Secure Shell Configuration Guide Cisco IOS XE Release 2

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

Configuring Secure Shell 1
   Finding Feature Information 1
   Prerequisites for Configuring SSH 1
   Restrictions for Configuring SSH 2
   Information About Secure Shell 2
      How SSH Works 2
      SSH Server 2
      SSH Integrated Client 3
      Related Features and Technologies 3
   How to Configure SSH 3
      Configuring SSH Server 4
      Verifying SSH 4
   Troubleshooting Tips 5
   Monitoring and Maintaining SSH 5
   SSH Configuration Examples 5
      SSH on a Cisco ASR1000 Series Router Example 6
   Additional References 6
   Feature Information for Configuring Secure Shell 8

Reverse SSH Enhancements 9
   Finding Feature Information 9
   Prerequisites for Reverse SSH Enhancements 9
   Restrictions for Reverse SSH Enhancements 9
   Information About Reverse SSH Enhancements 10
      Reverse Telnet 10
      Reverse SSH 10
   How to Configure Reverse SSH Enhancements 10
      Configuring Reverse SSH for Console Access 11
      Configuring Reverse SSH for Modem Access 12
      Troubleshooting Reverse SSH on the Client 14
Configuring Secure Shell

This chapter describes the Secure Shell (SSH) feature. The SSH feature consists of an application and a protocol.

- Finding Feature Information, page 1
- Prerequisites for Configuring SSH, page 1
- Restrictions for Configuring SSH, page 2
- Information About Secure Shell, page 2
- How to Configure SSH, page 3
- Troubleshooting Tips, page 5
- Monitoring and Maintaining SSH, page 5
- SSH Configuration Examples, page 5
- Additional References, page 6
- Feature Information for Configuring Secure Shell, page 8

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the Feature Information Table at the end of this document.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for Configuring SSH

Prior to configuring SSH, perform the following tasks:

- Download the required image on your router. (The SSH server requires you to have an IPsec (DES or 3DES) encryption software image downloaded on your router; the SSH client requires you to have an IPsec (DES or 3DES) encryption software image downloaded on your router.) For more information on downloading a software image, see the Cisco IOS XE Configuration Fundamentals Configuration Guide, Release 2.
- Configure a host name and host domain for your router.

To configure a host name and host domain, enter the `hostname hostname` and `ip domain-name domainname` commands in global configuration mode:
• Generate an RSA key pair for your router, which automatically enables SSH.

To generate an RSA key pair, enter the `crypto key generate rsa` command.

---

**Note**

To delete the RSA key-pair, use the `crypto key zeroize rsa` global configuration command. Once you delete the RSA key-pair, you automatically disable the SSH server.

---

• Configure user authentication for local or remote access. You can configure authentication with or without AAA. For more information on AAA, see the Authentication, Authorization, and Accounting chapters in the *Cisco IOS XE Security Configuration Guide: Securing User Services*, Release 2 and the *Cisco IOS Security Command Reference*.

**Restrictions for Configuring SSH**

• RSA authentication available in SSH clients is not supported in the SSH server for Cisco IOS XE software.

• SSH server and SSH client are supported on DES (56-bit) and 3DES (168-bit) data encryption software images only.

• Execution shell is the only application supported.

• The login banner is not supported in Secure Shell Version 1. It is supported in Secure Shell Version 2.

### Information About Secure Shell

Secure Shell (SSH) is an application and a protocol that provide a secure replacement to the Berkeley r-tools. The protocol secures the sessions using standard cryptographic mechanisms, and the application can be used similarly to the Berkeley `rexec` and `rsh` tools. There are currently two versions of SSH available: SSH Version 1 and SSH Version 2. This document describes SSH Version 1. For information about SSH Version 2, see the Secure Shell Version 2 Support document.

---

**Note**

Hereafter, unless otherwise noted, the term “SSH” will denote “SSH Version 1” only.

---

• How SSH Works, page 2

• Related Features and Technologies, page 3

### How SSH Works

• SSH Server, page 2

• SSH Integrated Client, page 3

### SSH Server

The SSH Server feature enables a SSH client to make a secure, encrypted connection to a Cisco router. This connection provides functionality that is similar to that of an inbound Telnet connection. Before SSH,
security was limited to Telnet security. SSH allows a strong encryption to be used with the Cisco IOS XE software authentication. The SSH server in Cisco IOS XE software will work with publicly and commercially available SSH clients.

**SSH Integrated Client**

The SSH Integrated Client feature is an application running over the SSH protocol to provide device authentication and encryption. The SSH client enables a Cisco router to make a secure, encrypted connection to another Cisco router or to any other device running the SSH server. This connection provides functionality that is similar to that of an outbound Telnet connection except that the connection is encrypted. With authentication and encryption, the SSH client allows for a secure communication over an insecure network.

The SSH client in the Cisco IOS XE software works with publicly and commercially available SSH servers. The SSH client supports the ciphers of Data Encryption Standard (DES), Triple DES (3DES), and password authentication. User authentication is performed like that in the Telnet session to the router. The user authentication mechanisms supported for SSH are RADIUS, TACACS+ and the use of locally stored user names and passwords.

**Note**

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

**Related Features and Technologies**

For more information about SSH-related features and technologies, review the following:

- Authentication, Authorization, and Accounting (AAA) feature. AAA is a suite of network security services that provide the primary framework through which access control can be set up on your Cisco router or access server. For more information on AAA, see the Authentication, Authorization, and Accounting chapters in the *Cisco IOS XE Security Configuration Guide: Securing User Services*, Release 2 and the *Cisco IOS Security Command Reference*.
- IP Security (IPsec) feature. IPsec is a framework of open standards that provides data confidentiality, data integrity, and data authentication between participating peers. IPsec provides these security services at the IP layer. IPsec uses Internet Key Exchange (IKE) to handle negotiation of protocols and algorithms based on local policy and to generate the encryption and authentication keys to be used by IPsec. IPsec can be used to protect one or more data flows between a pair of hosts, between a pair of security gateways, or between a security gateway and a host. For more information on IPsec, see the chapter Configuring Security for VPNs with IPsec and the *Cisco IOS Security Command Reference*.

**How to Configure SSH**

- Configuring SSH Server, page 4
- Verifying SSH, page 4
Configuring SSH Server

**Note**
The SSH client feature runs in user EXEC mode and has no specific configuration on the router.

**Note**
The SSH commands are optional and are disabled when the SSH server is disabled.

To enable and configure a Cisco Router for SSH, you can configure SSH parameters. If you do not configure SSH parameters, the default values will be used.

To configure SSH server, use the following command in global configuration mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router(config)# <strong>ip ssh</strong> {{timeout seconds }</td>
<td>(Required) Configures SSH control variables on your router. }</td>
</tr>
</tbody>
</table>

Verifying SSH

To verify that the SSH server is enabled and view the version and configuration data for your SSH connection, use the **show ip ssh** command. The following example shows that SSH is enabled:

```
Router# **show ip ssh**
SSH Enabled - version 1.5
Authentication timeout: 120 secs; Authentication retries: 3
```

The following example shows that SSH is disabled:

```
Router# **show ip ssh**
%SSH has not been enabled
```

To verify the status of your SSH server connections, use the **show ssh** command. The following example shows the SSH server connections on the router when SSH is enabled:

```
Router# **show ssh**
Connection  Version  Encryption  State  Username
 0  1.5  3DES  Session Started  guest
```
The following example shows that SSH is disabled:

```
Router# show ssh
%No SSH server connections running.
```

## Troubleshooting Tips

- If your SSH configuration commands are rejected as illegal commands, you have not successfully generated a RSA key pair for your router. Make sure you have specified a host name and domain. Then use the `crypto key generate rsa` command to generate a RSA key pair and enable the SSH server.
- When configuring the RSA key pair, you might encounter the following error messages:
  - No hostname specified
  - No domain specified

You must configure a host name for the router using the `hostname` global configuration command. For more information, see “Prerequisites for Configuring SSH, page 1.”

- The number of allowable SSH connections is limited to the maximum number of vty's configured for the router. Each SSH connection will use a vty resource.
- SSH uses either local security or the security protocol that is configured through AAA on your router for user authentication. When configuring AAA, you must ensure that AAA is disabled on the console for user authentication. AAA authorization is disabled on the console by default. If AAA authorization is enabled on the console, disable it by configuring the `no aaa authorization console` command during the AAA configuration stage.

## Monitoring and Maintaining SSH

To monitor and maintain your SSH connections, use the following commands in user EXEC mode:

<table>
<thead>
<tr>
<th>Command</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router# show ip ssh</td>
<td>Displays the version and configuration data for SSH.</td>
</tr>
<tr>
<td>Router# show ssh</td>
<td>Displays the status of SSH server connections.</td>
</tr>
</tbody>
</table>

## SSH Configuration Examples

This section provides the following configuration example showing output from the `show running configuration` EXEC command on a Cisco ASR1000 Series Aggregation Services Router.

- SSH on a Cisco ASR1000 Series Router Example, page 6
The crypto key generate rsa command is not displayed in the show running configuration output.

- SSH on a Cisco ASR1000 Series Router Example, page 6

### SSH on a Cisco ASR1000 Series Router Example

In the following example, SSH is configured on a Cisco ASR1000 series router with a timeout that is not to exceed 60 seconds, and no more than 2 authentication retries. Also, before configuring the SSH server feature on the router, TACACS+ is specified as the method of authentication.

```
hostname RouterASR1K
aaa new-model
aaa authentication login default tacacs+
aaa authentication login aaa7200kw none
enable password enableasr1kpw
username username1 password 0 password1
username username2 password 0 password2
ip subnet-zero
no ip domain-lookup
ip domain-name cisco.com
!
Enter the ssh commands.
ip ssh time-out 60
ip ssh authentication-retries 2
ip route 192.168.1.0 255.255.255.0 10.1.10.1
ip route 192.168.9.0 255.255.255.0 10.1.1.1
ip route 192.168.10.0 255.255.255.0 10.1.1.1
map-list atm
ip 10.1.10.1 atm-vc 7 broadcast
no cdp run
tacacs-server host 192.168.109.216 port 9000
tacacs-server key cisco
radius-server host 192.168.109.216 auth-port 1650 acct-port 1651
radius-server key cisco
line con 0
exec-timeout 0 0
login authentication aaa7200kw
transport input none
line aux 0
line vty 0 4
password enableasr1kpw
end
```

### Additional References

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS commands</td>
<td>Cisco IOS Master Commands List, All Releases</td>
</tr>
<tr>
<td>Security commands</td>
<td>Cisco IOS Security Command Reference</td>
</tr>
<tr>
<td>Related Topic</td>
<td>Document Title</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>AAA configuration</td>
<td>The following chapters of the <em>Cisco IOS XE Security Configuration Guide: Securing User Services</em>, Release 2:</td>
</tr>
<tr>
<td></td>
<td>• Configuring Authentication</td>
</tr>
<tr>
<td></td>
<td>• Configuring Authorization</td>
</tr>
<tr>
<td></td>
<td>• Configuring Accounting</td>
</tr>
<tr>
<td>IPSec configuration</td>
<td><em>Configuring Security for VPNs with IPsec</em></td>
</tr>
</tbody>
</table>

## Standards

<table>
<thead>
<tr>
<th>Standard</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>No new or modified MIBs are supported by this feature, and support for existing MIBs has not been modified by this feature. --</td>
</tr>
</tbody>
</table>

## MIBs

<table>
<thead>
<tr>
<th>MIB</th>
<th>MIBs Link</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: <a href="http://www.cisco.com/go/mibs">http://www.cisco.com/go/mibs</a></td>
</tr>
</tbody>
</table>

## RFCs

<table>
<thead>
<tr>
<th>RFC</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature. --</td>
</tr>
</tbody>
</table>

## Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.</td>
<td><a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a></td>
</tr>
</tbody>
</table>
Feature Information for Configuring Secure Shell

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 1  Feature Information for Configuring Secure Shell

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Configuration Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Secure Shell SSH Version 1</td>
<td></td>
<td>The SSH Version 1 Integrated Client feature is an application running over the SSH protocol to provide device authentication and encryption.</td>
</tr>
<tr>
<td>Integrated Client</td>
<td>Cisco IOS XE Release 2.1</td>
<td>In Cisco IOS XE Release 2.1, this feature was introduced on Cisco ASR 1000 Series Aggregation Services Routers.</td>
</tr>
<tr>
<td>Secure Shell SSH Version 1</td>
<td></td>
<td>The SSH Version 1 Server Support feature enables a SSH client to make a secure, encrypted connection to a Cisco router.</td>
</tr>
<tr>
<td>Server Support</td>
<td>Cisco IOS XE Release 2.1</td>
<td>In Cisco IOS XE Release 2.1, this feature was introduced on Cisco ASR 1000 Series Routers.</td>
</tr>
</tbody>
</table>

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.
Reverse SSH Enhancements

The Reverse SSH Enhancements feature provides an alternative method of configuring reverse Secure Shell (SSH). Using this feature, you can configure reverse SSH without having to list separate lines for every terminal or auxiliary line on which SSH has to be enabled. This feature also eliminates the rotary-group limitation. This feature is supported for SSH Version 1 and SSH Version 2.

- Finding Feature Information, page 9
- Prerequisites for Reverse SSH Enhancements, page 9
- Restrictions for Reverse SSH Enhancements, page 9
- Information About Reverse SSH Enhancements, page 10
- How to Configure Reverse SSH Enhancements, page 10
- Configuration Examples for Reverse SSH Enhancements, page 16
- Additional References, page 17
- Feature Information for Reverse SSH Enhancements, page 18

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the Feature Information Table at the end of this document.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for Reverse SSH Enhancements

- SSH must be enabled.
- The SSH client and server must be running the same version of SSH.

Restrictions for Reverse SSH Enhancements

- The -l keyword and userid: number ip-address delimiter and arguments are mandatory when configuring the alternative method of Reverse SSH for console access.
Information About Reverse SSH Enhancements

- Reverse Telnet, page 10
- Reverse SSH, page 10

Reverse Telnet

Cisco IOS XE software has for quite some time included a feature called Reverse Telnet, whereby you can telnet to a certain port range and connect to terminal or auxiliary lines. Reverse telnetting has often been used to connect a router that has many terminal lines to the consoles of other routers or to other devices. Telnetting makes it easy to reach the router console from anywhere simply by telnetting to the terminal server on a specific line. This telnetting approach can be used to configure a router even if all network connectivity to that router is disconnected. Reverse telnetting also allows modems that are attached to routers to be used for dial-out (usually with a rotary device).

Reverse SSH

Reverse telnetting can be accomplished using SSH. Unlike reverse telnetting, SSH provides for secure connections. The Reverse SSH Enhancements feature provides you with a simplified method of configuring SSH. Using this feature, you no longer have to configure a separate line for every terminal or auxiliary line on which you want to enable SSH. The previous method of configuring reverse SSH limited the number of ports that can be accessed to 100. The Reverse SSH Enhancements feature removes the port number limitation. For information on the alternative method of configuring reverse SSH, see the section “How to Configure Reverse SSH Enhancements, page 10.”

How to Configure Reverse SSH Enhancements

- Configuring Reverse SSH for Console Access, page 11
- Configuring Reverse SSH for Modem Access, page 12
- Troubleshooting Reverse SSH on the Client, page 14
- Troubleshooting Reverse SSH on the Server, page 15
Configuring Reverse SSH for Console Access

SUMMARY STEPS

1. enable
2. configure terminal
3. line line-number ending-line-number
4. no exec
5. login authentication listname
6. transport input ssh
7. exit
8. exit
9. ssh -l userid : {number} {ip-address}

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router&gt; enable</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> line line-number ending-line-number</td>
<td>Identifies a line for configuration and enters line configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router# line 1 3</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> no exec</td>
<td>Disables EXEC processing on a line.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router (config-line)# no exec</td>
<td></td>
</tr>
</tbody>
</table>
### Command or Action | Purpose
---|---
#### Step 5 login authentication listname
Example:
```
Router (config-line)# login authentication default
```
Defines a login authentication mechanism for the lines.
**Note** The authentication method must use a username and password.

#### Step 6 transport input ssh
Example:
```
Router (config-line)# transport input ssh
```
Defines which protocols to use to connect to a specific line of the router.
- The **ssh** keyword must be used for the Reverse SSH Enhancements feature.

#### Step 7 exit
Example:
```
Router (config-line)# exit
```
Exits line configuration mode.

#### Step 8 exit
Example:
```
Router (config)# exit
```
Exits global configuration mode.

#### Step 9 ssh -l userid : {number} {ip-address}
Example:
```
Router# ssh -l lab:1 router.example.com
```
Specifies the user ID to use when logging in on the remote networking device that is running the SSH server.
- **userid** --User ID.
- `:` --Signifies that a port number and terminal IP address will follow the userid argument.
- **number** --Terminal or auxiliary line number.
- **ip-address** --Terminal server IP address.

**Note** The `userid` argument and `:revolution{number}|ip-address` delimiter and arguments are mandatory when configuring the alternative method of Reverse SSH for modem access.

---

**Configuring Reverse SSH for Modem Access**

Reverse SSH is configured on a modem used for dial-out lines. To get any of the dial-out modems, you can use any SSH client and start a SSH session as shown (in Step 10) to get to the next available modem from the rotary device.
### SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `line line-number ending-line-number`
4. `no exec`
5. `login authentication listname`
6. `rotary group`
7. `transport input ssh`
8. `exit`
9. `exit`
10. `ssh -l userid :rotary {number} {ip-address}`

### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td></td>
<td>Router&gt; enable</td>
</tr>
<tr>
<td><strong>Step 2</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Router# configure terminal</td>
</tr>
<tr>
<td><strong>Step 3</strong> line line-number ending-line-number</td>
<td>Identifies a line for configuration and enters line configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Router# line 1 200</td>
</tr>
<tr>
<td><strong>Step 4</strong> no exec</td>
<td>Disables EXEC processing on a line.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Router (config-line)# no exec</td>
</tr>
<tr>
<td><strong>Step 5</strong> login authentication listname</td>
<td>Defines a login authentication mechanism for the lines.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td>Router (config-line)# login authentication default</td>
</tr>
<tr>
<td><strong>Note</strong></td>
<td>The authentication method must use a username and password.</td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
</tr>
<tr>
<td>------------------</td>
<td>---------</td>
</tr>
<tr>
<td><strong>Step 6</strong> rotary group</td>
<td>Defines a group of lines consisting of one or more virtual terminal lines or one auxiliary port line.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router (config-line)# rotary 1</td>
<td></td>
</tr>
<tr>
<td><strong>Step 7</strong> transport input ssh</td>
<td>Defines which protocols to use to connect to a specific line of the router.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router (config-line)# transport input ssh</td>
<td>• The <code>ssh</code> keyword must be used for the Reverse SSH Enhancements feature.</td>
</tr>
<tr>
<td><strong>Step 8</strong> exit</td>
<td>Exits line configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router (config-line)# exit</td>
<td></td>
</tr>
<tr>
<td><strong>Step 9</strong> exit</td>
<td>Exits global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router (config)# exit</td>
<td></td>
</tr>
<tr>
<td><strong>Step 10</strong> <code>ssh -l userid:rotary {number} {ip-address}</code></td>
<td>Specifies the user ID to use when logging in on the remote networking device that is running the SSH server.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router# ssh -l lab:rotary1 router.example.com</td>
<td>• <code>userid</code> --User ID.</td>
</tr>
<tr>
<td></td>
<td>• <code>:</code> --Signifies that a port number and terminal IP address will follow the userid argument.</td>
</tr>
<tr>
<td></td>
<td>• <code>number</code> --Terminal or auxiliary line number.</td>
</tr>
<tr>
<td></td>
<td>• <code>ip-address</code> --Terminal server IP address.</td>
</tr>
<tr>
<td><strong>Note</strong></td>
<td>The <code>userid</code> argument and <code>:rotary {number} {ip-address}</code> delimiter and arguments are mandatory when configuring the alternative method of Reverse SSH for modem access.</td>
</tr>
</tbody>
</table>

**Troubleshooting Reverse SSH on the Client**

To troubleshoot the reverse SSH configuration on the client (remote device), perform the following steps.

**SUMMARY STEPS**

1. enable
2. debug ip ssh client
DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td></td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td>Example:</td>
<td>Router&gt; enable</td>
</tr>
</tbody>
</table>

| Step 2 debug ip ssh client | Displays debugging messages for the SSH client. |
| Example:                   | Router# debug ip ssh client |

Troubleshooting Reverse SSH on the Server

To troubleshoot the reverse SSH configuration on the terminal server, perform the following steps. The steps may be configured in any order or independent of one another.

SUMMARY STEPS

1. enable
2. debug ip ssh
3. show ssh
4. show line

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td></td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td>Example:</td>
<td>Router&gt; enable</td>
</tr>
</tbody>
</table>

| Step 2 debug ip ssh | Displays debugging messages for the SSH server. |
| Example:            | Router# debug ip ssh |
### Command or Action | Purpose
--- | ---
**Step 3** | **show ssh** Displays the status of the SSH server connections.
**Example:**
Router# show ssh

**Step 4** | **show line** Displays parameters of a terminal line.
**Example:**
Router# show line

---

**Configuration Examples for Reverse SSH Enhancements**

- Example Reverse SSH Console Access, page 16
- Example Reverse SSH Modem Access, page 16

**Example Reverse SSH Console Access**

The following configuration example shows that reverse SSH has been configured for console access for terminal lines 1 through 3:

**Terminal Server Configuration**

```
line 1 3
no exec
login authentication default
transport input ssh
```

**Client Configuration**

The following commands configured on the SSH client will form the reverse SSH session with lines 1, 2, and 3, respectively:

```
ssh -l lab:1 router.example.com
ssh -l lab:2 router.example.com
ssh -l lab:3 router.example.com
```

**Example Reverse SSH Modem Access**

The following configuration example shows that dial-out lines 1 through 200 have been grouped under rotary group 1 for modem access:

```
line 1 200
no exec
login authentication default
rotary 1
```
The following command shows that reverse SSH will connect to the first free line in the rotary group:

```
ssh -l lab:rotary1 router.example.com
```
Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.</td>
<td><a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a></td>
</tr>
</tbody>
</table>

Feature Information for Reverse SSH Enhancements

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to [www.cisco.com/go/cfn](http://www.cisco.com/go/cfn). An account on Cisco.com is not required.

Table 2  Feature Information for Reverse SSH Enhancements

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Configuration Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Reverse SSH Enhancements</td>
<td>Cisco IOS XE Release 2.1</td>
<td>The Reverse SSH Enhancements feature provides an alternative method of configuring reverse Secure Shell (SSH). Using this feature, you can configure reverse SSH without having to list separate lines for every terminal or auxiliary line on which SSH has to be enabled. This feature also eliminates the rotary-group limitation. This feature is supported for SSH Version 1 and SSH Version 2. In Cisco IOS XE Release 2.1, this feature was introduced on Cisco ASR 1000 Series Aggregation Services Routers. The following command was introduced or modified: <code>ssh</code>.</td>
</tr>
</tbody>
</table>
Reverse SSH Enhancements

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.
Secure Copy

The Secure Copy (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.

- Finding Feature Information, page 21
- Prerequisites for Secure Copy, page 21
- Information About Secure Copy, page 21
- How to Configure SCP, page 22
- Configuration Examples for Secure Copy, page 24
- Additional References, page 25
- Feature Information for Secure Copy, page 26
- Glossary, page 27

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the Feature Information Table at the end of this document.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for Secure Copy

- Before enabling SCP, you must correctly configure SSH, authentication, and authorization on the router.
- Because SCP relies on SSH for its secure transport, the router must have an Rivest, Shamir, and Adelman (RSA) key pair.

Information About Secure Copy

- How SCP Works, page 22
How SCP Works

The behavior of SCP is similar to that of remote copy (rcp), which comes from the Berkeley r-tools suite, except that SCP relies on SSH for security. In addition, SCP requires that authentication, authorization, and accounting (AAA) authorization be configured so the router can determine whether the user has the correct privilege level.

SCP allows a user who has appropriate authorization to copy any file that exists in the Cisco IOS XE File System (IFS) to and from a router by using the `copy` command. An authorized administrator may also perform this action from a workstation.

How to Configure SCP

- Configuring SCP, page 22
- Verifying SCP, page 23
- Troubleshooting SCP, page 24

Configuring SCP

To enable and configure a Cisco router for SCP server-side functionality, perform the following steps.

SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `aaa new-model`
4. `aaa authentication login {default | list-name} method1[method2...]`
5. `aaa authorization {network | exec | commands level | reverse-access | configuration} {default | list-name} [method1 [method2...]]`
6. `username name [privilege level] {password encryption-type encrypted-password}`
7. `ip scp server enable`

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 <code>enable</code></td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td><code>Router&gt; enable</code></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td>Command or Action</td>
<td>Purpose</td>
</tr>
<tr>
<td>-------------------</td>
<td>---------</td>
</tr>
<tr>
<td><strong>Step 2</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> aaa new-model</td>
<td>Sets AAA authentication at login.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router (config)# aaa new-model</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> aaa authentication login {default</td>
<td>list-name} method1[method2...]</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router (config)# aaa authentication login default group tacacs+</td>
<td></td>
</tr>
<tr>
<td><strong>Step 5</strong> aaa authorization {network</td>
<td>exec</td>
</tr>
<tr>
<td><strong>Note</strong> The exec keyword runs authorization to determine if the user is allowed to run an EXEC shell; therefore, you must use it when you configure SCP.</td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router (config)# aaa authorization exec default group tacacs+</td>
<td></td>
</tr>
<tr>
<td><strong>Step 6</strong> username name [privilege level] [password encryption-type encrypted-password]</td>
<td>Establishes a username-based authentication system.</td>
</tr>
<tr>
<td><strong>Note</strong> You may skip this step if a network-based authentication mechanism--such as TACACS + or RADIUS--has been configured.</td>
<td></td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router (config)# username superuser privilege 2 password 0 superpassword</td>
<td></td>
</tr>
<tr>
<td><strong>Step 7</strong> ip scp server enable</td>
<td>Enables SCP server-side functionality.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router (config)# ip scp server enable</td>
<td></td>
</tr>
</tbody>
</table>

**Verifying SCP**

To verify SCP server-side functionality, perform the following steps.
SUMMARY STEPS
1. enable
2. show running-config

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td></td>
<td>• Enter your password if prompted.</td>
</tr>
</tbody>
</table>

Example:
Router> enable

Step 2 show running-config

Example:
Router# show running-config

Troubleshooting SCP

SUMMARY STEPS
1. enable
2. debug ip scp

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td></td>
<td>• Enter your password if prompted.</td>
</tr>
</tbody>
</table>

Example:
Router> enable

Step 2 debug ip scp

Example:
Router# debug ip scp

Configuration Examples for Secure Copy
Example SCP Server-Side Configuration Using Local Authentication

The following example shows how to configure the server-side functionality of SCP. This example uses a locally defined username and password:

```plaintext
aaa authentication and authorization must be configured properly for SCP to work.
aaa new-model
aaa authentication login default local
aaa authorization exec default local
username tiger privilege 15 password 0 lab

! SSH must be configured and functioning properly.
ip ssh time-out 120
ip ssh authentication-retries 3
ip scp server enable
```

Example SCP Server-Side Configuration Using Network-Based Authentication

The following example shows how to configure the server-side functionality of SCP using a network-based authentication mechanism:

```plaintext
aaa authentication and authorization must be configured properly for SCP to work.
aaa new-model
aaa authentication login default group tacacs+
aaa authorization exec default group tacacs+

! SSH must be configured and functioning properly.
ip ssh time-out 120
ip ssh authentication-retries 3
ip scp server enable
```

Additional References

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS commands</td>
<td>Cisco IOS Master Commands List, All Releases</td>
</tr>
<tr>
<td>Security commands: complete</td>
<td>Cisco IOS Security Command Reference</td>
</tr>
<tr>
<td>command syntax, command modes,</td>
<td></td>
</tr>
<tr>
<td>command history, defaults, usage</td>
<td></td>
</tr>
<tr>
<td>guidelines, and examples</td>
<td></td>
</tr>
<tr>
<td>Secure Shell</td>
<td>Configuring Secure Shell and Secure Shell Version 2 Support feature modules.</td>
</tr>
<tr>
<td>Configuring authentication and</td>
<td>Configuring Authentication, Configuring Authorization, and Configuring Accounting feature modules.</td>
</tr>
<tr>
<td>authorization</td>
<td></td>
</tr>
</tbody>
</table>
## Standards

<table>
<thead>
<tr>
<th>Standards</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>--</td>
</tr>
</tbody>
</table>

## MIBs

<table>
<thead>
<tr>
<th>MIBs</th>
<th>MIBs Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: <a href="http://www.cisco.com/go/mibs">http://www.cisco.com/go/mibs</a></td>
</tr>
</tbody>
</table>

## RFCs

<table>
<thead>
<tr>
<th>RFCs</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>No new or modified RFCs are supported by this feature, and support for existing RFCs has not been modified by this feature.</td>
<td>--</td>
</tr>
</tbody>
</table>

## Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.</td>
<td><a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a></td>
</tr>
</tbody>
</table>

---

### Feature Information for Secure Copy

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to [www.cisco.com/go/cfn](http://www.cisco.com/go/cfn). An account on Cisco.com is not required.
### Table 3  Feature Information for Secure Copy

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Configuration Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Secure Copy</td>
<td>Cisco IOS XE Release 2.1</td>
<td>The Secure Copy (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. In Cisco IOS XE Release 2.1, this feature was introduced on Cisco ASR 1000 Series Aggregation Services Routers. The following commands were introduced or modified: <code>debug ip scp</code>, <code>ip scp server enable</code>.</td>
</tr>
</tbody>
</table>

### Glossary

**AAA** --authentication, authorization, and accounting. Framework of security services that provide the method for identifying users (authentication), for remote access control (authorization), and for collecting and sending security server information used for billing, auditing, and reporting (accounting).

**rcp** --remote copy. Relying on Remote Shell (Berkeley r-tools suite) for security, rcp copies files, such as router images and startup configurations, to and from routers.

**SCP** --secure copy. Relying on SSH for security, SCP support allows the secure and authenticated copying of anything that exists in the Cisco IOS XE File Systems. SCP is derived from rcp.

**SSH** --Secure Shell. Application and a protocol that provide a secure replacement for the Berkeley r-tools. The protocol secures the sessions using standard cryptographic mechanisms, and the application can be used similarly to the Berkeley rexec and rsh tools. SSH Version 1 is implemented in the Cisco IOS XE software.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: [www.cisco.com/go/trademarks](http://www.cisco.com/go/trademarks). Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.
Secure Shell Version 2 Support

The Secure Shell Version 2 Support feature allows you to configure Secure Shell (SSH) Version 2. SSH runs on top of a reliable transport layer and provides strong authentication and encryption capabilities. Currently, the only reliable transport that is defined for SSH is TCP. SSH provides a means to securely access and securely execute commands on another computer over a network. The Secure Copy Protocol (SCP) feature that is provided with SSH also allows for the secure transfer of files.

• Finding Feature Information, page 29
• Prerequisites for Secure Shell Version 2 Support, page 29
• Restrictions for Secure Shell Version 2 Support, page 29
• Information About Secure Shell Version 2 Support, page 30
• How to Configure Secure Shell Version 2 Support, page 31
• Configuration Examples for Secure Shell Version 2 Support, page 42
• Where to Go Next, page 45
• Additional References, page 45
• Feature Information for Secure Shell Version 2 Support, page 47

Finding Feature Information

Your software release may not support all the features documented in this module. For the latest feature information and caveats, see the release notes for your platform and software release. To find information about the features documented in this module, and to see a list of the releases in which each feature is supported, see the Feature Information Table at the end of this document.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Prerequisites for Secure Shell Version 2 Support

Prior to configuring SSH, perform the following task:

• Download the required image on your router. The SSH server requires you to have a k9 (Triple Data Encryption Standard [3DES]) software image from downloaded on your router.

Restrictions for Secure Shell Version 2 Support

• SSH servers and SSH clients are supported in k9 software images.
Execution Shell, remote command execution, and Secure Copy Protocol (SCP) are the only applications supported.

Rivest, Shamir, and Adelman (RSA) key generation is an SSH server side requirement. Routers that act as SSH clients do not need to generate RSA keys.

The RSA key-pair size must be greater than or equal to 768.

The following functionality is not supported:

- RSA user authentication (in the SSH server or SSH client for Cisco IOS XE software)
- Public key authentication
- SSH server strict host key check
- Port forwarding
- Compression

Information About Secure Shell Version 2 Support

- Secure Shell Version 2, page 30
- Secure Shell Version 2 Enhancements, page 31
- SNMP Trap Generation, page 31

Secure Shell Version 2


The configuration for the SSH Version 2 server is similar to the configuration for SSH Version 1. The command `ip ssh version` has been introduced so that you may define which version of SSH that you want to configure. If you do not configure this command, SSH by default runs in compatibility mode; that is, both SSH Version 1 and SSH Version 2 connections are honored.

Note

SSH Version 1 is a protocol that has never been defined in a standard. If you do not want your router to fall back to the undefined protocol (Version 1), you should use the `ip ssh version` command and specify Version 2.

The `ip ssh rsa keypair-name` command was also introduced so that you can enable a SSH connection using RSA keys that you have configured. Previously, SSH was linked to the first RSA keys that were generated (that is, SSH was enabled when the first RSA key pair was generated). The behavior still exists, but by using the `ip ssh rsa keypair-name` command, you can overcome that behavior. If you configure the `ip ssh rsa keypair-name` command with a key-pair name, SSH is enabled if the key pair exists, or SSH will be enabled if the key pair is generated later. If you use this command to enable SSH, you are not forced to configure a host name and a domain name, which was required in SSH Version 1 of the Cisco IOS XE software.

Note

The login banner is supported in Secure Shell Version 2, but it is not supported in Secure Shell Version 1.
Secure Shell Version 2 Enhancements

The Secure Shell Version 2 Enhancements include a number of additional capabilities such as supporting VRF aware SSH, SSH debug enhancements, and Diffie-Hellman group exchange support.

The Cisco IOS XE SSH implementation has traditionally used 768 bit modulus but with an increasing need for higher key sizes to accommodate Diffie-Hellman (DH) Group 14 (2048 bits) and Group 16 (4096 bits) cryptographic applications a message exchange between the client and server to establish the favored DH group becomes necessary. The `ip ssh dh min size` command was introduced so you can configure modulus size on the SSH server. In addition to this the `ssh` command was extended to add VRF awareness to SSH client side functionality through which the VRF instance name in the client is provided with the IP address to look up the correct routing table and establish a connection.

Debugging has been enhanced by modifying SSH debug commands. The `debug ip ssh` command has been extended to allow you to simplify the debugging process. Previously this command printed all debug messages related to SSH regardless of what was specifically required. The behavior still exists, but if you configure the `debug ip ssh` command with a keyword messages are limited to information specified by the keyword.

SNMP Trap Generation

Simple Network Management Protocol (SNMP) traps will be generated automatically when an SSH session terminates if the traps have been enabled and SNMP debugging has been turned on. For information about enabling SNMP traps, see the Configuring SNMP Support feature module.

<table>
<thead>
<tr>
<th>Note</th>
</tr>
</thead>
<tbody>
<tr>
<td>When configuring the <code>snmp-server host</code> command, the IP address must be the address of the PC that has the SSH (telnet) client and that has IP connectivity to the SSH server. See Example Setting an SNMP Trap, page 43 for more information.</td>
</tr>
</tbody>
</table>

You must also turn on SNMP debugging using the `debug snmp packet` command to display the traps. The trap information includes information such as the number of bytes sent and the protocol that was used for the SSH session. See Example SNMP Debugging, page 44 for more information.

How to Configure Secure Shell Version 2 Support

- Configuring a Router for SSH Version 2 Using a Host Name and Domain Name, page 32
- Configuring a Router for SSH Version 2 Using RSA Key Pairs, page 33
- Starting an Encrypted Session with a Remote Device, page 34
- Enabling Secure Copy Protocol on the SSH Server, page 35
- Verifying the Status of the Secure Shell Connection Using the `show ssh` Command, page 37
- Verifying the Secure Shell Status Using the `show ip ssh` Command, page 39
- Monitoring and Maintaining Secure Shell Version 2, page 40
Configuring a Router for SSH Version 2 Using a Host Name and Domain Name

To configure your router for SSH Version 2 using a host name and domain name, perform the following steps. You may also configure SSH Version 2 by using the RSA key pair configuration. See Configuring a Router for SSH Version 2 Using RSA Key Pairs, page 33 for more information.

**SUMMARY STEPS**

1. enable
2. configure terminal
3. hostname hostname
4. ip domain-name name
5. crypto key generate rsa
6. ip ssh [time-out seconds | authentication-retries integer]
7. ip ssh version [1 | 2]

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td></td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router&gt; enable</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> hostname hostname</td>
<td>Configures a host name for your router.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router (config)# hostname cisco 7200</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> ip domain-name name</td>
<td>Configures a domain name for your router.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router (config)# ip domain-name example.com</td>
<td></td>
</tr>
</tbody>
</table>
### Configuring a Router for SSH Version 2 Using RSA Key Pairs

To enable SSH Version 2 without configuring a host name or domain name, perform the following steps. SSH Version 2 will be enabled if the key pair that you configure already exists or if it is generated later. You may also configure SSH Version 2 by using the host name and domain name configuration. See Configuring a Router for SSH Version 2 Using a Host Name and Domain Name, page 32 for more information.

#### SUMMARY STEPS

1. enable
2. configure terminal
3. ip ssh rsa keypair-name keypair-name
4. crypto key generate rsa usage-keys label key-label modulus modulus-size
5. ip ssh [time-out seconds | authentication-retries integer]
6. ip ssh version 2

#### DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step 1 enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td>Example:</td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td></td>
<td>Router&gt; enable</td>
</tr>
</tbody>
</table>
### Command or Action

<table>
<thead>
<tr>
<th>Step 2</th>
<th>configure terminal</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Example:</strong></td>
<td>Router# configure terminal</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 3</th>
<th>ip ssh rsa keypair-name keypair-name</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Example:</strong></td>
<td>Router (config)# ip ssh rsa keypair-name sshkeys</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Step 4</th>
<th>crypto key generate rsa usage-keys label key-label modulus modulus-size</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Example:</strong></td>
<td>Router (config)# crypto key generate rsa usage-keys label sshkeys modulus 768</td>
</tr>
</tbody>
</table>

| Step 5 | ip ssh [time-out seconds | authentication-retries integer] |
|--------|----------------------------------|
| **Example:** | Router (config)# ip ssh time-out 120 |

<table>
<thead>
<tr>
<th>Step 6</th>
<th>ip ssh version 2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Example:</strong></td>
<td>Router (config)# ip ssh version 2</td>
</tr>
</tbody>
</table>

### Starting an Encrypted Session with a Remote Device

To start an encrypted session with a remote networking device, perform the following step. (You do not have to enable your router. SSH can be run in disabled mode.)

**Note**
The device you wish to connect with must support a SSH server that has an encryption algorithm that is supported in Cisco IOS XE software.
SUMMARY STEPS

1. `ssh [-v {1 | 2}][-c {3des | aes128-cbc | aes192-cbc | aes256-cbc}][-m {hmac-md5 | hmac-md5-96 | hmac-sha1 | hmac-sha1-96}][-l userid][-o numberofpasswordpromts n][-p port-num][ip-addr | hostname] [command]`

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> `ssh [-v {1</td>
<td>2}][-c {3des</td>
</tr>
</tbody>
</table>

Example:

```
Router# ssh -v 2 -c aes256-cbc -m hmac-sha1-96 -l user2 10.76.82.24
```

Example:

```
Router# ssh -v 2 -c aes256-cbc -m hmac-sha1-96 user2@10.76.82.24
```

**Troubleshooting Tips**

The `ip ssh version` command can be used for troubleshooting your SSH configuration. By changing versions, you can determine which SSH version has a problem.

Enabling Secure Copy Protocol on the SSH Server

To configure server-side functionality for SCP, perform the following steps. This example shows a typical configuration that allows the router to securely copy files from a remote workstation.
SCP relies on AAA authentication and authorization to function correctly. Therefore AAA must be configured on the router.

**SUMMARY STEPS**

1. enable
2. configure terminal
3. aaa new-model
4. aaa authentication login default local
5. aaa authorization exec default local
6. username name privilege privilege-level password password
7. ip ssh time-out seconds
8. ip ssh authentication-retries integer
9. ip scp server enable

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong> enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td></td>
<td>• Enter your password if prompted.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router&gt; enable</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong> configure terminal</td>
<td>Enters global configuration mode.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router# configure terminal</td>
<td></td>
</tr>
<tr>
<td><strong>Step 3</strong> aaa new-model</td>
<td>Enables the authentication, authorization, and accounting (AAA) access control model.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router(config)# aaa new-model</td>
<td></td>
</tr>
<tr>
<td><strong>Step 4</strong> aaa authentication login default local</td>
<td>Sets authentication, authorization, and accounting (AAA) authentication at login to use the local username database for authentication.</td>
</tr>
<tr>
<td>Example:</td>
<td></td>
</tr>
<tr>
<td>Router(config)# aaa authentication login default local</td>
<td></td>
</tr>
</tbody>
</table>
### Command or Action

<table>
<thead>
<tr>
<th>Step</th>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td><code>aaa authorization exec default local</code></td>
<td>Sets the parameters that restrict user access to a network; runs the authorization to determine if the user ID allowed to run an EXEC shell; and specifies that the system uses the local database for authorization.</td>
</tr>
<tr>
<td>6</td>
<td><code>username name privilege privilege-level password</code></td>
<td>Establishes a username-based authentication system, specifies the username, the privilege level, and an unencrypted password.</td>
</tr>
<tr>
<td>7</td>
<td><code>ip ssh time-out seconds</code></td>
<td>Sets the time interval (in seconds) that the router waits for the SSH client to respond.</td>
</tr>
<tr>
<td>8</td>
<td><code>ip ssh authentication-retries integer</code></td>
<td>Sets the number of authentication attempts after which the interface is reset.</td>
</tr>
<tr>
<td>9</td>
<td><code>ip scp server enable</code></td>
<td>Enables the router to securely copy files from a remote workstation.</td>
</tr>
</tbody>
</table>

### Troubleshooting Tips

- **Troubleshooting Tips, page 37**

### Troubleshooting Tips

To troubleshoot SCP authentication problems, use the `debug ip scp` command.

### Verifying the Status of the Secure Shell Connection Using the show ssh Command

To display the status of the SSH connection on your router, use the `show ssh` command.
SUMMARY STEPS

1. enable
2. show ssh

DETAILED STEPS

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td>enable</td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router&gt; enable</td>
<td></td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td>show ssh</td>
<td>Displays the status of SSH server connections.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>Router# show ssh</td>
<td></td>
</tr>
</tbody>
</table>

Examples

**Version 1 and Version 2 Connections**

**Version 2 Connection with No Version 1**

**Version 1 Connection with No Version 2**

The following sample output from the `show ssh` command display status about various SSH Version 1 and Version 2 connections.

```
Router# show ssh
Connection   Version Encryption      State                   Username
0            1.5     3DES            Session started         lab

Connection Version Mode Encryption  Hmac                State
Username 1          2.0     IN   aes128-cbc  hmac-md5     Session started       lab
1          2.0     OUT  aes128-cbc  hmac-md5     Session started       lab

-------------------------------------------------------------------------

Router# show ssh
Connection   Version Encryption      State                   Username
0            1.5     3DES            Session started         lab

-------------------------------------------------------------------------

%No SSHv1 server connections running.

-------------------------------------------------------------------------

Router# show ssh
Connection   Version Encryption      State                   Username
0            1.5     3DES            Session started         lab
```
Verifying the Secure Shell Status Using the show ip ssh Command

To verify your SSH configuration, perform the following steps.

**SUMMARY STEPS**

1. `enable`
2. `show ip ssh`

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Step 1</strong></td>
<td></td>
</tr>
<tr>
<td><code>enable</code></td>
<td>Enables privileged EXEC mode.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>Router&gt; enable</code></td>
<td>Enter your password if prompted.</td>
</tr>
<tr>
<td><strong>Step 2</strong></td>
<td></td>
</tr>
<tr>
<td><code>show ip ssh</code></td>
<td>Displays the version and configuration data for SSH.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td><code>Router# show ip ssh</code></td>
<td></td>
</tr>
</tbody>
</table>

**Examples**

**Version 1 and Version 2 Connections**

**Version 2 Connection with No Version 1**

**Version 1 Connection with No Version 2**

The following examples from the `show ip ssh` command display the version of SSH that is enabled, the authentication timeout values, and the number of authentication retries.
Monitoring and Maintaining Secure Shell Version 2

To display debug messages about the SSH connections, use the `debug ip ssh` command.

**SUMMARY STEPS**

1. enable
2. debug ip ssh
3. debug snmp packet

**DETAILED STEPS**

<table>
<thead>
<tr>
<th>Command or Action</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| **Step 1** enable | Enables privileged EXEC mode.  
| Example: |  
| Router> enable |  
| **Step 2** debug ip ssh | Displays debugging messages for SSH.  
| Example: |  
| Router# debug ip ssh |  
| **Step 3** debug snmp packet | Displays information about every SNMP packet sent or received by the router.  
| Example: |  
| Router# debug snmp packet |  

**Example**

The following output from the `debug ip ssh` command shows that the digit 2 keyword has been assigned, signifying that it is an SSH Version 2 connection.

```
Router# debug ip ssh
00:33:55: SSH1: starting SSH control process
00:33:55: SSH1: sent protocol version id SSH-1.99-Cisco-1.25
00:33:55: SSH1: protocol version id is - SSH-2.0-OpenSSH_2.5.2p2
00:33:55: SSH2 1: send: len 280 (includes padlen 4)
00:33:55: SSH2 1: SSH2_MSG_KEXINIT sent
00:33:55: SSH2 1: ssh_receive: 536 bytes received
00:33:55: SSH2 1: input: packet len 632
00:33:55: SSH2 1: partial packet 8, need 624, maclen 0
00:33:55: SSH2 1: ssh_receive: 96 bytes received
00:33:55: SSH2 1: partial packet 8, need 624, maclen 0
00:33:55: SSH2 1: input: padlen 11
00:33:55: SSH2 1: received packet type 20
00:33:55: SSH2 1: SSH2_MSG_KEXINIT received
00:33:55: SSH2: kex: client->server aes128-cbc hmac-md5 none
```
00:33:55: SSH2: kex: server->client aes128-cbc hmac-md5 none
00:33:55: SSH2 1: expecting SSH2_MSG_KEXDH_INIT
00:33:55: SSH2 1: ssh_receive: 144 bytes received
00:33:55: SSH2 1: input: packet len 144
00:33:55: SSH2 1: partial packet 8, need 136, maclen 0
00:33:55: SSH2 1: input: padlen 5
00:33:55: SSH2 1: received packet type 30
00:33:55: SSH2 1: SSH2_MSG_KEXDH_INIT received
00:33:55: SSH2 1: signature length 111
00:33:55: SSH2 1: send: len 384 (includes padlen 7)
00:33:55: SSH2: kex_derive_keys complete
00:33:55: SSH2 1: send: len 16 (includes padlen 10)
00:33:55: SSH2 1: newkeys: mode 1
00:33:55: SSH2 1: SSH2_MSG_NEWKEYS sent
00:33:55: SSH2 1: waiting for SSH2_MSG_NEWKEYS
00:33:55: SSH2 1: ssh_receive: 16 bytes received
00:33:55: SSH2 1: input: packet len 16
00:33:55: SSH2 1: partial packet 8, need 8, maclen 0
00:33:55: SSH2 1: input: padlen 10
00:33:55: SSH2 1: newkeys: mode 0
00:33:55: SSH2 1: received packet type 2100:33:55: SSH2 1: SSH2_MSG_NEWKEYS received
00:33:55: SSH2 1: ssh_receive: 48 bytes received
00:33:55: SSH2 1: input: packet len 32
00:33:55: SSH2 1: MAC #3 ok
00:33:55: SSH2 1: send: len 48 (includes padlen 10)
00:33:55: SSH2 1: done calc MAC out #3
00:33:55: SSH2 1: ssh_receive: 64 bytes received
00:33:55: SSH2 1: input: packet len 48
00:33:55: SSH2 1: partial packet 16, need 32, maclen 16
00:33:55: SSH2 1: MAC #4 ok
00:33:55: SSH2 1: input: padlen 9
00:33:55: SSH2 1: received packet type 50
00:33:55: SSH2 1: send: len 32 (includes padlen 13)
00:33:55: SSH2 1: done calc MAC out #5
00:33:55: SSH2 1: ssh_receive: 160 bytes received
00:33:55: SSH2 1: input: packet len 64
00:33:55: SSH2 1: partial packet 16, need 48, maclen 16
00:33:55: SSH2 1: MAC #5 ok
00:33:55: SSH2 1: input: padlen 13
00:33:55: SSH2 1: received packet type 50
00:33:55: SSH2 1: send: len 16 (includes padlen 10)
00:33:55: SSH2 1: done calc MAC out #5
00:33:55: SSH2 1: authentication successful for lab
00:33:55: SSH2 1: input: packet len 64
00:33:55: SSH2 1: partial packet 16, need 48, maclen 16
00:33:55: SSH2 1: MAC #6 ok
00:33:55: SSH2 1: input: padlen 6
00:33:55: SSH2 1: received packet type 2
00:33:55: SSH2 1: ssh_receive: 64 bytes received
00:33:55: SSH2 1: input: packet len 48
00:33:55: SSH2 1: partial packet 16, need 32, maclen 16
00:33:55: SSH2 1: MAC #7 ok
00:33:55: SSH2 1: input: padlen 19
00:33:55: SSH2 1: received packet type 90
00:33:55: SSH2 1: channel open request
00:33:55: SSH2 1: send: len 32 (includes padlen 10)
00:33:55: SSH2 1: done calc MAC out #6
00:33:55: SSH2 1: ssh_receive: 192 bytes received
00:33:55: SSH2 1: input: packet len 64
00:33:55: SSH2 1: partial packet 16, need 48, maclen 16
00:33:55: SSH2 1: MAC #8 ok
00:33:55: SSH2 1: input: padlen 13
00:33:55: SSH2 1: received packet type 98
00:33:55: SSH2 1: pty-reg request
00:33:55: SSH2 1: setting TTY - requested: height 24, width 80; set: height 24, width 80
00:33:55: SSH2 1: input: packet len 96
00:33:55: SSH2 1: partial packet 16, need 80, maclen 16
00:33:55: SSH2 1: MAC #9 ok
00:33:55: SSH2 1: input: padlen 11
Configuration Examples for Secure Shell Version 2 Support

- Example Configuring Secure Shell Version 1, page 43
- Example Configuring Secure Shell Version 2, page 43
- Example Configuring Secure Shell Versions 1 and 2, page 43
- Example Starting an Encrypted Session with a Remote Device, page 43
Example Configuring Secure Shell Version 1

Router# configure terminal
Router (config)# ip ssh version 1
Router (config)# end

Example Configuring Secure Shell Version 2

Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)# ip ssh version 2
Router(config)# end

Example Configuring Secure Shell Versions 1 and 2

Router# configure terminal
Router (config)# no ip ssh version
Router (config)# end

Example Starting an Encrypted Session with a Remote Device

Router# ssh -v 2 -c aes256-cbc -m hmac-shal-160 -l shaship 10.76.82.24

Example Configuring Server-Side SCP

The following example shows how to configure server-side functionality for SCP. This example also configures AAA authentication and Authorization on the router. This example uses a locally defined username and password.

Router# configure terminal
Router (config)# aaa new-model
Router (config)# aaa authentication login default local
Router (config)# aaa authorization exec default local
Router (config)# username samplename privilege 15 password password1
Router (config)# ip ssh time-out 120
Router (config)# ip ssh authentication-retries 3
Router (config)# ip scp server enable
Router (config)# end

Example Setting an SNMP Trap

The following shows that an SNMP trap has been set. The trap notification is generated automatically when the SSH session terminates. For an example of SNMP trap debug output, see the section “Example SNMP Debugging, page 44.”

snmp-server
snmp-server host a.b.c.d public tty

Where a.b.c.d is the IP address of the SSH client.
Example SNMP Debugging

The following is sample output using the `debug snmp packet` command. The output provides SNMP trap information for an SSH session.

```
Router1# debug snmp packet
SNMP packet debugging is on
Router1# ssh -l lab 10.0.0.2
Password: [Connection to 10.0.0.2 closed by foreign host]
Router1# *Jul 18 10:18:42.619: SNMP: Queuing packet to 10.0.0.2
*Jul 18 10:18:42.619: SNMP: V1 Trap, ent cisco, addr 10.0.0.1, gentrap 6, spectrap 1
local.9.3.1.1.2.1 = 6
tcpConnEntry.1.10.0.0.1.22.10.0.0.2.55246 = 4
ltcpConnEntry.5.10.0.0.1.22.10.0.0.2.55246 = 1015
ltcpConnEntry.1.10.0.0.1.22.10.0.0.2.55246 = 1056
ltcpConnEntry.2.10.0.0.1.22.10.0.0.2.55246 = 1392
local.9.2.1.18.2 = lab
*Jul 18 10:18:42.879: SNMP: Packet sent via UDP to 10.0.0.2
```

Example SSH Debugging Enhancements

The following is sample output from the `debug ip ssh detail` command. The output provides debugging information regarding the SSH protocol and channel requests.

```
Router# debug ip ssh detail
00:04:22: SSH0: starting SSH control process
00:04:22: SSH0: sent protocol version id SSH-1.99-Cisco-1.25
00:04:22: SSH0: protocol version id is - SSH-1.99-Cisco-1.25
00:04:22: SSH2: SSH2_MSG_KEXINIT sent
00:04:22: SSH2: SSH2_MSG_KEXINIT received
00:04:22: SSH2: key: client->server enc:aes128-cbc mac:hmac-sha1
00:04:22: SSH2: key: server->client enc:aes128-cbc mac:hmac-sha1
00:04:22: SSH2: expecting SSH2_MSG_KEXDH_INIT
00:04:22: SSH2: SSH2_MSG_KEXDH_INIT received
00:04:22: SSH2: kex_derive_keys complete
00:04:22: SSH2: SSH2_MSG_NEWKEYS sent
00:04:22: SSH2: SSH2_MSG_NEWKEYS received
00:04:24: SSH2: authentication successful for lab
00:04:24: SSH2: channel open request
00:04:24: SSH2: pty-req request
00:04:24: SSH2: setting TTY - requested: height 24, width 80; set: height 24, width 80
00:04:24: SSH2: shell request
00:04:24: SSH2: shell message received
00:04:24: SSH2: starting shell for vty
00:04:38: SSH0: Session terminated normally
```

The following is sample output from the `debug ip ssh packet` command. The output provides debugging information regarding the ssh packet.

```
Router# debug ip ssh packet
00:05:43: SSH2: send:packet of length 280 (length also includes padlen of 4)
00:05:43: SSH2: ssh_receive: 64 bytes received
00:05:43: SSH2: input: total packet length of 280 bytes
00:05:43: SSH2: partial packet length (block size) 8 bytes, needed 272 bytes, maclen 0
00:05:43: SSH2: ssh_receive: 64 bytes received
00:05:43: SSH2: partial packet length (block size) 8 bytes, needed 272 bytes, maclen 0
00:05:43: SSH2: ssh_receive: 64 bytes received
00:05:43: SSH2: partial packet length (block size) 8 bytes, needed 272 bytes, maclen 0
00:05:43: SSH2: ssh_receive: 64 bytes received
00:05:43: SSH2: partial packet length (block size) 8 bytes, needed 272 bytes, maclen 0
00:05:43: SSH2: ssh_receive: 64 bytes received
00:05:43: SSH2: partial packet length (block size) 8 bytes, needed 272 bytes, maclen 0
00:05:43: SSH2: ssh_receive: 24 bytes received
```
Where to Go Next

You have to use a SSH remote device that supports SSH Version 2, and you have to connect to a router.

Additional References

- Related Documents, page 45
- Standards, page 46
- MIBs, page 46
- RFCs, page 46
- Technical Assistance, page 46

Related Documents

<table>
<thead>
<tr>
<th>Related Topic</th>
<th>Document Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cisco IOS commands</td>
<td>Cisco IOS Master Commands List, All Releases</td>
</tr>
<tr>
<td>Security commands: complete command syntax, command modes, command history, defaults, usage guidelines, and examples</td>
<td>Cisco IOS Security Command Reference</td>
</tr>
<tr>
<td>Configuring Secure Shell, a host name and host domain</td>
<td>Configuring Secure Shell feature module.</td>
</tr>
</tbody>
</table>
### Standards

<table>
<thead>
<tr>
<th>Standards</th>
<th>Title</th>
</tr>
</thead>
</table>

### MIBs

<table>
<thead>
<tr>
<th>MIBs</th>
<th>MIBs Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: <a href="http://www.cisco.com/go/mibs">http://www.cisco.com/go/mibs</a></td>
</tr>
</tbody>
</table>

### RFCs

<table>
<thead>
<tr>
<th>RFCs</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>--</td>
</tr>
</tbody>
</table>

### Technical Assistance

<table>
<thead>
<tr>
<th>Description</th>
<th>Link</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Cisco Support and Documentation website provides online resources to download documentation, software, and tools. Use these resources to install and configure the software and to troubleshoot and resolve technical issues with Cisco products and technologies. Access to most tools on the Cisco Support and Documentation website requires a Cisco.com user ID and password.</td>
<td><a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a></td>
</tr>
</tbody>
</table>
Feature Information for Secure Shell Version 2 Support

The following table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Use Cisco Feature Navigator to find information about platform support and Cisco software image support. To access Cisco Feature Navigator, go to www.cisco.com/go/cfn. An account on Cisco.com is not required.

Table 4 Feature Information for Secure Shell Version 2 Support

<table>
<thead>
<tr>
<th>Feature Name</th>
<th>Releases</th>
<th>Feature Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Secure Shell Version 2 Support</td>
<td>Cisco IOS XE Release 2.1</td>
<td>The Secure Shell Version 2 Support feature allows you to configure Secure Shell (SSH) Version 2. SSH runs on top of a reliable transport layer and provides strong authentication and encryption capabilities. In Cisco IOS Release 2.4, this feature was introduced on the Cisco ASR 1000 Series Aggregation Services Routers. The following commands were introduced or modified by this feature: debug ip ssh, ip ssh min dh size, ip ssh rsa keypair-name, ip ssh version, ssh.</td>
</tr>
<tr>
<td>Secure Shell Version 2 Enhancements</td>
<td>Cisco IOS XE Release 2.1</td>
<td>The Secure Shell Version 2 Enhancements include a number of additional capabilities such as support for VRF aware SSH, SSH debug enhancements, and Diffie-Hellman group 14 and group 16 exchange support. In Cisco IOS Release 2.4, this feature was introduced on the Cisco ASR 1000 Series Aggregation Services Routers.</td>
</tr>
</tbody>
</table>

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: www.cisco.com/go/trademarks. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)