

Configuring Secure Shell

This section provides the necessary information to configure Secure Shell on network devices.

- Secure Shell, on page 1
- How to Configure Secure Shell, on page 3
- Secure copy, on page 8
- Additional references, on page 10

Secure Shell

Secure Shell (SSH) is a protocol that provides a secure, remote connection to a device.

SSH provides more security for remote connections than Telnet does by providing strong encryption when a device is authenticated. This software release supports SSH Version 1 (SSHv1) and SSH Version 2 (SSHv2).

Prerequisites for configuring Secure Shell

Secure Shell (SSH) prerequisites are the mandatory configuration and software requirements that must be met before the device can support secure remote connections.

- An RSA public/private key pair must be generated on the device.
- An IPsec encryption software image (DES or 3DES) must be installed for both the SSH server and client.
- A valid hostname and host domain must be configured using global configuration commands.

Configuration requirements for Secure Shell

To successfully configure the device for SSH, ensure the following settings are applied:

- Configure the device identity by using the **hostname** and **ip domain-name** commands in global configuration mode.

Restrictions for configuring Secure Shell

Provides the necessary constraints and configuration requirements for enabling secure shell on the IR8100.

These are restrictions for configuring the IR8100 for secure shell.

- The router supports RSA authentication.
- SSH supports only the execution-shell application.
- The SSH server and the SSH client are supported only on Data Encryption Standard (DES) (56-bit) and 3DES (168-bit) data encryption software. In DES software images, DES is the only encryption algorithm available. In 3DES software images, both DES and 3DES encryption algorithms are available.
- This software release supports IP Security (IPSec).
- The IR8100 supports the Advanced Encryption Standard (AES) encryption algorithm with a 128-bit key, 192-bit key, or 256-bit key. However, symmetric cipher AES to encrypt the keys is not supported.
- The login banner is not supported in Secure Shell Version 1. It is supported in Secure Shell Version 2, which Cisco recommends due to its better security.
- The **-l** keyword and userid :{number} {ip-address} delimiter and arguments are mandatory when configuring the alternative method of Reverse SSH for console access.

Note Cisco highly recommends the 3DES encryption as it is stronger.

See the Cisco IOS-XE Device hardening guide at https://www.cisco.com/c/en/us/support/docs/ip/access-lists/13608-21.html for details.

SSH and router access

Secure Shell (SSH) is a protocol that provides a secure, remote connection to a device.

- Provides more security for remote connections than Telnet by using strong encryption.
- Supports SSH Version 1 (SSHv1) and SSH Version 2 (SSHv2).
- Functions consistently in both IPv4 and IPv6, supporting IPv6 addresses and secure transport.

SSH servers, integrated clients, and supported versions

The SSH Integrated Client is an application that runs over the SSH protocol to provide device authentication and encryption.

- Enables secure, encrypted connections to Cisco or non-Cisco devices running an SSH server.
- Supports ciphers including DES, 3DES, and password authentication.
- Requires the SSH server to be enabled for client functionality.

SSH authentication methods

The SSH client supports the following user authentication methods:

- TACACS+
- RADIUS

- Local authentication and authorization

Note The SSH client functionality is available only when the SSH server is enabled.

SSH configuration guidelines

An RSA key pair generated by a SSHv1 server can be used by an SSHv2 server, and the reverse.

RSA Key Generation Troubleshooting

Ensure that the hostname and domain are configured correctly before generating RSA keys to avoid CLI errors or warning messages.

- If you get CLI error messages after entering the **crypto key generate rsa** global configuration command, an RSA key pair has not been generated. Reconfigure the hostname and domain, and then enter the **crypto key generate rsa** command.
- When generating the RSA key pair, the message *No hostname specified* might appear. If it does, you must configure an IP hostname by using the **hostname** global configuration command.
- When generating the RSA key pair, the message No domain specified might appear. If it does, you must configure an IP domain name by using the **ip domain-name** global configuration command.

When configuring the local authentication and authorization authentication method, make sure that AAA is disabled on the console.

How to Configure Secure Shell

Set up the IR8140H to run SSH

To set up your device to run SSH, follow this procedure.

Note Configure user authentication for local or remote access. This step is required.

Procedure

Step 1 Use the **enable** command to enable the router and enter the privileged EXEC mode.

Example:

```
Router> enable
```

Note Enter password if prompted.

Step 2 Use the **configure terminal** command to enter the global configuration mode.

Example:

```
Router# configure terminal
```

Step 3 Use the **hostname** *hostname* command to configure a hostname for your device.

Example:

```
Router(config)# hostname your_hostname
```

Note You can follow this procedure only if you are configuring the device as an SSH server.

Step 4 Use the **ip domain-name** *domain_name* command to configure a host domain for your device.

Example:

```
Router(config)# ip domain-name your_domain_name
```

Step 5 Use the **crypto key generate rsa** command to enable the SSH server for local and remote authentication on the device and generate an RSA key pair.

Example:

```
Router(config)# crypto key generate rsa
```

Note You must follow this procedure only if you are configuring the device as an SSH server. Generating an RSA key pair for the device automatically enables SSH. A minimum modulus size of 1024 bits is recommended. When you generate RSA keys, you are prompted to enter a modulus length. A longer modulus length might be more secure, but it takes longer to generate and to use.

Step 6 Use the **end** command to exit to privileged EXEC mode.

Example:

```
Router(config)# end
```

Step 7 Use the **show running-config** command to verify your entries.

Example:

```
Router# show running-config
```

Step 8 (Optional) Use the **copy running-config startup-config** command to save your entries in the configuration file.

Example:

```
Router# copy running-config startup-config
```

Configure the SSH server

Before you begin

Note This procedure is only required if you are configuring the device as an SSH server.

Follow these steps to configure the SSH server.

Procedure

Step 1 Use the **configure terminal** to enter global configuration mode.

Example:

```
IR8100# configure terminal
```

Step 2 Use the **ip ssh version** [**2**] command to optionally configure the device to run SSH Version 2.

Example:

```
IR8100(config)#  ip ssh version 2
```

Note If you do not enter this command or do not specify a keyword, the SSH server selects the latest SSH version supported by the SSH client. For example, if the SSH client supports SSHv1 and SSHv2, the SSH server selects SSHv2.

Step 3 Use the **ip ssh** { **timeout** *seconds* | **authentication-retries** *number* } command to configures the SSH control parameters.

Example:

```
IR8100(config)# ip ssh timeout 90 ip ssh authentication-retries 2
```

Note

- Specify the time-out value in seconds; the default is 120 seconds. The range is 0 to 120 seconds. This parameter applies to the SSH negotiation phase. After the connection is established, the device uses the default time-out values of the CLI-based sessions.

  By default, up to five simultaneous, encrypted SSH connections for multiple CLI-based sessions over the network are available (session 0 to session 4). After the execution shell starts, the CLI-based session time-out value returns to the default of 10 minutes.

- Specify the number of times that a client can re-authenticate to the server. The default is 3; the range is 0 to 5.

- Repeat this Step 4 when configuring both parameters.

Step 4 Use one or both commands:

- **line vty** *line_number* [*ending line number*]
- **transport input ssh**

Example:

```
IR8100(config)# line vty 1 10
```

or

```
IR8100(config-line)#  transport input ssh
```

Note Optionally you can configure the virtual terminal line settings.

- Enters line configuration mode to configure the virtual terminal line settings. For the *line_number* and *ending_line_number* arguments, the range is from 0 to 15.
- Specifies that the device prevents non-SSH Telnet connections, limiting the device to only SSH connections.

Step 5 Use the **end** command to exit line configuration mode and return to privileged EXEC mode.

Example:

```
IR8100(config-line)# end
```

SSH configuration and status commands

This topic provides information about commands for displaying the SSH server configuration and status.

Table 1: Commands for displaying the SSH server configuration and status

Command	Purpose
show ip ssh	Shows the version and configuration information for the SSH server.
show ssh	Shows the status of the SSH server.

Configure the router for local authentication and authorization

Before you begin

Note To secure the router for HTTP access by using AAA methods, you must configure the router with the ip http authentication aaa global configuration command. Configuring AAA authentication does not secure the router for HTTP access by using AAA methods.

You can configure AAA to operate without a server by setting the switch to implement AAA in local mode. The router then handles authentication and authorization. No accounting is available in this configuration.

Follow these steps to configure AAA to operate without a server by setting the router to implement AAA in local mode:

Procedure

Step 1 Use the **configure terminal** command to enter global configuration mode.

Example:

```
IR8100# configure terminal
```

Step 2 Use the **aaa new-model** command to enable AAA.

Example:

```
IR8100(config)# aaa new-model
```

Step 3 Use the **aaa authentication login default local** command to set the login authentication to use the local username database.

Example:

```
IR8100(config)# aaa authentication login default local
```

Note The default keyword applies the local user database authentication to all ports.

Step 4 Use the **aaa authorization exec local** command to configure user AAA authorization, check the local database, and allow the user to run an EXEC shell.

Example:

```
IR8100(config-line)#  aaa authorization exec local
```

Step 5 Use the **aaa authorization network local** command to configure user AAA authorization for all network-related service requests.

Example:

```
IR8100(config-line)# aaa authorization network local
```

Step 6 Use the **username** *name* **privilege** *level* **password** *encryption-type password* command to enter the local database, and establishes a username-based authentication system.

Example:

```
IR8100(config-line)# username your_user_name privilege 1 password 7 secret567
```

Note Repeat this command for each user.

a. For *name* , specify the user ID as one word. Spaces and quotation marks are not allowed.

b. (Optional) For *level* , specify the privilege level the user has after gaining access. The range is 0 to 15. Level 15 gives privileged EXEC mode access. Level 0 gives user EXEC mode access.

c. For encryption-type, enter 0 to specify that an unencrypted password follows. Enter 7 to specify that a hidden password follows.

d. For password, specify the password the user must enter to gain access to the switch. The password must be from 1 to 25 characters, can contain embedded spaces, and must be the last option specified in the username command.

Step 7 Use the **end** command to exit line configuration mode and return to privileged EXEC mode.

Example:

```
IR8100(config-line)# end
```

Secure copy

The Secure Copy Protocol (SCP) feature provides a secure and authenticated method for copying router configuration or router image files.

SCP relies on Secure Shell (SSH), an application and a protocol that provide a secure replacement for the Berkeley r-tools.

Prerequisites for secure copy

These are the prerequisites for configuring the device for secure shell (SSH):

- Before enabling SCP, you must correctly configure SSH, authentication, and authorization on the switch.
- Because SCP relies on SSH for its secure transport, the router must have an RSA key pair.
- SCP relies on SSH for security.
- SCP requires that authentication, authorization, and accounting (AAA) authorization be configured so the router can determine whether the user has the correct privilege level.
- A user must have appropriate authorization to use SCP.
- A user who has appropriate authorization can use SCP to copy any file in the Cisco IOS File System (IFS) to and from a switch by using the **copy** command. An authorized administrator can also do this from a workstation.

Restrictions for configuring secure copy

This section describes the restrictions for configuring the IR1800 for secure copy (SCP).

- Before enabling SCP, you must correctly configure SSH, authentication, and authorization on the router.
- When using SCP, you cannot enter the password into the **copy** command. You must enter the password when prompted.

Configure secure copy server

To configure the Cisco IR8100 for Secure Copy (SCP) server-side functionality, perform these steps.

Procedure

Step 1 Use the **enable** command to enable privileged EXEC mode.

Example:

```
Router> enable
```

Step 2 Use the **configure terminal** command to enter global configuration mode.

Example:

```
Router# configure terminal
```

Step 3 Use the **aaa new-model** command to set AAA authentication at login.

Example:

```
Router(config)# aaa new-model
```

Step 4 Use the **aaa authentication login** { **default** | *list-name* } *method1* [*method2...*] command to enable the AAA access control system.

Example:

```
Router(config)# aaa authentication login default group tacacs+
```

Step 5 Use the **username** *name* [**privilege** *level*] **password** *encryption-type encrypted-password* command to establish a username-based authentication system.

Example:

```
Router(config)# username superuser privilege 2 password 0 superpassword
```

Note You may omit this step if a network-based authentication mechanism, such as TACACS+ or RADIUS, has been configured.

Step 6 Use the **ip scp server enable** command to enable SCP server-side functionality.

Example:

```
Router(config)# ip scp server enable
```

Step 7 Use the **exit** command to exit global configuration mode and return to privileged EXEC mode

Example:

```
Router(config)# exit
```

Step 8 Use the **show running-config** command to optionally display the SCP server-side functionality.

Example:

```
Router# show running-config
```

Step 9 Use the **debug ip scp** command to optionally troubleshoot SCP authentication problems.

Example:

```
Router# debug ip scp
```

```
IR8100#   copy scp <somefile> your_username@remotehost:/ <some/remote/directory>
```

Additional references

These sections provide references related to the SSH feature.

Table 2: References

Related Topic	Document Title
Configuring Identity Control policies and Identity Service templates for Session Aware networking.	Session Aware Networking Configuration Guide, Cisco IOS XE Release 3SE: https://www.cisco.com/en/US/docs/ios-xml/ios/san/configuration/xe-3se/3850/san-xe-3se-3850-book.pdf
Configuring RADIUS, TACACS+, Secure Shell, 802.1X and AAA.	Secure Shell Configuration Guide, Cisco IOS XE Gibraltar 16.11.x: https://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst9500/software/release/16-11/configuration_guide/sec/b_1611_sec_9500_cg/configuring_secure_shell__ssh_.html