

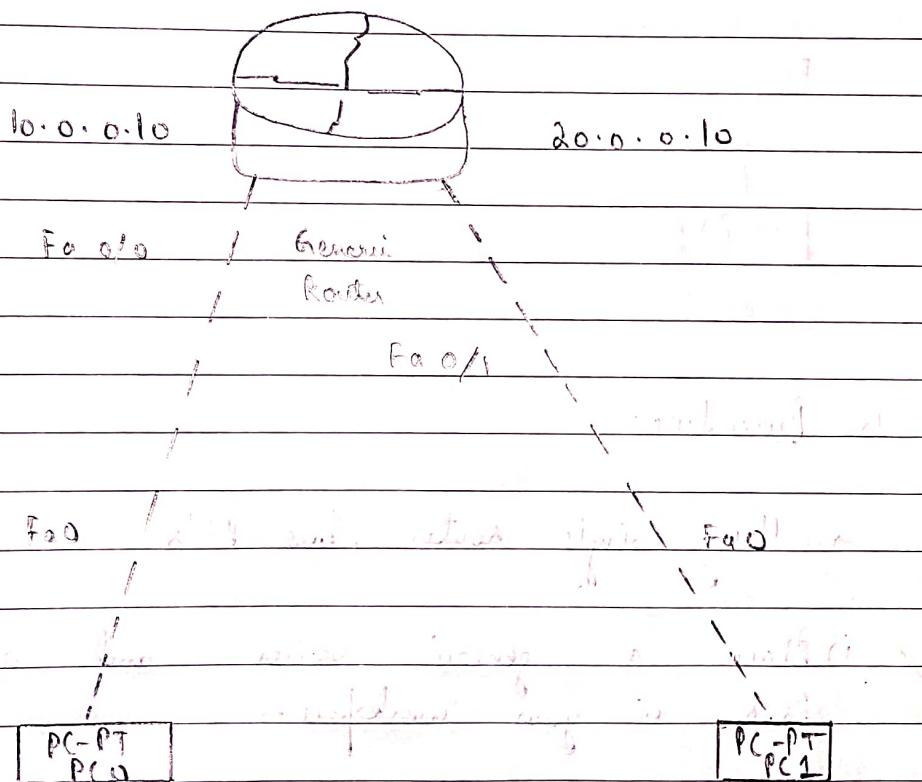
(Lab: Week 2)

Experiment using routers and PCs

↳ Aim: Configuring IP addresses to routers in packet tracer to explore the following messages: Ping response, destination unreachable, Request timed out, reply!

↳ Topology:

→ Using single router, two PCs:



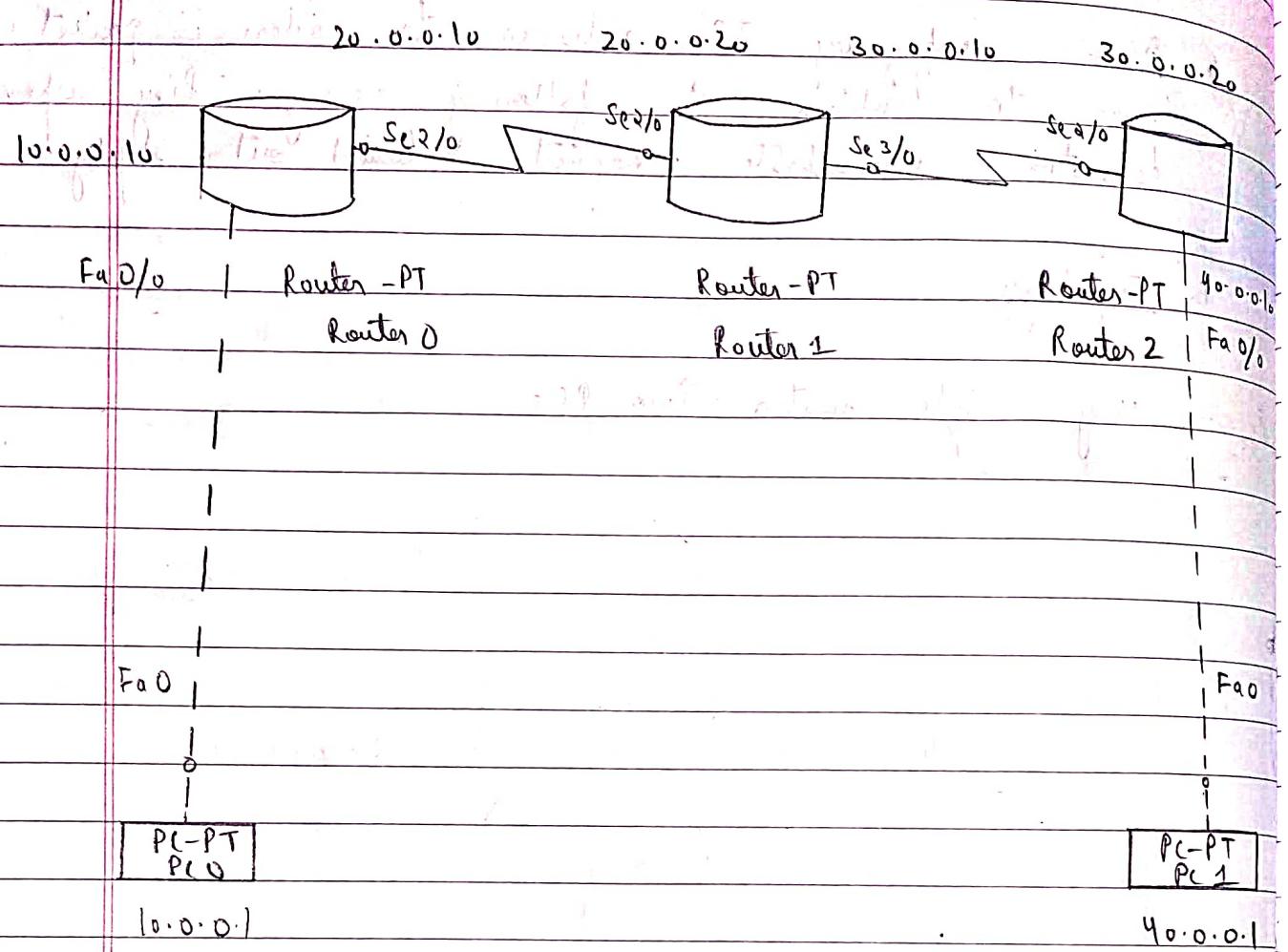
QUESTION: How does the destination unreachable message work?

ANSWER: If a host sends a packet to a destination that is not reachable, the router will return a destination unreachable message.

QUESTION: What happens if a host receives a destination unreachable message?

ANSWER: The host will drop the packet and send an ICMP error message back to the source.

→ Using three routers, two PCs:



↳ Procedure:

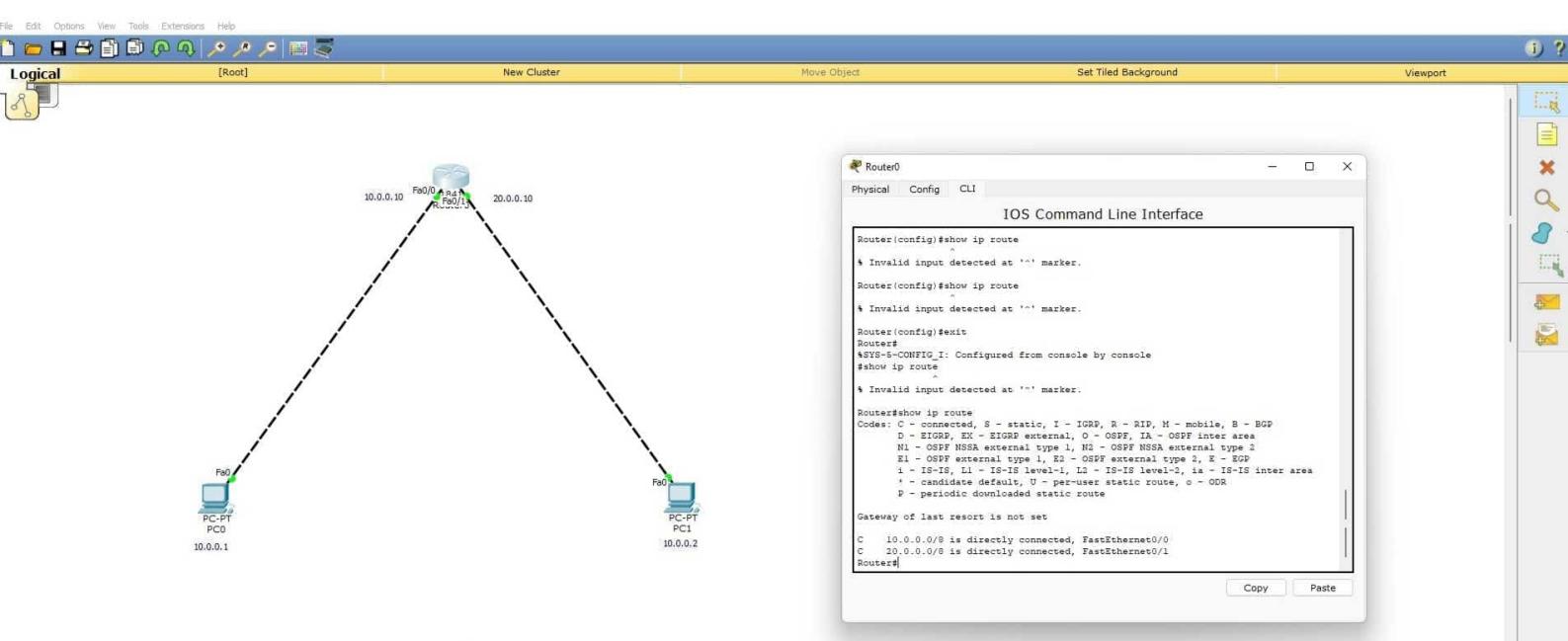
→ Using single router, two PCs

i) Place a generic router and add the generic PCs in your workspace.

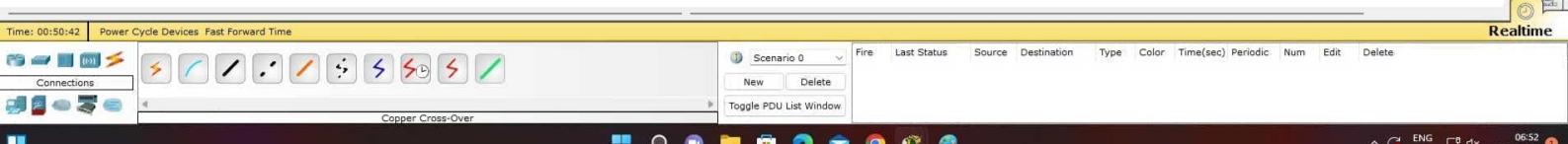
ii) Connect the router and PCs using copper wires.

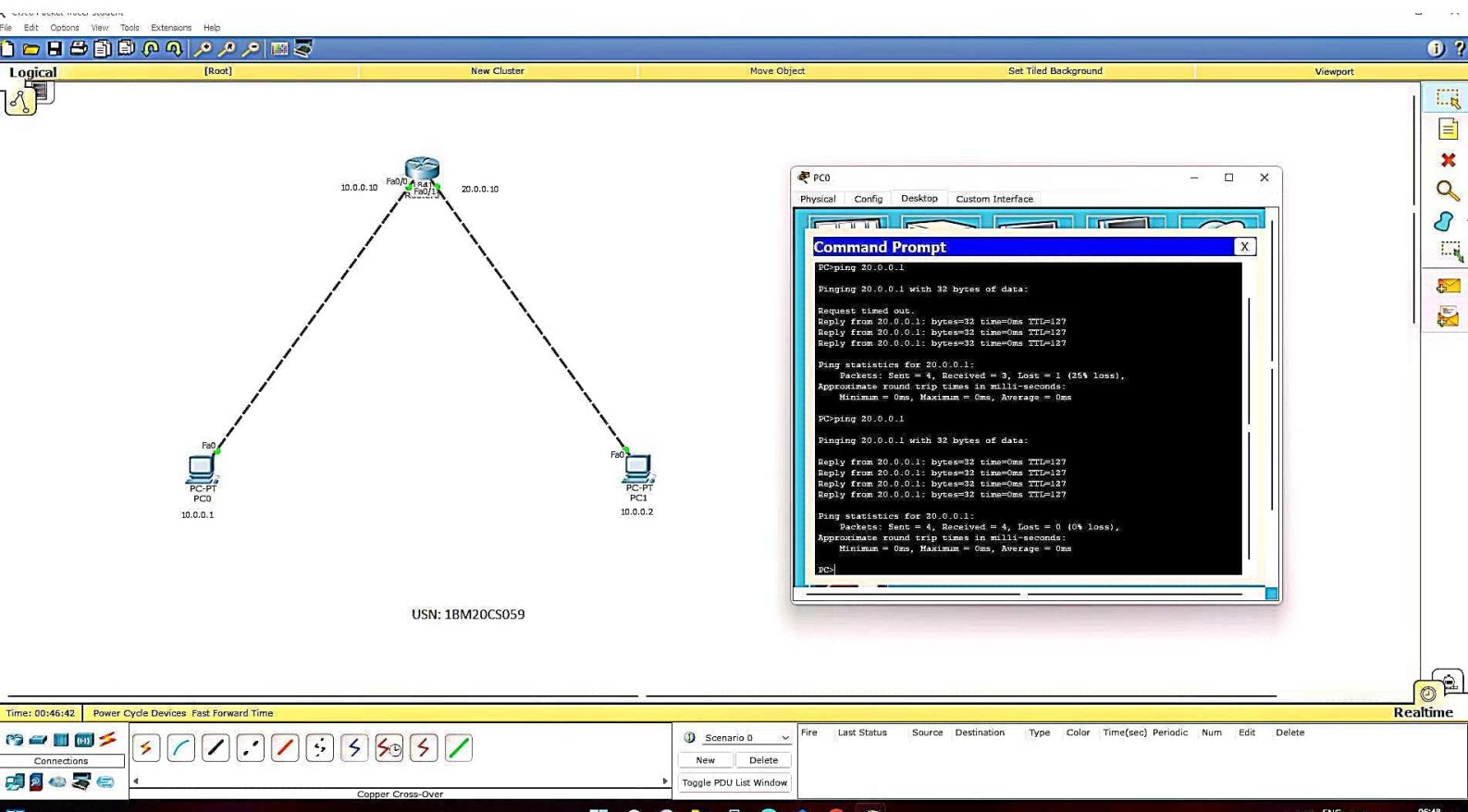
iii) Configure IP address of each PC and in the configuration tab under settings, set gateways for both PCs to the router.

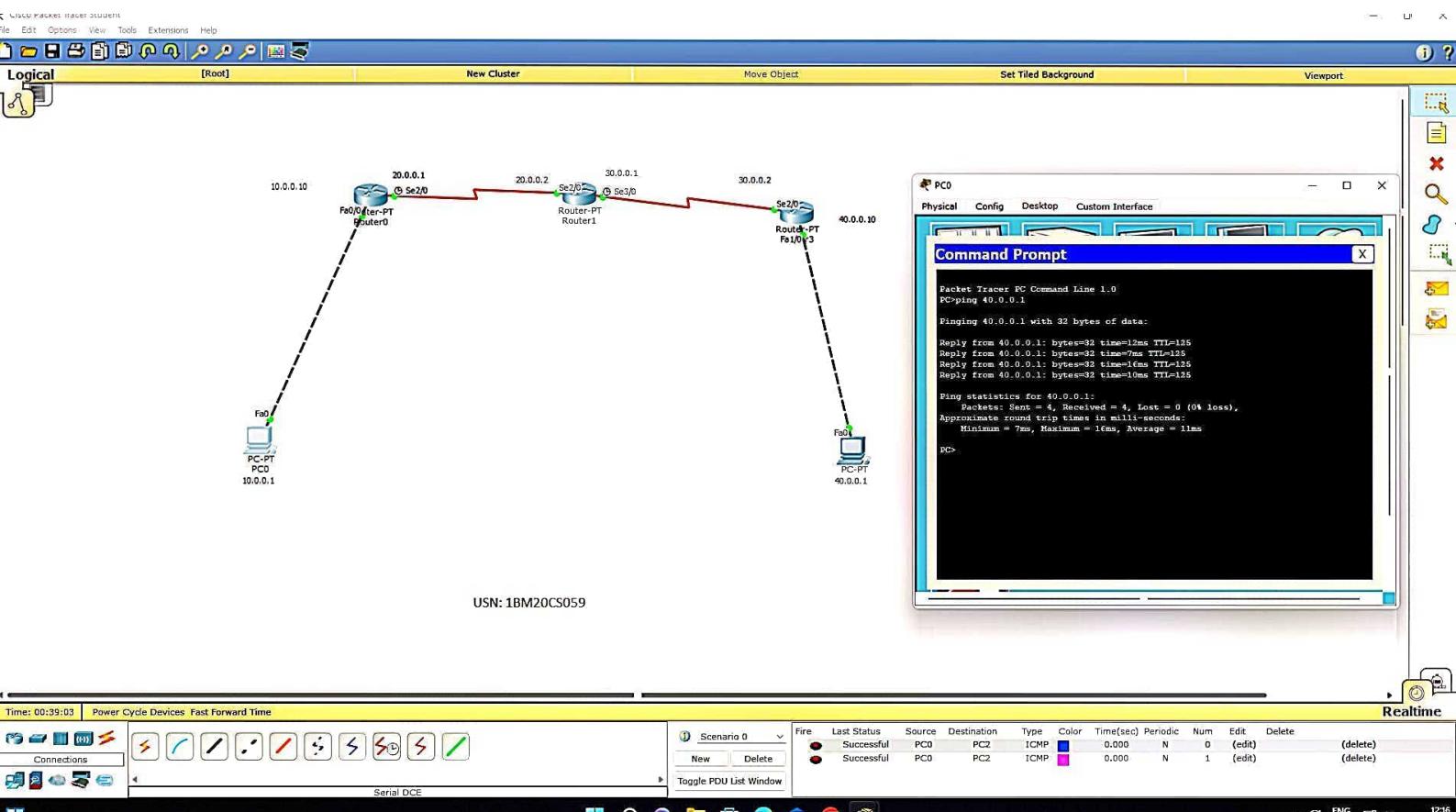
iv) Click on the generic router and go to the CLI

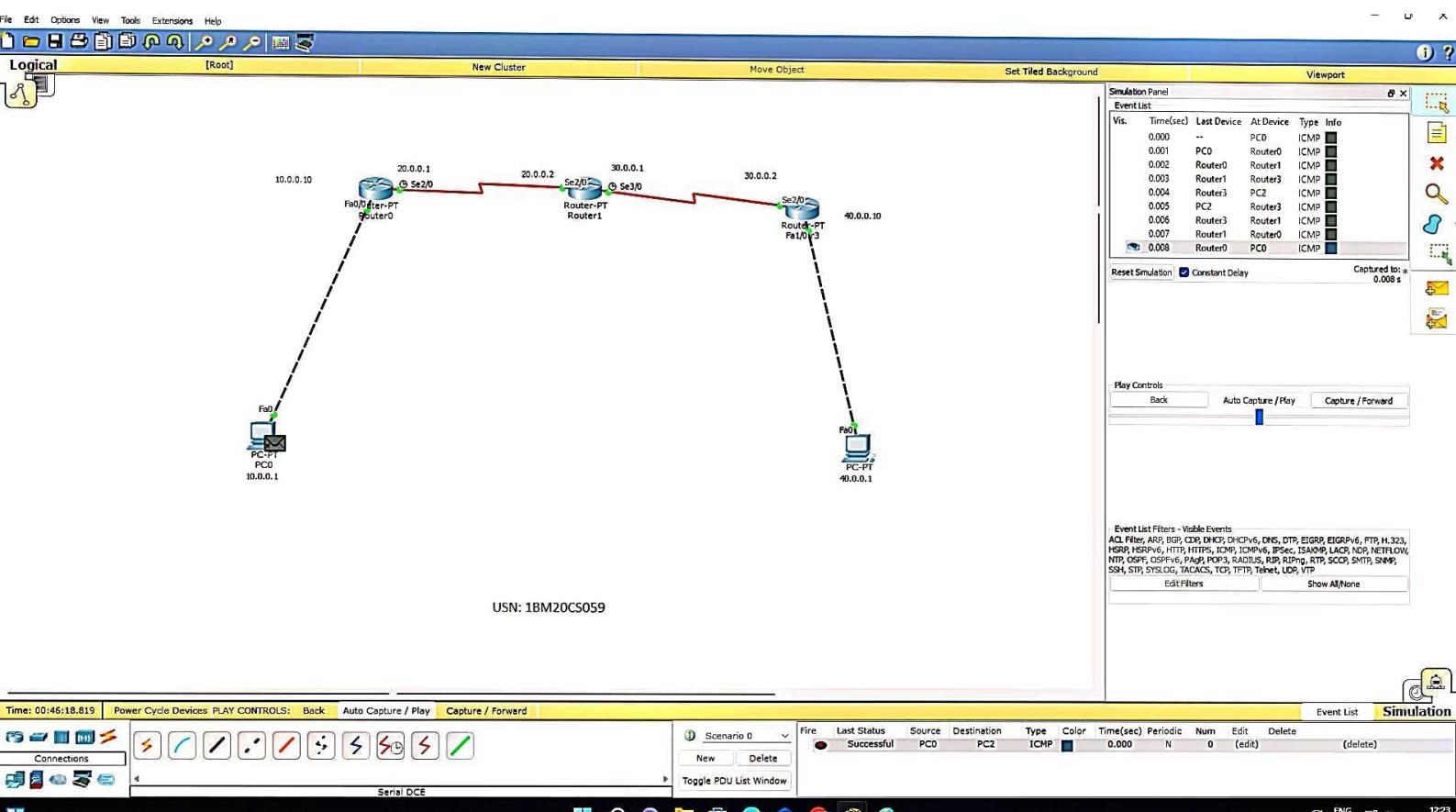


USN: 1BM20CS059









task : Enter the following commands to set up a connection between PCs (1) and generic router through gateway 10.0.0.10.

```

→ No
→ enable
# config t
(config) # interface fastethernet 0/0
(config-if) # ip address 10.0.0.10 255.0.0.0
# no shut
# exit

```

Now to set up connection between PC2 and the router through gateway 20.0.0.10.

```

# interface fastethernet 1/0
# ip address 20.0.0.10 255.0.0.0
# no shut
# exit

```

Once we enter no shut in both times, the amber light between the PC and router turns green indicating that the two devices are ready for use.

Simulation mode: Add a simple PDU by selecting the PCs and click on capture from right panel.

Real time mode: Select the PC you want to send the packet from which is PC0 in our case and open its command prompt from desktop tab. Specify the destination address. A response is sent from destination PC to source PC.

→ Using three routers, two PCs

i) Place 3 generic routers and 2 generic PCs in the workspace.

ii) Place a note for each device (PC and router) and specify the IP address.

iii) Connect the routers and PCs using copper crossover.

iv) Connect the routers using serial DCE.

v) Click on each PC, go to the configure tab, Set the IP address and subnet mask in fastethernet

vi) Next click on one settings in the config tab. Set the gateway as the IP address of the next router [e.g. 10.0.0.10]

vii) IP address of PC and its gateway address should belong to the same network.

For connecting two routers:

Click on Router #10. Go to CLI and enter the following commands:

→ no mod and ser 0/0 in higher mode

→ enable, interface fastethernet 0/0 type fiber

→ config +

→ interface serial 2/0

→ ip address 20.0.0.10 255.0.0.0

→ no shut

Click on router 1, open CLI and enter the following commands.

```

→ no
→ enable
→ config t
→ interface serial 2/0
→ ip address 20.0.0.20 255.0.0.0
→ no shut

```

After this procedure, the red lights between the two routers will now turn green [router 0 and router 1]. Indicating that they are now ready for communication.

For connecting two devices [1 PC and one router].

i) Since IP address of the PC is already configured, go to router.

ii) open CLI for router 0 and enter the following commands.

```

→ no
→ enable
→ config t
→ interface fastethernet 0/0
→ ip address 10.0.0.10 255.0.0.0
→ no shut

```

The red light turns green meaning that the router is ready for communication.

Teaching Router 0 of network 30:

```

→ no
→ enable
→ config t
→ interface serial 2/0
→ ip route 30.0.0.0 255.0.0.0 20.0.0.20
→ exit
→ show ip route
    
```

Teaching Router 1 of network 40:

```

→ no
→ enable
→ config t
→ interface serial 2/0
→ ip route 40.0.0.0 255.0.0.0 20.0.0.20
→ exit
→ show ip route
    
```

Similarly repeat this for router 1 and router 2.

Simulation mode: Add a simple PDU by selecting the PC 2 and click on auto capture from right panel.

Real time mode: Select the PC PC 0 and go to its command prompt and ping the router 0. Once the message has been sent successfully repeat this with routers 1 and 2 as well. Finally, ping PC 1.

- 5 Observation: ~~1) how CSMA/CD works at each node~~
 learning outcome: ~~2) what happens if a node receives a collision~~

→ 1 router:

When PC 0 pings PC 1 for the first time, we get the first packet lost as request timed out.

Now, if we ping PC 1 again from PC 0 we get all 4 packets without any loss. Now reverse pinging of PC 0 from PC 1 will also not lead to any loss, all packets are acknowledged.

→ 3 routers:

Before training the routers, we get the results in destination host unreachable. After training the routers, we get clear statistics on the result.

6 Result:

→ Using 3 routers, three PCs.

c:\> ping -c 1 20.0.0.1

Pinging 20.0.0.1 with 32 bytes of data:

Request timed out.

Reply from 20.0.0.1: bytes = 32 time < 1ms TTL=127

Reply from 20.0.0.1: bytes = 32 time < 1ms TTL=127

Reply from 20.0.0.1: bytes = 32 time < 1ms TTL=127

Ping statistics for 20.0.0.1:

c:\> ping -c 1 20.0.0.1

Pinging 20.0.0.1 with 32 bytes of data:

Reply from 20.0.0.1: bytes = 32 time < 1ms TTL=127

Reply from 20.0.0.1: bytes = 32 time < 1ms TTL=127

Reply from 20.0.0.1 : bytes=32 time<1ms TTL=127.

Reply from 20.0.0.1 : bytes=32 time<1ms TTL=127.

Ping statistics for 20.0.0.1:

packets: sent = 4 received = 4, lost = 0 (0% loss)

→ Using three routers, two PCs:

1. PC > Ping 40.0.0.1 with 32 bytes of data

Pinging 40.0.0.1 with 32 bytes of data

Reply from 10.0.0.10 : destination host unreachable.

Reply from 10.0.0.10 : destination host unreachable.

Reply from 10.0.0.10 : destination host unreachable

Reply from 10.0.0.10 : destination host unreachable

Ping statistics for 40.0.0.1

Packet: sent = 4, received = 0, lost = 4 (100% loss)

2. PC > Ping 10.0.0.10 with 32 bytes of data

Reply from 10.0.0.10 : bytes=32 time=0ms TTL=255

Ping statistics for 10.0.0.10:

Packet: sent = 4, received = 4, lost = 0 (0% lost)

3. PC > Ping 20.0.0.10 with 32 bytes of data

Reply from 20.0.0.10 : bytes=32 time=1ms TTL=255

Reply from 20.0.0.10 : bytes=32 time=1ms TTL=255

Reply from 20.0.0.10 : bytes=32 time=0ms TTL=255

Reply from 20.0.0.10 : bytes=32 time=0ms TTL=255

Ping statistics for 20.0.0.10:

Packets: sent = 4, received = 4, lost = 0 (0% loss)

4. PC > ping 20.0.0.10

Pinging 20.0.0.10 with 32 bytes of data

Reply from 20.0.0.10: bytes = 32 time = 1ms TTL = 254.

Reply from 20.0.0.10: bytes = 32 time = 1ms TTL = 254

Reply from 20.0.0.10: bytes = 32 time = 1ms TTL = 254

Reply from 20.0.0.10: bytes = 32 time = 8ms TTL = 254.

Ping statistics for 20.0.0.10:

Packets: sent = 4, received = 4, lost = 0 (0% loss)

5. PC > ping 40.0.0.1

Pinging 40.0.0.1 with 32 bytes of data

Request timed out

Reply from 40.0.0.1: bytes = 32, time = 10ms TTL = 125

Reply from 40.0.0.1: bytes = 32, time = 13ms TTL = 125

Reply from 40.0.0.1: bytes = 32, time = 8ms TTL = 125.

Ping statistics for 40.0.0.1:

Packets: sent = 4, received = 3, lost = 1 (25% loss)

6. PC > ping 40.0.0.1

Pinging 40.0.0.1 with 32 bytes of data:

✓ Reply from 40.0.0.1: bytes = 32 time = 2ms TTL = 125.

✓ Reply from 40.0.0.1: bytes = 32 time = 24ms TTL = 125.

✓ Reply from 40.0.0.1: bytes = 32 time = 9ms TTL = 125.

✓ Reply from 40.0.0.1: bytes = 32 time = 9ms TTL = 125.

Ping statistics for 40.0.0.1:

Packets: sent = 4, received = 4, lost = 0 (0% loss).