1.1.1.9 Lab – Mapping the Internet

Objectives

Part 1: Determine Network Connectivity to a Destination Host

Part 2: Trace a Route to a Remote Server Using Tracert

Background / Scenario

Route tracing computer software lists the networks that data traverses from the user's originating end device to a distant destination device.

This network tool is typically executed at the command line as:

tracert <destination network name or end device address>

(Microsoft Windows systems)

or

traceroute <destination network name or end device address>

(UNIX, Linux systems, and Cisco devices, such as switches and routers)

Both **tracert** and **traceroute** determine the route taken by packets across an IP network.

The **tracert** (or **traceroute**) tool is often used for network troubleshooting. By showing a list of routers traversed, the user can identify the path taken to reach a particular destination on the network or across internetworks. Each router represents a point where one network connects to another network and through which the data packet was forwarded. The number of routers is known as the number of hops the data traveled from source to destination.

The displayed list can help identify data flow problems when trying to access a service such as a website. It can also be useful when performing tasks, such as downloading data. If there are multiple websites (mirrors) available for the same data file, one can trace each mirror to get a good idea of which mirror would be the fastest to use.

Command-line based route tracing tools are usually embedded with the operating system of the end device. This activity should be performed on a computer that has Internet access and access to a command line.

Required Resources

PC with Internet access

Part 1: Determine Network Connectivity to a Destination Host

To trace the route to a distant network, the PC used must have a working connection to the Internet. Use the **ping** command to test whether a host is reachable. Packets of information are sent to the remote host with instructions to reply. Your local PC measures whether a response is received to each packet, and how long it takes for those packets to cross the network.

a. At the command-line prompt, type **ping www.cisco.com** to determine if it is reachable.

```
C:\>ping www.cisco.com

Pinging e144.dscb.akamaiedge.net [23.1.48.170] with 32 bytes of data:
Reply from 23.1.48.170: bytes=32 time=56ms TTL=57
Reply from 23.1.48.170: bytes=32 time=55ms TTL=57
Reply from 23.1.48.170: bytes=32 time=54ms TTL=57
Reply from 23.1.48.170: bytes=32 time=54ms TTL=57

Ping statistics for 23.1.48.170:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 54ms, Maximum = 56ms, Average = 54ms
```

b. Now ping one of the Regional Internet Registry (RIR) websites located in different parts of the world to determine if it is reachable:

Africa: **www.afrinic.net**

Australia: **www.apnic.net**

South America: **www.lacnic.net**

North America: **www.arin.net**

Note: At the time of writing, the European RIR www.ripe.net does not reply to ICMP echo requests.

The website you selected will be used in Part 2 for use with the **tracert** command.

Part 2: Trace a Route to a Remote Server Using Tracert

After you determine if your chosen websites are reachable by using **ping**, you will use **tracert** to determine the path to reach the remote server. It is helpful to look more closely at each network segment that is crossed.

Each hop in the **tracert** results displays the routes that the packets take when traveling to the final destination. The PC sends three ICMP echo request packets to the remote host. Each router in the path decrements the time to live (TTL) value by 1 before passing it onto the next system. When the decremented TTL value reaches 0, the router sends an ICMP Time Exceeded message back to the source with its IP address and the current time. When the final destination is reached, an ICMP echo reply is sent to the source host.

For example, the source host sends three ICMP echo request packets to the first hop (192.168.1.1) with the TTL value of 1. When the router 192.168.1.1 receives the echo request packets, it decrements the TTL value to 0. The router sends an ICMP Time Exceeded message back to the source. This process continues until the source hosts sends the last three ICMP echo request packets with TTL values of 8 (hop number 8 in the output below), which is the final destination. After the ICMP echo request packets arrive at the final destination, the router responds to the source with ICMP echo replies.

For hops 2 and 3, these IP addresses are private addresses. These routers are the typical setup for point-of-presence (POP) of ISP. The POP devices connect users to an ISP network.

A web-based whois tool is found at http://whois.domaintools.com/. It is used to determine the domains traveled from the source to destination.

a. At the command-line prompt, trace the route to www.cisco.com. Save the **tracert** output in a text file. Alternatively, you can redirect the output to a text file by using **>** or **>>**.

```
C:\Users\User1> tracert www.cisco.com
```

or

```
C:\Users\User1> tracert www.cisco.com > tracert-cisco.txt

Tracing route to e144.dscb.akamaiedge.net [23.67.208.170]
over a maximum of 30 hops:

  1     1 ms    <1 ms    <1 ms  192.168.1.1
  2    14 ms     7 ms     7 ms  10.39.0.1
  3    10 ms     8 ms     7 ms  172.21.0.118
  4    11 ms    11 ms    11 ms  70.169.73.196
  5    10 ms     9 ms    11 ms  70.169.75.157
  6    60 ms    49 ms     *     68.1.2.109
  7    43 ms    39 ms    38 ms  Equinix-DFW2.netarch.akamai.com [206.223.118.102]
  8    33 ms    35 ms    33 ms  a23-67-208-170.deploy.akamaitechnologies.com
[23.67.208.170]

Trace complete.
```

b. The web-based tool at http://whois.domaintools.com/ can be used to determine the owners of both the resulting IP address and domain names shown in the tracert tools output. Now perform a **tracert** to one of RIR web sites from Part 1 and save the results.

Africa: **www.afrinic.net**

Australia: **www.apnic.net**

Europe: **www.ripe.net**

South America: **www.lacnic.net**

North America: **www.arin.net**

List the domains below from your tracert results using the web-based whois tool.

whois.arin.net
whois.apnic.net
whois.apnic.net
whois.apnic.net
whois.apnic.net
whois.arin.net
whois.apnic.net
whois.arin.net

c. Compare the lists of domains crossed to reach the final destinations.

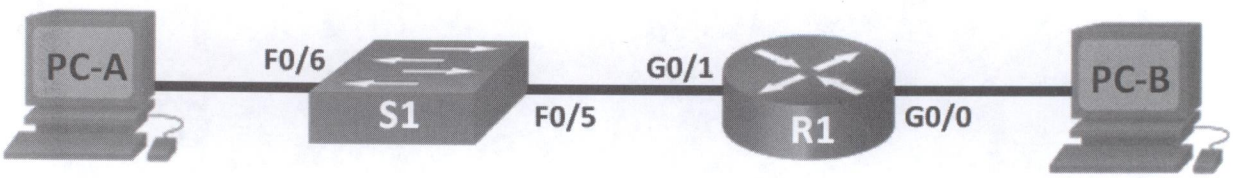
Reflection

What can affect **tracert** results?

1 <1 ms <1 ms 1 ms XiaoQiang [192.168.31.1]
2 1 ms 1 ms 2 ms 10.10.112.1
3 10 ms 10 ms 9 ms 103.26.246.237
4 9 ms * 12 ms 157.119.185.115
5 58 ms 57 ms 58 ms ix-xe-3-3-4-0.tcore2.mlv-mumbai.as6453.net [180.87.39.173]
6 * 59 ms 58 ms 180.87.39.26
7 92 ms 91 ms 90 ms 172.29.239.190
8 91 ms 92 ms 91 ms 14.140.193.146.STATIC-Kolkata-vsnl.net.in [14.140.193.146]
9 * 91 ms 91 ms a23-58-72-137.deploy.static.akamaitechnologies.com [23.58.72.137]

1.1.4.6 Lab – Configuring Basic Router Settings with IOS CLI

Topology



Addressing Table

Device	Interface	IP Address	Subnet Mask	Default Gateway
R1	G0/0	192.168.0.1	255.255.255.0	N/A
	G0/1	192.168.1.1	255.255.255.0	N/A
PC-A	NIC	192.168.1.3	255.255.255.0	192.168.1.1
PC-B	NIC	192.168.0.3	255.255.255.0	192.168.0.1

Objectives

Part 1: Set Up the Topology and Initialize Devices

- Cable equipment to match the network topology.
- Initialize and restart the router and switch.

Part 2: Configure Devices and Verify Connectivity

- Assign static IPv4 information to the PC interfaces.
- Configure basic router settings.
- Verify network connectivity.
- Configure the router for SSH.

Part 3: Display Router Information

- Retrieve hardware and software information from the router.
- Interpret the output from the startup configuration.
- Interpret the output from the routing table.
- Verify the status of the interfaces.

Part 4: Configure IPv6 and Verify Connectivity

Background / Scenario

This is a comprehensive lab to review previously covered IOS router commands. In Parts 1 and 2, you will cable the equipment and complete basic configurations and IPv4 interface settings on the router.

In Part 3, you will use SSH to connect to the router remotely and utilize IOS commands to retrieve information from the device to answer questions about the router. In Part 4, you will configure IPv6 on the router so that PC-B can acquire an IP address and then verify connectivity.

For review purposes, this lab provides the commands necessary for specific router configurations.

Note: The routers used with CCNA hands-on labs are Cisco 1941 Integrated Services Routers (ISRs) with Cisco IOS Release 15.2(4)M3 (universalk9 image). The switches used are Cisco Catalyst 2960 with Cisco IOS Release 15.0(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 output produced might vary from what is shown in the labs. Refer to the Router Interface Summary Table at the end of this lab for the correct interface identifiers.

Note: Make sure that the router and switch have been erased and have no startup configurations. Refer to Appendix A for the procedures to initialize and reload devices.

Required Resources

- 1 Router (Cisco 1941 with Cisco IOS Release 15.2(4)M3 universal image or comparable)
- 1 Switch (Cisco 2960 with Cisco IOS Release 15.0(2) lanbasek9 image or comparable)
- 2 PCs (Windows 7, Vista, or XP with 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

Note: The Gigabit Ethernet interfaces on Cisco 1941 ISRs are autosensing and an Ethernet straight-through cable can be used between the router and PC-B. If using another model Cisco router, it may be necessary to use an Ethernet crossover cable.

Part 1: Set Up the Topology and Initialize Devices

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

a. Attach the devices as shown in the topology diagram, and cable as necessary.

b. Power on all the devices in the topology.

Step 2: Initialize and reload the router and switch.

Note: Appendix A details the steps to initialize and reload the devices.

Part 2: Configure Devices and Verify Connectivity

Step 1: Configure the PC interfaces.

a. Configure the IP address, subnet mask, and default gateway settings on PC-A.

b. Configure the IP address, subnet mask, and default gateway settings on PC-B.

Step 2: Configure the router.

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

```
Router> enable
Router#
```

b. Enter into global configuration mode.

```
Router# config terminal
Router(config)#
```

c. Assign a device name to the router.

```
Router(config)# hostname R1
```

d. Disable DNS lookup to prevent the router from attempting to translate incorrectly entered commands as though they were hostnames.

```
R1(config)# no ip domain-lookup
```

e. Require that a minimum of 10 characters be used for all passwords.

```
R1(config)# security passwords min-length 10
```

Besides setting a minimum length, list other ways to strengthen passwords.

combining characters and digits, we can strengthen the password.

f. Assign **cisco12345** as the privileged EXEC encrypted password.

```
R1(config)# enable secret cisco12345
```

g. Assign **ciscoconpass** as the console password, establish a timeout, enable login, and add the **logging synchronous** command. The **logging synchronous** command synchronizes debug and Cisco IOS software output and prevents these messages from interrupting your keyboard input.

```
R1(config)# line con 0
R1(config-line)# password ciscoconpass
R1(config-line)# exec-timeout 5 0
R1(config-line)# login
R1(config-line)# logging synchronous
R1(config-line)# exit
R1(config)#
```

For the **exec-timeout** command, what do the **5** and **0** represent?

5 represents Minutes and 0 represents seconds

h. Assign **ciscovtypass** as the vty password, establish a timeout, enable login, and add the **logging synchronous** command.

```
R1(config)# line vty 0 4
R1(config-line)# password ciscovtypass
R1(config-line)# exec-timeout 5 0
R1(config-line)# login
R1(config-line)# logging synchronous
R1(config-line)# exit
R1(config)#
```

i. Encrypt the clear text passwords.

```
R1(config)# service password-encryption
```

j. Create a banner that warns anyone accessing the device that unauthorized access is prohibited.

```
R1(config)# banner motd #Unauthorized access prohibited!#
```

k. Configure an IP address and interface description. Activate both interfaces on the router.

```
R1(config)# int g0/0
R1(config-if)# description Connection to PC-B
R1(config-if)# ip address 192.168.0.1 255.255.255.0
R1(config-if)# no shutdown
R1(config-if)# int g0/1
R1(config-if)# description Connection to S1
R1(config-if)# ip address 192.168.1.1 255.255.255.0
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config)# exit
R1#
```

l. Set the clock on the router; for example:

```
R1# clock set 17:00:00 18 Feb 2013
```

m. Save the running configuration to the startup configuration file.

```
R1# copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
R1#
```

What would be the result of reloading the router prior to completing the **copy running-config startup-config** command?

Basically the running-config is working like a RAM which is volatile memory, whenever the router restarts the configurations will be erased, for solving this problem we copy the configuration of the router to startup-config which is working like a non volatile memory.

Step 3: **Verify network connectivity.**

a. Ping PC-B from a command prompt on PC-A.

Note: It may be necessary to disable the PCs firewall.

Were the pings successful? YES

After completing this series of commands, what type of remote access could be used to access R1?

We can use TELNET service as we configure with line vty command.