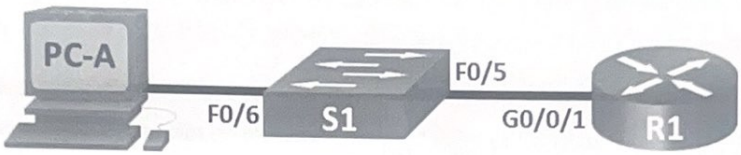
Networking Academy

Lab - Configure Network Devices with SSH

16.4.7

Topology



Addressing Table

Device	Interface	IP Address	Subnet Mask	Default Gateway
R1	G0/0/1	192.168.1.1	255.255.255.0	N/A
S1	VLAN 1	192.168.1.11	255.255.255.0	192.168.1.1 - D0
PC-A	NIC	192.168.1.3	255.255.255.0	192.168.1.1

Objectives

Part 1: Configure Basic Device Settings

Part 2: Configure the Router for SSH Access

Part 3: Configure the Switch for SSH Access

Part 4: SSH from the CLI on the Switch

Background / Scenario

In the past, Telnet was the most common network protocol used to remotely configure network devices. Telnet does not encrypt the information between the client and server. This allows a network sniffer to intercept passwords and configuration information.

Secure Shell (SSH) is a network protocol that establishes a secure terminal emulation connection to a router or other networking device. SSH encrypts all information that passes over the network link and provides authentication of the remote computer. SSH is rapidly replacing Telnet as the remote login tool of choice for network professionals. SSH is most often used to log in to a remote device and execute commands. However, it can also transfer files using the associated Secure FTP (SFTP) or Secure Copy (SCP) protocols.

The network devices that are communicating must be configured to support SSH in order for SSH to function. In this lab, you will enable the SSH server on a router and then connect to that router using a PC with an SSH client installed. On a local network, the connection is normally made using Ethernet and IP.

Note: The routers used with CCNA hands-on labs are Cisco 4221 with Cisco IOS XE Release 16.9.4 (universalk9 image). The switches used in the labs are Cisco Catalyst 2960s with Cisco IOS Release 15.2(2) (lanbasek9 image). Other routers, switches, and Cisco IOS versions can be used. Depending on the model and Cisco IOS version, the commands available and the output produced might vary from what is shown in the labs. Refer to the Router Interface Summary Table at the end of the lab for the correct interface identifiers.

Note: Make sure that the routers and switches have been erased and have no startup configurations. If you are unsure, contact your instructor.

Required Resources

- 1 Router (Cisco 4221 with Cisco IOS XE Release 16.9.4 universal image or comparable)
- 1 Switch (Cisco 2960 with Cisco IOS Release 15.2(2) lanbasek9 image or comparable)
- 1 PC (Windows with a terminal emulation program, such as Tera Term)
- Console cables to configure the Cisco IOS devices via the console ports
- Ethernet cables as shown in the topology

Instructions

Part 1: Configure Basic Device Settings

In Part 1, you will set up the network topology and configure basic settings, such as the interface IP addresses, device access, and passwords on the router.

Step 1: Cable the network as shown in the topology.

Step 2: Initialize and reload the router and switch.

Step 3: Configure the router.

a. Console into the router and enable privileged EXEC mode. >enable

b. Enter configuration mode. # conf term

c. Disable DNS lookup to prevent the router from attempting to translate incorrectly entered commands as though they were host names. (.)# no ip domain-lookup

d. Assign **class** as the privileged EXEC encrypted password. (.)# enable secret class

e. Assign **cisco** as the console password and enable login. (.)# line con 0 -> password cisco -> login

f. Assign **cisco** as the VTY password and enable login. (.)# line vty 0 4 -> password cisco -> login

g. Encrypt the plaintext passwords. (.)# service password-encryption

h. Create a banner that will warn anyone accessing the device that unauthorized access is prohibited. (.)# banner motd # ----- #

i. Configure and activate the G0/0/1 interface on the router using the information contained in the Addressing Table. (.)# int G0/0/1 -> ip address 192.168.1.1 255.255.255.0 -> no shutdown

j. Save the running configuration to the startup configuration file. -> # write memory

Step 4: Configure PC-A.

a. Configure PC-A with an IP address and subnet mask.

b. Configure a default gateway for PC-A.

Step 5: Verify network connectivity.

Ping R1 from PC-A. If the ping fails, troubleshoot the connection.

Part 2: Configure the Router for SSH Access

Using Telnet to connect to a network device is a security risk because all the information is transmitted in a clear text format. SSH encrypts the session data and provides device authentication, which is why SSH is recommended for remote connections. In Part 2, you will configure the router to accept SSH connections over the VTY lines.

Step 1: Configure device authentication.

The device name and domain are used as part of the crypto key when it is generated. Therefore, these names must be entered prior to issuing the **crypto key** command.

a. Configure device name. → hostname R1

b. Configure the domain for the device. → ip domain-name ciscolab.com

Step 2: Configure the encryption key method. → crypto key generate RSA 2048

Step 3: Configure a local database username. → #username admin secret Adm1nP@55

Configure a username using **admin** as the username and **Adm1nP@55** as the password.

Step 4: Enable SSH on the VTY lines. → line vty 0 4 → transport input ssh

a. Enable Telnet and SSH on the inbound VTY lines using the **transport input** command.

b. Change the login method to use the local database for user verification. → login local/end

Step 5: Save the running configuration to the startup configuration file. #write memory

Step 6: Establish an SSH connection to the router.

a. Start Tera Term from PC-A. ✓

b. Establish an SSH session to R1. Use the username **admin** and password **Adm1nP@55**. You should be able to establish an SSH session with R1. ✓

Part 3: Configure the Switch for SSH Access

In Part 3, you will configure the switch to accept SSH connections. After the switch has been configured, establish an SSH session using Tera Term.

Step 1: Configure the basic settings on the switch.

a. Console into the switch and enable privileged EXEC mode. > ena

b. Enter configuration mode. # conf term.

c. Disable DNS lookup to prevent the router from attempting to translate incorrectly entered commands as though they were host names. (.)# no ip domain-lookup

d. Assign **class** as the privileged EXEC encrypted password. (.)# enable secret class

e. Assign **cisco** as the console password and enable login. (.)# line con 0 → password cisco → login

f. Assign **cisco** as the VTY password and enable login. (.)# line vty 0 15 → password cisco → login

g. Encrypt the plain text passwords. (.)# service password-encryption

h. Create a banner that will warn anyone accessing the device that unauthorized access is prohibited. (.)# banner motd # ---#

i. Configure and activate the VLAN 1 interface on the switch according to the Addressing Table. (.)# int vlan 1 (.)# ip address 192.168.1.11 255.255.0 is not shutdown 255.255.255.0

j. Save the running configuration to the startup configuration file. (.)# ip default-gateway 192.168.1.1 #write memory

Step 2: Configure the switch for SSH connectivity.

Use the same commands that you used to configure SSH on the router in Part 2 to configure SSH for the switch.

a. Configure the device name as listed in the Addressing Table. ✓ S1

b. Configure the domain for the device. ✓

c. Configure the encryption key method. ✓

d. Configure a local database username. ✓

e. Enable Telnet and SSH on the VTY lines. ✓

f. Change the login method to use the local database for user verification. ✓

Step 3: Establish an SSH connection to the switch.

Start Tera Term from PC-A, and then SSH to the SVI interface on S1. 192.168.1.1

Are you able to establish an SSH session with the switch?

Yes, the connection was successful.

Part 4: SSH From the CLI on the Switch

The SSH client is built into the Cisco IOS and can be run from the CLI. In Part 4, you will SSH to the router from the CLI on the switch.

Step 1: View the parameters available for the Cisco IOS SSH client.

Use the question mark (**?**) to display the parameter options available with the **ssh** command.

* In Packet Tracer I could see -l and -v.

```
S1# ssh ?
  -c     Select encryption algorithm
  -l     Log in using this user name
  -m     Select HMAC algorithm
  -o     Specify options
  -p     Connect to this port
  -v     Specify SSH Protocol Version
  -vrf   Specify vrf name
  WORD   IP address or hostname of a remote system
```

Step 2: SSH to R1 from S1.

a. You must use the **–l admin** option when you SSH to R1. This allows you to log in as user **admin**. When prompted, enter **Adm1nP@55** for the password.

```
S1# ssh -l admin 192.168.1.1
Password:
Authorized Users Only!
R1>
```

b. You can return to S1 without closing the SSH session to R1 by pressing **Ctrl+Shift+6**. Release the **Ctrl+Shift+6** keys and press **x**. The switch privileged EXEC prompt displays. In the PAC is the same!

```
R1>
S1#
```

c. To return to the SSH session on R1, press Enter on a blank CLI line. You may need to press Enter a second time to see the router CLI prompt.

```
S1#
[Resuming connection 1 to 192.168.1.1 ... ]
```

```
R1>
```

d. To end the SSH session on R1, type **exit** at the router prompt.

```
R1# exit

[Connection to 192.168.1.1 closed by foreign host]
S1#
```

What versions of SSH are supported from the CLI?

Protocol Version 1
Protocol Version 2

Reflection Question

How would you provide multiple users, each with their own username, access to a network device?

I would add each username and password if I have the information on File such as a document or database and then I can copy and paste into devices like Router or switches.

Router Interface Summary Table

Router Model	Ethernet Interface #1	Ethernet Interface #2	Serial Interface #1	Serial Interface #2
1800	Fast Ethernet 0/0 (F0/0)	Fast Ethernet 0/1 (F0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
1900	Gigabit Ethernet 0/0 (G0/0)	Gigabit Ethernet 0/1 (G0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
2801	Fast Ethernet 0/0 (F0/0)	Fast Ethernet 0/1 (F0/1)	Serial 0/1/0 (S0/1/0)	Serial 0/1/1 (S0/1/1)
2811	Fast Ethernet 0/0 (F0/0)	Fast Ethernet 0/1 (F0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
2900	Gigabit Ethernet 0/0 (G0/0)	Gigabit Ethernet 0/1 (G0/1)	Serial 0/0/0 (S0/0/0)	Serial 0/0/1 (S0/0/1)
4221	Gigabit Ethernet 0/0/0 (G0/0/0)	Gigabit Ethernet 0/0/1 (G0/0/1)	Serial 0/1/0 (S0/1/0)	Serial 0/1/1 (S0/1/1)
4300	Gigabit Ethernet 0/0/0 (G0/0/0)	Gigabit Ethernet 0/0/1 (G0/0/1)	Serial 0/1/0 (S0/1/0)	Serial 0/1/1 (S0/1/1)