



Secure Shell Commands

This module describes the Cisco IOS XR software commands used to configure Secure Shell (SSH).

For detailed information about SSH concepts, configuration tasks, and examples, see the *Implementing Secure Shell on* module in the *System Security Configuration Guide for Cisco NCS 6000 Series Routers*.

- [clear ssh](#), on page 2
- [netconf-yang agent ssh](#) , on page 3
- [sftp](#), on page 4
- [sftp \(Interactive Mode\)](#), on page 7
- [show ssh](#), on page 10
- [show ssh history](#), on page 13
- [show ssh history details](#), on page 14
- [show ssh rekey](#), on page 15
- [show ssh session details](#), on page 16
- [show tech-support ssh](#), on page 18
- [ssh](#), on page 19
- [ssh algorithms cipher](#), on page 20
- [ssh client enable cipher](#) , on page 21
- [ssh client knownhost](#), on page 22
- [ssh client source-interface](#), on page 23
- [ssh client vrf](#), on page 24
- [ssh server](#), on page 25
- [ssh server algorithms host-key](#), on page 27
- [ssh disable hmac](#), on page 28
- [ssh server enable cipher](#), on page 29
- [ssh server rekey-time](#), on page 29
- [ssh server rekey-volume](#), on page 30
- [ssh server logging](#), on page 31
- [ssh server rate-limit](#), on page 32
- [ssh server session-limit](#), on page 33
- [ssh server v2](#), on page 34
- [ssh server netconf port](#), on page 34
- [ssh timeout](#), on page 35

clear ssh

To terminate an incoming or outgoing Secure Shell (SSH) connection, use the **clear ssh** command in XR EXEC mode.

clear ssh {*session-id* | **outgoing** *session-id*}

Syntax Description	<i>session-id</i>	Session ID number of an incoming connection as displayed in the show ssh command output. Range is from 0 to 1024.
	outgoing <i>session-id</i>	Specifies the session ID number of an outgoing connection as displayed in the show ssh command output. Range is from 1 to 10.

Command Default None

Command Modes XR EXEC mode

Command History	Release	Modification
	Release 5.0.0	This command was introduced.

Usage Guidelines Use the **clear ssh** command to disconnect incoming or outgoing SSH connections. Incoming connections are managed by the SSH server running on the local networking device. Outgoing connections are initiated from the local networking device.

To display the session ID for a connection, use the **show ssh** command.

Task ID	Task ID	Operations
	crypto	execute

Examples

In the following example, the **show ssh** command is used to display all incoming and outgoing connections to the router. The **clear ssh** command is then used to terminate the incoming session with the ID number 0.

```
RP/0/RP0/CPU0:router# show ssh

SSH version: Cisco-2.0
session      pty  location  state      userid    host      ver
-----
Incoming sessions
0           vty0 0/33/1  SESSION_OPEN  cisco    172.19.72.182  v2
1           vty1 0/33/1  SESSION_OPEN  cisco    172.18.0.5     v2
2           vty2 0/33/1  SESSION_OPEN  cisco    172.20.10.3   v1
3           vty3 0/33/1  SESSION_OPEN  cisco    3333:::50     v2

Outgoing sessions
1           0/33/1  SESSION_OPEN  cisco    172.19.72.182  v2
2           0/33/1  SESSION_OPEN  cisco    3333:::50     v2
```

```
RP/0/RP0/CPU0:router# clear ssh 0
```

The following output is applicable for the **clear ssh** command starting IOS-XR 5.3.2 releases and later.

```
RP/0/RP0/CPU0:router# show ssh
SSH version : Cisco-2.0
```

```

id chan pty      location          state          userid  host          ver
authentication connection type
-----
Incoming sessions
0 1 vty0 0/RSP0/CPU0 SESSION_OPEN lab 12.22.57.75 v2
rsa-pubkey Command-Line-Interface
0 2 vty1 0/RSP0/CPU0 SESSION_OPEN lab 12.22.57.75 v2
rsa-pubkey Command-Line-Interface
0 3 0/RSP0/CPU0 SESSION_OPEN cisco 12.22.57.75 v2
rsa-pubkey Sftp-Subsystem
1 vty7 0/RSP0/CPU0 SESSION_OPEN cisco 12.22.22.57 v1 password
Command-Line-Interface
3 1 0/RSP0/CPU0 SESSION_OPEN lab 12.22.57.75 v2 password
Netconf-Subsystem
4 1 vty3 0/RSP0/CPU0 SESSION_OPEN lab 192.168.1.55 v2 password
Command-Line-Interface

Outgoing sessions
1 0/RSP0/CPU0 SESSION_OPEN lab 192.168.1.51 v2 password
```

```
RP/0/RP0/CPU0:router# clear ssh 0
```

Related Commands

Command	Description
show ssh, on page 10	Displays the incoming and outgoing connections to the router.

netconf-yang agent ssh

To enable netconf agent over SSH (Secure Shell) , use the **netconf-yang agent ssh** command in XR Config mode. To disable netconf, use the **no** form of the command.

netconf-yang agent ssh

Syntax Description This command has no keywords or arguments.

Command Default None

Command Modes XR Config mode

Command History	Release	Modification
	Release 5.3.0	This command was introduced.

Usage Guidelines SSH is currently the supported transport method for Netconf.

Task ID	Task ID	Operation
	config-services	read, write

Example

This example shows how to use the **netconf-yang agent ssh** command:

```
RP/0/RP0/CPU0:router (config) # netconf-yang agent ssh
```

sftp

To start the secure FTP (SFTP) client, use the **sftp** command in XR EXEC mode.

```
sftp [ username @ host : remote-filename ] source-filename dest-filename [ source-interface type interface-path-id ] [ vrf vrf-name ]
```

Syntax Description	
<i>username</i>	(Optional) Name of the user performing the file transfer. The at symbol (@) following the username is required.
<i>hostname:remote-filename</i>	(Optional) Name of the Secure Shell File Transfer Protocol (SFTP) server. The colon (:) following the hostname is required.
<i>source-filename</i>	SFTP source, including the path.
<i>dest-filename</i>	SFTP destination, including the path.
source-interface	(Optional) Specifies the source IP address of a selected interface for all outgoing SSH connections.
<i>type</i>	Interface type. For more information, use the question mark (?) online help function.
<i>interface-path-id</i>	Physical interface or virtual interface. Note Use the show interfaces command in EXEC mode to see a list of all interfaces currently configured on the router. For more information about the syntax for the router, use the question mark (?) online help function.
vrf <i>vrf-name</i>	Specifies the name of the VRF associated with the source interface.

Command Default If no *username* argument is provided, the login name on the router is used. If no *hostname* argument is provided, the file is considered local.

Command Modes XR EXEC mode

Command History	Release	Modification
	Release 5.0.0	This command was introduced.

Usage Guidelines

SFTP provides for the secure (and authenticated) copying of files between a router and a remote host. Like the **copy** command, the **sftp** command can be invoked only in EXEC mode.

If a username is not provided, the login name on the router is used as the default. If a host name is not provided, the file is considered local.

If the source interface is specified in the **sftp** command, the **sftp** interface takes precedence over the interface specified in the **ssh client source-interface** command.

When the file destination is a local path, all of the source files should be on remote hosts, and vice versa.

When multiple source files exist, the destination should be a preexisting directory. Otherwise, the destination can be either a directory name or destination filename. The file source cannot be a directory name.

If you download files from different remote hosts, that is, the source points to different remote hosts, the SFTP client spawns SSH instances for each host, which may result in multiple prompts for user authentication.

Task ID	Task ID	Operations
	crypto	execute
	basic-services	execute

Examples

In the following example, user *abc* is downloading the file *ssh.diff* from the SFTP server *ena-view1* to *disk0*:

```
RP/0/RP0/CPU0:router#sftp abc@ena-view1:ssh.diff disk0
```

In the following example, user *abc* is uploading multiple files from *disk 0:/sam_** to */users/abc/* on a remote SFTP server called *ena-view1*:

```
RP/0/RP0/CPU0:router# sftp disk0:/sam_* abc@ena-view1:/users/abc/
```

In the following example, user *admin* is downloading the file *run* from *disk0a:* to *disk0:/v6copy* on a local SFTP server using an IPv6 address:

```
RP/0/RP0/CPU0:router#sftp admin@[2:2:2::2]:disk0a:/run disk0:/V6copy
Connecting to 2:2:2::2...
Password:
```

```
disk0a:/run
  Transferred 308413 Bytes
  308413 bytes copied in 0 sec (338172)bytes/sec
```

```
RP/0/RP0/CPU0:router#dir disk0:/V6copy
```

```
Directory of disk0:
```

```
70144      -rwx  308413      Sun Oct 16 23:06:52 2011  V6copy
```

```
2102657024 bytes total (1537638400 bytes free)
```

In the following example, user *admin* is uploading the file *v6copy* from *disk0:* to *disk0a:/v6back* on a local SFTP server using an IPv6 address:

```
RP/0/RP0/CPU0:router#sftp disk0:/V6copy admin@[2:2:2::2]:disk0a:/v6back
Connecting to 2:2:2::2...
Password:
```

```
/disk0:/V6copy
  Transferred 308413 Bytes
  308413 bytes copied in 0 sec (421329)bytes/sec
```

```
RP/0/RP0/CPU0:router#dir disk0a:/v6back
```

```
Directory of disk0a:
```

```
66016      -rwx  308413      Sun Oct 16 23:07:28 2011  v6back
```

```
2102788096 bytes total (2098987008 bytes free)
```

In the following example, user *admin* is downloading the file *sampfile* from *disk0:* to *disk0a:/sampfile_v4* on a local SFTP server using an IPv4 address:

```
RP/0/RP0/CPU0:router#sftp admin@2.2.2.2:disk0:/sampfile disk0a:/sampfile_v4
Connecting to 2.2.2.2...
Password:
```

```
disk0:/sampfile
  Transferred 986 Bytes
  986 bytes copied in 0 sec (493000)bytes/sec
```

```
RP/0/RP0/CPU0:router#dir disk0a:/sampfile_v4
```

```
Directory of disk0a:
```

```
131520     -rwx   986      Tue Oct 18 05:37:00 2011  sampfile_v4
```

```
502710272 bytes total (502001664 bytes free)
```

In the following example, user *admin* is uploading the file *sampfile_v4* from *disk0a:* to *disk0:/sampfile_back* on a local SFTP server using an IPv4 address:

```
RP/0/RP0/CPU0:router#sftp disk0a:/sampfile_v4 admin@2.2.2.2:disk0:/sampfile_back
Connecting to 2.2.2.2...
Password:
```

```
disk0a:/sampfile_v4
  Transferred 986 Bytes
  986 bytes copied in 0 sec (564000)bytes/sec
```

```
RP/0/RP0/CPU0:router#dir disk0:/sampfile_back
```

```
Directory of disk0:
```

```
121765     -rwx   986      Tue Oct 18 05:39:00 2011  sampfile_back
```

```
524501272 bytes total (512507614 bytes free)
```

Related Commands	Command	Description
	ssh client source-interface, on page 23	Specifies the source IP address of a selected interface for all outgoing SSH connections.
	ssh client vrf, on page 24	Configures a new VRF for use by the SSH client.

sftp (Interactive Mode)

To enable users to start the secure FTP (SFTP) client, use the **sftp** command in XR EXEC mode.

```
sftp [ username @ host : remote-filename ] [ source-interface type interface-path-id ]
[ vrf vrf-name ]
```

Syntax Description		
	<i>username</i>	(Optional) Name of the user performing the file transfer. The at symbol (@) following the username is required.
	<i>hostname:remote-filename</i>	(Optional) Name of the Secure Shell File Transfer Protocol (SFTP) server. The colon (:) following the hostname is required.
	port <i>port-num</i>	Specifies the non-default port number of the server to which the SFTP client on the router attempts a connection. The port number ranges from 1025 - 65535.
	source-interface	(Optional) Specifies the source IP address of a selected interface for all outgoing SSH connections.
	<i>type</i>	Interface type. For more information, use the question mark (?) online help function.
	<i>interface-path-id</i>	Physical interface or virtual interface. Note Use the show interfaces command in EXEC mode to see a list of all interfaces currently configured on the router. For more information about the syntax for the router, use the question mark (?) online help function.
	vrf <i>vrf-name</i>	Specifies the name of the VRF associated with the source interface.

Command Default If no *username* argument is provided, the login name on the router is used. If no *hostname* argument is provided, the file is considered local.

Command Modes XR EXEC mode

Command History	Release	Modification
	Release 5.0.0	This command was introduced.

Usage Guidelines

The SFTP client, in the interactive mode, creates a secure SSH channel where the user can enter any supported command. When a user starts the SFTP client in an interactive mode, the SFTP client process creates a secure SSH channel and opens an editor where user can enter any supported command.

More than one request can be sent to the SFTP server to execute the commands. While there is no limit on the number of 'non-acknowledged' or outstanding requests to the server, the server might buffer or queue these requests for convenience. Therefore, there might be a logical sequence to the order of requests.

The following unix based commands are supported in the interactive mode:

- **bye**
- **cd** *<path>*
- **chmod** *<mode>* *<path>*
- **exit**
- **get** *<remote-path>* [*local-path*]
- **help**
- **ls** [*-alt*] [*path*]
- **mkdir** *<path>*
- **put** *<local-path>* [*remote-path*]
- **pwd**
- **quit**
- **rename** *<old-path>* *<new-path>*
- **rmdir** *<path>*
- **rm** *<path>*

The following commands are not supported:

- **lcd**, **lls**, **lpwd**, **lumask**, **lmkdir**
- **ln**, **symlink**
- **chgrp**, **chown**
- **!**, **!command**
- **?**
- **mget**, **mput**

Task ID	Task ID	Operations
	crypto	execute
	basic-services	execute

Examples

In the following example, user *admin* is downloading and uploading a file from/to an external SFTP server using an IPv6 address:

```
RP/0/RP0/CPU0:router#sftp admin@[2:2:2::2]

Connecting to 2:2:2::2...
Password:
sftp> pwd
Remote working directory: /
sftp> cd /auto/tftp-server1-users5/admin
sftp> get frmRouter /disk0:/frmRouterdownload

/auto/tftp-server1-users5/admin/frmRouter
  Transferred 1578 Bytes
  1578 bytes copied in 0 sec (27684)bytes/sec
sftp> put /disk0:/frmRouterdownload againtoServer

/disk0:/frmRouterdownload
  Transferred 1578 Bytes
  1578 bytes copied in 0 sec (14747)bytes/sec
sftp>
```

In the following example, user *abc* is downloading and uploading a file from/to an external SFTP server using an IPv4 address:

```
RP/0/RP0/CPU0:router#sftp abc@2.2.2.2
Connecting to 2.2.2.2...
Password:
sftp> pwd
Remote working directory: /
sftp> cd /auto/tftp-server1-users5/abc
sftp> get frmRouter /disk0:/frmRouterdownload

/auto/tftp-server1-users5/abc/frmRouter
  Transferred 1578 Bytes
  1578 bytes copied in 0 sec (27684)bytes/sec
sftp> put /disk0:/frmRouterdownload againtoServer

/disk0:/frmRouterdownload
  Transferred 1578 Bytes
  1578 bytes copied in 0 sec (14747)bytes/sec
sftp>
```

Related Commands

Command	Description
ssh client source-interface, on page 23	Specifies the source IP address of a selected interface for all outgoing SSH connections.
ssh client vrf, on page 24	Configures a new VRF for use by the SSH client.

show ssh

To display all incoming and outgoing connections to the router, use the **show ssh** command in XR EXEC mode.

show ssh

Syntax Description This command has no keywords or arguments.

Command Default None

Command Modes XR EXEC mode

Command History	Release	Modification
	Release 5.0.0	This command was introduced.

Usage Guidelines Use the **show ssh** command to display all incoming and outgoing Secure Shell (SSH) Version 1 (SSHv1) and SSH Version 2 (SSHv2) connections.

The connection type field in the command output of **show ssh** command shows as **port-forwarded local** for SSH port-forwarded sessions.

Use the **show ssh server** command to see the details of the SSH server. The **Port Forwarding** column shows as **local** for the port-forwarded session. Whereas, for a regular SSH session, the field displays as **disabled**.

Task ID	Task ID	Operations
	crypto	read

Examples

This is sample output from the **show ssh** command when SSH is enabled:

```
RP/0/RP0/CPU0:router# show ssh

SSH version : Cisco-2.0

id  pty  location  state          userid  host          ver  authentication
-----
Incoming sessions

Outgoing sessions
1   0/3/CPU0  SESSION_OPEN  lab  12.22.57.  v2  password
2   0/3/CPU0  SESSION_OPEN  lab  12.22.57.75 v2  keyboard-interactive
```

The following output is applicable for the **show ssh** command starting IOS-XR 5.3.2 releases and later.

```
RP/0/RP0/CPU0:router# show ssh
SSH version : Cisco-2.0
```

```

id  chan  pty      location      state      userid  host      ver
authentication connection type
-----
Incoming sessions
0  1  vty0    0/RSP0/CPU0  SESSION_OPEN  lab     12.22.57.75  v2
rsa-pubkey  Command-Line-Interface
0  2  vty1    0/RSP0/CPU0  SESSION_OPEN  lab     12.22.57.75  v2
rsa-pubkey  Command-Line-Interface
0  3          0/RSP0/CPU0  SESSION_OPEN  cisco   12.22.57.75  v2
rsa-pubkey  Sftp-Subsystem
1          vty7    0/RSP0/CPU0  SESSION_OPEN  cisco   12.22.22.57  v1 password
          Command-Line-Interface
3  1          0/RSP0/CPU0  SESSION_OPEN  lab     12.22.57.75  v2 password
          Netconf-Subsystem
4  1  vty3    0/RSP0/CPU0  SESSION_OPEN  lab     192.168.1.55  v2 password
          Command-Line-Interface

Outgoing sessions
1          0/RSP0/CPU0  SESSION_OPEN  lab     192.168.1.51  v2 password

```

This table describes significant fields shown in the display.

Table 1: show ssh Field Descriptions

Field	Description
id	Session identifier for the incoming and outgoing SSH connections.
chan	Channel identifier for incoming (v2) SSH connections. NULL for SSH v1 sessions.
pty	pty-id allocated for the incoming session. Null for outgoing SSH connection.
location	Specifies the location of the SSH server for an incoming connection. For an outgoing connection, location specifies from which route processor the SSH session is initiated.
state	The SSH state that the connection is currently in.
userid	Authentication, authorization and accounting (AAA) username used to connect to or from the router.
host	IP address of the remote peer.
ver	Specifies if the connection type is SSHv1 or SSHv2.
authentication	Specifies the type of authentication method chosen by the user.
connection type	Specifies which application is performed over this connection (Command-Line-Interface, Remote-Command, Scp, Sftp-Subsystem, or Netconf-Subsystem)

The following is a sample output of SSH port-forwarded session:

```
Router#show ssh
Wed Oct 14 11:22:05.575 UTC
```

```

SSH version : Cisco-2.0

id chan pty location state userid host ver authentication connection type
-----
Incoming sessions
15 1 XXX 0/RP0/CPU0 SESSION_OPEN admin 192.168.122.1 v2 password
port-forwarded-local

Outgoing sessions

Router#

```

The following is a sample output of **show ssh server** command with SSH port forwarding enabled:

```

Router#show ssh server
Tue Sep 7 17:43:22.483 IST
-----
SSH Server Parameters
-----

Current supported versions := v2
                        SSH port := 22
                        SSH vrfs := vrfname:=default(v4-acl:=, v6-acl:=)
                        Netconf Port := 830
                        Netconf Vrfs := vrfname:=default(v4-acl:=, v6-acl:=)

Algorithms
-----
Hostkey Algorithms :=
x509v3-ssh-rsa,ecdsa-sha2-nistp521,ecdsa-sha2-nistp384,ecdsa-sha2-nistp256,rsa-sha2-512,rsa-sha2-256,ssh-rsa,ssh-dsa,ssh-ed25519

Key-Exchange Algorithms :=
ecdh-sha2-nistp521,ecdh-sha2-nistp384,ecdh-sha2-nistp256,diffie-hellman-group14-sha1

Encryption Algorithms :=
aes128-ctr,aes192-ctr,aes256-ctr,aes128-gcm@openssh.com,aes256-gcm@openssh.com
Mac Algorithms := hmac-sha2-512,hmac-sha2-256,hmac-sha1

Authentication Method Supported
-----
PublicKey := Yes
Password := Yes
Keyboard-Interactive := Yes
Certificate Based := Yes

Others
-----
DSCP := 0
Ratelimit := 600
Sessionlimit := 110
Rekeytime := 30
Server rekeyvolume := 1024
TCP window scale factor := 1
Backup Server := Disabled
Host Trustpoint :=
User Trustpoint := tes,test,x509user
Port Forwarding := local
Max Authentication Limit := 16
Certificate username := Common name(CN) User principle name(UPN)
Router#

```

Related Commands	Command	Description
	show sessions	Displays information about open Telnet or rlogin connections. For more information, see the <i>System Management Command Reference for Cisco NCS 6000 Series Routers</i>
	show ssh session details, on page 16	Displays the details for all the incoming and outgoing SSHv2 connections, to the router.

show ssh history

To display the last hundred SSH connections that were terminated, use the **show ssh history** command in XR EXEC mode.

show ssh history

Syntax Description This command has no keywords or arguments.

Command Default None

Command Modes XR EXEC mode

Command History	Release	Modification
	Release 6.4.1	This command was introduced.

Usage Guidelines No specific guidelines impact the use of this command.

Task ID	Task ID	Operations
	crypto	read

Examples

The following is sample output from the **show ssh history** command to display the last hundred SSH sessions that were terminated:

```
RP/0/RP0/CPU0:router# show ssh history

SSH version : Cisco-2.0

id      chan pty      location      userid      host      ver authentication
connection type
-----
Incoming sessions
1       1     XXXXX  0/RP0/CPU0   root      10.105.227.252  v2 password
Netconf-Subsystem
2       1     XXXXX  0/RP0/CPU0   root      10.105.227.252  v2 password
Netconf-Subsystem
3       1     XXXXX  0/RP0/CPU0   root      10.105.227.252  v2 password
```

show ssh history details

```

Netconf-Subsystem
4      1      XXXXX  0/RP0/CPU0    root    10.105.227.252    v2 password
Netconf-Subsystem
5      1      XXXXX  0/RP0/CPU0    root    10.105.227.252    v2 password
Netconf-Subsystem
6      1      XXXXX  0/RP0/CPU0    root    10.105.227.252    v2 password
Netconf-Subsystem
7      1      XXXXX  0/RP0/CPU0    root    10.105.227.252    v2 password
Netconf-Subsystem
8      1      XXXXX  0/RP0/CPU0    root    10.105.227.252    v2 password
Netconf-Subsystem
9      1      vty0   0/RP0/CPU0    root    10.196.98.106     v2 key-intr
Command-Line-Interface

```

Pty – VTU number used. This is represented as ‘XXXX’ when connection type is SFTP, SCP or Netconf.

show ssh history details

To display the last hundred SSH connections that were terminated, and also the start and end time of the session, use the **show ssh history details** command in XR EXEC mode.

show ssh history details

Syntax Description This command has no keywords or arguments.

Command Default None

Command Modes XR EXEC mode

Command History	Release	Modification
	Release 6.4.1	This command was introduced.

Usage Guidelines No specific guidelines impact the use of this command.

Task ID	Task ID	Operations
	crypto	read

Examples

The following is sample output from the **show ssh history details** command to display the last hundred SSH sessions that were terminated along with the start and end time of the sessions:

```

RP/0/RP0/CPU0:router# show ssh history details

SSH version : Cisco-2.0

id      key-exchange      pubkey      incipher      outcipher      inmac
outmac      start_time      end_time

```

```

Incoming Session
1      ecdh-sha2-nistp256      ssh-rsa      aes128-ctr aes128-ctr hmac-sha2-256
hmac-sha2-256 14-02-18 14:00:39      14-02-18 14:00:41
2      ecdh-sha2-nistp256      ssh-rsa      aes128-ctr aes128-ctr hmac-sha2-256
hmac-sha2-256 14-02-18 16:21:54      14-02-18 16:21:55
3      ecdh-sha2-nistp256      ssh-rsa      aes128-ctr aes128-ctr hmac-sha2-256
hmac-sha2-256 14-02-18 16:22:18      14-02-18 16:22:19
4      ecdh-sha2-nistp256      ssh-rsa      aes128-ctr aes128-ctr hmac-sha2-256
hmac-sha2-256 15-02-18 12:17:44      15-02-18 12:17:46
5      ecdh-sha2-nistp256      ssh-rsa      aes128-ctr aes128-ctr hmac-sha2-256
hmac-sha2-256 15-02-18 12:18:16      15-02-18 12:18:17
6      ecdh-sha2-nistp256      ssh-rsa      aes128-ctr aes128-ctr hmac-sha2-256
hmac-sha2-256 15-02-18 14:44:08      15-02-18 14:44:09
7      ecdh-sha2-nistp256      ssh-rsa      aes128-ctr aes128-ctr hmac-sha2-256
hmac-sha2-256 15-02-18 14:50:15      15-02-18 14:50:16
8      ecdh-sha2-nistp256      ssh-rsa      aes128-ctr aes128-ctr hmac-sha2-256
hmac-sha2-256 15-02-18 14:50:52      15-02-18 14:50:53
9      ecdh-sha2-nistp256      ssh-rsa      aes128-ctr aes128-ctr hmac-sha2-256
hmac-sha2-256 15-02-18 15:31:26      15-02-18 15:31:38

```

This table describes the significant fields shown in the display.

Table 2: Field Descriptions

Field	Description
session	Session identifier for the incoming and outgoing SSH connections.
key-exchange	Key exchange algorithm chosen by both peers to authenticate each other.
pubkey	Public key algorithm chosen for key exchange.
incipher	Encryption cipher chosen for the receiver traffic.
outcipher	Encryption cipher chosen for the transmitter traffic.
inmac	Authentication (message digest) algorithm chosen for the receiver traffic.
outmac	Authentication (message digest) algorithm chosen for the transmitter traffic.
start_time	Start time of the session.
end_time	End time of the session.

show ssh rekey

To display session rekey details such as session id, session rekey count, time to rekey, data to rekey, use the **show ssh rekey** command in XR EXEC mode.

```
show ssh rekey
```

Command Default None

Command Modes XR EXEC mode

Command History

Release	Modification
Release 5.0.0	This command was introduced.

Usage Guidelines The ssh rekey data is updated ten times between two consecutive rekeys.

Task ID

Task ID	Operations
crypto	read

Examples

The following sample output is from the **show ssh rekey** command:

```
# show ssh rekey

id      RekeyCount  TimeToRekey (min)  VolumeToRekey (MB)
-----
Incoming Session
0        8            59.5               1024.0
```

This table describes the fields shown in the display.

Table 3: show ssh rekey Field Descriptions

Field	Description
Rekey Count	Number of times the ssh rekey is generated.
TimeToRekey	Time remaining (in minutes) before the ssh rekey is regenerated based on the value set using the ssh server rekey-time command.
VolumeToRekey	Volume remaining (in megabytes) before the ssh rekey is regenerated based on the value set using the ssh server rekey-volume command.

show ssh session details

To display the details for all incoming and outgoing Secure Shell Version 2 (SSHv2) connections, use the **show ssh session details** command in XR EXEC mode.

show ssh session details

Syntax Description This command has no keywords or arguments.

Command Default None

Command Modes XR EXEC mode

Command History

Release	Modification
Release 5.0.0	This command was introduced.

Usage Guidelines Use the **show ssh session details** command to display a detailed report of the SSHv2 connections to or from the router, including the cipher chosen for the specific session.

Task ID

Task ID	Operations
crypto	read

Examples

The following is sample output from the **show ssh session details** command to display the details for all the incoming and outgoing SSHv2 connections:

```
RP/0/RP0/CPU0:router# show ssh session details

id key-exchange          pubkey    incipher  outcipher  inmac    outmac
-----
Incoming Session
0 diffie-hellman-group14  ssh-rsa  aes128-ctr  aes128-ctr  hmac-sha1  hmac-sha1
1 ecdh-sha2-nistp521     ssh-rsa  aes256-ctr  aes256-ctr  hmac-sha2-512  hmac-sha2-512
```

This table describes the significant fields shown in the display.

Table 4: show ssh session details Field Descriptions

Field	Description
session	Session identifier for the incoming and outgoing SSH connections.
key-exchange	Key exchange algorithm chosen by both peers to authenticate each other.
pubkey	Public key algorithm chosen for key exchange.
incipher	Encryption cipher chosen for the Rx traffic.
outcipher	Encryption cipher chosen for the Tx traffic.
inmac	Authentication (message digest) algorithm chosen for the Rx traffic.
outmac	Authentication (message digest) algorithm chosen for the Tx traffic.

Related Commands

Command	Description
show sessions	Displays information about open Telnet or rlogin connections.

Command	Description
show ssh, on page 10	Displays all the incoming and outgoing connections to the router.

show tech-support ssh

To automatically run show commands that display system information, use the show tech-support command, use the **show tech-support ssh** command in XR EXEC mode.

show tech-support ssh

Syntax Description This command has no keywords or arguments.

Command Default None

Command Modes XR EXEC mode

Command History

Release	Modification
Release 6.4.1	This command was introduced.

Usage Guidelines No specific guidelines impact the use of this command.

Task ID

Task ID	Operations
crypto	read

Examples The following is sample output from the **show tech-support ssh** command:

```
RP/0/RP0/CPU0:router# show tech-support ssh
++ Show tech start time: 2018-Feb-20.123016.IST ++
Tue Feb 20 12:30:27 IST 2018 Waiting for gathering to complete
.....
Tue Feb 20 12:32:35 IST 2018 Compressing show tech output
Show tech output available at 0/RP0/CPU0 :
/harddisk:/showtech/showtech-ssh-2018-Feb-20.123016.IST.tgz
++ Show tech end time: 2018-Feb-20.123236.IST ++
RP/0/RP0/CPU0:turin-sec1#
```

The **show tech-support ssh** command collects the output of these CLI:

Command	Description
show logging	Displays the contents of the logging buffer.
show context location all	

Command	Description
show running-config	Displays the contents of the currently running configuration or a subset of that configuration.
show ip int brief	Displays brief information about each interface.
show ssh	Displays all incoming and outgoing connections to the router.
show ssh session details	Displays the details for all the incoming and outgoing SSHv2 connections, to the router.
show ssh rekey	Displays session rekey details such as session id, session rekey count, time to rekey, data to rekey.
show ssh history	Displays the last hundred SSH connections that were terminated.
show tty trace info all all	
show tty trace error all all	

ssh

To start the Secure Shell (SSH) client connection and enable an outbound connection to an SSH server, use the **ssh** command in XR EXEC mode.

Syntax Description

<i>ipv4-address</i>	IPv4 address in A:B:C:D format.
<i>ipv6-address</i>	IPv6 address in X:X::X format.
<i>hostname</i>	Hostname of the remote node. If the hostname has both IPv4 and IPv6 addresses, the IPv4 address is used.
username <i>user-id</i>	(Optional) Specifies the username to use when logging in on the remote networking device running the SSH server. If no user ID is specified, the default is the current user ID.
cipher	
source interface	(Optional) Specifies the source IP address of a selected interface for all outgoing SSH connections.
<i>type</i>	Interface type. For more information, use the question mark (?)online help function.
<i>interface-path-id</i>	Physical interface or virtual interface.
Note	Use the show interfaces command in XR EXEC mode to see a list of all interfaces currently configured on the router.
	For more information about the syntax for the router, use the question mark(?)online help function.

command (Optional) Specifies a remote command. Adding this keyword prompts the SSHv2 server to parse and execute the `ssh command` in non-interactive mode instead of initiating the interactive session.

Command Default None

Command Modes XR EXEC mode

Command History

Release	Modification
Release 5.0.0	This command was introduced.

Usage Guidelines Use the `ssh` command to make an outbound client connection. The SSH client tries to make an SSHv2 connection to the remote peer. If the remote peer supports only the SSHv1 server, it internally spawns an SSHv1 connection to the remote server. The process of the remote peer version detection and spawning the appropriate client connection is transparent to the user.

If is specified in the `ssh` command, the `ssh` interface takes precedence over the interface specified in the `ssh client source-interface ssh client source-interface`, on page 23 command.

Use the `command` keyword to enable the SSHv2 server to parse and execute the `ssh` command in non-interactive mode instead of initiating an interactive session.

Task ID

Task ID	Operations
crypto	execute
basic-services	execute

Examples The following sample output is from the `ssh` command to enable an outbound SSH client connection:

```
RP/0/RP0/CPU0:router# ssh username userabc

Password:
Remote-host>
```

Related Commands

Command	Description
show ssh, on page 10	Displays all the incoming and outgoing connections to the router.

ssh algorithms cipher

To configure the list of supported SSH algorithms on the client or on the server, use the `ssh client algorithms cipher` command or `ssh server algorithms cipher` command in XR Config mode. To remove the configuration, use the `no` form of this command.

```
ssh {client | server} algorithms cipher {aes256-cbc | aes256-ctr | aes192-ctr | aes192-cbc |
aes128-ctr | aes128-cbc | aes128-gcm@openssh.com | aes256-gcm@openssh.com | 3des-cbc}
```

Syntax Description	client	Configures the list of supported SSH algorithms on the client.
	server	Configures the list of supported SSH algorithms on the server.

Command Default None

Command Modes XR Config mode

Command History	Release	Modification
	Release 7.0.1	This command was introduced.

Usage Guidelines No specific guidelines impact the use of this command.

Task ID	Task	Operation
	crypto	read, write

This example shows how to enable CTR cipher on the client and CBC cipher on the server:

```
Router1#ssh client algorithms cipher aes128-ctr aes192-ctr aes256-ctr
```

```
Router1#ssh server algorithms cipher aes128-cbc aes192-cbc aes256-cbc 3des-cbc
```

Related Commands	Command	Description
	ssh client enable cipher , on page 21	Enables CBC mode ciphers on the SSH client.
	ssh server enable cipher, on page 29	Enables CBC mode ciphers on the SSH server.

ssh client enable cipher

To enable the CBC mode ciphers 3DES-CBC and/or AES-CBC for an SSH client connection, use the **ssh client enable cipher** command in XR Config mode. To disable the ciphers, use the **no** form of this command.

```
ssh client enable cipher {aes-cbc | 3des-cbc}
```

Syntax Description	3des-cbc	Specifies that the 3DES-CBC cipher be enabled for the SSH client connection.
	aes-cbc	Specifies that the AES-CBC cipher be enabled for the SSH client connection.

Command Default CBC mode ciphers are disabled.

Command Modes XR Config mode

Command History	Release	Modification
	Release 6.3.1	This command was introduced.

Usage Guidelines The support for CBC ciphers were disabled by default, from Cisco IOS XR Software Release 6.1.2. Hence, **ssh client enable cipher** and **ssh server enable cipher** commands were introduced to explicitly enable CBC ciphers in required scenarios.

If a client tries to reach the router which acts as a server with CBC cipher, and if the CBC cipher is not explicitly enabled on that router, then the system displays an error message:

```
ssh root@x.x.x. -c aes128-cbc
Unable to negotiate with x.x.x.x port 22: no matching cipher found.
Their offer: aes128-ctr,aes192-ctr,aes256-ctr,aes128-gcm@openssh.com,aes256-gcm@openssh.com
```

You must configure **ssh server enable cipher aes-cbc** command in this case, to connect to the router using the CBC cipher.

Task ID	Task ID	Operation
	crypto read, write	

Examples The following example shows how to enable the 3DES-CBC and AES-CBC ciphers for an SSH client connection:

```
Router# configure
Router(config)# ssh client enable cipher aes-cbc 3des-cbc
Router(config)# commit
```

Related Commands	Command	Description
	ssh server enable cipher, on page 29	Enables CBC mode ciphers on the SSH server.

ssh client knownhost

To authenticate a server public key (pubkey), use the **ssh client knownhost** command in XR Config mode. To disable authentication of a server pubkey, use the **no** form of this command.

ssh client knownhost device: /filename

Syntax Description	device:/filename	Complete path of the filename (for example, slot0:/server_pubkey). The colon (:) and slash (/) are required.

Command Default	None

Command Modes XR Config mode

Command History	Release	Modification
	Release 5.0.0	This command was introduced.

Usage Guidelines The *server pubkey* is a cryptographic system that uses two keys at the client end—a public key known to everyone and a private, or secret, key known only to the owner of the keys. In the absence of certificates, the server pubkey is transported to the client through an out-of-band secure channel. The client stores this pubkey in its local database and compares this key against the key supplied by the server during the early stage of key negotiation for a session-building handshake. If the key is not matched or no key is found in the local database of the client, users are prompted to either accept or reject the session.

The operative assumption is that the first time the server pubkey is retrieved through an out-of-band secure channel, it is stored in the local database. This process is identical to the current model adapted by Secure Shell (SSH) implementations in the UNIX environment.

Task ID	Task ID	Operations
	crypto	read, write

Examples The following sample output is from the **ssh client knownhost** command:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# ssh client knownhost disk0:/ssh.knownhost
RP/0/RP0/CPU0:router(config)# commit
RP/0/RP0/CPU0:router# ssh host1 username user1234
Host key not found from the list of known hosts.
Are you sure you want to continue connecting (yes/no)? yes
Password:
RP/0/RP0/CPU0:host1# exit
RP/0/RP0/CPU0:router# ssh host1 username user1234
```

ssh client source-interface

To specify the source IP address of a selected interface for all outgoing Secure Shell (SSH) connections, use the **ssh client source-interface** command in XR Config mode. To disable use of the specified interface IP address, use the **no** form of this command.

ssh client source-interface *type interface-path-id*

Syntax Description	<i>type</i>	Interface type. For more information, use the question mark (?) online help function.
--------------------	-------------	---

interface-path-id Physical interface or virtual interface.

Note Use the **show interfaces** command to see a list of all interfaces currently configured on the router.

For more information about the syntax for the router, use the question mark (?) online help function.

Command Default No source interface is used.

Command Modes XR Config mode

Command History	Release	Modification
	Release 5.0.0	This command was introduced.

Usage Guidelines Use the **ssh client source-interface** command to set the IP address of the specified interface for all outgoing SSH connections. If this command is not configured, TCP chooses the source IP address when the socket is connected, based on the outgoing interface used—which in turn is based on the route required to reach the server. This command applies to outbound shell over SSH as well as Secure Shell File Transfer Protocol (SFTP) sessions, which use the ssh client as a transport.

The source-interface configuration affects connections only to the remote host in the same address family. The system database (Sysdb) verifies that the interface specified in the command has a corresponding IP address (in the same family) configured.

Task ID	Task ID	Operations
	crypto	read, write

Examples The following example shows how to set the IP address of the Management Ethernet interface for all outgoing SSH connections:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# ssh client source-interface MgmtEth 0//CPU0/0
```

ssh client vrf

To configure a new VRF for use by the SSH client, use the **ssh client vrf** command in XR Config mode. To remove the specified VRF, use the **no** form of this command.

ssh client vrf *vrf-name*

Syntax Description	<i>vrf-name</i>
	Specifies the name of the VRF to be used by the SSH client.

Command Default None

Command Modes XR Config mode

Command History	Release	Modification
	Release 5.0.0	This command was introduced.

Command History	Release	Modification
	Release 3.8.0	This command was introduced.

Usage Guidelines An SSH client can have only one VRF.

If a specific VRF is not configured for the SSH client, the default VRF is assumed when applying other SSH client-related commands, such as [ssh client knownhost, on page 22](#) or [ssh client source-interface, on page 23](#).

Task ID	Task ID	Operations
	crypto	read, write

Examples

The following example shows the SSH client being configured to start with the specified VRF:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# ssh client vrf green
```

Related Commands	Command	Description
	ssh client dscp <value from 0 - 63>	SSH Client supports setting DSCP value in the outgoing packets. If not configured, the default DSCP value set in packets is 16 (for both client and server).

ssh server

To bring up the Secure Shell (SSH) server and to configure one or more VRFs for its use, use the **ssh server** command in XR Config mode. To stop the SSH server from receiving any further connections for the specified VRF, use the **no** form of this command. Optionally ACLs for IPv4 and IPv6 can be used to restrict access to the server before the port is opened.

```
ssh server vrf vrf-name [ipv4 access-list ipv4 access list name ] [ipv6 access-list ipv6 access list name ] ]
ssh server v2
```

Syntax Description	vrf <i>vrf-name</i>	Specifies the name of the VRF to be used by the SSH server. The maximum VRF length is 32 characters. Note If no VRF is specified, the default VRF is assumed.
	ipv4 access-list <i>access list name</i>	Configures an IPv4 access-list for access restrictions to the ssh server.
	ipv6 access-list <i>access list name</i>	Configures an IPv6 access-list for access restrictions to the ssh server
	v2	Forces the SSH server version to be of only version 2.

Command Default The default SSH server version is 2 (SSHv2), which falls back to 1 (SSHv1) if the incoming SSH client connection is set to SSHv1.

Command Modes XR Config mode

Command History	Release	Modification
	Release 5.0.0	This command was introduced.

Usage Guidelines An SSH server must be configured at minimum for one VRF. If you delete all configured VRFs, including the default, the SSH server process stops. If you do not configure a specific VRF for the SSH client when applying other commands, such as **ssh client knownhost** or **ssh client source-interface**, the default VRF is assumed.

The SSH server listens for an incoming client connection on port 22. This server handles both Secure Shell Version 1 (SSHv1) and SSHv2 incoming client connections for both IPv4 and IPv6 address families. To accept only Secure Shell Version 2 connections, use the [ssh server v2, on page 34](#) command.

To verify that the SSH server is up and running, use the **show process sshd** command.

Task ID	Task ID	Operations
	crypto	read, write

Examples In the following example, the SSH server is brought up to receive connections for VRF “green”:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# ssh
```

Examples In the following example, the SSH server is configured to use IPv4 ACLs:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# ssh vrf vrf nameipv4 access-list access list name
```

Related Commands	Command	Description
	show processes	Displays information about the SSH server.
	ssh server v2, on page 34	Forces the SSH server version to be only 2 (SSHv2).
	ssh server dscp <value from 0 - 63>	SSH server supports setting DSCP value in the outgoing packets. If not configured, the default DSCP value set in packets is 16 (for both client and server).

ssh server algorithms host-key

To configure the allowed SSH host-key pair algorithms from the list of auto-generated host-key pairs on the SSH server, use the **ssh server algorithms host-key** command in XR Config mode. To remove the configuration, use the **no** form of this command.

```
ssh server algorithms host-key { dsa | ecdsa-nistp256 | ecdsa-nistp384 | ecdsa-nistp521 |
rsa }
```

Syntax Description	<ul style="list-style-type: none"> • dsa • ecdsa-nistp256 • ecdsa-nistp384 • ecdsa-nistp521 • rsa 	<p>Selects the specified host keys to be offered to the SSH client.</p> <p>While configuring this, you can specify the algorithms in any order.</p>
Command Default	In the absence of this configuration, the SSH server considers that it can send all the available algorithms to the user as host key algorithm, based on the availability of the key or the certificate.	
Command Modes	XR Config mode	
Usage Guidelines	<p>This configuration is optional. If this configuration is not present, it is considered that all the SSH host-key pairs are configured. In that case, the SSH client is allowed to connect to the SSH sever with any of the host-key pairs.</p> <p>You can also use the crypto key zeroize command to remove the SSH host keys that are not required.</p> <p>With the introduction of the automatic generation of SSH host-key pairs, the show crypto key mypubkey command output displays key information of all the keys that are auto-generated. Before its introduction, the output of this command displayed key information of only those host-key pairs that were explicitly configured using the crypto key generate command.</p>	
Task ID	Task ID	Operation
	crypto	read, write

This example shows how to select the **ecdsa** algorithm from the list of auto-generated host-key pairs on the SSH server:

```
Router(config)#ssh server algorithms host-key ecdsa-nistp521
```

ssh disable hmac

To disable HMAC cryptographic algorithm on the SSH server, use the **ssh server disable hmac** command, and to disable HMAC cryptographic algorithm on the SSH client, use the **ssh client disable hmac** command in XR Config mode. To disable this feature, use the **no** form of this command.

```
ssh {client | server} disable hmac {hmac-sha1 | hmac-sha2-512}
```

Syntax Description	
hmac-sha1	Disables the SHA-1 HMAC cryptographic algorithm.
hmac-sha2-512	Disables the SHA-2 HMAC cryptographic algorithm.
Note	This option is available only for the server .

Command Default None

Command Modes XR Config mode

Command History	Release	Modification
	Release 7.0.1	This command was introduced.

Usage Guidelines No specific guidelines impact the use of this command.

Task ID	Task ID	Operation
	crypto read, write	

This example shows how to disable SHA1 HMAC cryptographic algorithm on the SSH client:

```
Router#ssh client disable hmac hmac-sha1
```

This example shows how to disable SHA-2 HMAC cryptographic algorithm on the SSH server:

```
Router#ssh server disable hmac hmac-sha2-512
```

ssh server enable cipher

To enable CBC mode ciphers 3DES-CBC and/or AES-CBC for an SSH server connection, use the **ssh server enable cipher** command in XR Config mode. To disable the ciphers, use the **no** form of this command.

```
ssh server enable cipher {aes-cbc | 3des-cbc}
```

Syntax Description	3des-cbc Specifies that the 3DES-CBC cipher be enabled for the SSH server connection.				
	aes-cbc Specifies that the AES-CBC cipher be enabled for the SSH server connection.				
Command Default	CBC mode ciphers are disabled.				
Command Modes	XR Config mode				
Command History	<table border="1"> <thead> <tr> <th>Release</th> <th>Modification</th> </tr> </thead> <tbody> <tr> <td>Release 6.3.1</td> <td>This command was introduced.</td> </tr> </tbody> </table>	Release	Modification	Release 6.3.1	This command was introduced.
Release	Modification				
Release 6.3.1	This command was introduced.				
Usage Guidelines	The support for CBC ciphers were disabled by default, from Cisco IOS XR Software Release 6.1.2. Hence, ssh client enable cipher and ssh server enable cipher commands were introduced to explicitly enable CBC ciphers in required scenarios.				
Task ID	<table border="1"> <thead> <tr> <th>Task ID</th> <th>Operation</th> </tr> </thead> <tbody> <tr> <td>crypto</td> <td>read, write</td> </tr> </tbody> </table>	Task ID	Operation	crypto	read, write
Task ID	Operation				
crypto	read, write				
Examples	<p>The following example shows how to enable the 3DES-CBC and AES-CBC ciphers for an SSH server connection:</p> <pre>Router# configure Router(config)# ssh server enable cipher aes-cbc 3des-cbc Router(config)# commit</pre>				
Related Commands	<table border="1"> <thead> <tr> <th>Command</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>ssh client enable cipher , on page 21</td> <td>Enables CBC mode ciphers on the SSH client.</td> </tr> </tbody> </table>	Command	Description	ssh client enable cipher , on page 21	Enables CBC mode ciphers on the SSH client.
Command	Description				
ssh client enable cipher , on page 21	Enables CBC mode ciphers on the SSH client.				

ssh server rekey-time

To configure rekey of the ssh server key based on time, use the **ssh server** command in XR Config mode. Use the **no** form of this command to remove the rekey interval.

ssh server rekey-time *time in minutes*

Syntax Description	rekey-time <i>time in minutes</i>	Specifies the rekey-time interval in minutes. The range is between 30 to 1440 minutes.
	Note	If no time interval is specified, the default interval is considered to be 60 minutes.

Command Default None.

Command Modes XR Config mode

Command History	Release	Modification
	Release 5.0.0	This command was introduced.

Task ID	Task ID	Operations
	crypto	read, write

Examples

In the following example, the SSH server rekey-interval of 450 minutes is used:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# ssh server rekey-time 450
```

ssh server rekey-volume

To configure a volume-based rekey threshold for an SSH session, use the **ssh server** command in XR Config mode. Use the **no** form of this command to remove the volume-based rekey threshold.

ssh server rekey-volume *data in megabytes*

Syntax Description	rekey-volume <i>data in megabytes</i>	Specifies the volume-based rekey threshold in megabytes. The range is between 1024 to 4095 megabytes.
	Note	If no volume threshold is specified, the default size is considered to be 1024 MB.

Command Default None.

Command Modes XR Config mode

Command History	Release	Modification
	Release 5.0.0	This command was introduced.

Task ID	Task ID	Operations
	crypto	read, write

Examples

In the following example, the SSH server rekey-volume of 2048 minutes is used:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# ssh
```

ssh server logging

To enable SSH server logging, use the **ssh server logging** command in XR Config mode. To discontinue SSH server logging, use the **no** form of this command.

ssh server logging

Syntax Description This command has no keywords or arguments.

Command Default None

Command Modes XR Config mode

Command History	Release	Modification
	Release 5.0.0	This command was introduced.

Usage Guidelines Once you configure the logging, the following messages are displayed:

- Warning: The requested term-type is not supported
- SSH v2 connection from %s succeeded (*user:%s, cipher:%s, mac:%s, pty:%s*)

The warning message appears if you try to connect using an unsupported terminal type. Routers running the Cisco IOS XR software support only the vt100 terminal type.

The second message confirms a successful login.

Task ID	Task ID	Operations
	crypto	read, write

Examples

The following example shows the initiation of an SSH server logging:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router (config)# ssh server logging
```

Related Commands

Command	Description
ssh server, on page 25	Initiates the SSH server.

ssh server rate-limit

To limit the number of incoming Secure Shell (SSH) connection requests allowed per minute, use the **ssh server rate-limit** command in XR Config mode. To return to the default value, use the **no** form of this command.

ssh server rate-limit *rate-limit*

Syntax Description

rate-limit Number of incoming SSH connection requests allowed per minute. Range is from 1 to 120. When setting it to 60 attempts per minute, it basically means that we can only allow 1 per second. If you set up 2 sessions at the same time from 2 different consoles, one of them will get rate limited. This is connection attempts to the ssh server, not bound per interface/username or anything like that. So value of 30 means 1 session per 2 seconds and so forth.

Command Default

rate-limit: 60 connection requests per minute

Command Modes

XR Config mode

Command History

Release	Modification
Release 5.0.0	This command was introduced.

Usage Guidelines

Use the **ssh server rate-limit** command to limit the incoming SSH connection requests to the configured rate. Any connection request beyond the rate limit is rejected by the SSH server. Changing the rate limit does not affect established SSH sessions.

If, for example, the *rate-limit* argument is set to 30, then 30 requests are allowed per minute, or more precisely, a two-second interval between connections is enforced.

Task ID

Task ID	Operations
crypto	read, write

Examples

The following example shows how to set the limit of incoming SSH connection requests to 20 per minute:


```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# ssh server rate-limit 20
```

ssh server session-limit

To configure the number of allowable concurrent incoming Secure Shell (SSH) sessions, use the **ssh server session-limit** command in XR Config mode. To return to the default value, use the **no** form of this command.

ssh server session-limit *sessions*

Syntax Description

sessions Number of incoming SSH sessions allowed across the router. The range is from 1 to 100.

Note Although CLI output option has 1024, you are recommended to configure session-limit not more than 100. High session count may cause resource exhaustion .

Command Default

sessions: 64 per router

Command Modes

XR Config mode

Command History

Release	Modification
Release 5.0.0	This command was introduced.

Usage Guidelines

Use the **ssh server session-limit** command to configure the limit of allowable concurrent incoming SSH connections. Outgoing connections are not part of the limit.

Task ID

Task ID	Operations
crypto	read, write

Examples

The following example shows how to set the limit of incoming SSH connections to 50:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# ssh server session-limit 50
```

Related Commands

Command	Description
show processes	Displays information about the SSH server.

ssh server v2

To force the SSH server version to be only 2 (SSHv2), use the **ssh server v2** command in XR Config mode. To bring down an SSH server for SSHv2, use the **no** form of this command.

ssh server v2

Syntax Description This command has no keywords or arguments.

Command Default None

Command Modes XR Config mode

Command History	Release	Modification
	Release 5.0.0	This command was introduced.

Usage Guidelines Only SSHv2 client connections are allowed.

Task ID	Task ID	Operations
	crypto	read, write

Examples The following example shows how to initiate the SSH server version to be only SSHv2:

```
RP/0/RP0/CPU0:router#configure
RP/0/RP0/CPU0:router(config)# ssh server v2
```

ssh server netconf port

To configure a port for the netconf SSH server, use the **ssh server netconf port** command in XR Config mode. To return to the default port, use the **no** form of the command.

ssh server netconf port *port number*

Syntax Description	port	Port number for the netconf SSH server (default port number is 830).
	<i>port-number</i>	

Command Default The default port number is 830.

Command Modes XR Config mode

Command History	Release	Modification
	Release 5.3.0	This command was introduced.
	Release 6.0	The ssh server netconf command is no longer auto completed to configure the default port. This command is now optional

Usage Guidelines Starting with IOS-XR 6.0.0 it is no longer sufficient to configure a netconf port to enable netconf subsystem support. ssh server netconf needs to be at least configured for one vrf.

Task ID	Task ID	Operations
	crypto	read, write

Examples This example shows how to use the ssh server netconf port command with port 831:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# ssh
```

Related Commands	Command	Description
	ssh server netconf	Configures the vrf(s), where netconf subsystem requests are to be received.
	netconf-yang agent ssh	Configures the ssh netconf-yang backend for the netconf subsystem (Required to allow the system to service netconf-yang requests). For more information, see the <i>Cisco ASR 9000 Series Aggregation Services Router System Management Command Reference</i> .

ssh timeout

To configure the timeout value for authentication, authorization, and accounting (AAA) user authentication, use the **ssh timeout** command in XR Config mode. To set the timeout value to the default time, use the **no** form of this command.

ssh timeout *seconds*

Syntax Description *seconds* Time period (in seconds) for user authentication. The range is from 5 to 120.

Command Default *seconds: 30*

Command Modes XR Config mode

Command History	Release	Modification
	Release 5.0.0	This command was introduced.

Usage Guidelines Use the **ssh timeout** command to configure the timeout value for user authentication to AAA. If the user fails to authenticate itself within the configured time to AAA, the connection is terminated. If no value is configured, the default value of 30 seconds is used.

Task ID	Task ID	Operations
	crypto	read, write

Examples In the following example, the timeout value for AAA user authentication is set to 60 seconds:

```
RP/0/RP0/CPU0:router# configure
RP/0/RP0/CPU0:router(config)# ssh timeout 60
```