Reverse SSH Enhancements

Last Updated: January 23, 2012

The Reverse SSH Enhancements feature, which is supported for SSH Version 1 and 2, provides an alternative way to configure reverse Secure Shell (SSH) so that separate lines do not need to be configured for every terminal or auxiliary line on which SSH must be enabled. This feature also eliminates the rotary-group limitation.

Finding Feature Information

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

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

Prerequisites for Reverse SSH Enhancements

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

Americas Headquarters:
Cisco Systems, Inc., 170 West Tasman Drive, San Jose, CA 95134-1706 USA

Restrictions for Reverse SSH Enhancements

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

Information About Reverse SSH Enhancements

Reverse Telnet

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

Reverse SSH

Reverse telnet can be accomplished using SSH. Unlike reverse telnet, SSH provides for secure connections. The Reverse SSH Enhancements feature provides you with a simplified method of configuring SSH. Using this feature, you no longer have to configure a separate line for every terminal or auxiliary line on which you want to enable SSH. The previous method of configuring reverse SSH limited the number of ports that can be accessed to 100. The Reverse SSH Enhancements feature removes the port number limitation. For information on the alternative method of configuring reverse SSH, see How to Configure Reverse SSH Enhancements, page 2."

How to Configure Reverse SSH Enhancements

Configuring Reverse SSH for Console Access

To configure reverse SSH console access on the SSH server, perform the following steps.

SUMMARY STEPS

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

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**line** *line-number ending-line-number* **Example:** `Router# line 1 3`	Identifies a line for configuration and enters line configuration mode.
Step 4	**no exec** **Example:** `Router (config-line)# no exec`	Disables EXEC processing on a line.
Step 5	**login authentication** *listname* **Example:** `Router (config-line)# login authentication default`	Defines a login authentication mechanism for the lines. **Note** The authentication method must use a username and password.

	Command or Action	Purpose
Step 6	**transport input ssh** **Example:** `Router (config-line)# transport input ssh`	Defines which protocols to use to connect to a specific line of the router. • The **ssh** keyword must be used for the Reverse SSH Enhancements feature.
Step 7	**exit** **Example:** `Router (config-line)# exit`	Exits line configuration mode.
Step 8	**exit** **Example:** `Router (config)# exit`	Exits global configuration mode.
Step 9	**ssh -l** *userid* **:** {*number*} {*ip-address*} **Example:** `Router# ssh -l lab:1 router.example.com`	Specifies the user ID to use when logging in on the remote networking device that is running the SSH server. • *userid* --User ID. • **:** --Signifies that a port number and terminal IP address will follow the userid argument. • *number* --Terminal or auxiliary line number. • *ip-address* --Terminal server IP address. **Note** The *userid* argument and **:rotary**{*number*}{*ip-address*} delimiter and arguments are mandatory when configuring the alternative method of Reverse SSH for modem access.

Configuring Reverse SSH for Modem Access

To configure Reverse SSH for modem access, perform the steps shown in the “SUMMARY STEPS” section below.

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

SUMMARY STEPS

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

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**configure terminal** **Example:** `Router# configure terminal`	Enters global configuration mode.
Step 3	**line** *line-number ending-line-number* **Example:** `Router# line 1 200`	Identifies a line for configuration and enters line configuration mode.
Step 4	**no exec** **Example:** `Router (config-line)# no exec`	Disables EXEC processing on a line.
Step 5	**login authentication** *listname* **Example:** `Router (config-line)# login authentication default`	Defines a login authentication mechanism for the lines. **Note** The authentication method must use a username and password.

	Command or Action	Purpose
Step 6	**rotary** *group* **Example:** `Router (config-line)# rotary 1`	Defines a group of lines consisting of one or more virtual terminal lines or one auxiliary port line.
Step 7	**transport input ssh** **Example:** `Router (config-line)# transport input ssh`	Defines which protocols to use to connect to a specific line of the router. • The **ssh** keyword must be used for the Reverse SSH Enhancements feature.
Step 8	**exit** **Example:** `Router (config-line)# exit`	Exits line configuration mode.
Step 9	**exit** **Example:** `Router (config)# exit`	Exits global configuration mode.
Step 10	**ssh -l** *userid* **:rotary** {*number*} {*ip-address*} **Example:** `Router# ssh -l lab:rotary1 router.example.com`	Specifies the user ID to use when logging in on the remote networking device that is running the SSH server. • *userid* --User ID. • **:** --Signifies that a port number and terminal IP address will follow the *userid* argument. • *number* --Terminal or auxiliary line number. • *ip-address* --Terminal server IP address. **Note** The *userid* argument and **:rotary**{*number*}{*ip-address*} delimiter and arguments are mandatory when configuring the alternative method of Reverse SSH for modem access.

Troubleshooting Reverse SSH on the Client

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

SUMMARY STEPS

1. **enable**
2. **debug ip ssh client**

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**debug ip ssh client** **Example:** `Router# debug ip ssh client`	Displays debugging messages for the SSH client.

Troubleshooting Reverse SSH on the Server

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

SUMMARY STEPS

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

DETAILED STEPS

	Command or Action	Purpose
Step 1	**enable** **Example:** `Router> enable`	Enables privileged EXEC mode. • Enter your password if prompted.
Step 2	**debug ip ssh** **Example:** `Router# debug ip ssh`	Displays debugging messages for the SSH server.

	Command or Action	Purpose
Step 3	**show ssh** **Example:** `Router# show ssh`	Displays the status of the SSH server connections.
Step 4	**show line** **Example:** `Router# show line`	Displays parameters of a terminal line.

Configuration Examples for Reverse SSH Enhancements

Example Reverse SSH Console Access

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

Terminal Server Configuration

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

Client Configuration

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

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

Example Reverse SSH Modem Access

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

```
line 1 200
   no exec
   login authentication default
   rotary 1
```

```
transport input ssh
exit
```

The following command shows that reverse SSH will connect to the first free line in the rotary group:

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

Additional References

Related Documents

Related Topic	Document Title
Cisco IOS commands	Cisco IOS Master Commands List, All Releases
Configuring Secure Shell	See the following modules: • Configuring Secure Shell • Secure Shell Version 2 Support • SSH Terminal-Line Access
Security commands	*Cisco IOS Security Command Reference*

Standards

Standards	Title
No new or modified standards are supported by this feature.	--

MIBs

MIBs	MIBs Link
None	To locate and download MIBs for selected platforms, Cisco software releases, and feature sets, use Cisco MIB Locator found at the following URL: http://www.cisco.com/go/mibs

RFCs

RFCs	Title
None	--

Technical Assistance

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

Feature Information for Reverse SSH Enhancements

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

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

Table 1 Feature Information for Reverse SSH Enhancements

Feature Name	Releases	Feature Information
Reverse SSH Enhancements	12.3(11)T	The Reverse SSH Enhancements feature, which is supported for SSH Version 1 and 2, provides an alternative way to configure reverse Secure Shell (SSH) so that separate lines do not need to be configured for every terminal or auxiliary line on which SSH must be enabled. This feature also eliminates the rotary-group limitation. This feature was introduced in Cisco IOS Release 12.3(11)T. The following command was introduced: **ssh**.