAA server authenticates orphaned users, but most commands are not authorized.

Procedure

Step 1 **configure**

Example:

```
RP/0/RP0/CPU0:router#configure
```

Enters configuration mode.

Step 2 **username** *user-name*

Example:

```
RP/0/RP0/CPU0:router(config)#username user1
```

Creates a name for a new user (or identifies a current user) and enters username configuration submode.

Step 3 **secret{0|5|8|9|10}** *secret*

Example:

```
RP/0/RP0/CPU0:router(config-un)#secret 0 sec1
```

Specifies a password for the user named in step 2.

- Use the secret command to create a secure login password for the user names specified in step 2.
- Entering 0 followed by the password command specifies that an unencrypted (clear-text) password follows. Entering 5, 8, 9, 10 followed by the password command specifies that an encrypted password follows.

Step 4 **group** *group-name*

Example:

```
RP/0/RP0/CPU0:router(config-un)#group test
```

Step 5 Repeat step 4 for each user group the user specified in step 2 must be associated with.

Step 6 **commit**

Use the **commit** to save the configuration changes and remain within the configuration session.

Step 7 **end**

Use the **end** to take one of the following actions:

- Yes—Saves configuration changes and exits the configuration session.
- No—Exits the configuration session without committing the configuration changes.
- Cancel—Remains in the configuration session, without committing the configuration changes.

Configure Secret Type 8 and Type 9

When configuring a secret, user has the following two options:

- You can provide an already encrypted value, which is stored directly in the system without any further encryption.
- You can provide a cleartext password that is internally encrypted and stored in the system.

The Type 5, Type 8, and Type 9 encryption methods provide the above mentioned options for users to configure their passwords.

Example:

The following output is an example of directly configuring a Type 8 encrypted password:

```
RP/0/RP0/CPU0:router(config)# username demo8
RP/0/RP0/CPU0:router(confg)# secret?
RP/0/RP0/CPU0:router(config-un)#secret 8
$8$dsYGNam3K1SIJO$7nv/35M/qr6t.dVc7UY9zrJDWRVqncHub1PE9UlMQFs
RP/0/RP0/CPU0:router(config-un)#commit
```

The following output is an example of configuring a clear-text password that is encrypted using the Type 8 encryption method:

```
RP/0/RP0/CPU0:router(config)# username demo8
RP/0/RP0/CPU0:router(config-un)#secret 0 enc-type 8 PASSWORD
```

The following output is an example of directly configuring a Type 9 encrypted password:

```
RP/0/RP0/CPU0:router(config)#username cisco
RP/0/RP0/CPU0:router(config-un)#secret ?
RP/0/RP0/CPU0:router(config-un)#secret 9
$9$q8j4v/mflSOg5v$nGAhRkf0ek3wSYjDG/VKhwpb2znvPaWusuZtkx9Z1sM
RP/0/RP0/CPU0:router(config-un)#commit
```

The following output is an example of configuring a clear-text password that is encrypted using the Type 9 encryption method:

```
RP/0/RP0/CPU0:router(config)#username cisco
RP/0/RP0/CPU0:router(config-un)#secret 0 enc-type 9 cisco123
RP/0/RP0/CPU0:router(config-un)#commit
```

Configure Secret Type 10

You can use the following options to configure secret Type 10 (that uses SHA512 hashing algorithm) for a user:

Configuration Example:

Directly configuring a Type 10 encrypted password:

```
RP/0/RP0/CPU0:router#configure
RP/0/RP0/CPU0:router(config)#username root secret 10
$6$9UvJidvsTEqgkAPU$3CL1Ei/F.E4v/Hi.UaqLwX8UsSEr9ApG6c5pzhMJmZtgW4jObAQ7meAwyhu5VM/aRFJqe/jxZG17h6xPrvJWf1
RP/0/RP0/CPU0:router(config-un)#commit
```

Configuring a clear-text password that is encrypted using Type 10 encryption method:

```
RP/0/RP0/CPU0:router#configure
RP/0/RP0/CPU0:router(config)#username user10 secret 0 enc-type 10 testpassword
RP/0/RP0/CPU0:router(config-un)#commit
```