

Secure Shell and Secure Sockets Layer Commands

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

For detailed information about SSH and SSL concepts, configuration tasks, and examples, see the *Implementing Secure Shell* chapter in the Software configuration module in the *System Security Configuration Guide for Cisco ASR 9000 Series RoutersSystem Security Configuration Guide for Cisco 8000 Series Routers.*

- clear netconf-yang agent session, on page 3
- clear ssh, on page 4
- disable auth-methods, on page 6
- netconf-yang agent ssh , on page 7
- sftp, on page 8
- sftp (Interactive Mode), on page 11
- show netconf-yang clients, on page 14
- show netconf-yang statistics, on page 15
- show ssh, on page 17
- show ssh history, on page 20
- show ssh history details, on page 22
- show ssh rekey, on page 24
- show ssh session details, on page 25
- show ssl, on page 27
- show tech-support ssh, on page 29
- ssh-server authorized-keys, on page 31
- ssh algorithms cipher, on page 34
- ssh client auth-method, on page 35
- ssh client enable cipher , on page 37
- ssh client knownhost, on page 39
- ssh client source-interface, on page 40
- ssh client vrf, on page 42
- ssh server disable hmac, on page 43
- ssh, on page 44
- ssh server, on page 46
- ssh server algorithms host-key, on page 48

- ssh server certificate, on page 50
- ssh server enable cipher, on page 51
- ssh server logging, on page 52
- ssh server max-auth-limit, on page 53
- ssh server port-forwarding local, on page 54
- ssh server netconf , on page 55
- ssh server netconf disable ssh-port, on page 56
- ssh server netconf port, on page 57
- ssh server packet-flow-netio ingress, on page 58
- ssh server rate-limit, on page 59
- ssh server rekey-time, on page 60
- ssh server rekey-volume, on page 61
- ssh server session-limit, on page 62
- ssh server set-dscp-connection-phase, on page 63
- ssh server timeout, on page 64
- ssh server trustpoint, on page 65
- ssh server v2, on page 66
- ssh timeout, on page 67
- show shell history location, on page 68

clear netconf-yang agent session

To clear the specified netconf agent session, use the **clear netconf-yang agent session** in EXEC mode.

clear netconf-yang agent session *session-id*

Syntax Description

session-id	The session-id which needs to be cleared.

Command Default

None

Command Modes

EXEC modeXR EXEC mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

The **show netconf-yang clients** command can be used to get the required session-id(s).

Task ID

Task ID	Operation
config-services	read, write

Example

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

```
RP/0/RP0RSP0/CPU0:router (config) #  clear netconf-yang agent session 32125
```

clear ssh

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

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

Syntax Description

session-id	Session ID number of an incoming connection as displayed in the **show ssh** command output. Range is from 1 to 4294967295.
outgoing *session-id*	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

EXEC modeXR EXEC mode

Command History

Release	Modification
Release 7.0.12	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/RP0RSP0/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    0/33/1     SESSION_OPEN         cisco       123.100.100.18       v2
password     Command-Line-Interface

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/RP0RSP0/CPU0:router# clear ssh 0


RP/0/RP0RSP0/CPU0:router# show ssh

SSH version : Cisco-2.0

id  chan pty     location        state           userid    host                 ver
authentication connection type
-------------------------------------------------------------------------------------------------
Incoming sessions

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

disable auth-methods

To selectively disable the authentication methods for the SSH server, use the **disable auth-methods** command in ssh server configuration mode. To remove the configuration, use the **no** form of this command.

disable auth-methods { **keyboard-interactive** | **password** | **public-key** }

Syntax Description

keyboard-interactive	Disables keyboard-interactive authentication method for the SSH server
password	Disables password authentication method for the SSH server
public-key	Disables publick-key authentication method for the SSH server

Command Default

Allows all the authentication methods, by default.

Command Modes

ssh server

Command History

Release	Modification
Release 7.8.1	This command was introduced.

Usage Guidelines

If this configuration is not present, you can consider that the SSH server on the router allows all the authentication methods.

The public-key authentication method includes certificate-based authentication as well.

Task ID

Task ID	Operation
crypto	read, write

This example shows how to disable the public-key authentication method for the SSH server on the router.

```
Router#configure
Router(config)# ssh server
Router(config-ssh)# disable auth-methods public-key
Router(config-ssh)# commit
```

netconf-yang agent ssh

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

netconf-yang agent ssh
no netconf-yang agent ssh

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

Global Configuration

Command History

Release	Modification
Release 7.0.12	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/RP0RSP0/CPU0:router (config) #  netconf-yang agent ssh
```

sftp

To start the secure FTP (SFTP) client, use the **sftp** command.

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

Syntax Description

username	(Optional) Name of the user performing the file transfer. The at symbol (@) following the username is required.
hostname:remote-filename	(Optional) Name of the Secure Shell File Transfer Protocol (SFTP) server. The colon (:) following the hostname is required.
source-filename	SFTP source, including the path.
dest-filename	SFTP destination, including the path.
source-interface	(Optional) Specifies the source IP address of a selected interface for all outgoing SSH connections.
type	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 in EXEC modeXR 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 *vrf-name*	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

EXEC modeXR EXEC mode

Command History

Release	Modification
Release 7.0.12	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 modeXR 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.

From Cisco IOS XR Software Release 7.10.1 and later, you can use public-key based user authentication for Cisco IOS XR routers configured as SSH clients as well. This feature thereby allows you to use password-less authentication for secure file transfer and copy operations using SFTP and SCP protocols.

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/RP0RSP0/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/RP0RSP0/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/RP0RSP0/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/RP0RSP0/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/RP0RSP0/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/RP0RSP0/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/RP0RSP0/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/RP0RSP0/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/RP0RSP0/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/RP0RSP0/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)
```

sftp (Interactive Mode)

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

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

Syntax Description

username	(Optional) Name of the user performing the file transfer. The at symbol (@) following the username is required.
hostname:remote-filename	(Optional) Name of the Secure Shell File Transfer Protocol (SFTP) server. The colon (:) following the hostname is required.
source-interface	(Optional) Specifies the source IP address of a selected interface for all outgoing SSH connections.
type	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 in EXEC modeXR 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 *vrf-name*	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

EXEC modeXR EXEC mode

Command History

Release	Modification
Release 7.0.12	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

From Cisco IOS XR Software Release 7.10.1 and later, you can use public-key based user authentication for Cisco IOS XR routers configured as SSH clients as well. This feature thereby allows you to use password-less authentication for secure file transfer and copy operations using SFTP and SCP protocols.

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/RP0RSP0/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:/frmRouterdownoad

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

/disk0:/frmRouterdownoad
  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/RP0RSP0/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:/frmRouterdownoad

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

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

show netconf-yang clients

To display the client details for netconf-yang, use the **show netconf-yang clients** command in EXEC mode.

show netconf-yang clients

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
config-services	read

Example

This example shows how to use the **show netconf-yang clients** command:

```
RP/0/RP0RSP0/CPU0:router (config) #  sh netconf-yang clients
Netconf clients
client session ID|   NC version|    client connect time|        last OP time|        last
OP type|    <lock>|
 22969|                    1.1|          0d  0h  0m  2s|            11:11:24|
close-session|        No|
 15389|                    1.1|          0d  0h  0m  1s|            11:11:25|
get-config|        No|
```

Table 1: Field descriptions

Field name	Description
Client session ID	Assigned session identifier
NC version	Version of the Netconf client as advertised in the hello message
Client connection time	Time elapsed since the client was connected
Last OP time	Last operation time
Last OP type	Last operation type
Lock (yes or no)	To check if the session holds a lock on the configuration datastore

show netconf-yang statistics

To display the statistical details for netconf-yang, use the **show netconf-yang statistics** command in EXEC mode.

show netconf-yang statistics

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
config-services	read

Example

This example shows how to use the **show netconf-yang statistics** command:

```
RP/0/RP0RSP0/CPU0:router (config) #  sh netconf-yang statistics
Summary statistics
                        # requests|             total time|   min time per request|   max
 time per request|   avg time per request|
other                              0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
close-session                      4|       0h  0m  0s   3ms|       0h  0m  0s   0ms|
 0h  0m  0s   1ms|       0h  0m  0s   0ms|
kill-session                       0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
get-schema                         0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
get                                0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
get-config                         1|       0h  0m  0s   1ms|       0h  0m  0s   1ms|
 0h  0m  0s   1ms|       0h  0m  0s   1ms|
edit-config                        3|       0h  0m  0s   2ms|       0h  0m  0s   0ms|
 0h  0m  0s   1ms|       0h  0m  0s   0ms|
commit                             0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
cancel-commit                      0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
lock                               0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
unlock                             0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
discard-changes                    0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
```

```
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
validate                          0|       0h  0m  0s   0ms|       0h  0m  0s   0ms|
 0h  0m  0s   0ms|       0h  0m  0s   0ms|
xml parse                         8|       0h  0m  0s   4ms|       0h  0m  0s   0ms|
 0h  0m  0s   1ms|       0h  0m  0s   0ms|
netconf processor                 8|       0h  0m  0s   6ms|       0h  0m  0s   0ms|
 0h  0m  0s   1ms|       0h  0m  0s   0ms|
```

Table 2: Field descriptions

Field name	Description
Requests	Total number of processed requests of a given type
Total time	Total processing time of all requests of a given type
Min time per request	Minimum processing time for a request of a given type
Max time per request	Maximum processing time for a request of a given type
Avg time per request	Average processing time for a request type

show ssh

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

show ssh

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

EXEC modeXR EXEC mode

Command History

Release	Modification
Release 7.0.12	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

The following output is applicable for the **show ssh** command starting release 6.0 and later.

```
RP/0/RP0RSP0/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    0/33/1     SESSION_OPEN         cisco       123.100.100.18       v2
password     Command-Line-Interface

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

This table describes significant fields shown in the display.

Table 3: show ssh Field Descriptions

Field	Description
session	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#
```

show ssh history

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

show ssh history

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

EXEC modeXR EXEC mode

Command History

Release	Modification
Release 7.0.12	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 teminated:

```
RP/0/RP0RSP0/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
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 – VTY 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 EXEC modeXR EXEC mode.

show ssh history details

Syntax Description This command has no keywords or arguments.

Command Default None

Command Modes EXEC modeXR EXEC mode

Command History

Release	Modification
Release 7.0.12	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 teminated along with the start and end time of the sessions:

```
RP/0/RP0RSP0/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 4: 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.

show ssh rekey

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	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 5: 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.

show ssh session details

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

EXEC modeXR EXEC mode

Command History

Release	Modification
Release 7.0.12	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/RP0RSP0/CPU0:router# show ssh session details

SSH version: Cisco-2.0
session     key-exchange   pubkey   incipher   outcipher    inmac     outmac
--------------------------------------------------------------------------------
Incoming Session

0           diffie-hellman ssh-dss  3des-cbc   3des-cbc     hmac-md5  hmac-md5

Outgoing connection

1           diffie-hellman ssh-dss  3des-cbc   3des-cbc     hmac-md5  hmac-md5
```

This table describes the significant fields shown in the display.

Table 6: 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.

Field	Description
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.

show ssl

To display active Secure Socket Layer (SSL) sessions, use the **show ssl** command.

show ssl [*process-id*]

Syntax Description

process-id	(Optional) Process ID (PID) of the SSL application. The range is from 1 to 1000000000.

Command Default

None

Command Modes

EXEC

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

To display a specific process, enter the process ID number. To get a specific process ID number, enter **run pidin** from the command line or from a shell.

The absence of any argument produces a display that shows all processes that are running SSL.

Task ID

Task ID	Operations
crypto	read

Examples

The following sample output is from the **show ssl** command:

```
RP/0/RP0RSP0/CPU0:router# show ssl

PID         Method      Type        Peer                Port    Cipher-Suite
=============================================================================

1261711     sslv3       Server      172.16.0.5          1296    DES-CBC3-SHA
```

This table describes the fields shown in the display.

Table 7: show ssl Field Descriptions

Field	Description
PID	Process ID of the SSL application.
Method	Protocol version (sslv2, sslv3, sslv23, or tlsv1).
Type	SSL client or server.
Peer	IP address of the SSL peer.
Port	Port number on which the SSL traffic is sent.

Field	Description
Cipher-Suite	Exact cipher suite chosen for the SSL traffic. The first portion indicates the encryption, the second portion the hash or integrity method. In the sample display, the encryption is Triple DES and the Integrity (message digest algorithm) is SHA.

Related Commands

Command	Description
run pidin	Displays the process ID for all processes that are running.

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 EXEC modeXR EXEC mode.

show tech-support ssh

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

EXEC modeXR EXEC mode

Command History

Release	Modification
Release 7.0.12	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/RP0RSP0/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	
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.

Command	Description
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-server authorized-keys

To support user based public key authentication, use the **ssh-server authorized-keys** command in Global Executive mode.

ssh-server authorized-keys { **sftp** | **user** | **zeroize** }

Syntax Description

sftp	Indicates the SFTP protocol to import authorized keys.
user	Specifies admin or specific user.
zeroize	Removes the authorized keys.

Command Default

Enabled, by default

Command Modes

Global Executive mode

Command History

Release	Modification
Release 26.2.1	A new keyword, **sftp**, was added to the command.
Release 25.2.1	The command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
crypto	read, write

This example displays steps to generate public key authentication:

```
Router# ssh-server authorized keys
Router# user
Router# test
Router# commit
Enter the public key
Public key
```

This example displays steps to remove authorized public key:

```
Router# ssh-server authorized keys
Router# zeroize
Router# commit
All the public keys for the user cisco will be removed
Do you really want to remove all the public keys for the user?? [yes/no]: yes
```

This example displays steps to show authorized public key:

```
Router# show ssh server authorized-keys user
Mon Jan 12 12:31:07.908 UTC
----------------------------------------
sk-ssh-ed25519@xxxx.com
AAAAGnNrLXNzaC1lZDI1NTE5QG9wZW5zc2guY29tAAAAILycNWbmfereElmEbl+XCVtnkLpWxeWgM2dAzuRF0cRkXXAABHNzaDo=
 ssh:
----------------------------------------
```

These examples show how to use `sftp` to import authorized keys.

```
Router# ssh-server authorized-keys sftp root@10.105.227.128:/root/keys/ca.pub
This is a Cisco managed device to be used only for authorized purposes.
Your use is monitored for security, asset protection, and policy compliance.

Password:
RP/0/RP0/CPU0:May 25 09:00:04.361 UTC: ssh_syslog_proxy[1216]:
%SECURITY-SSHD_SYSLOG_PRX-6-INFO_GENERAL : ssh_csm[44791]: SSH Client successfully established
 session as root to 10.105.227.128
Connected to root@10.105.227.128.
Fetching /root/keys/ca.pub to /tmp/sftplib_download_cafyauto
ca.pub                                        0%    0     0.0KB/s   --:-- ET
ca.pub
                                            100%  411     0.4KB/s   00:00
```

Example with the **port** option:

```
Router# ssh-server authorized-keys sftp root@10.105.227.128:/root/keys/ed25519.pub port
2222 user test
This is a Cisco managed device to be used only for authorized purposes.
Your use is monitored for security, asset protection, and policy compliance.

Password:
Connected to root@10.105.227.128.
Fetching /root/keys/ed25519.pub to /tmp/sftplib_download_test
ed25519.pub                   100%  111     0.1KB/s   00:00
                                  100%  111     0.1KB/s   00:00
```

Example with the **source-interface** option:

```
Router# ssh-server authorized-keys user user100 sftp
root@10.105.227.128:/root/keys/venthota1.pub source-interface mgmtEth 0/RP0/CPU0/0

RP/0/RP0/CPU0:Jun  2 17:31:25.228 UTC: ssh_syslog_proxy[1220]:
%SECURITY-SSHD_SYSLOG_PRX-6-INFO_GENERAL : ssh_csm[509]
This is a Cisco managed device to be used only for authorized purposes.
Your use is monitored for security, asset protection, and policy compliance.

Password:
RP/0/RP0/CPU0:Jun  2 17:31:29.274 UTC: ssh_syslog_proxy[1220]:
%SECURITY-SSHD_SYSLOG_PRX-6-INFO_GENERAL : ssh_csm[509]: SSH Client successfully established
 session as root to 10.105.227.128
Connected to root@10.105.227.128.
Fetching /root/keys/venthota1.pub to /tmp/sftplib_download_user100

venthota1.pub                 100%  787     0.8KB/s   00:00
RP/0/RP0/CPU0:Jun  2 17:31:29.431 UTC: ssh_syslog_proxy[1220]:
%SECURITY-SSHD_SYSLOG_PRX-6-INFO_GENERAL : ssh_csm[509]: SSH Client terminated session as
root to 10.105.227.128
```

Example with the **vrf** option:

```
Router# ssh-server authorized-keys sftp root@10.105.227.128:/root/keys/venthota1.pub vrf
vrf1

RP/0/RP0/CPU0:Jun  2 17:35:52.732 UTC: ssh_syslog_proxy[1220]:
%SECURITY-SSHD_SYSLOG_PRX-6-INFO_GENERAL : ssh_csm[4529]
```

```
This is a Cisco managed device to be used only for authorized purposes.
Your use is monitored for security, asset protection, and policy compliance.

Password:

RP/0/RP0/CPU0:Jun  2 17:35:58.101 UTC: ssh_syslog_proxy[1220]:
%SECURITY-SSHD_SYSLOG_PRX-6-INFO_GENERAL : ssh_csm[4529]: SSH Client successfully established
 session as root to 10.105.227.128
Connected to root@10.105.227.128.
Fetching /root/keys/venthota1.pub to /tmp/sftplib_download_cafyauto

venthota1.pub               100%  787     0.8KB/s   00:00
RP/0/RP0/CPU0:Jun  2 17:35:58.257 UTC: ssh_syslog_proxy[1220]:
%SECURITY-SSHD_SYSLOG_PRX-6-INFO_GENERAL : ssh_csm[4529]: SSH Client terminated session as
 root to 10.105.227.128
```

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 Global Configuration modeXR 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

Global Configuration modeXR Config mode

Command History

Release	Modification
Release 7.0.1	This command was introduced.

Usage Guidelines

Starting Cisco IOS XR Release 26.2.1, the **3DES-CBC** cipher and **Diffie-Hellman Group 1 SHA1** key exchange are insecure and deprecated.

Task ID

Task ID	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 37	Enables CBC mode ciphers on the SSH client.
ssh server enable cipher, on page 51	Enables CBC mode ciphers on the SSH server.

ssh client auth-method

To set the preferred order of SSH client authentication methods to be negotiated with the SSH server while establishing SSH sessions, use the **ssh client auth-method** command in the Global Configuration modeXR Config mode. To revert to the default order of SSH client authentication methods, use the **no** form of this command.

```
ssh client auth-method list-of-auth-method
```

Syntax Description

list-of-auth-method	Specifies the list of SSH client authentication methods in the respective order. The available options are: • **keyboard-interactive** • **password** • **public-key**

Command Default

None

Command Modes

Global ConfigurationXR Config

Command History

Release	Modification
Release 7.9.2/Release 7.10.1	This command was introduced.

Usage Guidelines

The default order of SSH client authentication methods on Cisco IOS XR routers is as follows:

- On routers running Cisco IOS XR SSH:
  - **public-key**, **password** and **keyboard-interactive**
- On routers running CiscoSSH (open source-based SSH):
  - **public-key**, **keyboard-interactive** and **password**

Task ID

Task ID	Operation
crypto	read, write

This example shows how to set the order of SSH client authentication methods in such a way that public key authentication is negotiated first, followed by keyboard-interactive, and then password-based authentication.

```
Router#configure
```

```
Router(config)#ssh client auth-method public-key keyboard-interactive password
Router(config-ssh)#commit
```

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 Global Configuration modeXR 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

Global Configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

The support for CBC ciphers are disabled by default. 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.

Starting Cisco IOS XR Release 26.2.1, the **3DES-CBC** cipher and **Diffie-Hellman Group 1 SHA1** key exchange are insecure and deprecated.

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 algorithms cipher, on page 34	Configures the list of supported SSH algorithms on the client or on the server.
ssh server enable cipher, on page 51	Enables CBC mode ciphers on the SSH server.

ssh client knownhost

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

ssh client knownhost device : / filename
no 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

Global Configuration modeXR Config mode

Command History

Release	Modification
Release 7.0.12	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/RP0RSP0/CPU0:router# configure
RP/0/RP0RSP0/CPU0:router(config)# ssh client knownhost disk0:/ssh.knownhost
RP/0/RP0RSP0/CPU0:router(config)# commit
RP/0/RP0RSP0/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/RP0RSP0/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. To disable use of the specified interface IP address, use the **no** form of this command.

ssh client source-interface *type interface-path-id*
no ssh client source-interface *type interface-path-id*

Syntax Description

type	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

Global Configuration modeXR Config mode

Command History

Release	Modification
Release 7.0.12	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/RP0RSP0/CPU0:router# configure
RP/0/RP0RSP0/CPU0:router(config)# ssh client source-interface MgmtEth 0/RP0/CPU0/0
```

ssh client vrf

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

ssh client vrf *vrf-name*
no ssh client vrf *vrf-name*

Syntax Description

vrf-name	Specifies the name of the VRF to be used by the SSH client.

Command Default

None

Command Modes

Global Configuration modeXR Config mode

Command History

Release	Modification
Release 7.0.12	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 39 or ssh client source-interface, on page 40.

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/RP0RSP0/CPU0:router# configure
RP/0/RP0RSP0/CPU0:router(config)# ssh client vrf green
```

ssh server 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 Global Configuration modeXR 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

Global Configuration modeXR Config mode

Command History

Release	Modification
Release 7.0.12	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

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

ssh [**vrf** *vrf-name*] {**ipv4-address** | **ipv6-address** | **hostname**} [**username** *user-id*] [**cipher aes** {**128-cbc** | **192-cbc** | **256-cbc**}][**source-interface** *type interface-path-id*][**command***command-name*]

Syntax Description

vrf *vrf-name*	Specifies the name of the VRF associated with this connection.
ipv4-address	IPv4 address in A:B:C:D format.
ipv6-address	IPv6 address in X:X::X format.
hostname	Hostname of the remote node. If the hostname has both IPv4 and IPv6 addresses, the IPv6 address is used.
username*user-id*	(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.
cipheraes	(Optional) Specifies Advanced Encryption Standard (AES) as the cipher for the SSH client connection. **Note** If there is no specification of a particular cipher by the administrator, the client proposes 3DES as the default to ensure compatibility.
128-CBC	128-bit keys in CBC mode.
192-CBC	192-bit keys in CBC mode.
256-CBC	256-bit keys in CBC mode.
source interface	(Optional) Specifies the source IP address of a selected interface for all outgoing SSH connections.
type	Interface type. For more information, use the question mark (**?**)online help function.
interface-path-id	Physical interface or virtual interface. **Note** Use the**showinterfaces** command in EXEC modeXR 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

3DES cipher

Command Modes

EXEC modeXR EXEC mode

Command History

Release	Modification
Release 7.0.12	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 a VRF is specified in the **ssh** command, the **ssh** interface takes precedence over the interface specified in the ssh client source-interface, on page 40 command.

When you configure the **cipher aes** keyword, an SSH client makes a proposal, including one or more of the key sizes you specified, as part of its request to the SSH server. The SSH server chooses the best possible cipher, based both on which ciphers that server supports and on the client proposal.

Note AES encryption algorithm is not supported on the SSHv1 server and client. Any requests for an AES cipher sent by an SSHv2 client to an SSHv1 server are ignored, with the server using 3DES instead.

A VRF is required to run SSH, although this may be either the default VRF or a VRF specified by the user. If no VRF is specified while configuring the ssh client source-interface, on page 40 or ssh client knownhost, on page 39 commands, the default VRF is assumed.

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/RP0RSP0/CPU0:router# ssh vrf green  username userabc

Password:
Remote-host>
```

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. To stop the SSH server from receiving any further connections for the specified VRF, use the **no** form of this command.

ssh server [**vrf vrf-name** | **v2**]
no ssh server [**vrf vrf-name** | **v2**]

Syntax Description

vrf *vrf-name*	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.
v2	Forces the SSH server version to be only 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

Global Configuration modeXR Config mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.
Release 25.4.1	The sshv1 command is deprecated.

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 66 command.

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

Starting with Cisco IOS XR Release 25.3.1, SSH version 1 (SSHv1) is deprecated and cannot be configured. You must use SSH version 2 (SSHv2) for secure device access.

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/RP0RSP0/CPU0:router# configure
RP/0/RP0RSP0/CPU0:router(config)# ssh server vrf green
```

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 Global Configuration modeXR 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** | **ed25519** | **rsa** | **x509v3-ssh-rsa** }

Syntax Description

• **dsa** • **ecdsa-nistp256** • **ecdsa-nistp384** • **ecdsa-nistp521** • **ed25519** • **rsa** • **x509v3-ssh-rsa**	Selects the specified host keys to be offered to the SSH client. While configuring this, you can specify the algorithms in any order.

Command Default

None

Command Modes

Global Configuration modeXR Config mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.
Release 7.3.1	The support for **ed25519** and **x509v3-ssh-rsa** algorithms was introduced.

Usage Guidelines

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.

You can also use the **crypto key zeroize** command to remove the SSH algorithms that are not required.

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.

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#ssh server algorithms host-key ecdsa-nistp521
```

Similarly, this example shows how to select the **ed25519** algorithm:

```
Router(config)#ssh server algorithms host-key ed25519
```

Similarly, this example shows how to select the **x509v3-ssh-rsa** algorithm:

```
Router(config)#ssh server algorithms host-key x509v3-ssh-rsa
```

ssh server certificate

To configure the certificate-related parameters of SSH server, use the **ssh server certificate** command in Global Configuration modeXR Config mode. To remove the configuration, use the **no** form of this command.

ssh server certificate username { **common-name** | **user-principle-name** }

Syntax Description

username	Specifies which field in the certificate to be used as the username.
common-name	Configures the user common name (CN) from the subject name field.
user-principle-name	Configures the user principle name (UPN) from subject alternate name.

Command Default

In the absence of this configuration, the SSH server considers common name (CN) as the username.

Command Modes

Global Configuration modeXR Config mode

Command History

Release	Modification
Release 7.3.1	This command was introduced.

Usage Guidelines

The user name must match the user name provided in the CLI.

Task ID

Task ID	Operation
crypto	read, write

This example shows how to specify which field in the certificate is to be used as the username. Here, it specifies the user common name to be picked up from the subject name field.

```
Router#configure
Router(config)#ssh server certificate username common-name
Router(config)#commit
```

Here, it specifies the user principle name to be picked up from the subject alternate name field.

```
Router#configure
Router(config)#ssh server certificate username user-principle-name
Router(config)#commit
```

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 Global Configuration modeXR 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

Global Configuration

Command History

Release	Modification
Release 7.0.12	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.

Starting Cisco IOS XR Release 26.2.1, the **3DES-CBC** cipher and **Diffie-Hellman Group 1 SHA1** key exchange are insecure and deprecated.

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 server connection:

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

Related Commands

Command	Description
ssh algorithms cipher, on page 34	Configures the list of supported SSH algorithms on the client or on the server.
ssh client enable cipher , on page 37	Enables CBC mode ciphers on the SSH client.

ssh server logging

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

ssh server logging
no ssh server logging

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

Global Configuration modeXR Config mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

Only SSHv2 client connections are allowed.

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/RP0RSP0/CPU0:router# configure
RP/0/RP0RSP0/CPU0:router(config)# ssh server logging
```

ssh server max-auth-limit

To configure the maximum number of authentication attempts allowed for SSH connection, use the **ssh server max-auth-limit** command in Global Configuration modeXR Config mode. To remove the configuration, use the **no** form of this command.

ssh server max-auth-limit *limit*

Syntax Description

limit	Specifies the maximum authentication attempts allowed for SSH connection. The limit ranges from 3 to 20; default being 20 (prior to Cisco IOS XR Software Release 7.3.2, the limit range was from 4 to 20).

Command Default

The default authentication limit is 20.

Command Modes

Global Configuration modeXR Config mode

Command History

Release	Modification
Release 7.3.2	The command was modified to change the minimum value of limit range from 4 to 3.
Release 7.3.1	This command was introduced

Usage Guidelines

The SSH server limits the number of authentication attempts using the password authentication method to a maximum of 3 due to security reasons. You cannot change this particular limit of 3 by configuring the maximum authentication attempts limit for SSH.

For example, even if you configure the maximum authentication attempts limit as 5, the number of authentication attempts allowed using the password authentication method still remain as 3.

Task ID

Task ID	Operations
crypto	read, write

Examples

This example shows how to configure the maximum number of authentication attempts allowed for SSH connection:

```
Router# configure
Router(config)# ssh server max-auth-limit 5
Router(config)# commit
```

ssh server port-forwarding local

To enable SSH port forwarding feature on SSH server, use the **ssh server port-forwarding local** command in Global Configuration modeXR Config mode. To disable the feature, use the **no** form of this command.

ssh server port-forwarding local

Syntax Description

This command has no keywords or arguments.

Command Default

Disabled, by default.

Command Modes

Global Configuration modeXR Config mode

Command History

Release	Modification
Release 7.3.2	This command was introduced with CiscoSSH, an OpenSSH-based implementation of SSH.
Release 7.3.15	This command was introduced with Cisco IOS XR SSH.

Usage Guidelines

The Cisco IOS XR software supports SSH port forwarding only on SSH server; not on SSH client. Hence, to utilize this feature, the SSH client running at the end host must already have the support for SSH port forwarding or tunneling.

Task ID

Task ID	Operations
crypto	read, write

Examples

This example shows how to enable SSH port forwarding feature on SSH server:

```
Router#configure
Router(config)#ssh server port-forwarding local
Router(config)#commit
```

Related Commands

Command	Description
show ssh, on page 17	Displays all incoming and outgoing SSH connections on the router.

ssh server netconf

To configure a port for the netconf SSH server, use the **ssh server netconf port** in the Global Configuration modeXR Config mode. To disable netconf for the configured port, use the **no** form of the command.

ssh server netconf [**port** *port-number*]
no ssh server netconf [**port** *port-number*]

Syntax Description

port-number	(Optional) Port number for the netconf SSH server (default port number is 830).

Command Default

Default port number is 830.

Command Modes

Global Configuration modeXR Config mode

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Task ID

Task ID	Operation
crypto	read, write

Example

This example shows how to use the **ssh server netconf port** command:

```
RP/0/RP0RSP0/CPU0:router (config) #  ssh server netconf port 830
```

ssh server netconf disable ssh-port

To disable a SSH port for the netconf SSH server, use the **ssh server netconf disable ssh-port** command in the global configuration mode.

ssh server netconf disable ssh-port

Syntax Description

ssh-port	SSH-port (Netconf will not work on SSH port)

Command Default

None

Command Modes

Global configuration

Command History

Release	Modification
Release 25.2.1	This command was introduced.

Usage Guidelines

You must configure the **ssh server netconf** command for at least one VRF, in order to disable a netconf port to enable netconf subsystem support.

Task ID

Task ID	Operations
crypto	read, write

Examples

This example shows how to disable a SSH port for the netconf SSH server:

```
Router#config
Router#ssh server netconf disable ssh-port
Router#commit
```

ssh server netconf port

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

ssh server netconf port *port number*
no ssh server netconf port*port number*

Syntax Description

port *port-number*	Port number for the netconf SSH server (default port number is 830).

Command Default

The default port number is 830.

Command Modes

Global configuration

Command History

Release	Modification
Release 7.0.12	This command was introduced.

Usage Guidelines

You must configure the **ssh server netconf** command for at least one VRF, in order to configure a netconf port to enable netconf subsystem support.

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/RP0RSP0/CPU0:router# configure
RP/0/RP0RSP0/CPU0:router(config)# ssh server netconf port 831
```

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 *System Management Command Reference for Cisco ASR 9000 Series RoutersSystem Management Command Reference for Cisco 8000 Series Routers*.

ssh server packet-flow-netio ingress

To allow filtering of the ingress SSH and Netconf traffic while still having the ingress ACL configured on the management interface, use the **ssh server packet-flow-netio ingress** command in the Global Configuration modeXR Config mode. To remove the configuration, use the **no** form of the command.

ssh server packet-flow-netio ingress

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

Global Configuration

Command History

Release	Modification
Release 25.1.1	This command was introduced.

Usage Guidelines

For software versions prior to Release 25.1.1, we recommend to configure the ingress ACL under the **ssh server** configuration mode instead of configuring it under the management interface, to filter out the ingress SSH and Netconf traffic.

For SSH:

ssh server vrf *vrf-name* **ipv4 access-list** *ipv4-access-list-name* **ipv6 access-list** *ipv6-access-list-name*

For Netconf:

ssh server netconf vrf *vrf-name* **ipv4 access-list** *ipv4-access-list-name* **ipv6 access-list** *ipv6-access-list-name*

Task ID

Task ID	Operation
config-services	read, write

Example

This example shows how to allow filtering of the ingress SSH and Netconf traffic while still having the ingress ACL configured on the management interface:

```
Router(config)#ssh server packet-flow-netio ingress
Router(config)#commit
```

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. To return to the default value, use the **no** form of this command.

ssh server rate-limit *rate-limit*
no ssh server 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

Global Configuration modeXR Config mode

Command History

Release	Modification
Release 7.0.12	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/RP0RSP0/CPU0:router# configure
RP/0/RP0RSP0/CPU0:router(config)# ssh server rate-limit 20
```

ssh server rekey-time

To configure rekey of the ssh server key based on time. Use the **no** form of this command to remove the rekey interval.

ssh server rekey-time *time in minutes*
no ssh server rekey-time

Syntax Description

rekey-time *time in minutes*	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 30 minutes.

Command Default

None.

Command Modes

Global configuration

Command History

Release	Modification
Release 6.2.1	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/RP0RSP0/CPU0:router# configure
RP/0/RP0RSP0/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 **no** form of this command to remove the volume-based rekey threshold.

ssh server rekey-volume *data in megabytes*
no ssh server rekey-volume

Syntax Description

rekey-volume *data in megabytes*	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

Global configuration

Command History

Release	Modification
Release 7.0.12	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/RP0RSP0/CPU0:router# configure
RP/0/RP0RSP0/CPU0:router(config)# ssh server rekey-volume 2048
```

ssh server session-limit

To configure the number of allowable concurrent incoming Secure Shell (SSH) sessions, use the **ssh server session-limit** command. 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 110.

Note
Although CLI output option has 110, 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

Global Configuration modeXR Config mode

Command History

Release	Modification
Release 7.0.12	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/RP0RSP0/CPU0:router# configure
RP/0/RP0RSP0/CPU0:router(config)# ssh server session-limit 50
```

ssh server set-dscp-connection-phase

To set the DSCP marking from TCP connection phase itself for SSH packets originating from Cisco IOS XR routers that function as SSH servers, use the **ssh server set-dscp-connection-phase** command in Global Configuration modeXR Config mode. To remove the configuration and to continue marking the SSH packets from the authentication phase, use the **no** form of this command.

ssh server set-dscp-connection-phase

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

Global Configuration modeXR Config mode

Command History

Release	Modification
Release 24.1.1	This command was introduced.

Usage Guidelines

- By default, the DSCP marking for the SSH packets originating from Cisco IOS XR routers with CiscoSSH that function as SSH servers is done from the authentication phase. Whereas, for routers with Cisco IOS XR SSH, the DSCP marking for the SSH packets is done from TCP connection phase itself.
- Although the **ssh server set-dscp-connection-phase** command is available on routers with CiscoSSH and routers with Cisco IOS XR SSH, this configuration is relevant only on routers with CiscoSSH due to the above mentioned reason.

Task ID

Task ID	Operations
crypto	read, write

Examples

This example shows how to set the DSCP marking from TCP connection phase itself for SSH server packets originating from Cisco IOS XR routers with CiscoSSH:

```
Router#configure
Router(config)#ssh server set-dscp-connection-phase
Router(config-ssh)#commit
```

ssh server timeout

To configure timeout for unused SSH connections and SSH channels on your routers that function as SSH servers, use the **ssh server timeout** command in Global Configuration modeXR Config mode. To remove the timeout configuration, use the **no** form of this command.

ssh server timeout { **channel** | **connection** } *timeout-period*

Syntax Description

timeout-period	Timeout period, in seconds, for unused SSH connections and idle SSH channels. The range is 0 to 86400.

Command Default

Disabled

Command Modes

Global Configuration modeXR Config mode

Command History

Release	Modification
Release 25.3.1	This command was introduced.

Usage Guidelines

The unused connection timeout and channel timeout are applicable only for connections that are newly created after the timeout period is configured.

The channel timeout begins counting down after the set period of inactivity. The unused connection timeout begins counting down when there is no active channel within the SSH connection.

Task ID

Task ID	Operations
crypto	read, write

Examples

This example shows how to configure a timeout of 600 seconds for an unused SSH connection:

```
Router#configure
Router(config)# ssh server timeout connection 600
Router(config-ssh)#commit
```

This example shows how to configure a timeout of 300 seconds for an idle SSH channel:

```
Router#configure
Router(config)#ssh server timeout channel 300
Router(config-ssh)#commit
```

ssh server trustpoint

To configure the trustpoint for SSH certificates, use the **ssh server trustpoint** command in Global Configuration modeXR Config mode. To disable this feature, use the **no** form of this command.

ssh server trustpoint { **host** | **user** } *trustpoint-name*

Syntax Description

host	Configures the trustpoint from where server takes its certificate.
user	Configures the trustpoints used for user certificate validation.
trustpoint-name	Specifies the name of the trustpoint.

Command Default

None

Command Modes

Global Configuration modeXR Config mode

Command History

Release	Modification
Release 7.3.1	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 configure the trustpoint from where SSH server takes its certificate:

```
Router#configure
Router(config)#ssh server trustpoint host test-host-tp
Router(config)#commit
```

This example shows how to configure the trustpoint used for user certificate validation:

```
Router#configure
Router(config)#ssh server trustpoint user test-user-tp
Router(config)#commit
```

ssh server v2

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

ssh server v2
no ssh server v2

Syntax Description

This command has no keywords or arguments.

Command Default

None

Command Modes

Global Configuration modeXR Config mode

Command History

Release	Modification
Release 7.0.12	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/RP0RSP0/CPU0:router#configure
RP/0/RP0RSP0/CPU0:router(config)# ssh server v2
```

ssh timeout

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

ssh timeout *seconds*
no 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

Global Configuration modeXR Config mode

Command History

Release	Modification
Release 7.0.12	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/RP0RSP0/CPU0:router# configure
RP/0/RP0RSP0/CPU0:router(config)# ssh timeout 60
```

show shell history location

To display the **run** and **bash** shell command history for a specified node location, use the **show shell history location** command in EXEC mode. Enter the location in the `rack/slot/module` format.

show shell history location *location*

Command Default

None

Command Modes

EXEC Mode

Command History

Release	Modification
Release 26.3.1	This command was introduced.

Usage Guidelines

No specific guidelines impact the use of this command.

Examples

This output displays the **run** and **bash** commands, including timestamp and user.

```
Router#show shell history location 0/RP0/CPU0
Mon Jul 13 12:27:24.451 UTC

Time                          User       Origin         Command
----------------------------  ---------- ----------  --------------------------------
Aug  3 2026 17:41:23.541 UTC  user       run        /bin/sh -c -- "cat /proc/57392/cmdline"
Aug  3 2026 17:41:23.546 UTC  user       run         /usr/bin/cat /proc/57392/cmdline
Aug  3 2026 17:41:23.548 UTC  user       run         /bin/sh -c -- "ps -p 57392 -o ppid="
Aug  3 2026 17:41:23.551 UTC  user       run         /usr/bin/ps -p 57392 -o ppid=
Aug  3 2026 17:41:23.579 UTC  user       run        /bin/sh -c -- "cat /proc/57365/cmdline"
Aug  3 2026 17:41:23.583 UTC  user       run         /usr/bin/cat /proc/57365/cmdline
Aug  3 2026 17:41:23.585 UTC  user       run         /bin/sh -c -- "ps -p 57365 -o ppid="
Aug  3 2026 17:12:35.222 UTC  user       bash        /bin/hostname
Aug  3 2026 17:12:35.223 UTC  user       bash        /bin/less -V
Aug  3 2026 17:12:35.225 UTC  user       bash        /bin/tty
Aug  3 2026 17:12:40.733 UTC  user       bash        /bin/ls -lrt
Aug  3 2026 17:12:43.637 UTC  user       bash        /bin/uname -a
Aug  3 2026 17:12:46.309 UTC  user       bash        /bin/whoami
```