



## **Secure Shell Configuration Guide, Cisco IOS XE Release 3SE (Catalyst 3850 Switches)**

### **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

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED "AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

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.

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: <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)

© 2015 Cisco Systems, Inc. All rights reserved.



## CONTENTS

---

### CHAPTER 1

#### Configuring Secure Shell 1

- Finding Feature Information 1
- Prerequisites for Configuring SSH 2
- Restrictions for Configuring SSH 2
- Information About Secure Shell (SSH) 3
  - SSH Server 3
  - SSH Integrated Client 3
  - RSA Authentication Support 3
- How to Configure SSH 4
  - Configuring an SSH Server 4
  - Invoking an SSH Client 5
    - Troubleshooting Tips 6
- Configuration Examples for SSH 6
  - Example SSH on a Cisco 7200 Series Router 6
  - Example SSH on a Cisco 7500 Series Router 7
  - Example SSH on a Cisco 12000 Series Router 9
  - Example: Verifying SSH 10
- Additional References 11
- Feature Information for Configuring Secure Shell 12

---

### CHAPTER 2

#### Secure Copy 13

- Finding Feature Information 13
- Prerequisites for Secure Copy 13
- Information About Secure Copy 14
  - How Secure Copy Works 14
- How to Configure Secure Copy 14
  - Configuring Secure Copy 14
- Configuration Examples for Secure Copy 16

|                                                                          |    |
|--------------------------------------------------------------------------|----|
| Example: Secure Copy Configuration Using Local Authentication            | 16 |
| Example SCP Server-Side Configuration Using Network-Based Authentication | 16 |
| Additional References                                                    | 17 |
| Feature Information for Secure Copy                                      | 17 |
| Glossary                                                                 | 18 |

---

**CHAPTER 3****Secure Shell Version 2 Support 19**

|                                                                                 |    |
|---------------------------------------------------------------------------------|----|
| Finding Feature Information                                                     | 19 |
| Prerequisites for Secure Shell Version 2 Support                                | 20 |
| Restrictions for Secure Shell Version 2 Support                                 | 20 |
| Information About Secure Shell Version 2 Support                                | 20 |
| Secure Shell Version 2                                                          | 20 |
| Secure Shell Version 2 Enhancements                                             | 21 |
| Secure Shell Version 2 Enhancements for RSA Keys                                | 21 |
| SNMP Trap Generation                                                            | 22 |
| SSH Keyboard Interactive Authentication                                         | 23 |
| How to Configure Secure Shell Version 2 Support                                 | 23 |
| Configuring a Device for SSH Version 2 Using a Hostname and Domain Name         | 23 |
| Configuring a Device for SSH Version 2 Using RSA Key Pairs                      | 25 |
| Configuring the Cisco SSH Server to Perform RSA-Based User Authentication       | 26 |
| Configuring the Cisco IOS SSH Client to Perform RSA-Based Server Authentication | 28 |
| Starting an Encrypted Session with a Remote Device                              | 30 |
| Troubleshooting Tips                                                            | 31 |
| Enabling Secure Copy Protocol on the SSH Server                                 | 31 |
| Verifying the Status of the Secure Shell Connection                             | 33 |
| Verifying the Secure Shell Status                                               | 35 |
| Monitoring and Maintaining Secure Shell Version 2                               | 36 |
| Configuration Examples for Secure Shell Version 2 Support                       | 39 |
| Example: Configuring Secure Shell Version 1                                     | 39 |
| Example: Configuring Secure Shell Version 2                                     | 39 |
| Example: Configuring Secure Shell Versions 1 and 2                              | 39 |
| Example: Starting an Encrypted Session with a Remote Device                     | 39 |
| Example: Configuring Server-Side SCP                                            | 39 |
| Example: Setting an SNMP Trap                                                   | 40 |
| Examples: SSH Keyboard Interactive Authentication                               | 40 |

|                                                                       |    |
|-----------------------------------------------------------------------|----|
| Example: Enabling Client-Side Debugs                                  | 40 |
| Example: Enabling ChPass with a Blank Password Change                 | 41 |
| Example: Enabling ChPass and Changing the Password on First Login     | 41 |
| Example: Enabling ChPass and Expiring the Password After Three Logins | 41 |
| Example: SNMP Debugging                                               | 42 |
| Examples: SSH Debugging Enhancements                                  | 42 |
| Additional References for Secure Shell Version 2 Support              | 43 |
| Feature Information for Secure Shell Version 2 Support                | 44 |





## CHAPTER

# 1

## Configuring Secure Shell

---

The Secure Shell (SSH) feature is an application and a protocol that provides a secure replacement to the Berkeley r-tools. The protocol secures sessions using standard cryptographic mechanisms, and the application can be used similarly to the Berkeley rexec and rsh tools. Two versions of SSH are available: SSH Version 1 and SSH Version 2. Unless otherwise noted, the term “SSH” denotes “SSH Version 1” only. For information about SSH Version 2, see the “Secure Shell Version 2 Support” feature module.

- [Finding Feature Information, page 1](#)
- [Prerequisites for Configuring SSH, page 2](#)
- [Restrictions for Configuring SSH, page 2](#)
- [Information About Secure Shell \(SSH\), page 3](#)
- [How to Configure SSH, page 4](#)
- [Configuration Examples for SSH, page 6](#)
- [Additional References, page 11](#)
- [Feature Information for Configuring Secure Shell, page 12](#)

### Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see [Bug Search Tool](#) and 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.

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.

## Prerequisites for Configuring SSH

**Note**

Unless otherwise noted, the term “SSH” denotes “SSH Version 1” only.

- Download the required image on the device. The Secure Shell (SSH) server requires an IPsec (Data Encryption Standard [DES] or 3DES) encryption software image; the SSH client requires an IPsec (DES or 3DES) encryption software image.) For information about downloading a software image, see the *Loading and Managing System Images Configuration Guide*.
- Configure a hostname and host domain for your device by using the **hostname** and **ip domain-name** commands in global configuration mode.
- Generate a Rivest, Shamir, and Adleman (RSA) key pair for your device. This key pair automatically enables SSH and remote authentication when the **crypto key generate rsa** command is entered in global configuration mode.

**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 authentication, authorization, and accounting (AAA). For more information, see the *Authentication, Authorization, and Accounting Configuration Guide*.

## Restrictions for Configuring SSH

**Note**

Unless otherwise noted, the term “SSH” denotes “SSH Version 1” only.

- The Secure Shell (SSH) server and SSH client are supported on Data Encryption Standard (DES) (56-bit) and 3DES (168-bit) data encryption software images only. In DES software images, DES is the only encryption algorithm available. In 3DES software images, both DES and 3DES encryption algorithms are available.
- 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 (SSH)

**Note**

Unless otherwise noted, the term “SSH” denotes “SSH Version 1” only.

### SSH Server

**Note**

Unless otherwise noted, the term “SSH” denotes “SSH Version 1” only.

The Secure Shell (SSH) Server feature enables an SSH client to make a secure, encrypted connection to a Cisco device. 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 software authentication. The SSH server in Cisco software works with publicly and commercially available SSH clients.

### SSH Integrated Client

**Note**

Unless otherwise noted, the term “SSH” denotes “SSH Version 1” only.

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

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

**Note**

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

### RSA Authentication Support

Rivest, Shamir, and Adleman (RSA) authentication available in Secure Shell (SSH) clients is not supported on the SSH server for Cisco software by default. For more information about RSA authentication support, see the “Configuring a Router for SSH Version 2 Using RSA Pairs” section of the “Secure Shell Version 2 Support” module.

# How to Configure SSH

## Configuring an SSH Server



**Note** Unless otherwise noted, the term “SSH” denotes “SSH Version 1” only.

### SUMMARY STEPS

1. `enable`
2. `configure terminal`
3. `ip ssh {time-out seconds | authentication-retries integer}`
4. `ip ssh rekey {time time | volume volume}`
5. `exit`
6. `show ip ssh`

### DETAILED STEPS

|               | Command or Action                                                                                                                            | Purpose                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|---------------|----------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Step 1</b> | <b>enable</b><br><br><b>Example:</b><br>Device> enable                                                                                       | Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>                                                                                                                                                                                                                                                                                                                                                                            |
| <b>Step 2</b> | <b>configure terminal</b><br><br><b>Example:</b><br>Device# configure terminal                                                               | Enters global configuration mode.                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| <b>Step 3</b> | <b>ip ssh {time-out <i>seconds</i>   authentication-retries <i>integer</i>}</b><br><br><b>Example:</b><br>Device(config)# ip ssh time-out 30 | Configures Secure Shell (SSH) control parameters. <p><b>Note</b> This command can also be used to establish the number of password prompts provided to the user. The number is the lower of the following two values:</p> <ul style="list-style-type: none"> <li>• Value proposed by the client using the <b>ssh -o numberofpasswordprompt</b> command.</li> <li>• Value configured on the device using the <b>ip ssh authentication-retries <i>integer</i></b> command, plus one.</li> </ul> |

|               | Command or Action                                                                                                              | Purpose                                                                                                                    |
|---------------|--------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------|
| <b>Step 4</b> | <b>ip ssh rekey</b> {time <i>time</i>   volume <i>volume</i> }<br><br><b>Example:</b><br>Device(config)# ip ssh rekey time 108 | (Optional) Configures a time-based rekey or a volume-based rekey for SSH.                                                  |
| <b>Step 5</b> | <b>exit</b><br><br><b>Example:</b><br>Device(config)# exit                                                                     | Returns to privileged EXEC mode.                                                                                           |
| <b>Step 6</b> | <b>show ip ssh</b><br><br><b>Example:</b><br>Device# show ip ssh                                                               | (Optional) Verifies that the SSH server is enabled and displays the version and configuration data for the SSH connection. |

## Invoking an SSH Client



### Note

Unless otherwise noted, the term “SSH” denotes “SSH Version 1” only.

Perform this task to invoke the Secure Shell (SSH) client. The SSH client runs in user EXEC mode and has no specific configuration tasks.

### SUMMARY STEPS

1. **enable**
2. **ssh -l** *username* **-vrf** *vrf-name* *ip-address*

### DETAILED STEPS

|               | Command or Action                                                                                                                              | Purpose                                                                                                                    |
|---------------|------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------|
| <b>Step 1</b> | <b>enable</b><br><br><b>Example:</b><br>Device> enable                                                                                         | Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>         |
| <b>Step 2</b> | <b>ssh -l</b> <i>username</i> <b>-vrf</b> <i>vrf-name</i> <i>ip-address</i><br><br><b>Example:</b><br>Device# ssh -l user1 -vrf vrf1 192.0.2.1 | Invokes the SSH client to connect to an IP host or address in the specified virtual routing and forwarding (VRF) instance. |

## Troubleshooting Tips

**Note**

Unless otherwise noted, the term “SSH” denotes “SSH Version 1” only.

- If your Secure Shell (SSH) configuration commands are rejected as illegal commands, you have not successfully generated an Rivest, Shamir, and Adleman (RSA) key pair for your device. Make sure that you have specified a hostname and domain. Then use the **crypto key generate rsa** command to generate an RSA key pair and enable the SSH server.
- When configuring the RSA key pair, you might encounter the following error messages:
  - No hostname specified.  
You must configure a hostname for the device using the **hostname** global configuration command. See the “IPsec and Quality of Service” module for more information.
  - No domain specified.  
You must configure a host domain for the device using the **ip domain-name** global configuration command. See the “IPsec and Quality of Service” module for more information
- The number of allowable SSH connections is limited to the maximum number of vtys configured for the device. Each SSH connection uses a vty resource.
- SSH uses either local security or the security protocol that is configured through AAA on your device for user authentication. When configuring Authentication, Authorization, and Accounting (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.

## Configuration Examples for SSH

### Example SSH on a Cisco 7200 Series Router

In the following example, SSH is configured on a Cisco 7200 with a timeout that is not to exceed 60 seconds and no more than 2 authentication retries. Before the SSH server feature is configured on the router, TACACS+ is specified as the method of authentication.

```
hostname Router72K
aaa new-model
aaa authentication login default tacacs+
aaa authentication login aaa7200kw none
enable password password
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
controller E1 2/0
controller E1 2/1
interface Ethernet1/0
ip address 192.168.110.2 255.255.255.0 secondary
ip address 192.168.109.2 255.255.255.0
no ip directed-broadcast
no ip route-cache
no ip mroute-cache
no keepalive
no cdp enable
interface Ethernet1/1
no ip address
no ip directed-broadcast
no ip route-cache
no ip mroute-cache
shutdown
no cdp enable
interface Ethernet1/2
no ip address
no ip directed-broadcast
no ip route-cache
no ip mroute-cache
shutdown
no cdp enable
no ip classless
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 password
end

```

## Example SSH on a Cisco 7500 Series Router

In the following example, SSH is configured on a Cisco 7500 with a timeout that is not to exceed 60 seconds and no more than 5 authentication retries. Before the SSH server feature is configured on the router, RADIUS is specified as the method of authentication.

```

hostname Router75K
aaa new-model
aaa authentication login default radius
aaa authentication login aaa7500kw none
enable password password

username username1 password 0 password1
username username2 password 0 password2
ip subnet-zero
no ip cef
no ip domain-lookup
ip domain-name cisco.com
! Enter ssh commands.
ip ssh time-out 60

```

```
ip ssh authentication-retries 5
controller E1 3/0
channel-group 0 timeslots 1

controller E1 3/1
channel-group 0 timeslots 1
channel-group 1 timeslots 2
interface Ethernet0/0/0
no ip address
no ip directed-broadcast
no ip route-cache distributed
shutdown

interface Ethernet0/0/1
no ip address
no ip directed-broadcast
no ip route-cache distributed
shutdown
interface Ethernet0/0/2
no ip address
no ip directed-broadcast
no ip route-cache distributed
shutdown

interface Ethernet0/0/3
no ip address
no ip directed-broadcast
no ip route-cache distributed
shutdown

interface Ethernet1/0
ip address 192.168.110.2 255.255.255.0 secondary
ip address 192.168.109.2 255.255.255.0
no ip directed-broadcast
no ip route-cache
no ip mroute-cache

interface Ethernet1/1
ip address 192.168.109.2 255.255.255.0
no ip directed-broadcast
no ip route-cache
no ip mroute-cache
shutdown

interface Ethernet1/2
no ip address
no ip directed-broadcast
no ip route-cache
no ip mroute-cache

interface Ethernet1/3
no ip address
no ip directed-broadcast
no ip route-cache
no ip mroute-cache
shutdown

interface Ethernet1/4
no ip address
no ip directed-broadcast
no ip route-cache
no ip mroute-cache
shutdown
interface Ethernet1/5
no ip address
no ip directed-broadcast
no ip route-cache
no ip mroute-cache
shutdown

interface Serial2/0
ip address 10.1.1.2 255.0.0.0
no ip directed-broadcast
```

```

encapsulation ppp
no ip route-cache
no ip mroute-cache
ip classless
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
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 aaa7500kw
transport input none
line aux 0
transport input all
line vty 0 4

end

```

## Example SSH on a Cisco 12000 Series Router

In the following example, SSH is configured on a Cisco 12000 with a timeout that is not to exceed 60 seconds and no more than two authentication retries. Before the SSH server feature is configured on the router, TACACS+ is specified as the method of authentication.

```

hostname Router12K
aaa new-model
aaa authentication login default tacacs+ local
aaa authentication login aaa12000kw local
enable password password

username username1 password 0 password1
username username2 password 0 password2
redundancy
main-cpu
auto-sync startup-config
ip subnet-zero
no ip domain-lookup
ip domain-name cisco.com
! Enter ssh commands.
ip ssh time-out 60
ip ssh authentication-retries 2
interface ATM0/0
no ip address
no ip directed-broadcast
no ip route-cache cef
shutdown
interface POS1/0
ip address 10.100.100.2 255.255.255.0
no ip directed-broadcast
encapsulation ppp
no ip route-cache cef
no keepalive
crc 16
no cdp enable

interface POS1/1
no ip address
no ip directed-broadcast
no ip route-cache cef
shutdown
crc 32

interface POS1/2
no ip address
no ip directed-broadcast
no ip route-cache cef

```

```

shutdown
crc 32

interface POS1/3
no ip address
no ip directed-broadcast
no ip route-cache cef
shutdown
crc 32

interface POS2/0
ip address 10.1.1.1 255.255.255.0
no ip directed-broadcast
encapsulation ppp
no ip route-cache cef
crc 16

interface Ethernet0
ip address 172.17.110.91 255.255.255.224
no ip directed-broadcast

router ospf 1
network 0.0.0.0 255.255.255.255 area 0.0.0.0
ip classless
ip route 0.0.0.0 0.0.0.0 172.17.110.65

logging trap debugging
tacacs-server host 172.17.116.138
tacacs-server key cisco

radius-server host 172.17.116.138 auth-port 1650 acct-port 1651
radius-server key cisco

line con 0
exec-timeout 0 0
login authentication aaal2000kw
transport input none
line aux 0
line vty 0 4

no scheduler max-task-time
no exception linecard slot 0 sqe-registers
no exception linecard slot 1 sqe-registers
no exception linecard slot 2 sqe-registers
no exception linecard slot 3 sqe-registers
no exception linecard slot 4 sqe-registers
no exception linecard slot 5 sqe-registers
no exception linecard slot 6 sqe-registers
end

```

## Example: Verifying SSH



### Note

Unless otherwise noted, the term “SSH” denotes “SSH Version 1” only.

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

```

Device# show ip ssh

SSH Enabled - version 1.5
Authentication timeout: 120 secs; Authentication retries: 3
The following example shows that SSH is disabled:

```

```

Device# 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 device when SSH is enabled:

```
Device# show ssh
```

```
Connection      Version      Encryption State Username
0 1.5 3DES Session Started guest
```

The following example shows that SSH is disabled:

```
Device# show ssh
```

```
%No SSH server connections running.
```

## Additional References

### Related Documents

| Related Topic                                       | Document Title                                                           |
|-----------------------------------------------------|--------------------------------------------------------------------------|
| Cisco IOS commands                                  | <a href="#">Cisco IOS Master Command List, All Releases</a>              |
| Authentication, authorization, and accounting (AAA) | <i>Authentication, Authorization, and Accounting Configuration Guide</i> |
| IPsec                                               | “IPsec and Quality of Service” module                                    |
| SSH Version 2                                       | “Secure Shell Version 2 Support” module                                  |
| Downloading a software image                        | <i>Loading and Managing System Images Configuration Guide</i>            |

### Technical Assistance

| Description                                                                                                                                                                                                                                                                                                                                                                           | Link                                                                                                              |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------|
| 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. | <a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a> |

## 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 [http://www.cisco.com/go/featurenavigator](#). An account on Cisco.com is not required.

**Table 1: Feature Information for Configuring Secure Shell**

| Feature Name | Releases                                 | Feature Information                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
|--------------|------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Secure Shell | Cisco IOS XE 3.2SE<br>Cisco IOS XE 3.3SE | <p>The Secure Shell (SSH) feature is an application and a protocol that provides a secure replacement to the Berkeley r-tools. The protocol secures sessions using standard cryptographic mechanisms, and the application can be used similarly to the Berkeley rexec and rsh tools. Two versions of SSH are available: SSH Version 1 and SSH Version 2. This document describes SSH Version 1.</p> <p>This document also includes information about the Secure Shell SSH Version 1 Integrated Client feature and the Secure Shell SSH Version 1 Server Support feature. Both features are part of the Secure Shell functionality.</p> <p>In Cisco IOS XE Release 3.2SE, this feature was supported on the following platforms:</p> <ul style="list-style-type: none"> <li>• Catalyst 3850 Series Switches</li> <li>• Cisco 5760 Wireless LAN Controller</li> </ul> <p>In Cisco IOS XE Release 3.3SE, this feature was supported on the following platforms:</p> <ul style="list-style-type: none"> <li>• Catalyst 3650 Series Switches</li> </ul> |



## CHAPTER 2

# Secure Copy

---

The Secure Copy (SCP) feature provides a secure and authenticated method for copying device configurations or device image files. SCP relies on Secure Shell (SSH), an application and protocol that provide a secure replacement for the Berkeley r-tools suite (Berkeley university's own set of networking applications). This document provides the procedure to configure a Cisco device for SCP server-side functionality.

- [Finding Feature Information, page 13](#)
- [Prerequisites for Secure Copy, page 13](#)
- [Information About Secure Copy, page 14](#)
- [How to Configure Secure Copy, page 14](#)
- [Configuration Examples for Secure Copy, page 16](#)
- [Additional References, page 17](#)
- [Feature Information for Secure Copy, page 17](#)
- [Glossary, page 18](#)

## Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see [Bug Search Tool](#) and 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.

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.

## Prerequisites for Secure Copy

- Before enabling Secure Copy (SCP), you must correctly configure Secure Shell (SSH), authentication, and authorization on the device.

- Because SCP relies on SSH for its secure transport, the device must have a Rivest, Shamir, and Adelman (RSA) key pair.

## Information About Secure Copy

### How Secure Copy Works

The behavior of Secure Copy (SCP) is similar to that of remote copy (RCP), which comes from the Berkeley r-tools suite (Berkeley university's own set of networking applications), except that SCP relies on Secure Shell (SSH) for security. In addition, SCP requires that authentication, authorization, and accounting (AAA) authorization be configured so that the device can determine whether the user has the correct privilege level.

SCP allows a user only with a privilege level of 15 to copy any file that exists in the Cisco IOS File System (IFS) to and from a device by using the **copy** command. An authorized administrator may also perform this action from a workstation.




---

**Note** Enable the SCP option while using the pscp.exe file with the Cisco software.

---

## How to Configure Secure Copy

### Configuring Secure Copy

To configure a Cisco device for Secure Copy (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**
8. **exit**
9. **show running-config**
10. **debug ip scp**

## DETAILED STEPS

|        | Command or Action                                                                                                                                                                                                                 | Purpose                                                                                                                                                                                                                                                 |
|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Step 1 | <b>enable</b><br><br><b>Example:</b><br>Device> enable                                                                                                                                                                            | Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>                                                                                                                                      |
| Step 2 | <b>configure terminal</b><br><br><b>Example:</b><br>Device# configure terminal                                                                                                                                                    | Enters global configuration mode.                                                                                                                                                                                                                       |
| Step 3 | <b>aaa new-model</b><br><br><b>Example:</b><br>Device(config)# aaa new-model                                                                                                                                                      | Sets AAA authentication at login.                                                                                                                                                                                                                       |
| Step 4 | <b>aaa authentication login {default   list-name} method1 [ method2... ]</b><br><br><b>Example:</b><br>Device(config)# aaa authentication login default group tacacs+                                                             | Enables the AAA access control system.                                                                                                                                                                                                                  |
| Step 5 | <b>aaa authorization {network   exec   commands level   reverse-access   configuration} {default   list-name} [method1 [ method2... ]]</b><br><br><b>Example:</b><br>Device(config)# aaa authorization exec default group tacacs+ | Sets parameters that restrict user access to a network. <p><b>Note</b> The <b>exec</b> keyword runs authorization to determine if the user is allowed to run an EXEC shell; therefore, you must use the <b>exec</b> keyword when you configure SCP.</p> |
| Step 6 | <b>username name [privilege level] password encryption-type encrypted-password</b><br><br><b>Example:</b><br>Device(config)# username superuser privilege 2 password 0 superpassword                                              | Establishes a username-based authentication system. <p><b>Note</b> You may omit this step if a network-based authentication mechanism, such as TACACS+ or RADIUS, has been configured.</p>                                                              |
| Step 7 | <b>ip scp server enable</b><br><br><b>Example:</b><br>Device(config)# ip scp server enable                                                                                                                                        | Enables SCP server-side functionality.                                                                                                                                                                                                                  |

|         | Command or Action                                                                | Purpose                                                              |
|---------|----------------------------------------------------------------------------------|----------------------------------------------------------------------|
| Step 8  | <b>exit</b><br><br><b>Example:</b><br>Device(config)# exit                       | Exits global configuration mode and returns to privileged EXEC mode. |
| Step 9  | <b>show running-config</b><br><br><b>Example:</b><br>Device# show running-config | (Optional) Displays the SCP server-side functionality.               |
| Step 10 | <b>debug ip scp</b><br><br><b>Example:</b><br>Device# debug ip scp               | (Optional) Troubleshoots SCP authentication problems.                |

## Configuration Examples for Secure Copy

### Example: Secure Copy Configuration Using Local Authentication

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

```
! AAA authentication and authorization must be configured properly in order for SCP to work.
aaa new-model
aaa authentication login default local
aaa authorization exec default local
username user1 privilege 15 password 0 lab
! SSH must be configured and functioning properly.
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:

```
! 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

### Related Documents

| Related Topic                                | Document Title                                                           |
|----------------------------------------------|--------------------------------------------------------------------------|
| Cisco IOS commands                           | <a href="#">Cisco IOS Master Command List, All Releases</a>              |
| Secure Shell Version 1 and 2 support         | <i>Secure Shell Configuration Guide</i>                                  |
| Authentication and authorization commands    | <a href="#">Cisco IOS Security Command Reference: Commands A to C</a>    |
| Configuring authentication and authorization | <i>Authentication, Authorization, and Accounting Configuration Guide</i> |

### Technical Assistance

| Description                                                                                                                                                                                                                                                                                                                                                                           | Link                                                                                                              |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------|
| 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. | <a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a> |

## 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 [http://www.cisco.com/go/featurenavigator](#). An account on Cisco.com is not required.

Table 2: Feature Information for Secure Copy

| Feature Name | Releases                                 | Feature Information                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|--------------|------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Secure Copy  | Cisco IOS XE 3.2SE<br>Cisco IOS XE 3.3SE | <p>The Secure Copy (SCP) feature provides a secure and authenticated method for copying device configurations or device image files. SCP relies on Secure Shell (SSH), an application and protocol that provide a secure replacement for the Berkeley r-tools suite.</p> <p>In Cisco IOS XE Release 3.2SE, this feature was supported on the following platforms:</p> <ul style="list-style-type: none"> <li>• Catalyst 3850 Series Switches</li> <li>• Cisco 5760 Wireless LAN Controller</li> </ul> <p>In Cisco IOS XE Release 3.3SE, this feature was supported on the following platforms:</p> <ul style="list-style-type: none"> <li>• Catalyst 3650 Series Switches</li> </ul> <p>The following commands were introduced or modified: <b>debug ip scp</b>, <b>ip scp server enable</b>.</p> |

## Glossary

**AAA**—authentication, authorization, and accounting. A 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. Relies on Remote Shell (Berkeley r-tools suite) for security; RCP copies files such as device images and startup configurations to and from devices.

**SCP**—secure copy. Relies on SSH for security; SCP support allows secure and authenticated copying of anything that exists in the Cisco IOS File System (IFS). SCP is derived from RCP.

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





## Secure Shell Version 2 Support

---

The Secure Shell Version 2 Support feature allows you to configure Secure Shell (SSH) Version 2. (SSH Version 1 support was implemented in an earlier Cisco software release.) SSH runs on top of a reliable transport layer and provides strong authentication and encryption capabilities. 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 allows for the secure transfer of files.

- [Finding Feature Information, page 19](#)
- [Prerequisites for Secure Shell Version 2 Support, page 20](#)
- [Restrictions for Secure Shell Version 2 Support, page 20](#)
- [Information About Secure Shell Version 2 Support, page 20](#)
- [How to Configure Secure Shell Version 2 Support, page 23](#)
- [Configuration Examples for Secure Shell Version 2 Support, page 39](#)
- [Additional References for Secure Shell Version 2 Support, page 43](#)
- [Feature Information for Secure Shell Version 2 Support, page 44](#)

### Finding Feature Information

Your software release may not support all the features documented in this module. For the latest caveats and feature information, see [Bug Search Tool](#) and 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.

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.

## Prerequisites for Secure Shell Version 2 Support

- Before configuring SSH, ensure that the required image is loaded on your device. The SSH server requires you to have a k9 (Triple Data Encryption Standard [3DES]) software image depending on your release.
- You have to use a SSH remote device that supports SSH Version 2 and connect to a Cisco device.
- SCP relies on authentication, authorization, and accounting (AAA) to function correctly. Therefore, AAA must be configured on the device to enable the secure copy protocol on the SSH Server.

**Note**

The SSH Version 2 server and the SSH Version 2 client are supported on your Cisco software, depending on your release. (The SSH client runs both the SSH Version 1 protocol and the SSH Version 2 protocol. The SSH client is supported in both k8 and k9 images depending on your release.)

For more information about downloading a software image, refer to the *Configuration Fundamentals Configuration Guide*.

## Restrictions for Secure Shell Version 2 Support

- Secure Shell (SSH) servers and SSH clients are supported in Triple Data Encryption Standard (3DES) software images.
- Execution Shell, remote command execution, and Secure Copy Protocol (SCP) are the only applications supported.
- Rivest, Shamir, and Adleman (RSA) key generation is an SSH server-side requirement. Devices that act as SSH clients need not generate RSA keys.
- The RSA key pair size must be greater than or equal to 768 bits.
- The following features are not supported:
  - Port forwarding
  - Compression

## Information About Secure Shell Version 2 Support

### Secure Shell Version 2

The Secure Shell Version 2 Support feature allows you to configure SSH Version 2.

The configuration for the SSH Version 2 server is similar to the configuration for SSH Version 1. The **ip ssh version** command defines the SSH version to be configured. 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 device 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 enables an SSH connection using the Rivest, Shamir, and Adleman (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). This behavior still exists, but by using the **ip ssh rsa keypair-name** command, you can overcome this 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 hostname and a domain name, which was required in SSH Version 1 of the Cisco software.

**Note**

---

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

---

## Secure Shell Version 2 Enhancements

The SSH Version 2 Enhancements feature includes a number of additional capabilities such as supporting Virtual Routing and Forwarding (VRF)-Aware SSH, SSH debug enhancements, and Diffie-Hellman (DH) group exchange support.

**Note**

---

The VRF-Aware SSH feature is supported depending on your release.

---

The Cisco SSH implementation has traditionally used 768-bit modulus, but with an increasing need for higher key sizes to accommodate DH Group 14 (2048 bits) and Group 16 (4096 bits) cryptographic applications, a message exchange between the client and the server to establish the favored DH group becomes necessary. The **ip ssh dh min size** command configures the modulus size on the SSH server. In addition to this, the **ssh** command was extended to add VRF awareness to the 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 was enhanced by modifying SSH debug commands. The **debug ip ssh** command was extended to simplify the debugging process. Before the simplification of the debugging process, 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.

## Secure Shell Version 2 Enhancements for RSA Keys

Cisco SSH Version 2 supports keyboard-interactive and password-based authentication methods. The SSH Version 2 Enhancements for RSA Keys feature also supports RSA-based public key authentication for the client and the server.

User authentication—RSA-based user authentication uses a private/public key pair associated with each user for authentication. The user must generate a private/public key pair on the client and configure a public key on the Cisco SSH server to complete the authentication.

An SSH user trying to establish credentials provides an encrypted signature using the private key. The signature and the user's public key are sent to the SSH server for authentication. The SSH server computes a hash over the public key provided by the user. The hash is used to determine if the server has a matching entry. If a match is found, an RSA-based message verification is performed using the public key. Hence, the user is authenticated or denied access based on the encrypted signature.

Server authentication—While establishing an SSH session, the Cisco SSH client authenticates the SSH server by using the server host keys available during the key exchange phase. SSH server keys are used to identify the SSH server. These keys are created at the time of enabling SSH and must be configured on the client.

For server authentication, the Cisco SSH client must assign a host key for each server. When the client tries to establish an SSH session with a server, the client receives the signature of the server as part of the key exchange message. If the strict host key checking flag is enabled on the client, the client checks if it has the host key entry corresponding to the server. If a match is found, the client tries to validate the signature by using the server host key. If the server is successfully authenticated, the session establishment continues; otherwise, it is terminated and displays a “Server Authentication Failed” message.




---

**Note** Storing public keys on a server uses memory; therefore, the number of public keys configurable on an SSH server is restricted to ten users, with a maximum of two public keys per user.

---




---

**Note** RSA-based user authentication is supported by the Cisco server, but Cisco clients cannot propose public key as an authentication method. If the Cisco server receives a request from an open SSH client for RSA-based authentication, the server accepts the authentication request.

---




---

**Note** For server authentication, configure the RSA public key of the server manually and configure the **ip ssh stricthostkeycheck** command on the Cisco SSH client.

---

## SNMP Trap Generation

Depending on your release, Simple Network Management Protocol (SNMP) traps are generated automatically when an SSH session terminates if the traps have been enabled and SNMP debugging has been enabled. For information about enabling SNMP traps, see the “Configuring SNMP Support” module in the *SNMP Configuration Guide*.




---

**Note** When you configure the **snmp-server host** 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. For an example of an SNMP trap generation configuration, see the “” section.

---

You must also enable 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. For an example of SNMP debugging, see the “[Example: SNMP Debugging](#)” section.

## SSH Keyboard Interactive Authentication

The SSH Keyboard Interactive Authentication feature, also known as Generic Message Authentication for SSH, is a method that can be used to implement different types of authentication mechanisms. Basically, any currently supported authentication method that requires only user input can be performed with this feature. The feature is automatically enabled.

The following methods are supported:

- Password
- SecurID and hardware tokens printing a number or a string in response to a challenge sent by the server
- Pluggable Authentication Module (PAM)
- S/KEY (and other One-Time-Pads)

For examples of various scenarios in which the SSH Keyboard Interactive Authentication feature has been automatically enabled, see the “[Examples: SSH Keyboard Interactive Authentication, on page 40](#)” section.

## How to Configure Secure Shell Version 2 Support

### Configuring a Device for SSH Version 2 Using a Hostname and Domain Name

#### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **hostname *name***
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]**
8. **exit**

#### DETAILED STEPS

|        | Command or Action                                      | Purpose                                                                                                            |
|--------|--------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------|
| Step 1 | <b>enable</b><br><br><b>Example:</b><br>Device> enable | Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul> |

|               | Command or Action                                                                                                                                | Purpose                                                                                                                                                                            |
|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Step 2</b> | <b>configure terminal</b><br><br><b>Example:</b><br>Device# configure terminal                                                                   | Enters global configuration mode.                                                                                                                                                  |
| <b>Step 3</b> | <b>hostname <i>name</i></b><br><br><b>Example:</b><br>Device(config)# hostname cisco7200                                                         | Configures a hostname for your device.                                                                                                                                             |
| <b>Step 4</b> | <b>ip domain-name <i>name</i></b><br><br><b>Example:</b><br>cisco7200(config)# ip domain-name example.com                                        | Configures a domain name for your device.                                                                                                                                          |
| <b>Step 5</b> | <b>crypto key generate rsa</b><br><br><b>Example:</b><br>cisco7200(config)# crypto key generate rsa                                              | Enables the SSH server for local and remote authentication.                                                                                                                        |
| <b>Step 6</b> | <b>ip ssh [time-out <i>seconds</i>   authentication-retries <i>integer</i>]</b><br><br><b>Example:</b><br>cisco7200(config)# ip ssh time-out 120 | (Optional) Configures SSH control variables on your device.                                                                                                                        |
| <b>Step 7</b> | <b>ip ssh version [1   2]</b><br><br><b>Example:</b><br>cisco7200(config)# ip ssh version 1                                                      | (Optional) Specifies the version of SSH to be run on your device.                                                                                                                  |
| <b>Step 8</b> | <b>exit</b><br><br><b>Example:</b><br>cisco7200(config)# exit                                                                                    | Exits global configuration mode and enters privileged EXEC mode. <ul style="list-style-type: none"> <li>• Use <b>no hostname</b> command to return to the default host.</li> </ul> |

## Configuring a Device for SSH Version 2 Using RSA Key Pairs

### 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**
7. **exit**

### DETAILED STEPS

|        | Command or Action                                                                                                                                                                                                           | Purpose                                                                                                                                                                                                                                                                                                                                                                |
|--------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Step 1 | <b>enable</b><br><br><b>Example:</b><br>Device> enable                                                                                                                                                                      | Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>                                                                                                                                                                                                                                                     |
| Step 2 | <b>configure terminal</b><br><br><b>Example:</b><br>Device# configure terminal                                                                                                                                              | Enters global configuration mode.                                                                                                                                                                                                                                                                                                                                      |
| Step 3 | <b>ip ssh rsa keypair-name</b> <i>keypair-name</i><br><br><b>Example:</b><br>Device(config)# ip ssh rsa keypair-name<br>sshkeys                                                                                             | Specifies the RSA key pair to be used for SSH. <p><b>Note</b> A Cisco device can have many RSA key pairs.</p>                                                                                                                                                                                                                                                          |
| Step 4 | <b>crypto key generate rsa</b> <b>usage-keys</b> <b>label</b> <i>key-label</i> <b>modulus</b> <i>modulus-size</i><br><br><b>Example:</b><br>Device(config)# crypto key generate rsa<br>usage-keys label sshkeys modulus 768 | Enables the SSH server for local and remote authentication on the device. <ul style="list-style-type: none"> <li>• For SSH Version 2, the modulus size must be at least 768 bits.</li> </ul> <p><b>Note</b> To delete the RSA key pair, use the <b>crypto key zeroize rsa</b> command. When you delete the RSA key pair, you automatically disable the SSH server.</p> |
| Step 5 | <b>ip ssh</b> [ <b>time-out</b> <i>seconds</i>   <b>authentication-retries</b> <i>integer</i> ]                                                                                                                             | Configures SSH control variables on your device.                                                                                                                                                                                                                                                                                                                       |

|               | Command or Action                                                                  | Purpose                                                          |
|---------------|------------------------------------------------------------------------------------|------------------------------------------------------------------|
|               | <b>Example:</b><br>Device(config)# ip ssh time-out 12                              |                                                                  |
| <b>Step 6</b> | <b>ip ssh version 2</b><br><br><b>Example:</b><br>Device(config)# ip ssh version 2 | Specifies the version of SSH to be run on the device.            |
| <b>Step 7</b> | <b>exit</b><br><br><b>Example:</b><br>Device(config)# exit                         | Exits global configuration mode and enters privileged EXEC mode. |

## Configuring the Cisco SSH Server to Perform RSA-Based User Authentication

### SUMMARY STEPS

1. enable
2. configure terminal
3. hostname *name*
4. ip domain-name *name*
5. crypto key generate rsa
6. ip ssh pubkey-chain
7. username *username*
8. key-string
9. key-hash *key-type key-name*
10. end

### DETAILED STEPS

|               | Command or Action                                      | Purpose                                                                                                            |
|---------------|--------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------|
| <b>Step 1</b> | <b>enable</b><br><br><b>Example:</b><br>Device> enable | Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul> |



|               | Command or Action                                                                                                        | Purpose                                                                                                                                                                                                                                                                                                                                                                         |
|---------------|--------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Step 2</b> | <b>configure terminal</b><br><br><b>Example:</b><br>Device# configure terminal                                           | Enters global configuration mode.                                                                                                                                                                                                                                                                                                                                               |
| <b>Step 3</b> | <b>hostname <i>name</i></b><br><br><b>Example:</b><br>Device(config)# hostname host1                                     | Specifies the hostname.                                                                                                                                                                                                                                                                                                                                                         |
| <b>Step 4</b> | <b>ip domain-name <i>name</i></b><br><br><b>Example:</b><br>host1(config)# ip domain-name name1                          | Defines a default domain name that the Cisco software uses to complete unqualified hostnames.                                                                                                                                                                                                                                                                                   |
| <b>Step 5</b> | <b>crypto key generate rsa</b><br><br><b>Example:</b><br>host1(config)# crypto key generate rsa                          | Generates RSA key pairs.                                                                                                                                                                                                                                                                                                                                                        |
| <b>Step 6</b> | <b>ip ssh pubkey-chain</b><br><br><b>Example:</b><br>host1(config)# ip ssh pubkey-chain                                  | Configures SSH-RSA keys for user and server authentication on the SSH server and enters public-key configuration mode. <ul style="list-style-type: none"> <li>• The user authentication is successful if the RSA public key stored on the server is verified with the public or the private key pair stored on the client.</li> </ul>                                           |
| <b>Step 7</b> | <b>username <i>username</i></b><br><br><b>Example:</b><br>host1(conf-ssh-pubkey)# username user1                         | Configures the SSH username and enters public-key user configuration mode.                                                                                                                                                                                                                                                                                                      |
| <b>Step 8</b> | <b>key-string</b><br><br><b>Example:</b><br>host1(conf-ssh-pubkey-user)#<br>key-string                                   | Specifies the RSA public key of the remote peer and enters public-key data configuration mode. <p><b>Note</b> You can obtain the public key value from an open SSH client; that is, from the .ssh/id_rsa.pub file.</p>                                                                                                                                                          |
| <b>Step 9</b> | <b>key-hash <i>key-type key-name</i></b><br><br><b>Example:</b><br>host1(conf-ssh-pubkey-data)# key-hash<br>ssh-rsa key1 | (Optional) Specifies the SSH key type and version. <ul style="list-style-type: none"> <li>• The key type must be ssh-rsa for the configuration of private public key pairs.</li> <li>• This step is optional only if the <b>key-string</b> command is configured.</li> <li>• You must configure either the <b>key-string</b> command or the <b>key-hash</b> command.</li> </ul> |

|                | Command or Action                                                                      | Purpose                                                                                                                                                                                                                                                                                        |
|----------------|----------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                |                                                                                        | <b>Note</b> You can use a hashing software to compute the hash of the public key string, or you can also copy the hash value from another Cisco device. Entering the public key data using the <b>key-string</b> command is the preferred way to enter the public key data for the first time. |
| <b>Step 10</b> | <b>end</b><br><br><b>Example:</b><br><br><pre>host1 (conf-ssh-pubkey-data) # end</pre> | Exits public-key data configuration mode and returns to privileged EXEC mode. <ul style="list-style-type: none"> <li>• Use <b>no hostname</b> command to return to the default host.</li> </ul>                                                                                                |

## Configuring the Cisco IOS SSH Client to Perform RSA-Based Server Authentication

### SUMMARY STEPS

1. **enable**
2. **configure terminal**
3. **hostname** *name*
4. **ip domain-name** *name*
5. **crypto key generate rsa**
6. **ip ssh pubkey-chain**
7. **server** *server-name*
8. **key-string**
9. **exit**
10. **key-hash** *key-type key-name*
11. **end**
12. **configure terminal**
13. **ip ssh stricthostkeycheck**

### DETAILED STEPS

|               | Command or Action                                                        | Purpose                                                                                                            |
|---------------|--------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------|
| <b>Step 1</b> | <b>enable</b><br><br><b>Example:</b><br><br><pre>Device&gt; enable</pre> | Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul> |

|                | Command or Action                                                                                 | Purpose                                                                                                                                                                                                                             |
|----------------|---------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Step 2</b>  | <b>configure terminal</b><br><br><b>Example:</b><br>Device# configure terminal                    | Enters global configuration mode.                                                                                                                                                                                                   |
| <b>Step 3</b>  | <b>hostname <i>name</i></b><br><br><b>Example:</b><br>Device(config)# hostname host1              | Specifies the hostname.                                                                                                                                                                                                             |
| <b>Step 4</b>  | <b>ip domain-name <i>name</i></b><br><br><b>Example:</b><br>host1(config)# ip domain-name name1   | Defines a default domain name that the Cisco software uses to complete unqualified hostnames.                                                                                                                                       |
| <b>Step 5</b>  | <b>crypto key generate rsa</b><br><br><b>Example:</b><br>host1(config)# crypto key generate rsa   | Generates RSA key pairs.                                                                                                                                                                                                            |
| <b>Step 6</b>  | <b>ip ssh pubkey-chain</b><br><br><b>Example:</b><br>host1(config)# ip ssh pubkey-chain           | Configures SSH-RSA keys for user and server authentication on the SSH server and enters public-key configuration mode.                                                                                                              |
| <b>Step 7</b>  | <b>server <i>server-name</i></b><br><br><b>Example:</b><br>host1(conf-ssh-pubkey)# server server1 | Enables the SSH server for public-key authentication on the device and enters public-key server configuration mode.                                                                                                                 |
| <b>Step 8</b>  | <b>key-string</b><br><br><b>Example:</b><br>host1(conf-ssh-pubkey-server)#<br>key-string          | Specifies the RSA public-key of the remote peer and enters public key data configuration mode.<br><br><b>Note</b> You can obtain the public key value from an open SSH client; that is, from the <code>.ssh/id_rsa.pub</code> file. |
| <b>Step 9</b>  | <b>exit</b><br><br><b>Example:</b><br>host1(conf-ssh-pubkey-data)# exit                           | Exits public-key data configuration mode and enters public-key server configuration mode.                                                                                                                                           |
| <b>Step 10</b> | <b>key-hash <i>key-type key-name</i></b>                                                          | (Optional) Specifies the SSH key type and version.                                                                                                                                                                                  |

|                | Command or Action                                                                                                  | Purpose                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|----------------|--------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                | <p><b>Example:</b></p> <pre>host1(conf-ssh-pubkey-server)# key-hash ssh-rsa key1</pre>                             | <ul style="list-style-type: none"> <li>The key type must be <code>ssh-rsa</code> for the configuration of private/public key pairs.</li> <li>This step is optional only if the <b>key-string</b> command is configured.</li> <li>You must configure either the <b>key-string</b> command or the <b>key-hash</b> command.</li> </ul> <p><b>Note</b> You can use a hashing software to compute the hash of the public key string, or you can copy the hash value from another Cisco device. Entering the public key data using the <b>key-string</b> command is the preferred way to enter the public key data for the first time.</p> |
| <b>Step 11</b> | <p><b>end</b></p> <p><b>Example:</b></p> <pre>host1(conf-ssh-pubkey-server)# end</pre>                             | Exits public-key server configuration mode and returns to privileged EXEC mode.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |
| <b>Step 12</b> | <p><b>configure terminal</b></p> <p><b>Example:</b></p> <pre>host1# configure terminal</pre>                       | Enters global configuration mode.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| <b>Step 13</b> | <p><b>ip ssh stricthostkeycheck</b></p> <p><b>Example:</b></p> <pre>host1(config)# ip ssh stricthostkeycheck</pre> | <p>Ensures that server authentication takes place.</p> <ul style="list-style-type: none"> <li>The connection is terminated in case of a failure.</li> <li>Use <b>no hostname</b> command to return to the default host.</li> </ul>                                                                                                                                                                                                                                                                                                                                                                                                   |

## Starting an Encrypted Session with a Remote Device



### Note

The device with which you want to connect must support a Secure Shell (SSH) server that has an encryption algorithm that is supported in Cisco software. Also, you need not enable your device. SSH can be run in disabled mode.

## SUMMARY STEPS

1. `ssh [-v {1 | 2}] [-c {aes128-ctr | aes192-ctr | aes256-ctr | aes128-cbc | 3des | aes192-cbc | aes256-cbc}] [-I user-id | -I user-id:vrf-name number ip-address ip-address | -I user-id:rotary number ip-address] [-m {hmac-md5-128 | hmac-md5-96 | hmac-sha1-160 | hmac-sha1-96}] [-o numberofpasswordprompts n | -p port-num] {ip-addr | hostname} [command | -vrf]`

## DETAILED STEPS

|        | Command or Action                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | Purpose                                                      |
|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------|
| Step 1 | <pre>ssh [-v {1   2}] [-c {aes128-ctr   aes192-ctr   aes256-ctr   aes128-cbc   3des   aes192-cbc   aes256-cbc}] [-I user-id   -I user-id:vrf-name number ip-address ip-address   -I user-id:rotary number ip-address] [-m {hmac-md5-128   hmac-md5-96   hmac-sha1-160   hmac-sha1-96}] [-o numberofpasswordprompts n   -p port-num] {ip-addr   hostname} [command   -vrf]</pre> <p><b>Example:</b></p> <pre>Device# ssh -v 2 -c aes256-ctr -m hmac-sha1-96 -I user2 10.76.82.24</pre> | Starts an encrypted session with a remote networking device. |

## Troubleshooting Tips

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

## Enabling Secure Copy Protocol on the SSH Server



### Note

The following task configures the server-side functionality for SCP. This task shows a typical configuration that allows the device to securely copy files from a remote workstation.

## SUMMARY STEPS

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

## DETAILED STEPS

|               | Command or Action                                                                                                                 | Purpose                                                                                                                                                                                                                      |
|---------------|-----------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Step 1</b> | <b>enable</b><br><br><b>Example:</b><br>Device> enable                                                                            | Enables privileged EXEC mode.<br><br><ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul>                                                                                                    |
| <b>Step 2</b> | <b>configure terminal</b><br><br><b>Example:</b><br>Device# configure terminal                                                    | Enters global configuration mode.                                                                                                                                                                                            |
| <b>Step 3</b> | <b>aaa new-model</b><br><br><b>Example:</b><br>Device(config)# aaa new-model                                                      | Enables the AAA access control model.                                                                                                                                                                                        |
| <b>Step 4</b> | <b>aaa authentication login default local</b><br><br><b>Example:</b><br>Device(config)# aaa authentication login<br>default local | Sets AAA authentication at login to use the local username database for authentication.                                                                                                                                      |
| <b>Step 5</b> | <b>aaa authorization exec defaultlocal</b><br><br><b>Example:</b><br>Device(config)# aaa authorization exec<br>default local      | Sets the parameters that restrict user access to a network, runs the authorization to determine if the user ID is allowed to run an EXEC shell, and specifies that the system must use the local database for authorization. |

|                | Command or Action                                                                                                                                                                                                    | Purpose                                                                                                                                                                                                                                                                                       |
|----------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Step 6</b>  | <p><b>username</b> <i>username</i> <b>privilege</b> <i>privilege-level</i> <b>password</b> <i>password</i></p> <p><b>Example:</b></p> <pre>Device(config)# username samplename privilege 15 password password1</pre> | <p>Establishes a username-based authentication system, and specifies the username, privilege level, and an unencrypted password.</p> <p><b>Note</b> The minimum value for the <i>privilege-level</i> argument is 15. A privilege level of less than 15 results in the connection closing.</p> |
| <b>Step 7</b>  | <p><b>ip ssh time-out</b> <i>seconds</i></p> <p><b>Example:</b></p> <pre>Device(config)# ip ssh time-out 120</pre>                                                                                                   | <p>Sets the time interval (in seconds) that the device waits for the SSH client to respond.</p>                                                                                                                                                                                               |
| <b>Step 8</b>  | <p><b>ip ssh authentication-retries</b> <i>integer</i></p> <p><b>Example:</b></p> <pre>Device(config)# ip ssh authentication-retries 3</pre>                                                                         | <p>Sets the number of authentication attempts after which the interface is reset.</p>                                                                                                                                                                                                         |
| <b>Step 9</b>  | <p><b>ip scpserverenable</b></p> <p><b>Example:</b></p> <pre>Device(config)# ip scp server enable</pre>                                                                                                              | <p>Enables the device to securely copy files from a remote workstation.</p>                                                                                                                                                                                                                   |
| <b>Step 10</b> | <p><b>exit</b></p> <p><b>Example:</b></p> <pre>Device(config)# exit</pre>                                                                                                                                            | <p>Exits global configuration mode and returns to privileged EXEC mode.</p>                                                                                                                                                                                                                   |
| <b>Step 11</b> | <p><b>debug ip scp</b></p> <p><b>Example:</b></p> <pre>Device# debug ip scp</pre>                                                                                                                                    | <p>(Optional) Provides diagnostic information about SCP authentication problems.</p>                                                                                                                                                                                                          |

## Verifying the Status of the Secure Shell Connection

### SUMMARY STEPS

1. enable
2. show ssh
3. exit

## DETAILED STEPS

|               | Command or Action                                          | Purpose                                                                                                                   |
|---------------|------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------|
| <b>Step 1</b> | <b>enable</b><br><br><b>Example:</b><br>Device> enable     | Enables privileged EXEC mode.<br><br><ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul> |
| <b>Step 2</b> | <b>show ssh</b><br><br><b>Example:</b><br>Device# show ssh | Displays the status of SSH server connections.                                                                            |
| <b>Step 3</b> | <b>exit</b><br><br><b>Example:</b><br>Device# exit         | Exits privileged EXEC mode and returns to user EXEC mode.                                                                 |

## Examples

The following sample output from the **show ssh** command displays status of various SSH Version 1 and Version 2 connections for Version 1 and Version 2 connections:

```
-----
Device# 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
-----
```

The following sample output from the **show ssh** command displays status of various SSH Version 1 and Version 2 connections for a Version 2 connection with no Version 1 connection:

```
-----
Device# show ssh

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
%No SSHv1 server connections running.
-----
```

The following sample output from the **show ssh** command displays status of various SSH Version 1 and Version 2 connections for a Version 1 connection with no Version 2 connection:

```
-----
Device# show ssh
```



```

Connection      Version Encryption      State      Username
0               1.5          3DES          Session started      lab
%No SSHv2 server connections running.
-----

```

## Verifying the Secure Shell Status

### SUMMARY STEPS

1. **enable**
2. **show ip ssh**
3. **exit**

### DETAILED STEPS

|               | Command or Action                                                | Purpose                                                                 |
|---------------|------------------------------------------------------------------|-------------------------------------------------------------------------|
| <b>Step 1</b> | <b>enable</b><br><br><b>Example:</b><br>Device> enable           | Enables privileged EXEC mode.<br><br>• Enter your password if prompted. |
| <b>Step 2</b> | <b>show ip ssh</b><br><br><b>Example:</b><br>Device# show ip ssh | Displays the version and configuration data for SSH.                    |
| <b>Step 3</b> | <b>exit</b><br><br><b>Example:</b><br>Device# exit               | Exits privileged EXEC mode and returns to user EXEC mode.               |

### Examples

The following sample output from the **show ip ssh** command displays the version of SSH that is enabled, the authentication timeout values, and the number of authentication retries for Version 1 and Version 2 connections:

```

-----
Device# show ip ssh

SSH Enabled - version 1.99
Authentication timeout: 120 secs; Authentication retries: 3
-----

```

The following sample output from the **show ip ssh** command displays the version of SSH that is enabled, the authentication timeout values, and the number of authentication retries for a Version 2 connection with no Version 1 connection:

```

-----

```

```
Device# show ip ssh
```

```
SSH Enabled - version 2.0
Authentication timeout: 120 secs; Authentication retries: 3
```

The following sample output from the **show ip ssh** command displays the version of SSH that is enabled, the authentication timeout values, and the number of authentication retries for a Version 1 connection with no Version 2 connection:

```
-----
Device# show ip ssh
```

```
3d06h: %SYS-5-CONFIG_I: Configured from console by console
SSH Enabled - version 1.5
Authentication timeout: 120 secs; Authentication retries: 3
-----
```

## Monitoring and Maintaining Secure Shell Version 2

### SUMMARY STEPS

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

### DETAILED STEPS

|               | Command or Action                                                            | Purpose                                                                                                            |
|---------------|------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------|
| <b>Step 1</b> | <b>enable</b><br><br><b>Example:</b><br>Device> enable                       | Enables privileged EXEC mode. <ul style="list-style-type: none"> <li>• Enter your password if prompted.</li> </ul> |
| <b>Step 2</b> | <b>debug ip ssh</b><br><br><b>Example:</b><br>Device# debug ip ssh           | Enables debugging of SSH.                                                                                          |
| <b>Step 3</b> | <b>debug snmp packet</b><br><br><b>Example:</b><br>Device# debug snmp packet | Enables debugging of every SNMP packet sent or received by the device.                                             |

### Example

The following sample output from the **debug ip ssh** command shows the connection is an SSH Version 2 connection:

```
Device# 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
00:33:56: SSH2 1: ssh_receive: 48 bytes received
00:33:56: SSH2 1: input: packet len 32
00:33:56: SSH2 1: partial packet 16, need 16, maclen 16
00:33:56: SSH2 1: MAC #3 ok
00:33:56: SSH2 1: input: padlen 10
00:33:56: SSH2 1: received packet type 5
00:33:56: SSH2 1: send: len 32 (includes padlen 10)
00:33:56: SSH2 1: done calc MAC out #3
00:33:56: SSH2 1: ssh_receive: 64 bytes received
00:33:56: SSH2 1: input: packet len 48
00:33:56: SSH2 1: partial packet 16, need 32, maclen 16
00:33:56: SSH2 1: MAC #4 ok
00:33:56: SSH2 1: input: padlen 9
00:33:56: SSH2 1: received packet type 50
00:33:56: SSH2 1: send: len 32 (includes padlen 13)
00:33:56: SSH2 1: done calc MAC out #4
00:34:04: SSH2 1: ssh_receive: 160 bytes received
00:34:04: SSH2 1: input: packet len 64
00:34:04: SSH2 1: partial packet 16, need 48, maclen 16
00:34:04: SSH2 1: MAC #5 ok
00:34:04: SSH2 1: input: padlen 13
00:34:04: SSH2 1: received packet type 50
00:34:04: SSH2 1: send: len 16 (includes padlen 10)
00:34:04: SSH2 1: done calc MAC out #5
00:34:04: SSH2 1: authentication successful for lab
00:34:04: SSH2 1: input: packet len 64
00:34:04: SSH2 1: partial packet 16, need 48, maclen 16
00:34:04: SSH2 1: MAC #6 ok
00:34:04: SSH2 1: input: padlen 6
```

```
00:34:04: SSH2 1: received packet type 2
00:34:04: SSH2 1: ssh_receive: 64 bytes received
00:34:04: SSH2 1: input: packet len 48
00:34:04: SSH2 1: partial packet 16, need 32, maclen 16
00:34:04: SSH2 1: MAC #7 ok
00:34:04: SSH2 1: input: padlen 19
00:34:04: SSH2 1: received packet type 90
00:34:04: SSH2 1: channel open request
00:34:04: SSH2 1: send: len 32 (includes padlen 10)
00:34:04: SSH2 1: done calc MAC out #6
00:34:04: SSH2 1: ssh_receive: 192 bytes received
00:34:04: SSH2 1: input: packet len 64
00:34:04: SSH2 1: partial packet 16, need 48, maclen 16
00:34:04: SSH2 1: MAC #8 ok
00:34:04: SSH2 1: input: padlen 13
00:34:04: SSH2 1: received packet type 98
00:34:04: SSH2 1: pty-req request
00:34:04: SSH2 1: setting TTY - requested: height 24, width 80; set: height 24,
width 80
00:34:04: SSH2 1: input: packet len 96
00:34:04: SSH2 1: partial packet 16, need 80, maclen 16
00:34:04: SSH2 1: MAC #9 ok
00:34:04: SSH2 1: input: padlen 11
00:34:04: SSH2 1: received packet type 98
00:34:04: SSH2 1: x11-req request
00:34:04: SSH2 1: ssh_receive: 48 bytes received
00:34:04: SSH2 1: input: packet len 32
00:34:04: SSH2 1: partial packet 16, need 16, maclen 16
00:34:04: SSH2 1: MAC #10 ok
00:34:04: SSH2 1: input: padlen 12
00:34:04: SSH2 1: received packet type 98
00:34:04: SSH2 1: shell request
00:34:04: SSH2 1: shell message received
00:34:04: SSH2 1: starting shell for vty
00:34:04: SSH2 1: send: len 48 (includes padlen 18)
00:34:04: SSH2 1: done calc MAC out #7
00:34:07: SSH2 1: ssh_receive: 48 bytes received
00:34:07: SSH2 1: input: packet len 32
00:34:07: SSH2 1: partial packet 16, need 16, maclen 16
00:34:07: SSH2 1: MAC #11 ok
00:34:07: SSH2 1: input: padlen 17
00:34:07: SSH2 1: received packet type 94
00:34:07: SSH2 1: send: len 32 (includes padlen 17)
00:34:07: SSH2 1: done calc MAC out #8
00:34:07: SSH2 1: ssh_receive: 48 bytes received
00:34:07: SSH2 1: input: packet len 32
00:34:07: SSH2 1: partial packet 16, need 16, maclen 16
00:34:07: SSH2 1: MAC #12 ok
00:34:07: SSH2 1: input: padlen 17
00:34:07: SSH2 1: received packet type 94
00:34:07: SSH2 1: send: len 32 (includes padlen 17)
00:34:07: SSH2 1: done calc MAC out #9
00:34:07: SSH2 1: ssh_receive: 48 bytes received
00:34:07: SSH2 1: input: packet len 32
00:34:07: SSH2 1: partial packet 16, need 16, maclen 16
00:34:07: SSH2 1: MAC #13 ok
00:34:07: SSH2 1: input: padlen 17
00:34:07: SSH2 1: received packet type 94
00:34:07: SSH2 1: send: len 32 (includes padlen 17)
00:34:07: SSH2 1: done calc MAC out #10
00:34:08: SSH2 1: ssh_receive: 48 bytes received
00:34:08: SSH2 1: input: packet len 32
00:34:08: SSH2 1: partial packet 16, need 16, maclen 16
00:34:08: SSH2 1: MAC #14 ok
00:34:08: SSH2 1: input: padlen 17
00:34:08: SSH2 1: received packet type 94
00:34:08: SSH2 1: send: len 32 (includes padlen 17)
00:34:08: SSH2 1: done calc MAC out #11
00:34:08: SSH2 1: ssh_receive: 48 bytes received
00:34:08: SSH2 1: input: packet len 32
00:34:08: SSH2 1: partial packet 16, need 16, maclen 16
00:34:08: SSH2 1: MAC #15 ok
00:34:08: SSH2 1: input: padlen 17
```

```
00:34:08: SSH2 1: received packet type 94
00:34:08: SSH2 1: send: len 32 (includes padlen 16)
00:34:08: SSH2 1: done calc MAC out #12
00:34:08: SSH2 1: send: len 48 (includes padlen 18)
00:34:08: SSH2 1: done calc MAC out #13
00:34:08: SSH2 1: send: len 16 (includes padlen 6)
00:34:08: SSH2 1: done calc MAC out #14
00:34:08: SSH2 1: send: len 16 (includes padlen 6)
00:34:08: SSH2 1: done calc MAC out #15
00:34:08: SSH1: Session terminated normally
```

## Configuration Examples for Secure Shell Version 2 Support

### Example: Configuring Secure Shell Version 1

```
Device# configure terminal
Device(config)# ip ssh version 1
```

### Example: Configuring Secure Shell Version 2

```
Device# configure terminal
Device(config)# ip ssh version 2
```

### Example: Configuring Secure Shell Versions 1 and 2

```
Device# configure terminal
Device(config)# no ip ssh version
```

### Example: Starting an Encrypted Session with a Remote Device

```
Device# 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 the server-side functionality for SCP. This example also configures AAA authentication and authorization on the device. This example uses a locally defined username and password.

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

## Example: Setting an SNMP Trap

The following example shows that an SNMP trap is set. The trap notification is generated automatically when the SSH session terminates. In the example, a.b.c.d is the IP address of the SSH client. For an example of SNMP trap debug output, see the “[Example: SNMP Debugging, on page 42](#)” section.

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

## Examples: SSH Keyboard Interactive Authentication

### Example: Enabling Client-Side Debugs

The following example shows that the client-side debugs are turned on, and the maximum number of prompts is six (three for the SSH keyboard interactive authentication method and three for the password authentication method).

```
Password:
Password:
Password:
Password:
Password:
Password: cisco123
Last login: Tue Dec 6 13:15:21 2005 from 10.76.248.213
user1@courier:~> exit
logout
[Connection to 10.76.248.200 closed by foreign host]
Device1# debug ip ssh client

SSH Client debugging is on

Device1# ssh -l lab 10.1.1.3

Password:
*Nov 17 12:50:53.199: SSH0: sent protocol version id SSH-1.99-Cisco-1.25
*Nov 17 12:50:53.199: SSH CLIENT0: protocol version id is - SSH-1.99-Cisco-1.25
*Nov 17 12:50:53.199: SSH CLIENT0: sent protocol version id SSH-1.99-Cisco-1.25
*Nov 17 12:50:53.199: SSH CLIENT0: protocol version exchange successful
*Nov 17 12:50:53.203: SSH0: protocol version id is - SSH-1.99-Cisco-1.25
*Nov 17 12:50:53.335: SSH CLIENT0: key exchange successful and encryption on
*Nov 17 12:50:53.335: SSH2 CLIENT 0: using method keyboard-interactive
Password:
Password:
Password:
*Nov 17 12:51:01.887: SSH2 CLIENT 0: using method password authentication
Password:
Password: lab
Device2>

*Nov 17 12:51:11.407: SSH2 CLIENT 0: SSH2_MSG_USERAUTH_SUCCESS message received
*Nov 17 12:51:11.407: SSH CLIENT0: user authenticated
*Nov 17 12:51:11.407: SSH2 CLIENT 0: pty-req request sent
*Nov 17 12:51:11.411: SSH2 CLIENT 0: shell request sent
*Nov 17 12:51:11.411: SSH CLIENT0: session open
```

## Example: Enabling ChPass with a Blank Password Change

In the following example, the ChPass feature is enabled, and a blank password change is accomplished using the SSH Keyboard Interactive Authentication method. A TACACS+ access control server (ACS) is used as the back-end AAA server.

```
Device1# ssh -l cisco 10.1.1.3
Password:
Old Password: cisco
New Password: cisco123
Re-enter New password: cisco123

Device2> exit

[Connection to 10.1.1.3 closed by foreign host]
```

## Example: Enabling ChPass and Changing the Password on First Login

In the following example, the ChPass feature is enabled and TACACS+ ACS is used as the back-end server. The password is changed on the first login using the SSH keyboard interactive authentication method.

```
Device1# ssh -l cisco 10.1.1.3
Password: cisco
Your password has expired.
Enter a new one now.
New Password: cisco123
Re-enter New password: cisco123

Device2> exit

[Connection to 10.1.1.3 closed by foreign host]

Device1# ssh -l cisco 10.1.1.3
Password:cisco1
Your password has expired.
Enter a new one now.
New Password: cisco
Re-enter New password: cisco12
The New and Re-entered passwords have to be the same.
Try again.
New Password: cisco
Re-enter New password: cisco

Device2>
```

## Example: Enabling ChPass and Expiring the Password After Three Logins

In the following example, the ChPass feature is enabled and TACACS+ ACS is used as the back-end AAA server. The password expires after three logins using the SSH keyboard interactive authentication method.

```
Device# ssh -l cisco. 10.1.1.3
Password: cisco

Device2> exit

[Connection to 10.1.1.3 closed by foreign host]
```

```

Device1# ssh -l cisco 10.1.1.3
Password: cisco
Device2> exit
Device1# ssh -l cisco 10.1.1.3
Password: cisco
Device2> exit
[Connection to 10.1.1.3 closed by foreign host]
Device1# ssh -l cisco 10.1.1.3
Password: cisco
Your password has expired.
Enter a new one now.
New Password: cisco123
Re-enter New password: cisco123
Device2>

```

## Example: SNMP Debugging

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

```

Device1# debug snmp packet
SNMP packet debugging is on
Device1# ssh -l lab 10.0.0.2
Password:
Device2# exit
[Connection to 10.0.0.2 closed by foreign host]
Device1#
*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
Device1#

```

## Examples: SSH Debugging Enhancements

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

```

Device# 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 0: SSH2_MSG_KEXINIT sent
00:04:22: SSH2 0: SSH2_MSG_KEXINIT received
00:04:22: SSH2:kex: client->server enc:aes128-cbc mac:hmac-sha1

```



```

00:04:22: SSH2:kex: server->client enc:aes128-cbc mac:hmac-shal
00:04:22: SSH2 0: expecting SSH2_MSG_KEXDH_INIT
00:04:22: SSH2 0: SSH2_MSG_KEXDH_INIT received
00:04:22: SSH2: kex_derive_keys_complete
00:04:22: SSH2 0: SSH2_MSG_NEWKEYS sent
00:04:22: SSH2 0: waiting for SSH2_MSG_NEWKEYS
00:04:22: SSH2 0: SSH2_MSG_NEWKEYS received
00:04:24: SSH2 0: authentication successful for lab
00:04:24: SSH2 0: channel open request
00:04:24: SSH2 0: pty-req request
00:04:24: SSH2 0: setting TTY - requested: height 24, width 80; set: height 24, width 80
00:04:24: SSH2 0: shell request
00:04:24: SSH2 0: shell message received
00:04:24: SSH2 0: 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 about the SSH packet.

```
Device# debug ip ssh packet
```

```

00:05:43: SSH2 0: send:packet of length 280 (length also includes padlen of 4)
00:05:43: SSH2 0: ssh_receive: 64 bytes received
00:05:43: SSH2 0: input: total packet length of 280 bytes
00:05:43: SSH2 0: partial packet length(block size)8 bytes,needed 272 bytes, maclen 0
00:05:43: SSH2 0: ssh_receive: 64 bytes received
00:05:43: SSH2 0: partial packet length(block size)8 bytes,needed 272 bytes, maclen 0
00:05:43: SSH2 0: ssh_receive: 64 bytes received
00:05:43: SSH2 0: partial packet length(block size)8 bytes,needed 272 bytes, maclen 0
00:05:43: SSH2 0: ssh_receive: 64 bytes received
00:05:43: SSH2 0: partial packet length(block size)8 bytes,needed 272 bytes, maclen 0
00:05:43: SSH2 0: ssh_receive: 24 bytes received
00:05:43: SSH2 0: partial packet length(block size)8 bytes,needed 272 bytes, maclen 0
00:05:43: SSH2 0: input: padlength 4 bytes
00:05:43: SSH2 0: ssh_receive: 64 bytes received
00:05:43: SSH2 0: input: total packet length of 144 bytes
00:05:43: SSH2 0: partial packet length(block size)8 bytes,needed 136 bytes, maclen 0
00:05:43: SSH2 0: ssh_receive: 64 bytes received
00:05:43: SSH2 0: partial packet length(block size)8 bytes,needed 136 bytes, maclen 0
00:05:43: SSH2 0: ssh_receive: 16 bytes received
00:05:43: SSH2 0: partial packet length(block size)8 bytes,needed 136 bytes, maclen 0
00:05:43: SSH2 0: input: padlength 6 bytes
00:05:43: SSH2 0: signature length 143
00:05:43: SSH2 0: send:packet of length 448 (length also includes padlen of 7)
00:05:43: SSH2 0: send:packet of length 16 (length also includes padlen of 10)
00:05:43: SSH2 0: newkeys: mode 1
00:05:43: SSH2 0: ssh_receive: 16 bytes received
00:05:43: SSH2 0: input: total packet length of 16 bytes
00:05:43: SSH2 0: partial packet length(block size)8 bytes,needed 8 bytes, maclen 0
00:05:43: SSH2 0: input: padlength 10 bytes
00:05:43: SSH2 0: newkeys: mode 0
00:05:43: SSH2 0: ssh_receive: 52 bytes received
00:05:43: SSH2 0: input: total packet length of 32 bytes
00:05:43: SSH2 0: partial packet length(block size)16 bytes,needed 16 bytes, maclen 20
00:05:43: SSH2 0: MAC compared for #3 :ok

```

## Additional References for Secure Shell Version 2 Support

### Related Documents

| Related Topic      | Document Title                                              |
|--------------------|-------------------------------------------------------------|
| Cisco IOS commands | <a href="#">Cisco IOS Master Command List, All Releases</a> |

| Related Topic                                                                           | Document Title                                              |
|-----------------------------------------------------------------------------------------|-------------------------------------------------------------|
| AAA<br>Hostname and host domain configuration tasks<br>Secure shell configuration tasks | <i>Security Configuration Guide: Securing User Services</i> |
| Downloading a software image<br>Configuration fundamentals                              | <i>Configuration Fundamentals Configuration Guide</i>       |
| IPsec configuration tasks                                                               | <i>Security Configuration Guide: Secure Connectivity</i>    |
| SNMP traps configuration tasks                                                          | <i>SNMP Configuration Guide</i>                             |

### Standards

| Standards                                   | Title                                                   |
|---------------------------------------------|---------------------------------------------------------|
| IETF Secure Shell Version 2 Draft Standards | <a href="#">Internet Engineering Task Force website</a> |

### Technical Assistance

| Description                                                                                                                                                                                                                                                                                                                                                                           | Link                                                                                                              |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------|
| 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. | <a href="http://www.cisco.com/cisco/web/support/index.html">http://www.cisco.com/cisco/web/support/index.html</a> |

## 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 [http://www.cisco.com/cisco/web/featurenavigator](#). An account on Cisco.com is not required.

**Table 3: Feature Information for Secure Shell Version 2 Support**

| Feature Name                   | Releases                                 | Feature Information                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|--------------------------------|------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Secure Shell Version 2 Support | Cisco IOS XE 3.2SE<br>Cisco IOS XE 3.3SE | <p>The Secure Shell Version 2 Support feature allows you to configure Secure Shell (SSH) Version 2 (SSH Version 1 support was implemented in an earlier Cisco IOS software release). SSH runs on top of a reliable transport layer and provides strong authentication and encryption capabilities. SSH version 2 also supports AES counter-based encryption mode.</p> <p>In Cisco IOS XE Release 3.2SE, this feature was supported on the following platforms:</p> <ul style="list-style-type: none"> <li>• Catalyst 3850 Series Switches</li> <li>• Cisco 5760 Wireless LAN Controller</li> </ul> <p>In Cisco IOS XE Release 3.3SE, this feature was supported on the following platforms:</p> <ul style="list-style-type: none"> <li>• Catalyst 3650 Series Switches</li> </ul> <p>The following commands were introduced or modified: <b>debug ip ssh</b>, <b>ip ssh min dh size</b>, <b>ip ssh rsa keypair-name</b>, <b>ip ssh version</b>, <b>ssh</b>.</p> |

| Feature Name                                     | Releases                                 | Feature Information                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
|--------------------------------------------------|------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Secure Shell Version 2 Client and Server Support | Cisco IOS XE 3.2SE<br>Cisco IOS XE 3.3SE | <p>The Cisco IOS image was updated to provide for the automatic generation of SNMP traps when an SSH session terminates.</p> <p>In Cisco IOS XE Release 3.2SE, this feature was supported on the following platforms:</p> <ul style="list-style-type: none"> <li>• Catalyst 3850 Series Switches</li> <li>• Cisco 5760 Wireless LAN Controller</li> </ul> <p>In Cisco IOS XE Release 3.3SE, this feature was supported on the following platforms:</p> <ul style="list-style-type: none"> <li>• Catalyst 3650 Series Switches</li> </ul>                                                                                                                                                                                                         |
| SSH Keyboard Interactive Authentication          | Cisco IOS XE 3.2SE<br>Cisco IOS XE 3.3SE | <p>The SSH Keyboard Interactive Authentication feature, also known as Generic Message Authentication for SSH, is a method that can be used to implement different types of authentication mechanisms. Basically, any currently supported authentication method that requires only user input can be performed with this feature.</p> <p>In Cisco IOS XE Release 3.2SE, this feature was supported on the following platforms:</p> <ul style="list-style-type: none"> <li>• Catalyst 3850 Series Switches</li> <li>• Cisco 5760 Wireless LAN Controller</li> </ul> <p>In Cisco IOS XE Release 3.3SE, this feature was supported on the following platforms:</p> <ul style="list-style-type: none"> <li>• Catalyst 3650 Series Switches</li> </ul> |

| Feature Name                        | Releases                                 | Feature Information                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
|-------------------------------------|------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Secure Shell Version 2 Enhancements | Cisco IOS XE 3.2SE<br>Cisco IOS XE 3.3SE | <p>The Secure Shell Version 2 Enhancements feature includes a number of additional capabilities such as support for VRF-aware SSH, SSH debug enhancements, and DH Group 14 and Group 16 exchange support.</p> <p>In Cisco IOS XE Release 3.2SE, this feature was supported on the following platforms:</p> <ul style="list-style-type: none"><li>• Catalyst 3850 Series Switches</li><li>• Cisco 5760 Wireless LAN Controller</li></ul> <p>In Cisco IOS XE Release 3.3SE, this feature was supported on the following platforms:</p> <ul style="list-style-type: none"><li>• Catalyst 3650 Series Switches</li></ul> <p>The following commands were introduced or modified: <b>debug ip ssh</b>, <b>ip ssh dh min size</b>.</p> |

| Feature Name                                      | Releases                                 | Feature Information                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|---------------------------------------------------|------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Secure Shell Version 2 Enhancements for RSA Keys. | Cisco IOS XE 3.2SE<br>Cisco IOS XE 3.3SE | <p>The Secure Shell Version 2 Enhancements for RSA Keys feature includes a number of additional capabilities to support RSA key-based user authentication for SSH and SSH server host key storage and verification.</p> <p>In Cisco IOS XE Release 3.2SE, this feature was supported on the following platforms:</p> <ul style="list-style-type: none"> <li>• Catalyst 3850 Series Switches</li> <li>• Cisco 5760 Wireless LAN Controller</li> </ul> <p>In Cisco IOS XE Release 3.3SE, this feature was supported on the following platforms:</p> <ul style="list-style-type: none"> <li>• Catalyst 3650 Series Switches</li> </ul> <p>The following commands were introduced or modified: <b>ip ssh pubkey-chain</b>, <b>ip ssh stricthostkeycheck</b>.</p> |