CST 235 Network and System Administration

Understanding Routing Information Protocol

Name: ________Ben Walker____

Date: _________6-28-18____

1. Routing Information Protocol (RIP)

RIP is one of the oldest distance-vector routing protocols, which uses the hop count as a routing metric. RIP avoids routing loops by employing a limit on the number of hops permitted in a path from the source to a destination. The maximum number of hops permitted for RIP is 15. A hop count of 16 is considered an infinite distance, in other words, the route is considered unreachable. Despite this limitation, RIP works great for basic route communications between devices.

How RIP works? With RIP, a router sends its full routing table to all other connected routers every 30 seconds. Triggered updates can also occur if a router goes down before the 30-second timer has expired.

What is different in RIP version 2? RIPv2 boasts the following enhancements:

- Support for variable length subnet masks (VLSM) (Because of this, RIP doesn't assume that all networks are classful.)
- Multicast routing updates
- Authentication with an encrypted password for routing updates.

2. Goal

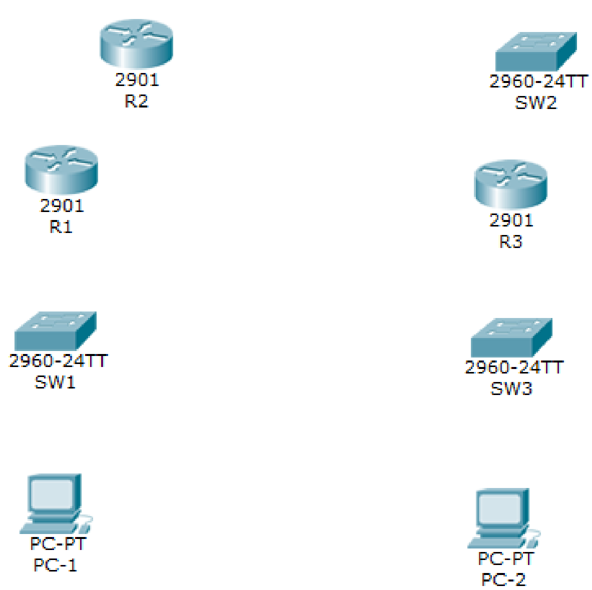
This lab helps you gain a better understanding of how to configure and troubleshoot RIP version 2. In this lab, we will configure a basic lab topology, enable RIPv2 on the routers and configure them to advertise their networks. We will also learn to use Cisco IOS show and debug commands to verify and troubleshoot our lab.

3. Lab Procedures

3.1 Topology

Step 1: Create three routers, three switches, and two PCs.

Step 2: Change the display and host name according to the following figure.



3.2 making the connection

Step 3: Use copper straight-Through to connect:

- PC-1 FastEthernet0 to SW1 GigabitEthernet 0/2
- SW1 GigabitEthernet 0/1 to R1 GigabitEthernet 0/0
- PC-2 FastEthernet0 to SW3 GigabitEthernet 0/2
- SW3 GigabitEtherent 0/1 to R3 GigabitEthernet 0/1
- R3 GigabitEthernet 0/0 to SW2 GigabitEthernet 0/2
- SW2 GigabitEthernet 0/1 to R2 GigabitEthernet 0/0

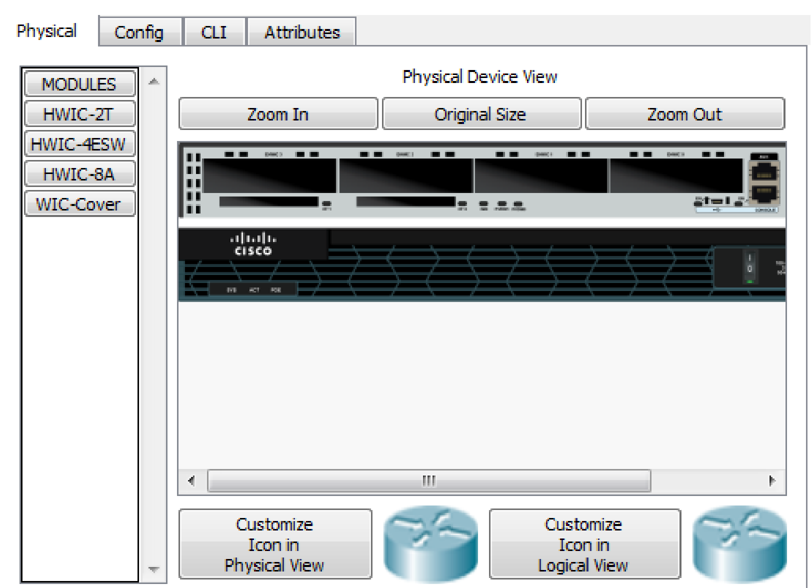
Why using Straight-Through cable?

Since the connections are between different devices, a standard straight through is needed. For device to device such as router to router, it is sometimes more optimal to use crossover cables, however they require the correct hardware for it to work. Copper straight through is also usually less expensive to run than fiber but can still handle fairly high bandwidth. And if wireless was used, again special hardware may be required, and this may null the whole test in general because all things may already be able to connect with each other depending on the setup. Using something like phone line would not handle enough bandwidth.

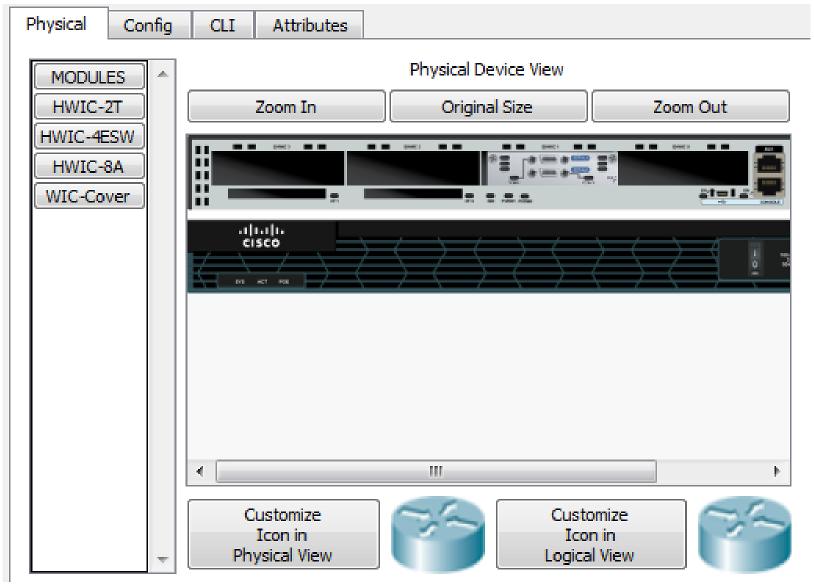
Step 4: Use Serial DCE to connect serial ports

Step 4.1: Create Serial ports

Click the router, for example R2. Click Physical tab. You will see the following figure



Then, turn off the switch. If you cannot see the switch, zoom in to find it. After you turn off the switch, we can add the serial port to the router. Click HWIC-2T (The HWIC-2T is a Cisco 2-port Serial High-Speed WAN Interface Card, providing 2 serial ports), and drag it to any slot. After that, turn on the switch. Then, the back of the router looks like the following figure.

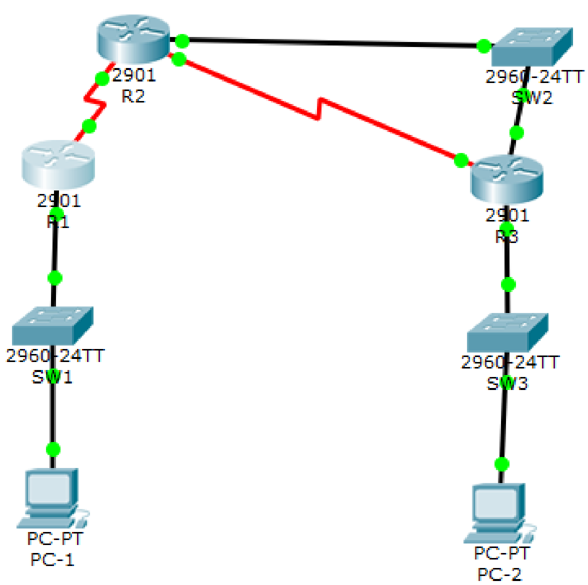


Repeat the same process for R1 and R3.

<u>Step 4.2</u>: Connect Serial ports by Serial DCE

- R1 S0/1/0 to R2 S0/1/0 (note: click R2 first to make the interface as DCE)
- R2 S0/1/1 to R3 S0/1/0 (note: click R2 first to make the interface as DCE)

Your topology should be the same as the following figure.



<u>Step 5</u>: Configure no-domain lookup on all routers and switches

Click R1, then choose the CLI tab. First, you need to enter the configure terminal mode. Then, use the command “no ip domain-lookup” to disable the DNS lookup.

```
Continue with configuration dialog? [yes/no]: no


Press RETURN to get started!



Router>enable
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#no ip domain-lookup
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
```

Repeat the same process for all routers (R2 and R3) and switches (SW1, SW2, and SW3).

3.3 LAN Configuration

Step 6: Assign IP address and subnet to GigabitEthernet interfaces of R1, R2 and R3. Here is the command for R1. You need to repeat the same process for R2 and R3.

```
Router>enable
Router#config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface gigabitethernet 0/0
Router(config-if)#ip address 172.16.10.1 255.255.255.0
Router(config-if)#exit
```

- R1: GigabitEthernet 0/0: 172.16.10.1/24
- R2: GigabitEthernet 0/0: 192.168.2.2/27
- R3:
  - GigabitEthernet 0/0: 192.168.2.3/27
  - GigabitEthernet 0/1: 172.16.30.1/24

Step 7: Assign IP address, subnet, and default gateway to hosts.

Click the PC icon, then choose desktop tab. Enter the IP address and subnet mask under IP Configuration.

- PC-1:
  - ip address and mask: 172.16.10.10/24
  - Default gateway: 172.16.10.1
- PC-2:
  - ip address and mask: 172.16.30.10/24
  - Default gateway: 172.16.30.1

Step 8: Insure GigabitEthernet interfaces are not administratively down by access each interface and issue a “no shutdown” command. Here is the example for R1. Please repeat the same process for R2 and R3.

Step 8.1: Change the status from administratively down to administratively up

```
R1(config)#interface gigabitethernet 0/0
R1(config-if)#no shutdown
```

Step 8.2: Check the status by “show interface” command for all routers. Here is the example for R1. Please repeat the process for R2 and R3.

```
R1(config-if)#exit
R1(config)#exit
R1#
%SYS-5-CONFIG_I: Configured from console by console

R1#show interface
GigabitEthernet0/0 is up, line protocol is up (connected)
  Hardware is CN Gigabit Ethernet, address is 0001.64dd.c901 (bia
0001.64dd.c901)
  Internet address is 172.16.10.1/24
  MTU 1500 bytes, BW 1000000 Kbit, DLY 10 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Full-duplex, 100Mb/s, media type is RJ45
```

3.4 WAN Configuration

Step 9: Assign IP address and subnet to serial interfaces of R1, R2 and R3. Here is the example for R3. Please repeat the process for R1 and R2.

```
R3>enable
R3#config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R3(config)#interface S0/1/0
R3(config-if)#ip address 192.168.1.2 255.255.255.248
R3(config-if)#no shutdown

R3(config-if)#
%LINK-5-CHANGED: Interface Serial0/1/0, changed state to up

R3(config-if)#exit
R3(config)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/1/0,
changed state to up
```

- R1
  - S0/1/0: 192.168.0.1/30
- R2
  - S0/1/0: 192.168.0.2/30
  - S0/1/1: 192.168.1.1/29
- R3
  - S0/1/0: 192.168.1.2/29

What is serial interface?

A serial interface is a short distance communication interface with high speed.

Step 10: Assign IP address and subnet to loopback of R1 and R3

Here is the example for R1 Loopback interface 100. Repeat the same process for R1 Loopback interface 101, and R3 Loopback interface 100 and 101.

```
R1#config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#interface Loopback 100
R1(config-if)#ip address 10.10.100.1 255.255.255.224
R1(config-if)#exit
```

- R1
  - Loopback 100: 10.10.100.1/27
  - Loopback 101: 10.10.101.1/26
- R3
  - Loopback 100: 10.30.100.1/29
  - Loopback 101: 10.30.101.1/25

What is loopback address?

Loopback address is an address that sends outgoing signal back to the same machine.

Step 11: Configure R2 interface Serial0/1/0 and Serial0/1/1 as DCE to provide clocking to R1 and R3 at a clock speed of 2 Mbps. Here is the example for interface S0/1/0. Repeat the same process for S0/1/1.

```
R2(config)#interface S0/1/0
R2(config-if)#clock rate 2000000
R2(config-if)#no shutdown
```

Tips: use “show controller S0/1/0” and “show controller S0/1/1” on R1, R2, and R3 to check which interface is DCE and which one is DTC.

What is DCE and what is DTE?

DTE: R1 S0/3/0, R2 S0/3/1

DCE: R2 S0/3/0, R3 S0/3/0

DTE stands for Data Terminal Equipment, synchronizes clock.

DCE stands for Data Communication Equipment, provides clock signal.

Step 12: Insure all interfaces are not administratively down.

Apply “no shutdown” for each interface and then use “show interface” to check the status.

Step 13: Using ping insure connectivity to the connected ports.

For example:

```
R2#ping 192.168.0.2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.0.2, timeout is 2
seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max =
2/4/13 ms
```

Can you successfully ping from PC-1 to PC-2? Show screenshot and explain why or why not.

```
Packet Tracer PC Command Line 1.0
C:\>ping 172.16.30.10

Pinging 172.16.30.10 with 32 bytes of data:

Reply from 172.16.10.1: Destination host unreachable.
Reply from 172.16.10.1: Destination host unreachable.
Reply from 172.16.10.1: Destination host unreachable.
Reply from 172.16.10.1: Destination host unreachable.

Ping statistics for 172.16.30.10:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>
```

It cannot ping these computers, because they are using different subnet masks and are in different IP ranges, essentially putting them on different networks from R2.

3.5 RIP Configuration

Step 14: Enable RIPv2 on R1, R2 and R3

Step 15: Configure RIPv2 routing for the GigabitEthernet and the Serial interfaces of R1, R2 and R3.

Here is the example for R1 for both Step 14 and 15. Repeat the same process for R2 and R3.

```
R1>enable
R1#config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#router rip
R1(config-router)#version 2
R1(config-router)#network 172.16.10.0
R1(config-router)#network 192.168.0.0
R1(config-router)#network 10.10.100.0
R1(config-router)#network 10.10.101.0
```

Step 16: Verify that RIPv2 has been enabled on R1, R2, and R3 using the appropriate commands.

Issue “show ip protocols” and “show ip route” on R1, R2 and R3. For example:

```
R1#show ip protocols
Routing Protocol is "rip"
Sending updates every 30 seconds, next due in 24 seconds
Invalid after 180 seconds, hold down 180, flushed after 240
Outgoing update filter list for all interfaces is not set
Incoming update filter list for all interfaces is not set
Redistributing: rip
Default version control: send version 2, receive 2
  Interface             Send  Recv  Triggered RIP  Key-chain
  GigabitEthernet0/0    2     2
  Serial0/1/0           2     2
  Loopback100           2     2
  Loopback101           2     2
Automatic network summarization is in effect
Maximum path: 4
Routing for Networks:
           10.0.0.0
           172.16.0.0
           192.168.0.0
Passive Interface(s):
Routing Information Sources:
           Gateway         Distance      Last Update
           192.168.0.2          120      00:01:44
```

Can you successfully ping from PC-1 to PC-2? Show screenshot and explain why or why not.

```
C:\>ping 172.16.30.10

Pinging 172.16.30.10 with 32 bytes of data:

Reply from 172.16.10.1: Destination host unreachable.
Reply from 172.16.10.1: Destination host unreachable.
Reply from 172.16.10.1: Destination host unreachable.
Reply from 172.16.10.1: Destination host unreachable.

Ping statistics for 172.16.30.10:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>ping 172.16.30.10

Pinging 172.16.30.10 with 32 bytes of data:

Reply from 172.16.10.1: Destination host unreachable.
Reply from 172.16.10.1: Destination host unreachable.
Reply from 172.16.10.1: Destination host unreachable.
Reply from 172.16.10.1: Destination host unreachable.

Ping statistics for 172.16.30.10:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>
```

With RIP being enabled for version 2 on those ranges the paths the classes are still in effect. They do not allow communication outside their ranges.

3.6 Automatic Summarization

In your studies, you learned that RIPv2 performs automatic summarization at Classful boundaries.

Step 17: Disable automatic summarization on R1, R2, and R3.

Here is the example to disable automatic summarization on R1. Repeat the same process for R2 and R3.

```
R1#config terminal
Enter configuration commands, one per line.  End with CNTL/Z.
R1(config)#router rip
R1(config-router)#no auto-summary
R1(config-router)#exit
```

Display the routing tables of R2 and R3 and verify that the 10.10.100.0/27 and 10.10.101.0/26 routes from R1 are now present.

Can you successfully ping from PC-1 to PC-2? Show screenshot and explain why or why not.

```
Pinging 172.16.30.10 with 32 bytes of data:

Reply from 172.16.10.1: Destination host unreachable.
Reply from 172.16.10.1: Destination host unreachable.
Reply from 172.16.10.1: Destination host unreachable.
Reply from 172.16.10.1: Destination host unreachable.

Ping statistics for 172.16.30.10:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>ping 172.16.30.10

Pinging 172.16.30.10 with 32 bytes of data:

Reply from 172.16.30.10: bytes=32 time=2ms TTL=125
Reply from 172.16.30.10: bytes=32 time=18ms TTL=125
Reply from 172.16.30.10: bytes=32 time=16ms TTL=125
Reply from 172.16.30.10: bytes=32 time=15ms TTL=125

Ping statistics for 172.16.30.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 2ms, Maximum = 18ms, Average = 12ms

C:\>
```

Since RIP is now classless, it allows building routes beyond subnets and differing networks.

3.7 Verifying RIPv2 Updates Using Debug

RIPv2 update debugging is used by network engineers to troubleshoot network and routing problems. Like RIPv1, RIPv2 is a distance vector protocol that uses hop count with a maximum of 15 hops, but unlike RIPv1, RIPv2 sends updates using Multicast.

Enable debugging on R1 and R2 and verify that RIPv2 updates are being sent out of all RIPv2-enabled networks. Keep in mind that by default RIP sends updates every 30 seconds, so you will typically see updates within that time frame. Familiarize yourself with the output of the debugs.

Real World Tip: you may wish to start a capture on your terminal session to capture the debug output to a file so you can review it later.

Step 18: Issue “debug ip rip events” on R1 and R2

Here is the example for R1. Repeat the same process for R2.

```
R1#debug ip rip events
RIP event debugging is on
R1#RIP: received v2 update from 192.168.0.2 on Serial0/1/0
      10.30.100.0/29 via 0.0.0.0 in 2 hops
      10.30.101.0/25 via 0.0.0.0 in 2 hops
      172.16.30.0/24 via 0.0.0.0 in 2 hops
      192.168.1.0/29 via 0.0.0.0 in 1 hops
      192.168.2.0/27 via 0.0.0.0 in 1 hops
RIP: sending  v2 update to 224.0.0.9 via GigabitEthernet0/0
(172.16.10.1)
RIP: build update entries
      10.10.100.0/27 via 0.0.0.0, metric 1, tag 0
      10.10.101.0/26 via 0.0.0.0, metric 1, tag 0
```

Show the result you got by screenshot. Explain why or why not RIP works.

```
RIP: received v2 update from 192.168.0.2 on Serial0/3/0
      10.30.100.0/29 via 0.0.0.0 in 2 hops
      10.30.101.0/25 via 0.0.0.0 in 2 hops
      172.16.30.0/24 via 0.0.0.0 in 2 hops
      192.168.1.0/29 via 0.0.0.0 in 1 hops
      192.168.2.0/27 via 0.0.0.0 in 1 hops
```

```
RIP: received v2 update from 192.168.0.1 on Serial0/3/0
      10.10.100.0/27 via 0.0.0.0 in 1 hops
      10.10.101.0/26 via 0.0.0.0 in 1 hops
      172.16.10.0/24 via 0.0.0.0 in 1 hops
RIP: received v2 update from 192.168.2.3 on GigabitEthernet0/0
      10.30.100.0/29 via 0.0.0.0 in 1 hops
      10.30.101.0/25 via 0.0.0.0 in 1 hops
      172.16.30.0/24 via 0.0.0.0 in 1 hops
      192.168.1.0/29 via 0.0.0.0 in 1 hops
RIP: received v2 update from 192.168.1.2 on Serial0/3/1
      10.30.100.0/29 via 0.0.0.0 in 1 hops
      10.30.101.0/25 via 0.0.0.0 in 1 hops
      172.16.30.0/24 via 0.0.0.0 in 1 hops
      192.168.2.0/27 via 0.0.0.0 in 1 hops
```

RIP works because it is able to communicate through each router, and build routes. Even if one router can’t see all connections, each can see the connections it has, and so the routes can be built beyond the scope of just one of the routers.

<u>Step 19</u>: Disable debugging on all interfaces.

Here is the example for R1. Repeat the same process for R2.

```
R1#no debug all
All possible debugging has been turned off
R1#
```

3.7 Passive Interfaces for RIPv2 Updates (Bonus 10 points)

In this part of the lab, you will show how to prevent RIPv2 from sending unnecessary updates by using passive interfaces.

By default, RIPv2 sends updates via Multicast on all interfaces for which RIPv2 has been enabled. For example, it is not possible to ever have another device connected to a Loopback interface, so it is a waste of router processing power to have RIPv2 continuously send updates to a Loopback interface.

Step 20: using the “show ip protocols” command verify which interfaces RIPv2 is sending updates on.

Show your screenshot and summarize interfaces that are enabled for RIPv2 updates.

Step 21: Enable debugging on R1 and show that RIPv2 updates are being sent on all RIPv2-enabled interfaces. Disable debugging when done.

Show screenshot and highlight important parts.

Step 22: Prevent RIPv2 from sending updates on the Loopback interfaces of R1 and R3.

Show your commands by using screenshot

Step 23: Verify your configuration by using the “show IP protocols” command and enabling debugging. Disable debugging when done.

Show screenshot and highlight important parts.