Equipment

The physical topology is preconfigured in Packet Tracer using three 1841s and three PCs.

The 1841 routers have two WIC-2T cards inserted. Interfaces S0/0/0, S0/0/1, and S0/1/0 are used to interconnect the three routers:

- S0/0/0 interfaces connect Router1 (DCE) and Router2 – the clock rate is 4,000,000 bits/sec
- S0/0/1 interfaces connect Router2 (DCE) and Router3 – the clock rate is 4,000,000 bits/sec
- S0/1/0 interfaces connect Router3 (DCE) and Router1 – the clock rate is 4,000,000 bits/sec

The built-in F0/0 interface on each router connects to the FastEthernet interface on the connected PC.

Objective

Configure IPv6 RIP to facilitate internetworking.

Note: Routerx in the PT topology has been preconfigured with hostname “Rx” for x=1,2,3.
Step 1: Enable IPv6 unicast routing on each router.

a. Configure EUI-64 network addresses on each FastEthernet segment. Enable stateless auto-configuration on each PC. Configure global unicast and unique local IPv6 networks on the serial links. Use the following network addresses:

- R1 Fast Ethernet: 2011:314:271:1::/64
- R3 Fast Ethernet: 2011:314:271:3::/64
- R1-to-R2 Serial: FC00::12:0/112 (use host address ::x for Rx, x=1,2)
- R2-to-R3 Serial: FC00::23:0/112 (use host address ::x for Rx, x=2,3)
- R3-to-R1 Serial: FC00::31:0/112 (use host address ::x for Rx, x=3,1)

For example, here is the configuration for R1:

```
R1> enable
R1# configure terminal
R1(config)# ipv6 unicast-routing
R1(config)# interface FastEthernet 0/0
R1(config-if)# ipv6 address 2011:314:271:1::/64 eui-64
R1(config-if)# ipv6 enable
R1(config-if)# interface Serial 0/0/0
R1(config-if)# ipv6 address FC00::12:1/112
```

b. Repeat the parallel commands on R2 and R3.

Step 2: Determine global unicast IPv6 addresses.

a. On each router, determine the global unicast IPv6 address for the FastEthernet interface. Record the address.

```
R1# show ipv6 interface brief
FastEthernet0/0 [up/up]
    FE80::201:97FF:FE72:B401
FastEthernet0/1 [administratively down/down]
    FE80::202:16FF:FEEB:3D01
    FC00::12:1
Serial0/0/0 [up/up]
    FE80::207:ECFF:FE56:BE01
    FC00::31:1
Serial0/0/1 [administratively down/down]
Serial0/1/0 [up/up]
    FE80::207:ECFF:FE56:BE01
    FC00::31:1
Vlan1 [administratively down/down]
```

b. Find and record the global unicast IPv6 address on R2.

```
R2# show ipv6 interface brief
FastEthernet0/0 [up/up]
    FE80::260:70FF:FESA:4501
FastEthernet0/1 [administratively down/down]
Serial0/0/0 [up/up]
    FE80::2E0:F7FF:FE24:2401
    FC00::12:2
Serial0/0/1 [up/up]
```
FE80::2E0:F7FF:FE24:2402
FC00::23:2
Serial0/1/0 [administratively down/down]
Serial0/1/1 [administratively down/down]
Vlan1 [administratively down/down]

Step 3: On each router, configure IPv6 RIP routes for each of the three connected networks.

Configuring RIPng however is easier than RIPv2. When configuring RIPng you do not use the network x.x.x.x command in router configuration mode to specify which interfaces participate in the RIP process anymore. As of RIPng you now configure RIP interface participation on a Per interface basis.

To configure an interface to participate in RIPng you use the ipv6 rip NAME enable command on a per interface basis. The name specified in the syntax is a locally significant name used to identify the RIPng process on that router as you can have multiple RIPng processes on a single router whereas RIP for IPv4 you can only have a single process.

a. Configure the connected networks on R1. Enable the RIPng process “JINJA”.

```bash
R1(config)# interface F0/0
R1(config-if)# ipv6 rip JINJA enable
```

```bash
R1(config-if)# interface Serial0/0/0
R1(config-if)# ipv6 rip JINJA enable
R1(config-if)# interface Serial0/1/0
R1(config-if)# ipv6 rip JINJA enable
```

b. Repeat the parallel commands on R2 and R3.

Step 4: Verify routing tables.

a. Verify the configuration with the command show ipv6 route.

```bash
R1# show ipv6 route
IPv6 Routing Table - 10 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
U - Per-user Static route, M - MIPv6
I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
```
Step 5: Verify connectivity.

On R1, verify connectivity to PC2, PC3, and network FC00::23:0/112.

a. To determine the IPv6 addresses of PC2 and PC3, click the PC, click the Desktop tab, click the Command Prompt button, and then type the command `ipv6config` to view the PC’s IPv6 address (results will vary). Record the addresses.

PC2:
```
PC> ipv6config
```
```
Default Gateway...................: FE80::260:70FF:FE8A:4501
```

PC3:
```
PC> ipv6config
```
```
Default Gateway...................: FE80::290:CFF:FE65:8D01
```

b. Ping PC2 and PC3 address from R1.

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2011:314:271:2:260:5CFF:FE74:4CD4, timeout is 2 seconds:
```

b. Repeat for routers R2 and R3.
Success rate is 100 percent (5/5), round-trip min/avg/max = 62/62/63 ms


Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2011:314:271:3:250:fff:fe13:a3c7, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 62/62/63 ms

c. Ping the interface addresses on the FC00::23:0/112 network from R1.

R1# ping fc00::23:2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to FC00::23:2, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 16/31/62 ms

R1# ping fc00::23:3

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to FC00::23:3, timeout is 2 seconds:
!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 31/37/48 ms

d. Repeat verification from R2 and R3 to all other PCs and networks.

At this point, there is IPv6 connectivity between all devices in the topology.

To view the RIPng protocol timers and other related information you use the command `show ipv6 rip NAME`. To view the RIPng database you’ll use the `show ipv6 rip NAME database` command in user or privileged mode.

There is one new command for IPv6 that is not in RIP for IPv4 which is the `show ipv6 rip NAME next-hops` command. This command will display how many routes are are pointing towards each next-hop (neighboring router).