VISVESVARAYATECHNOLOGICALUNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.

LAB REPORT

on

COMPUTER NETWORKS

Submitted by

ASHISH SERU (1BM21CS035)

in partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING

B.M.S. COLLEGE OF ENGINEERING

(Autonomous Institution under VTU)

BENGALURU-560019

JUN-2023 to SEP-2023

B. M. S. College of Engineering,
Bull Temple Road, Bangalore 560019
(Affiliated To Visvesvaraya Technological University, Belgaum)
Department of Computer Science and Engineering

CERTIFICATE

This is to certify that the Lab work entitled “COMPUTER NETWORKS” carried out by ASHISH SERU (1BM21CS035), who is a bonafide student of B. M. S. College of Engineering. It is in partial fulfillment for the award of Bachelor of Engineering in Computer Science and Engineering of the Visvesvaraya Technological University, Belgaum during the year 2023. The Lab report has been approved as it satisfies the academic requirements in respect of a Computer Networks - (22CS4PCCON) work prescribed for the said degree.

M Lakshmi Neelima
Assistant Professor
Departmentof CSE
BMSCE, Bengaluru

Dr. Jyothi S Nayak
Professor and Head
Department of CSE
BMSCE, Bengaluru

Index

WEEK 1

Create a topology and simulate sending a simple PDU from source to destination using hub and switch as connecting devices and demonstrate ping messages.

OBSERVATION:

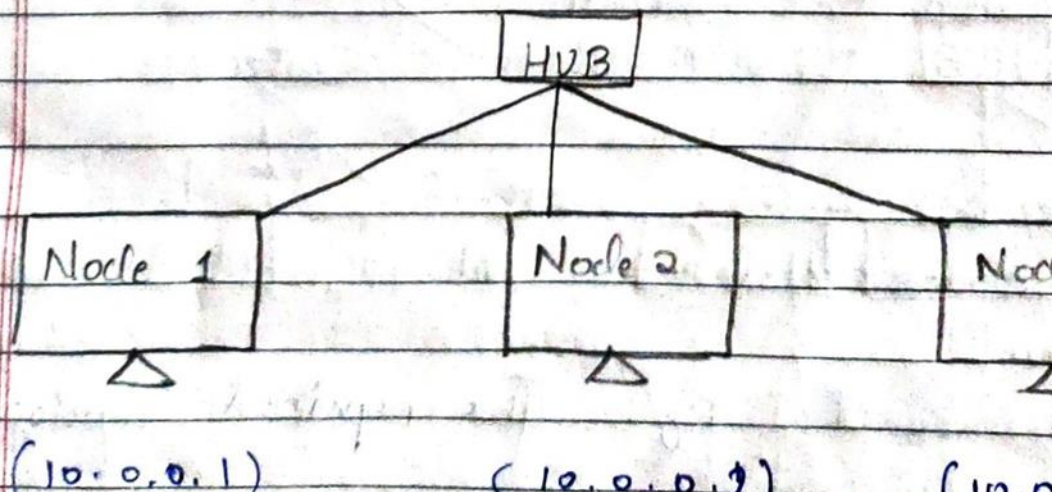
Create a topology and stimulate
a simple PDU from source to
using switch & hub as connec
domains.

→ Aim

To create a topology and stim
sending a simple PDU from sour
destination using a simple HUB and
as connecting device.

Procedure

1. Start creating the topology.
2. Select the HUB in the logical
and connect three PC's to the
3. Set IP address for all 3 PC's
clicking on the PC's individua
choosing the config and entering
IP address.



Output :

1. In first network (HUB),
ping 10.0.0.3 (Pinged PC 1 with PC 3)

→ Pinging 10.0.0.3 with 32 bytes of data

Reply from 10.0.0.3 : bytes = 32 time = 0s TTL = 128
Reply from 10.0.0.3 : bytes = 32 time = 0ms TTL = 128
Reply from 10.0.0.3 : bytes = 32 time = 0ms TTL = 128
Reply from 10.0.0.3 : bytes = 32 time = 0ms TTL = 128

Pinging statistics for 10.0.0.3
Packets : sent = 4, recieved = 4, lost = 0 (0% loss)
Approximate round trips in milliseconds : min = 0ms, max = 0ms, Avg = 0 ms

2. In second network (SWITCH)
Ping 10.0.0.6 (pinging PC 4 to PC 6)
→ Pinging 10.0.0.6 with 32 bytes of data.
Reply from 10.0.0.6 bytes = 32 time = 0ms TTL = 128
Reply from 10.0.0.6 bytes = 32 time = 0ms TTL = 128
Reply from 10.0.0.6 bytes = 32 time = 0ms TTL = 128
Reply from 10.0.0.6 bytes = 32 time = 2ms TTL = 128
Ping statistics for 10.0.0.6
packets sent = 4, recieved = 4, lost = 0 (0% loss)
Approx. round trip times in milli-seconds
minimum = 0 ms, max = 2 ms, Avg = 0 ms.

3. In complete network (both hub & switch)
Ping 10.0.0.6 (pinging PC 1 to PC 6)

→ Pinging 10.0.0.6 with 32 bytes of data.
Reply from 10.0.0.6 bytes = 32 time = 0ms TTL=
Reply from 10.0.0.6 bytes = 32 time = 0ms TTL=
Reply from 10.0.0.6 bytes = 32 time = 0ms TTL=
Reply from 10.0.0.6 bytes = 32 time = 0ms TTL=
Ping statistics for 10.0.0.6
Packets : sent = 4, recieved = 4, lost = 0, 10 % loss
Approximate round trips times in milli-second
minimum = 0ms, max = 0ms, Avg = 0ms.

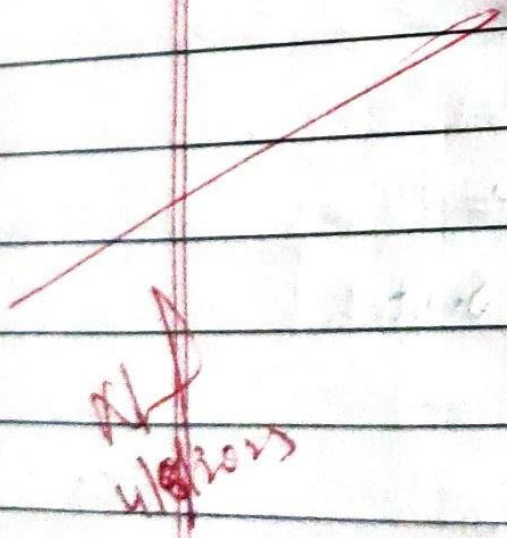
Observation

1. When a packet is sent from PC 1 to PC 3 that are connected to the HUB, it is broadcasted to all the PC, but acknowledgment is recieved from only the addressed PC.

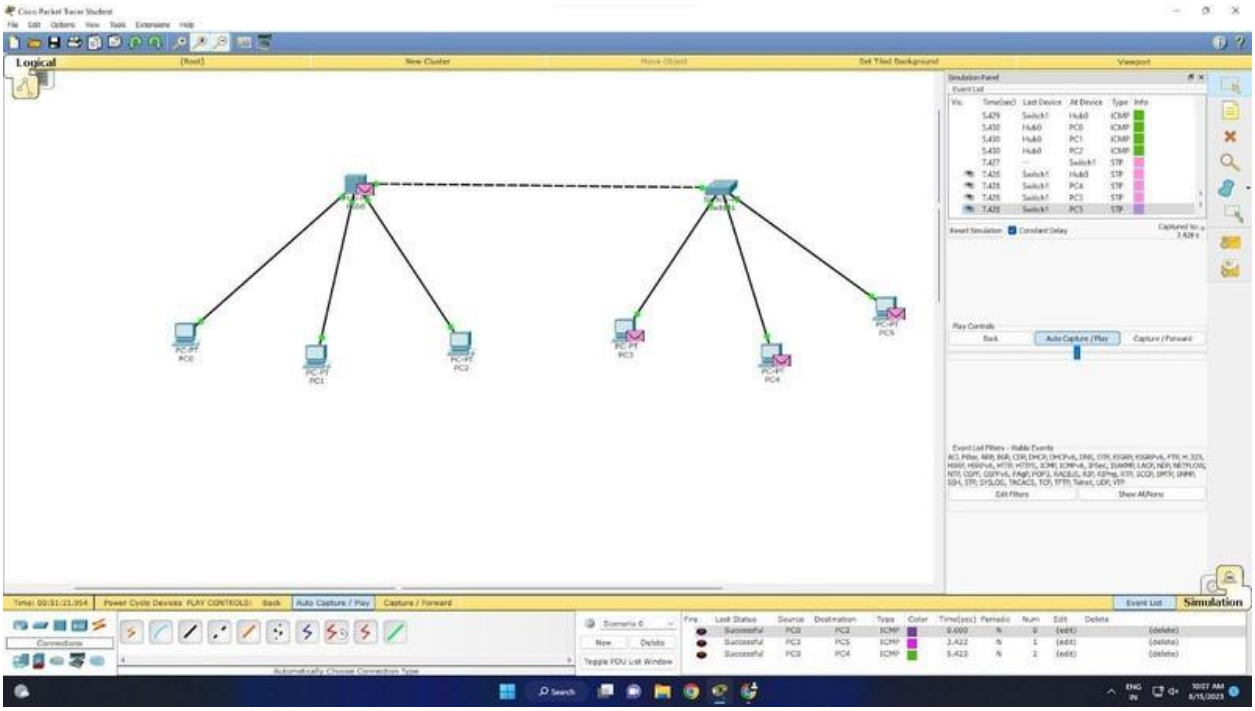
2. When Packet is sent from PC 4 to PC 6 connected to switch at first it is broadcasted to all and acknowledgement is only recieved from the addressed PC but from next time it is only unincasted that is sent to the add PC.

3. When Packet is sent from PC 1 to PC 6 that are connected to HUB and switch, which are also connected to each other the packet is sent to all PC connected to HUB but no acknowledgement is recieved and it is only sent to the addressed PC

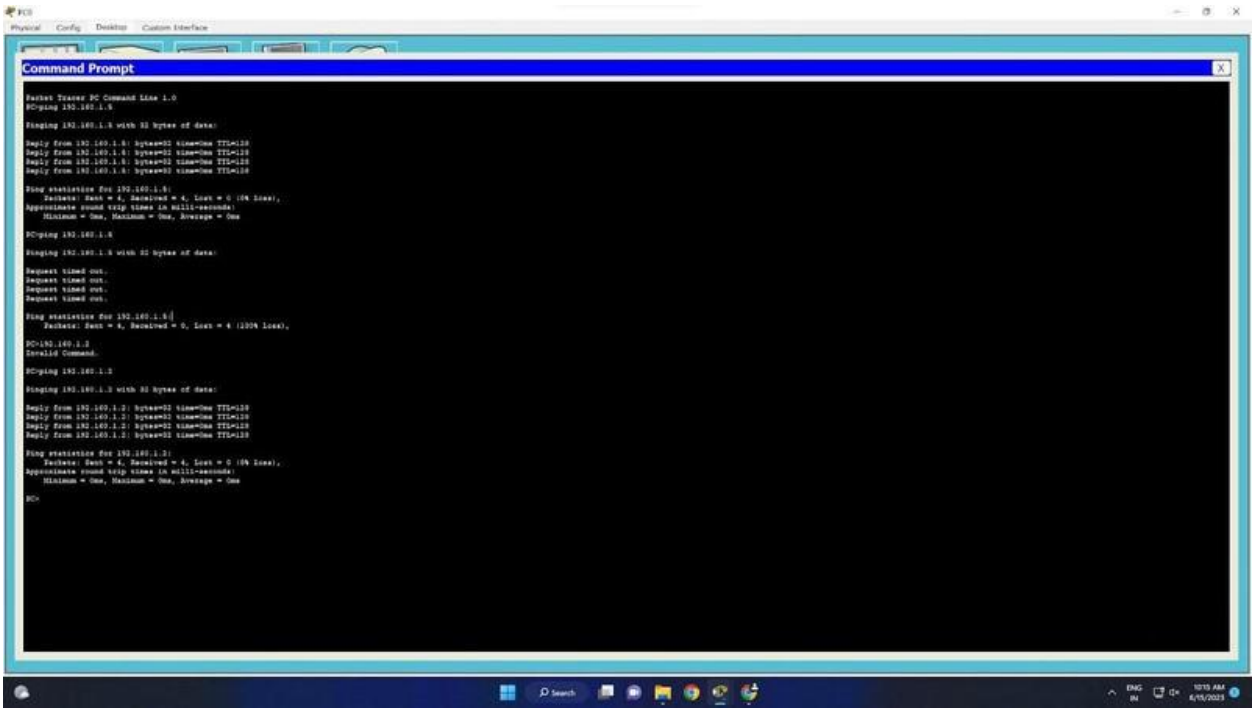
connected through switch and acknowledgement is also recieved only for it.

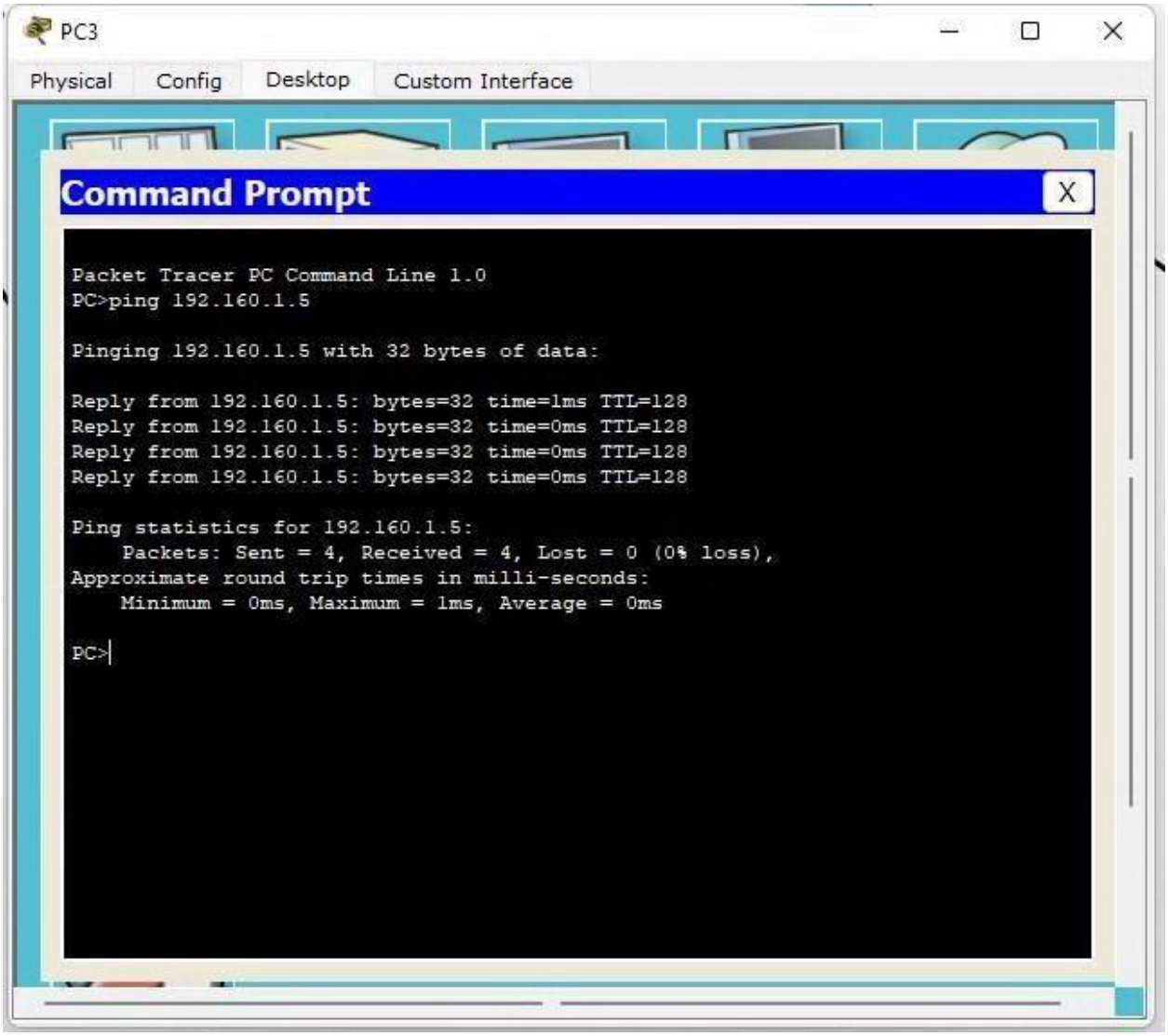


TOPOLOGY:



OUTPUT:




PC3
Physical
Config
Desktop
Custom Interface
Command Prompt
X
Packet Tracer PC Command Line 1.0
PC>ping 192.160.1.5
Pinging 192.160.1.5 with 32 bytes of data:
Reply from 192.160.1.5: bytes=32 time=1ms TTL=128
Reply from 192.160.1.5: bytes=32 time=0ms TTL=128
Reply from 192.160.1.5: bytes=32 time=0ms TTL=128
Reply from 192.160.1.5: bytes=32 time=0ms TTL=128
Ping statistics for 192.160.1.5:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 1ms, Average = 0ms
PC>

WEEK 2

Configure IP address to routers (one and three) in packet tracer. Explore the following messages: ping responses, destination unreachable, request timed out, reply.

OBSERVATION:

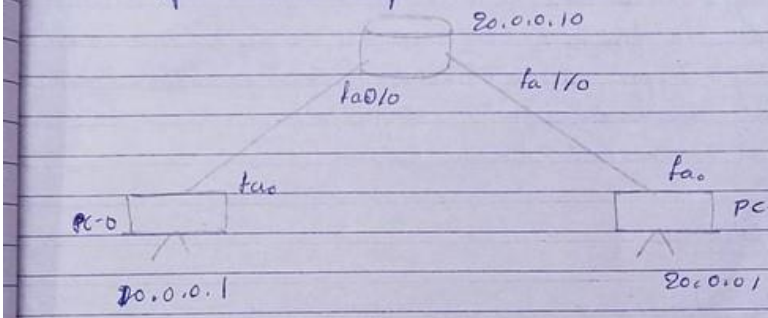
6/13 Lab-2

Program 2.1

Aim: Configure IP address to a single router. Explore the following messages: ping message, destination unreachable request timed out, reply

Topology

experimental setup



Procedure:-

① Select one generic router and 2 generic PC's. Connect the PC's to router using copper cross-over cable.

② Set the IP address of both PC's by clicking on PC & config tab. Along with IP address & gateway in the settings option on config tab

③ To set the IP addresses of router, click on it & go to CLI tab and type the following commands.

Step 1: type no & press enter
Step 2: Type enable & press enter
Step 3: type config T & press enter
Step 4: type interface fast Ethernet 0/0 & press enter
Step 5: type IP address 10.0.0.10 255.0.0.0 & press enter
Step 6: type No shut & press enter

Step 7: type EXIT
Step 8: type interface fast Ethernet 1/0 & press enter
Step 9: type IP address 20.0.0.10 25.0.0.0 & press enter
step 10: type NO shut & press enter
Step 11: type EXIT
Step 12: type Exit
Step 13: type show IP route [for using the connection sta
close the tab 1 & Click on PC to go to command prompt. Type ping 20.0.0.1. to send packets across
Atleast send packets in simulation mode to get a successful transmission

PING output

Packet tracer PC command line 1.0
PC > Ping 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=0ms TTL=12
Reply from 20.0.0.1 : bytes=32 time=0ms TTL=12
Reply from 20.0.0.1 : bytes=32 time=10ms TTL=12

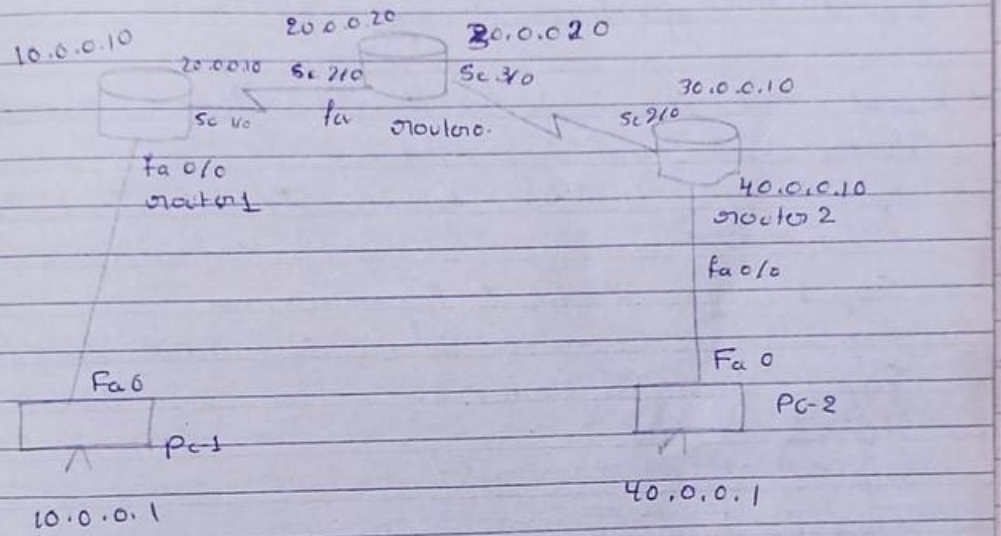
Ping statistics for 20.0.0.1

Packets : Sent=4, Recieved=3, Lost=1 (25% Loss)
Approximate round trip times in milliseconds
minimum=0ms, maximum=10ms Average=3

Program 2.2

Aim:- Configure IP address to three routers in packet tracer. Explore the following messages, ping response, destination unreachable, request timed out, reply

Topology



Procedure:-

① Connect 2 PC's & 3 routers using copper cross over cable for PC to router & Serial DCE cable to connect to routers to routers

② Set the IP address of both PC's and gateway numbers

③ Now for setting IP address & gateway number to routers select one router & perform following command.

Step 1 - type no & press enter

Step 2 - type Enable & press enter

Step 3 - type config T & press enter.

Step 4: type interface fast ethernet 0/0 & press enter

Step 5: type IP address 10.0.0.10 255.0.0.0 & press enter

Step 6: type No shut & press enter

Step 7: type Exit
Step 8: type interface 8 c 2/0 & press enter
Step 9: type IP address 20.0.0.1c 255.0.0.0 & press enter
Step 10: type no shut & press enter
Step 11: type Exit
Step 12: type Exit

+ Repeat these commands for the other two routers with respective IP/gateway address
* Now to introduce the other two IP address to the first router we type following commands.
Step 1: type config T & press enter
Step 2: type IP route 30.0.0 255.0.0.0 20.0.0.20
Step 3: type IP route 40.0.0 255.0.0 20.0.0.20
Step 4: Exit
Step 5: Exit
Step 6: type Show IP route.
* Repeat these steps for other 2 routers with appropriate addresses
* Go to command prompt by clicking on PC & config tab. Type Ping message to send packets to the destination address

PING OUTPUT
output-1
Packet tracer PC command line 1.0
PC > Ping 40.0.0.1
Pinging 40.0.0.1 with 32 bytes of data.
Reply from 10.0.0.10: Destination host unreachable.

Request timed out
Ping statistics for 40.0.0.1
Packets: sent = 4, Received = 0, los 4 (100% loss)

Output 2

Packet trace PC command line 1.0
PC > Ping 10.0.0.1
Pinging 10.0.0.1 with 32 bytes of data:
Reply from 10.0.0.1: bytes = 32 time = 2ms TTL = 128

Ping statistics for 10.0.0.1:
Packet: Sent = 4, Received = 4 lost = 0 (0% loss)
Approximate round trip times in milliseconds
minimum = 2ms, Maximum = 8ms Average = 3ms

Observation

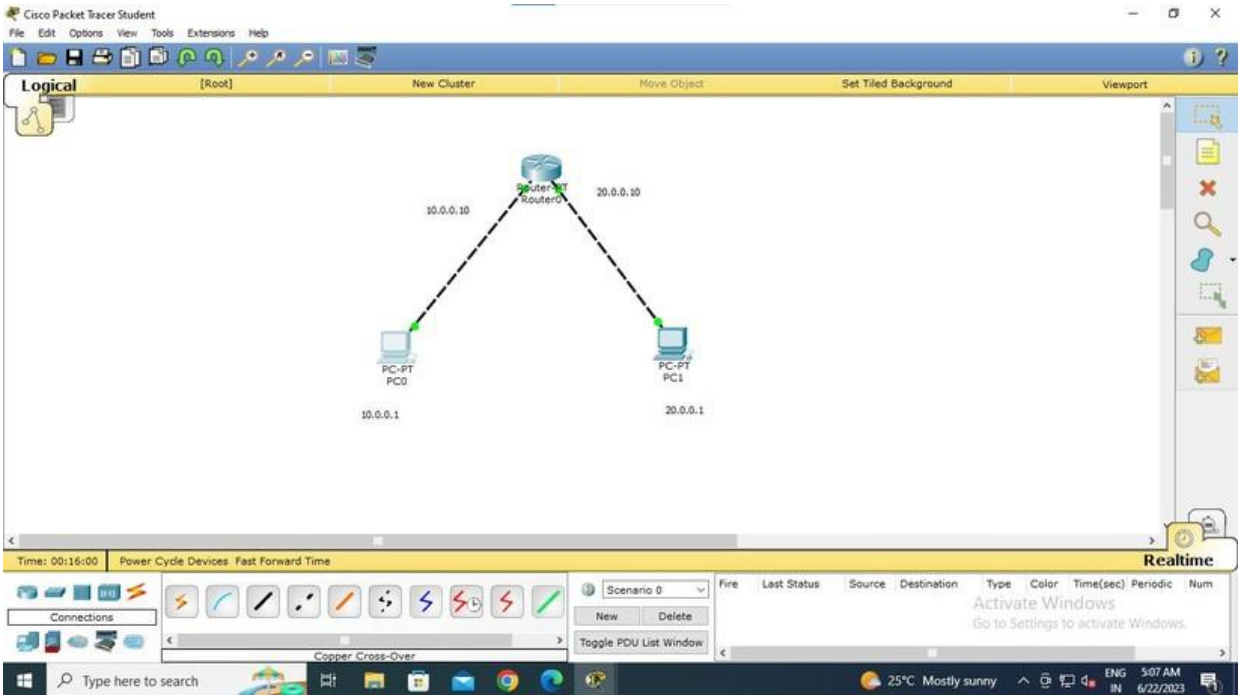
* In program 2.1 when we ping destination address we get allocated with 32 bytes. In this first 8 bytes are used to trace about the routes & their addresses. Rest bytes are used for sending packets to destination address: Then again if we ping all bytes are used for message sending and there will be no timed out message.

* In program 2.2 when the routes doesn't know about the remaining address, and we ping a message we get host unreachable message. Once the routers have access / knowledge about other addressed message will be sent successfully.

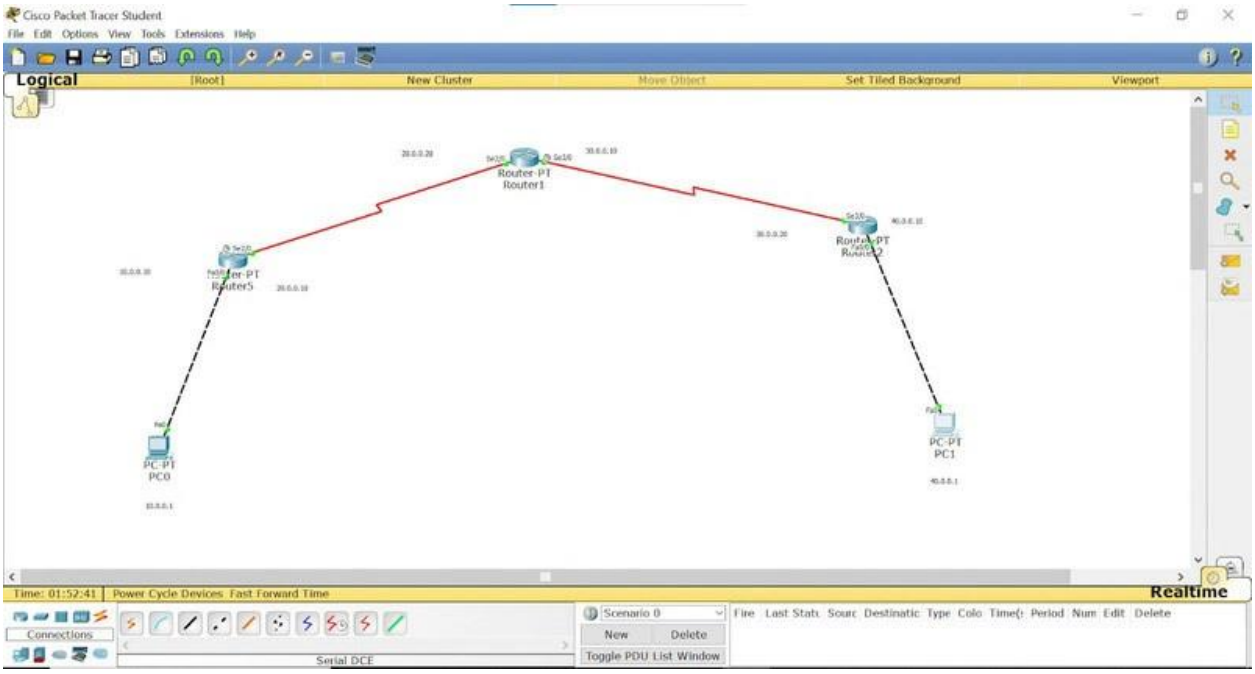
13/4/2023

TOPOLOGY:

PROGRAM 2.1

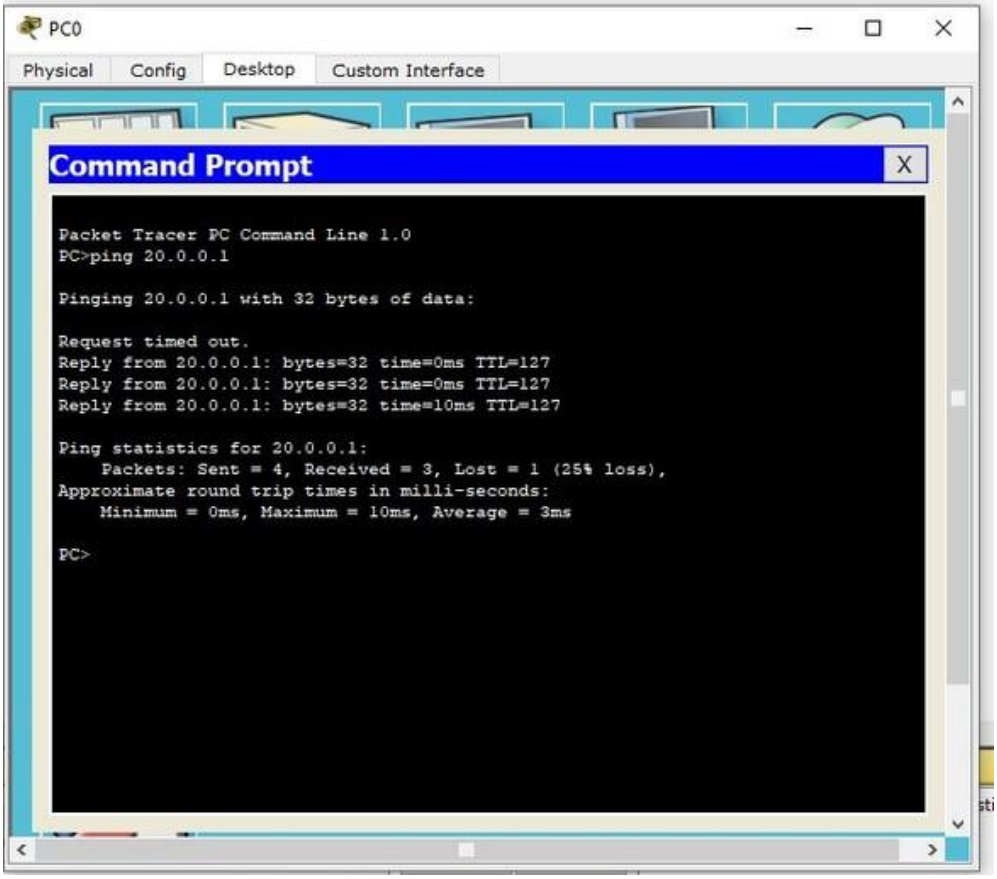


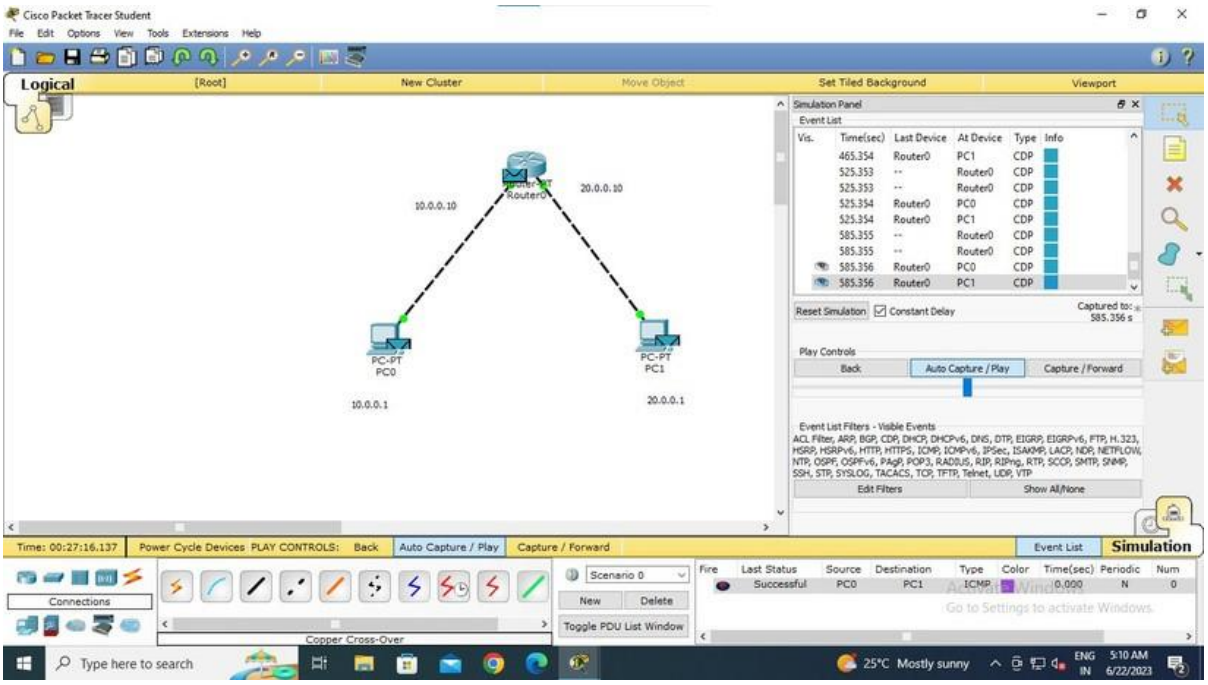
PROGRAM 2.2



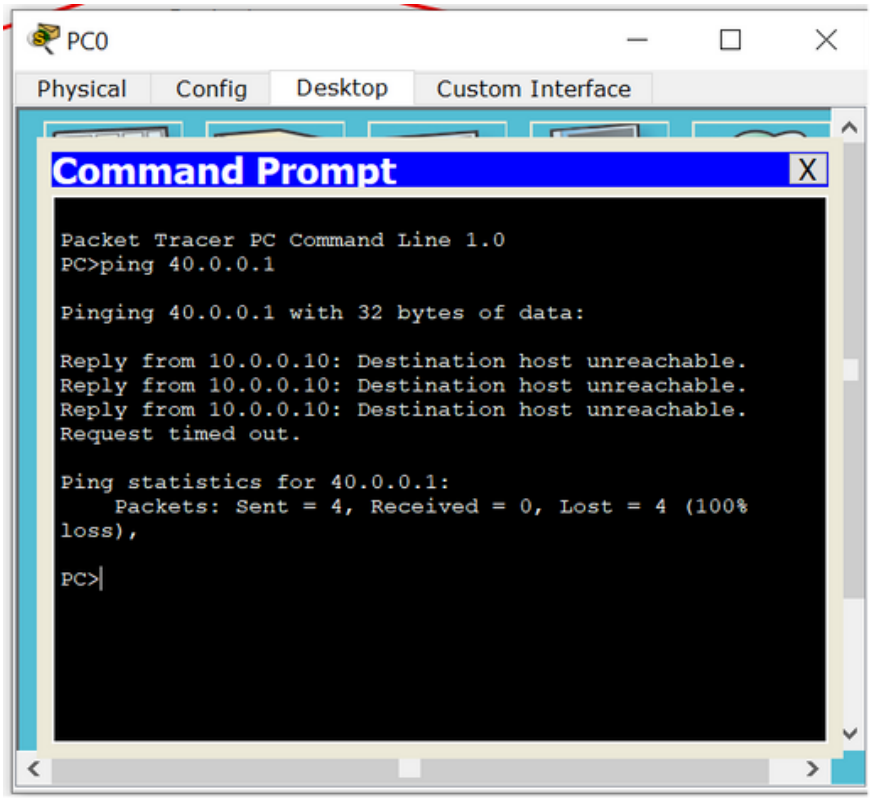
OUTPUT:

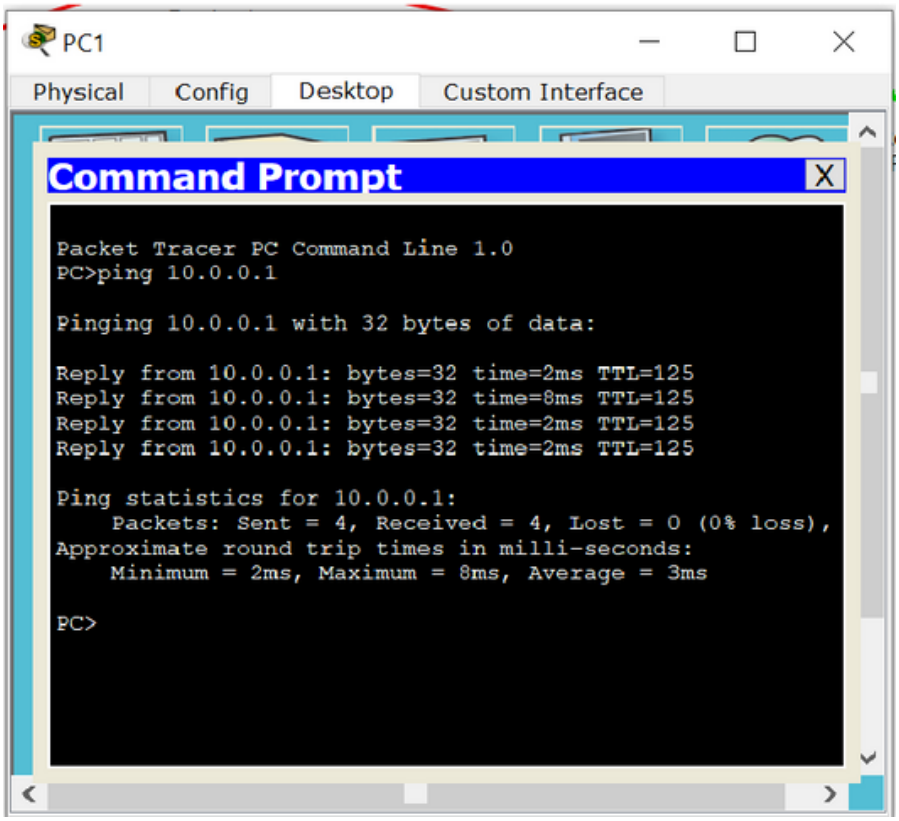
PROGRAM 2.1

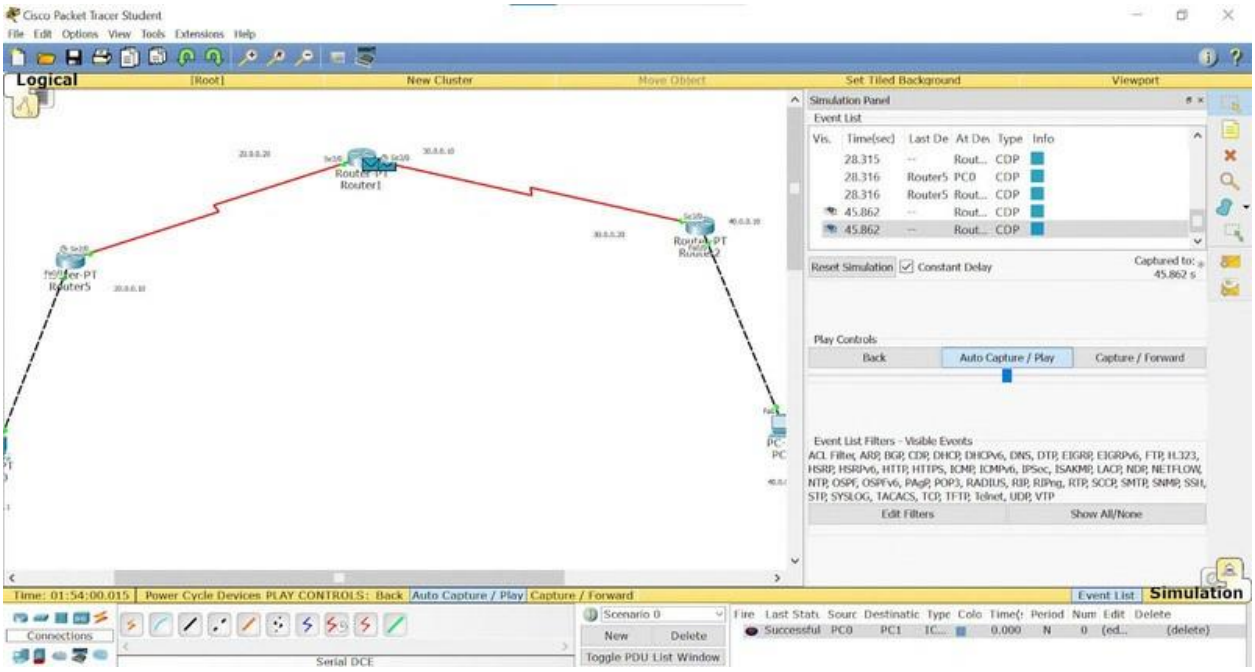




PROGRAM 2.2

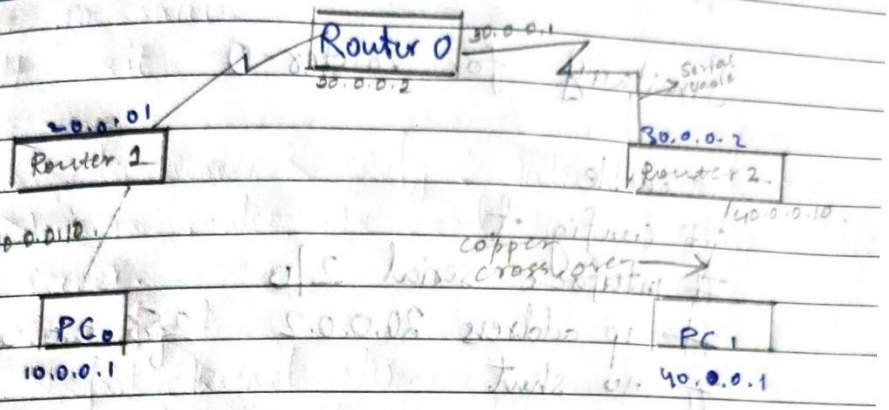

PC0
Physical
Config
Desktop
Custom Interface
Command Prompt
Packet Tracer PC Command Line 1.0
PC>ping 40.0.0.1
Pinging 40.0.0.1 with 32 bytes of data:
Reply from 10.0.0.10: Destination host unreachable.
Reply from 10.0.0.10: Destination host unreachable.
Reply from 10.0.0.10: Destination host unreachable.
Request timed out.
Ping statistics for 40.0.0.1:
Packets: Sent = 4, Received = 0, Lost = 4 (100%
loss),
PC>


PC1
Physical
Config
Desktop
Custom Interface
Command Prompt
Packet Tracer PC Command Line 1.0
PC>ping 10.0.0.1
Pinging 10.0.0.1 with 32 bytes of data:
Reply from 10.0.0.1: bytes=32 time=2ms TTL=125
Reply from 10.0.0.1: bytes=32 time=8ms TTL=125
Reply from 10.0.0.1: bytes=32 time=2ms TTL=125
Reply from 10.0.0.1: bytes=32 time=2ms TTL=125
Ping statistics for 10.0.0.1:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 2ms, Maximum = 8ms, Average = 3ms
PC>


Cisco Packet Tracer Student
File Edit Options View Tools Extensions Help
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Router-PT
Router1
Router-PT
Router5
Router-PT
Router2
PC-PT
PC
Simulation Panel
Event List
Vis. Time(sec) Last De At Des Type Info
28.315 -- Rout... CDP
28.316 Router5 PC0 CDP
28.316 Router5 Rout... CDP
45.862 -- Rout... CDP
45.862 -- Rout... CDP
Reset Simulation
Constant Delay
Captured to: 45.862 s
Play Controls
Back
Auto Capture / Play
Capture / Forward
Event List Filters - Visible Events
ACL Filter, ARP, BGP, CDP, DHCP, DHCPv6, DNS, DTP, EIGRP, EIGRPv6, FTP, H.323, HSRP, HSRPv6, HTTP, HTTPS, ICMP, ICMPv6, IPSec, ISAKMP, LACP, NDP, NETFLOW, NTP, OSPF, OSPFv6, PAgP, POP3, RADIUS, RIP, RIPng, RTP, SCCP, SMTP, SNMP, SSH, STP, SYSLOG, TACACS, TCP, TFTP, Telnet, UDP, VTP
Edit Filters
Show All/None
Time: 01:54:00.015
Power Cycle Devices PLAY CONTROLS: Back
Auto Capture / Play
Capture / Forward
Event List
Simulation
Connections
Serial DCE
Scenario 0
New
Delete
Toggle PDU List Window
Fire Last Stat Sourc Destinatic Type Colo Time(s Period Num Edit Delete
Successful PC0 PC1 IC... 0.000 N 0 (ed... (delete)

Aim: Config default and static route for a connection of routers.

Topology:



Procedure

1. Select 3 generic routers and two PC's as end devices. Connect the PCs to different routers, with copper cross over and connect both routers to main router with serial cables.
2. Set the IP addresses of PC and gateways.
3. Set gateway addresses in all the routers taking interface as fastethernet for the PCs and serial for routers.
4. Connect PCs to interfaces.
5. Config - steps

> enable (router 1)
config t
interface fastethernet 0/0
ip address 10.0.0.10 255.0.0.0
no shut

WEEK 3

Configure default route, static route to the

Router. OBSERVATION:

exit
interface serial 2/0
ip address 20.0.0.1 255.0.0.0
no shut
exit

Similarly for router-0

> enable
config t
interface serial 2/0
ip address 20.0.0.2 255.0.0.0
no shut
exit
interface serial 3/0
ip address 30.0.0.1 255.0.0.0
no shut
exit

For router-2

> enable
config t
interface fastethernet 0/0
ip address 40.0.0.10 255.0.0.0
no shut
exit
interface serial 2/0
ip address 30.0.0.2 255.0.0.0
no shut
exit

6) We need to set IP routes for all routes via routers.
For router 1 and router-2, we do default routing and for router-0, static routing is done.

For router-1
config t
ip route 0.0.0.0 0.0.0.0 20.0.0.2
no shut
exit
Show ip rout
C 10.0.0.0/8 is directly connected. FastEthernet 0/0
C 10.0.0.0/8 is directly connected. Serial 2/0
S* 00.0.0/0 [1/0] via 20.0.0.2

Similarly for Router-2
config t
ip route 0.0.0.0 0.0.0.0 30.0.0.1
exit
Show ip rout

For router-0 (static routing)
config t
ip rout 10.0.0.0 255.0.0.0 20.0.0.0
ip rout 40.0.0.0 25.0.0.0 30.0.0.0
exit
Show ip rout
S 10.0.0.0/8 [1/0] via 20.0.0.0
C 20.0.0.0/8 is directly connected, serial 2/0
C 30.0.0.0/8 is directly connected, serial 3/0
S 40.0.0.0/8 [1/0] via 30.0.0.0

7) Now, we ping 10.0.0.1 from the command prompt of 40.0.0.1

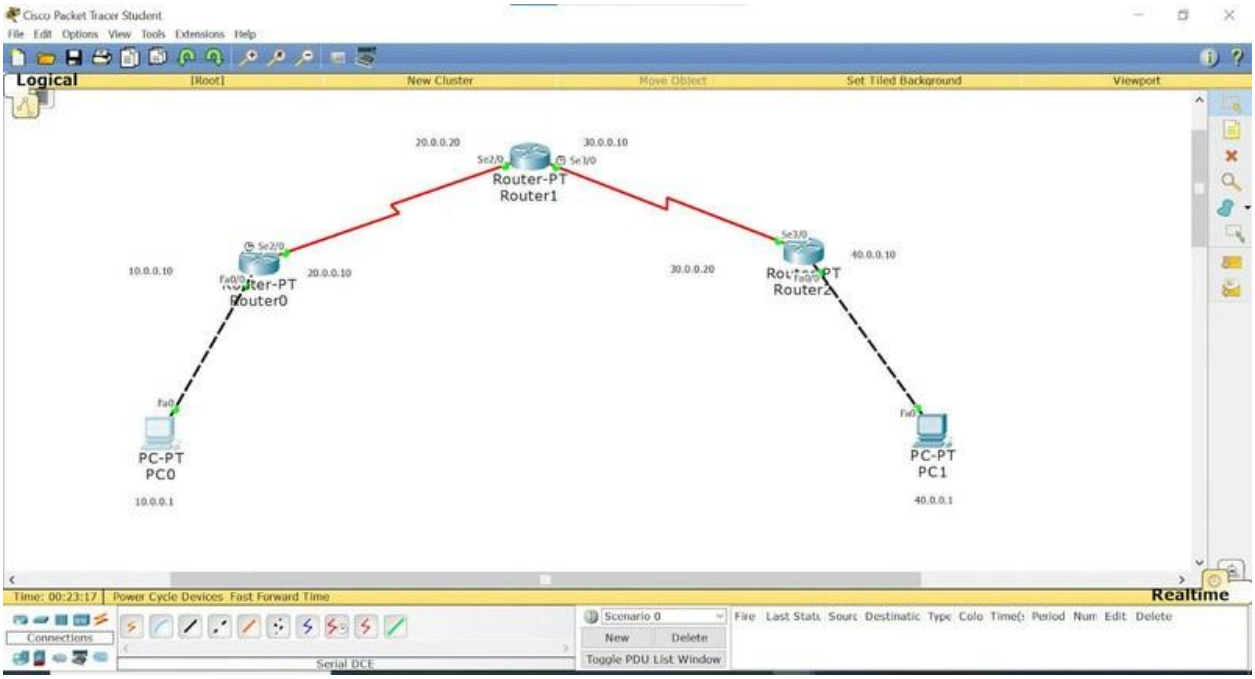
Request timed out
Reply from 40.0.0.1 bytes=32 time=0ms TTL=
Reply from 40.0.0.1 bytes=32 time=2ms TTL=
Reply from 40.0.0.1 bytes=32 time=2ms TTL=
Ping status for 40.0.0.1
Packets send = 4, Recieved = 3, Lost = 1 (125%

Observation

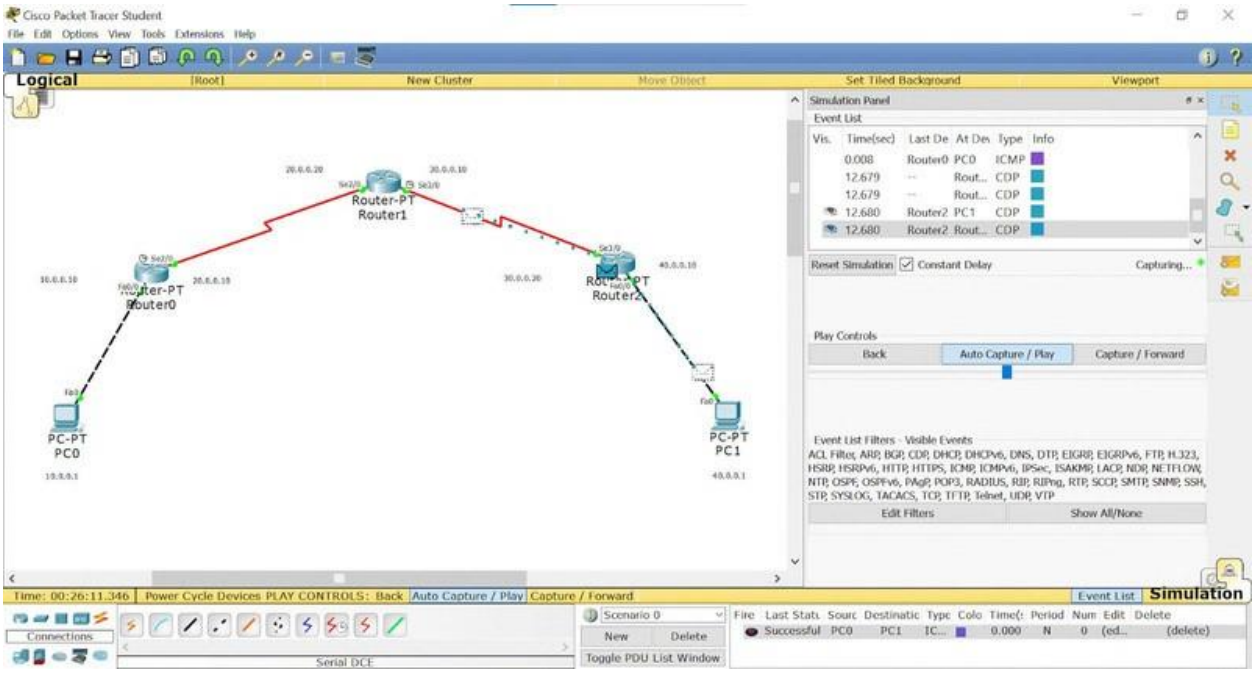
Through default routing, we configure network

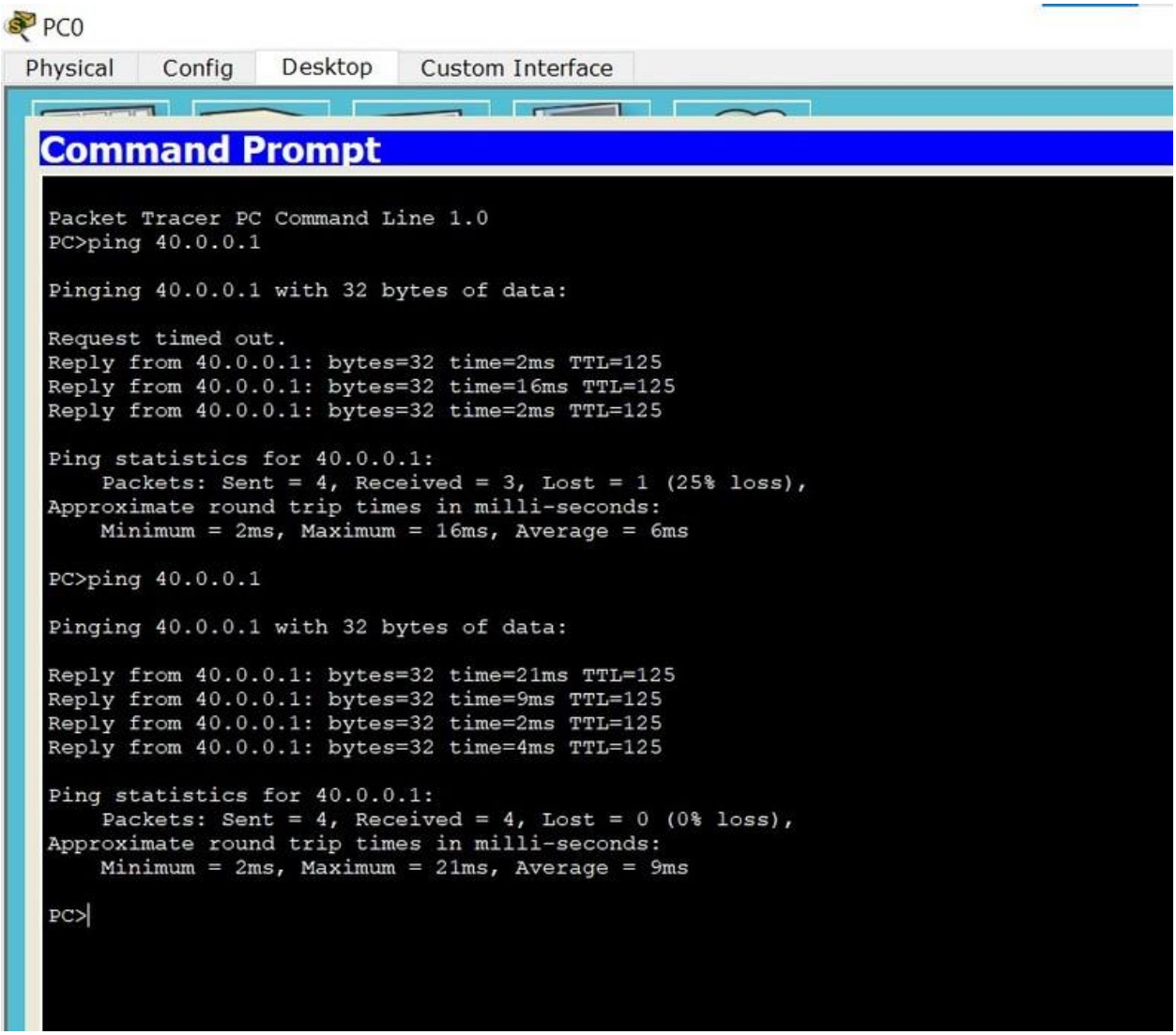
4/9/2023

TOPOLOGY:



OUTPUT:




PC0
Physical
Config
Desktop
Custom Interface
Command Prompt
Packet Tracer PC Command Line 1.0
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=2ms TTL=125
Reply from 40.0.0.1: bytes=32 time=16ms TTL=125
Reply from 40.0.0.1: bytes=32 time=2ms TTL=125
Ping statistics for 40.0.0.1:
Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
Minimum = 2ms, Maximum = 16ms, Average = 6ms
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=21ms TTL=125
Reply from 40.0.0.1: bytes=32 time=9ms TTL=125
Reply from 40.0.0.1: bytes=32 time=2ms TTL=125
Reply from 40.0.0.1: bytes=32 time=4ms TTL=125
Ping statistics for 40.0.0.1:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 2ms, Maximum = 21ms, Average = 9ms
PC>

WEEK 4

Configure DHCP within a LAN and outside

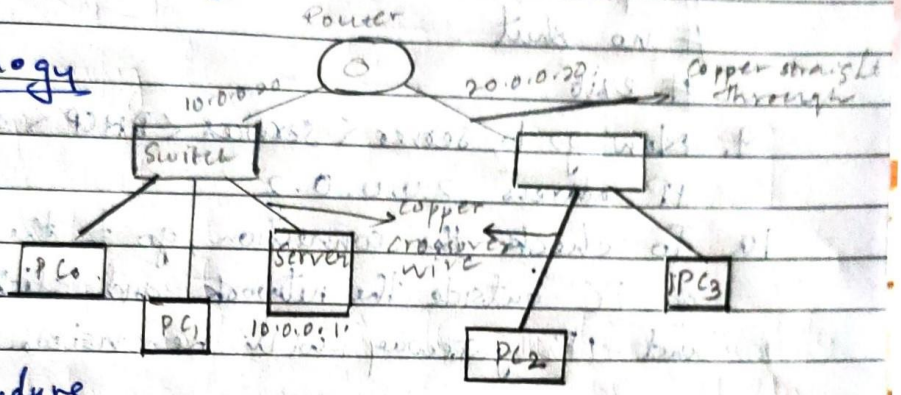
LAN. OBSERVATION:

Configure DHCP within LAN and outside the LAN

Aim

Connection of server LAN within and outside the network using switches and routers.

Topology



Procedure

1. Select two more PC and a server connecting to switch and another network with only end devices and switch
2. Connect both switches to router.
3. Set IP address of server as 10.0.0.1
4. Now go to services < select DHCP < save current IP address 20.0.0.2
5. Now check the IP address of other devices in the network in IP configuration in desktop.
6. Now in CLI of router, enable following:

```
>enable
#config t
#interface fastethernet 4/0
#ipaddress 10.0.0.10 25.0.0.0
#no shut
#exit
#interface fastethernet 0/0
#ip address 20.0.0.20 255.0.0.0
#no shut
#exit
```

7. Go to server < config < gateway 10.0.0.20
8. Now in router, we need to set up ipaddress of server
config t
fast ethernet 0/0
ip helper address 10.0.0.1
no shut
exit
9. Now go to server < services < DHCP < add no IP address 20.0.0.2
10. To check the connection, go to the IP config of PC outside the network and click on DHCP and IP gateway will be visible.

Result

From server from PC2 to PC0 whose IP address is 10.0.0.2.

PC > ping 10.0.0.2

pinging 10.0.0.2 with 32 bytes of data:

request timed out.

Reply from 10.0.0.2 bytes=32 time=6ms TTL=125

10.0.0.2 bytes=32 time=2ms TTL=125

10.0.0.2 time=12ms TTL=125

Ping Statistics for 10.0.0.2, Packet sent = 4, Recieved=3, Lost=1

Approximate round trip time in milliseconds:

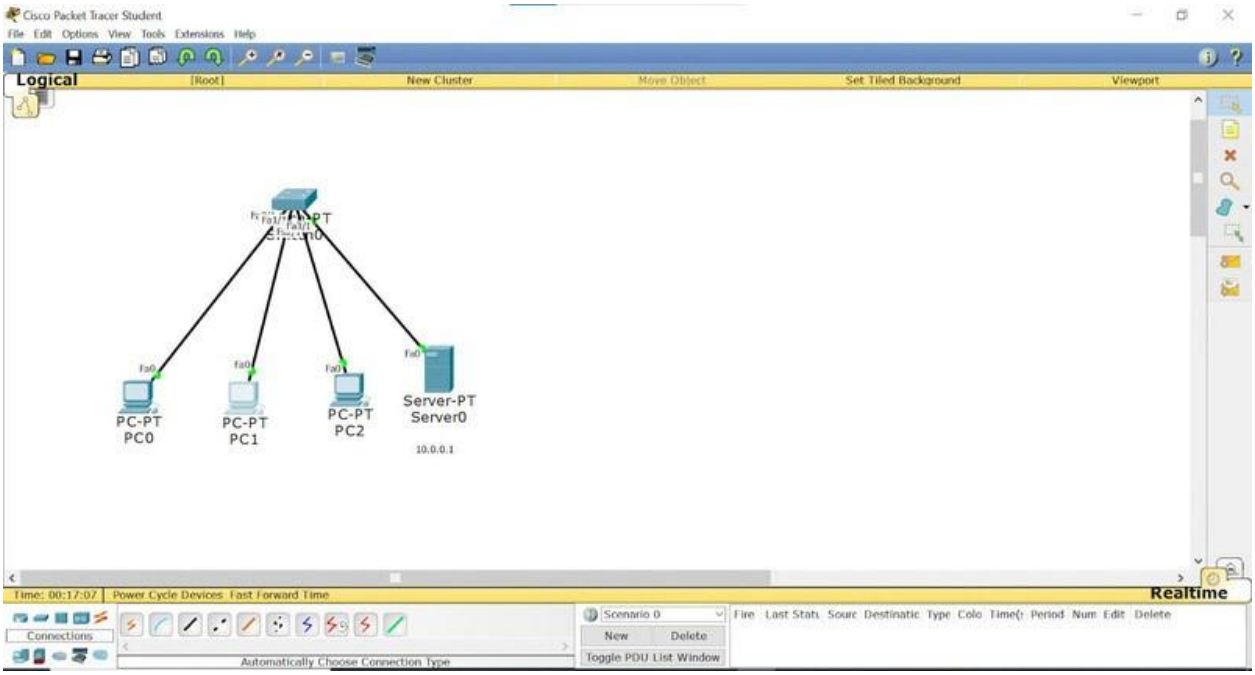
Minimum = 2ms, Maximum = 12ms, Average = 6ms

Observation

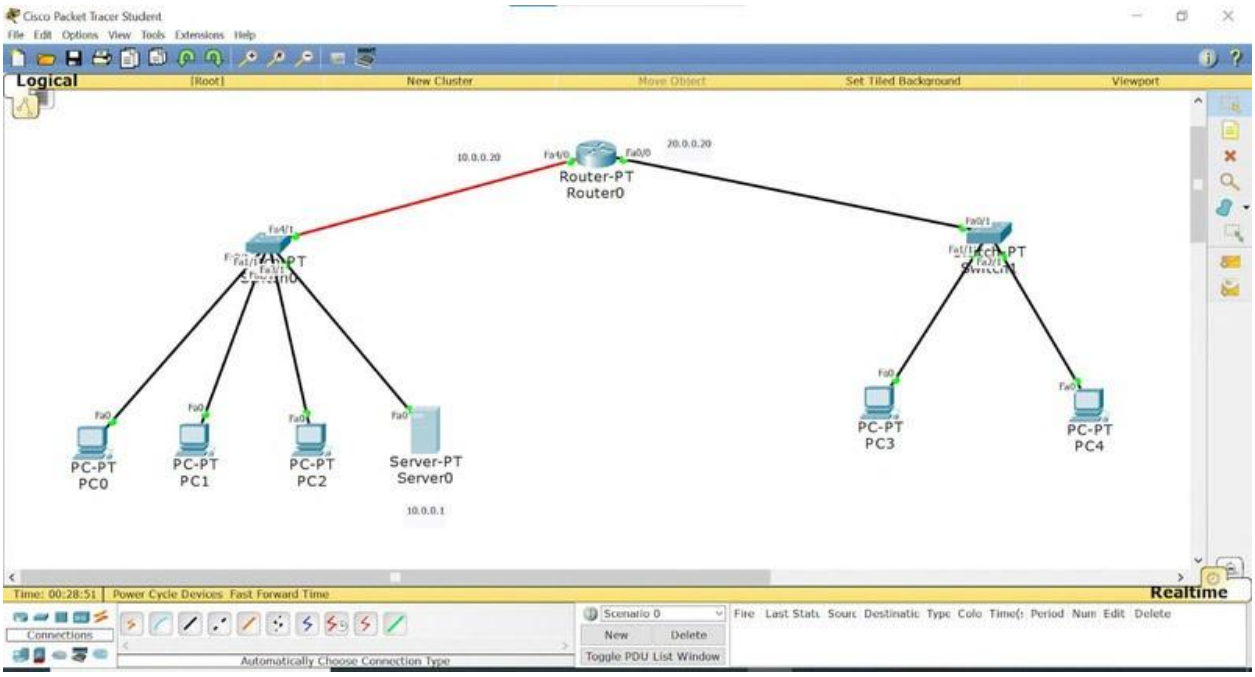
IP address of PC2 & PC3 are also automatically set by server 0. IP address of PC2 is to 10.0.0.2 and PC3 to 20.0.0.3. We could successfully ping PC2 to PC0 without loss.

TOPOLOGY:

PROGRAM 4.1:

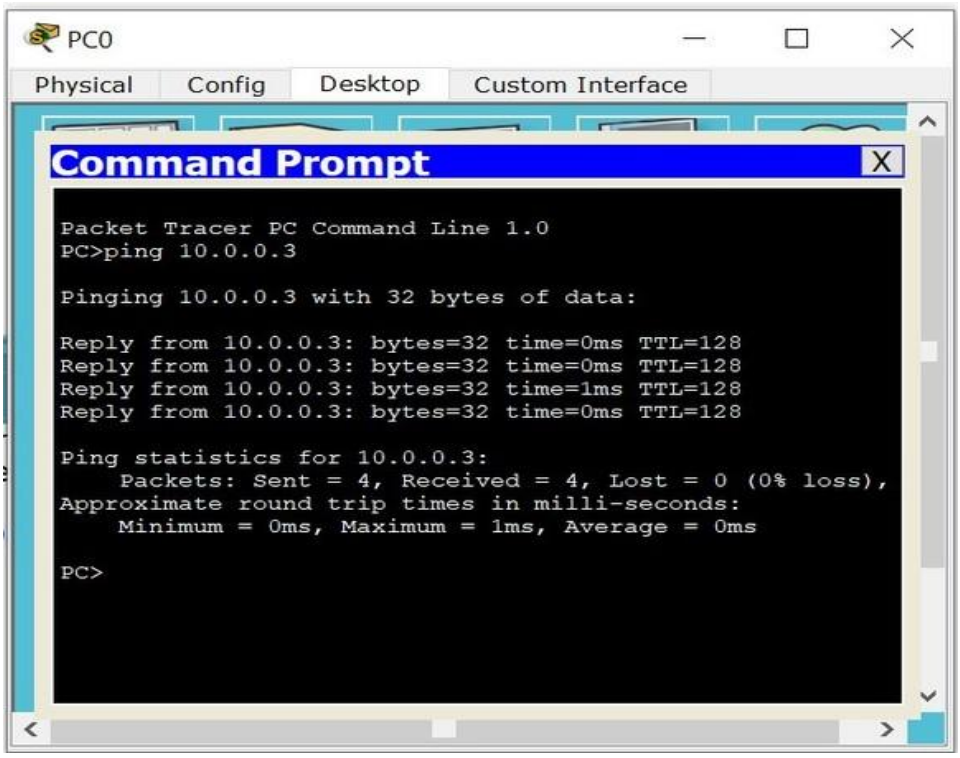


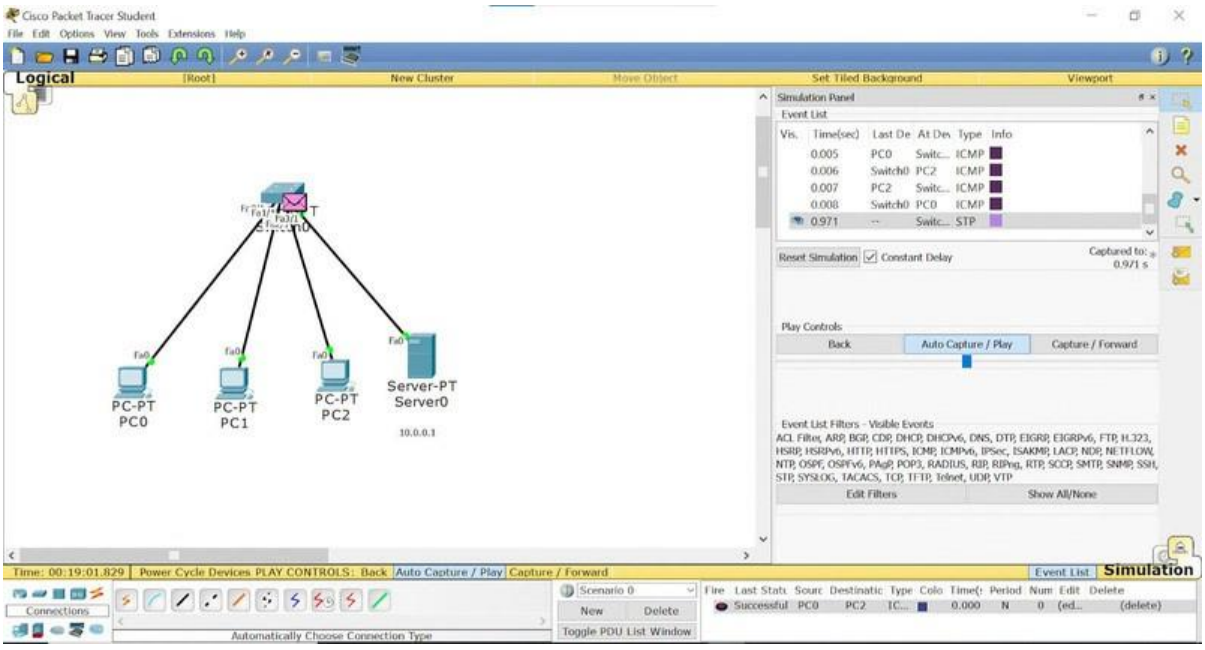
PROGRAM 4.2:



OUTPUT:

PROGRAM 4.1:


PC0
Physical
Config
Desktop
Custom Interface
Command Prompt
Packet Tracer PC Command Line 1.0
PC>ping 10.0.0.3
Pinging 10.0.0.3 with 32 bytes of data:
Reply from 10.0.0.3: bytes=32 time=0ms TTL=128
Reply from 10.0.0.3: bytes=32 time=0ms TTL=128
Reply from 10.0.0.3: bytes=32 time=1ms TTL=128
Reply from 10.0.0.3: bytes=32 time=0ms TTL=128
Ping statistics for 10.0.0.3:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 1ms, Average = 0ms
PC>


Cisco Packet Tracer Student
Logical
PC-PT PC0
PC-PT PC1
PC-PT PC2
Server-PT Server0
10.0.0.1
Simulation

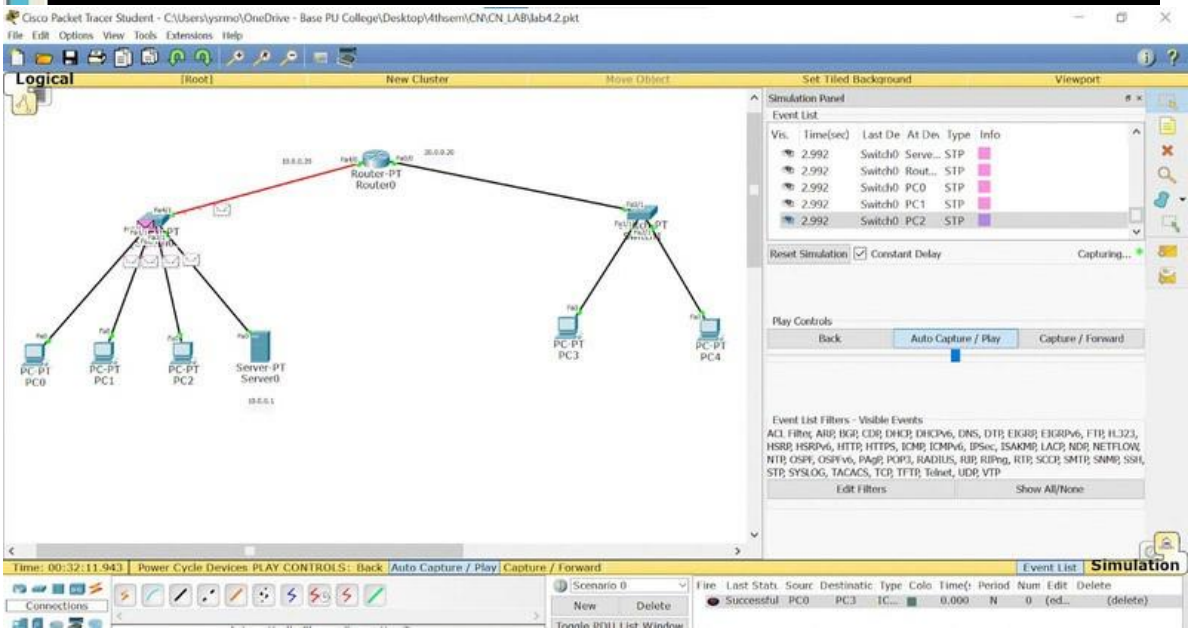
PROGRAM 4.2:

PC0

Physical | Config | Desktop | Custom Interface

Command Prompt

```
Packet Tracer PC Command Line 1.0
PC>ping 20.0.0.2

Pinging 20.0.0.2 with 32 bytes of data:

Request timed out.
Reply from 20.0.0.2: bytes=32 time=0ms TTL=127
Reply from 20.0.0.2: bytes=32 time=0ms TTL=127
Reply from 20.0.0.2: bytes=32 time=0ms TTL=127

Ping statistics for 20.0.0.2:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

PC>ping 20.0.0.3

Pinging 20.0.0.3 with 32 bytes of data:

Request timed out.
Reply from 20.0.0.3: bytes=32 time=0ms TTL=127
Reply from 20.0.0.3: bytes=32 time=0ms TTL=127
Reply from 20.0.0.3: bytes=32 time=0ms TTL=127

Ping statistics for 20.0.0.3:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

PC>
```



WEEK 5

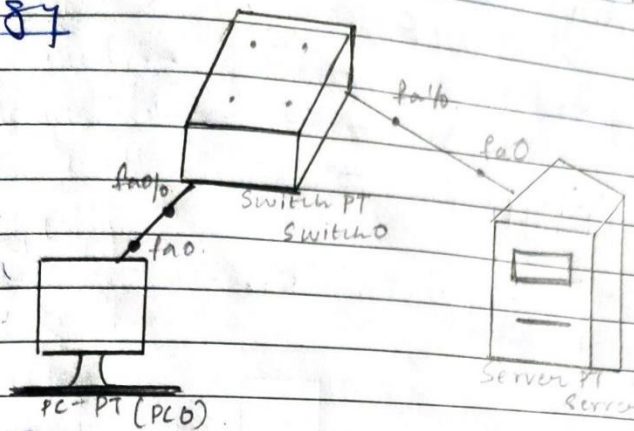
Configure Web Server, DNS within a

LAN. OBSERVATION:

Bafna Gold
Date: Page:

Configure webserver, DNS within LAN.

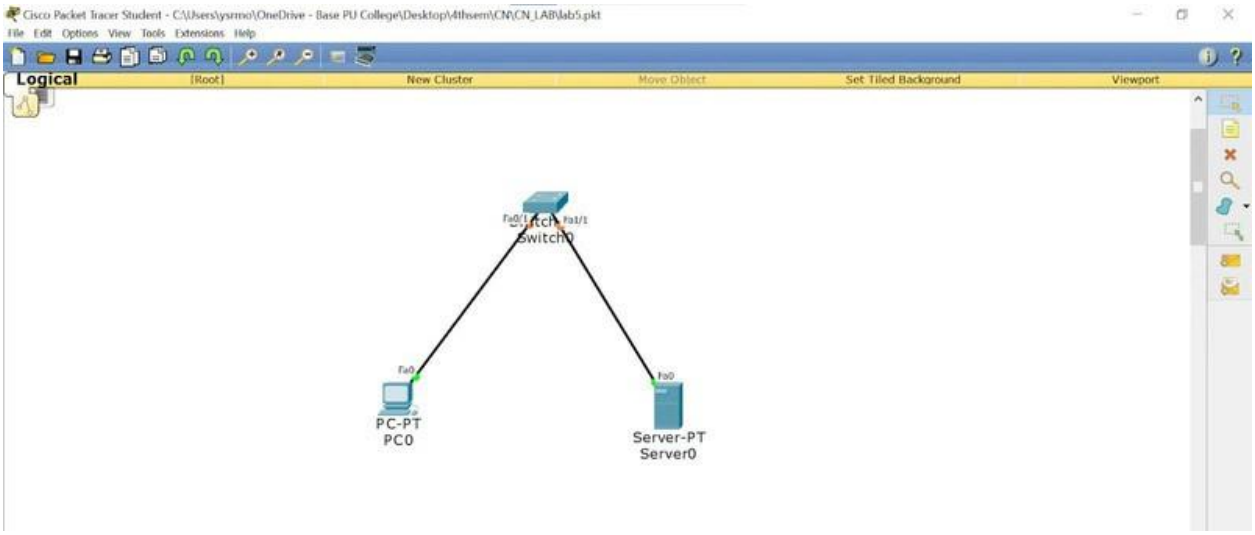
Topology



Procedure

1. Connect a switch, PC and a server to form a LAN.
2. Set PC's IP address by checking on it and go to config, in fastethernet, set IP address as 10.0.0.1 and subnet mask.
3. Set server IP address as 10.0.0.2 and subnet mask respectively.
4. Go to PC's desktop and click on web browser's in the URL type 10.0.0.2 you will get a default display.
5. To make a CV here, we need to make changes in server services.
6. Go to server → Services → HTTP → index.html
7. Go to PC → desktop → webbrowser and type 10.0.0.2, you see the CV or content that is changed.
8. Next, go to server → Services → DNS and switch on IP address as 10.0.0.2. Press add and save it.
9. Again go back to PC → desktop → webbrowser & type the given domain name. Here we can see the CV which had been created earlier.

TOPOLOGY:


Logical
Switch0
PC-PT
PC0
Server-PT
Server0

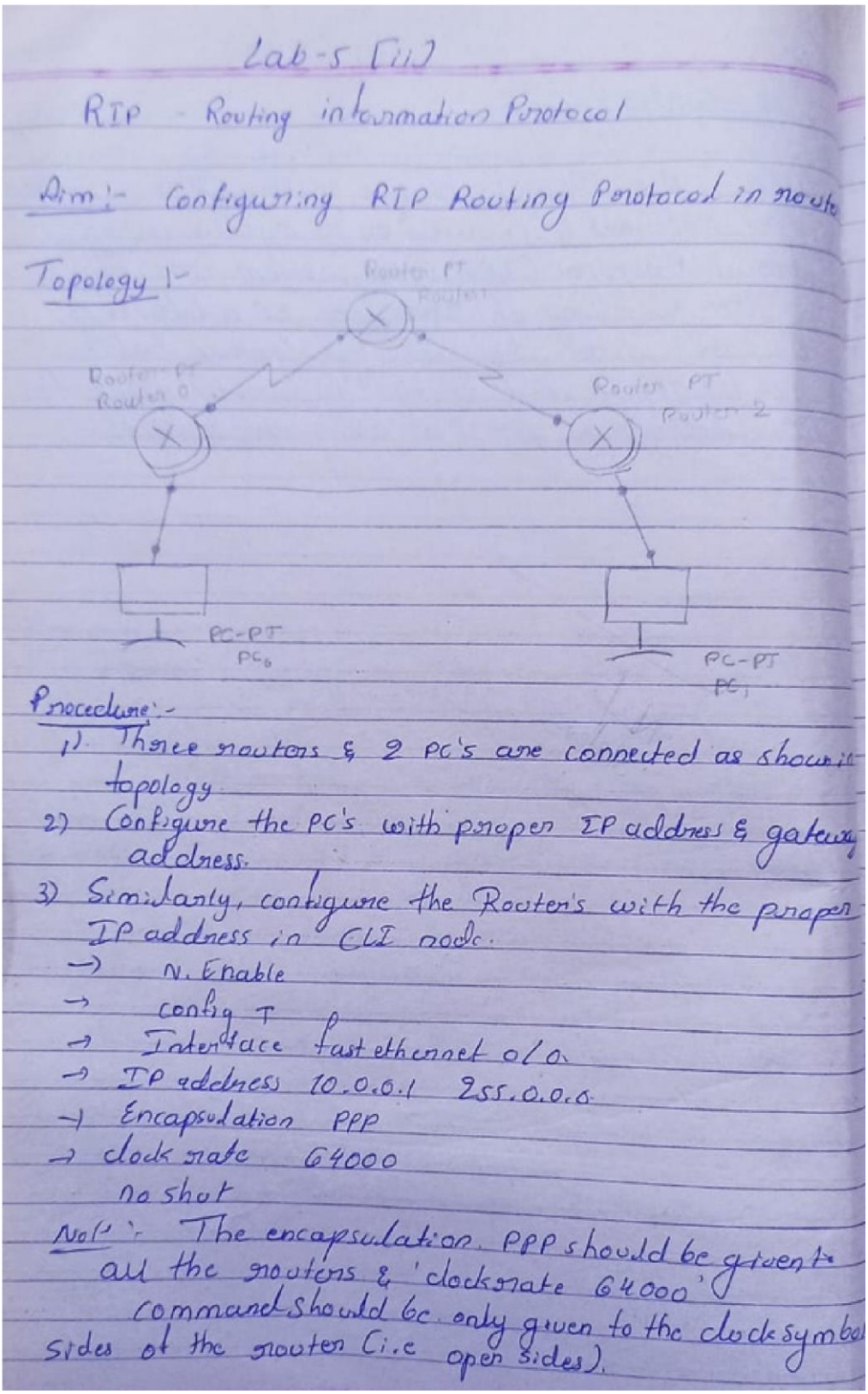
OUTPUT:


PC0
Physical
Config
Desktop
Custom Interface
Web Browser
X
URL
http://chinmay
Go
Stop
Hi welcome to DNS

WEEK 6

Configure RIP routing Protocol in

Routers. OBSERVATION:



Lab-5 [ii]

RIP - Routing information Protocol

Aim:- Configuring RIP Routing Protocol in router

Topology:-

Procedure:-

1) Three routers & 2 PC's are connected as shown in topology.
2) Configure the PC's with proper IP address & gateway address.
3) Similarly, configure the Router's with the proper IP address in CLI mode.
   - → en. Enable
   - → config T
   - → Interface fastethernet 0/0
   - → IP address 10.0.0.1 255.0.0.0
   - → Encapsulation PPP
   - → clock rate 64000
   - no shut

Note:- The encapsulation PPP should be given to all the routers & 'clockrate 64000' command should be only given to the clock symbol sides of the router (i.e open sides).

→ For making the routers to know about the others devices, in the previous 2 experiments we used a static & the other with dynamic address but here we use a Routing protocol Algorithm that itself makes the router to know other devices

→ router rip
→ network 20.0.0.0 } router 2
→ network 30.0.0.0 }

→ router rip
→ network 30.0.0.0 } router 3
→ network 40.0.0.0 }

→ router rip
→ network 10.0.0.0 } router 1
→ network 20.0.0.0 }

Ping output

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:0ms TTL:128
Reply from 40.0.0.1 : bytes=32 time:0ms TTL:128
Reply from 40.0.0.1 : bytes=32 time:0ms TTL:128
Reply from 40.0.0.1 : bytes=32 time:0ms TTL:128

Ping statistics from 40.0.0.1

Packets sent=4 Received 4 Lost=0 (0% Loss)
Approximate round trip times in ms
Minimum=0ms, maximum:0ms Average=0ms

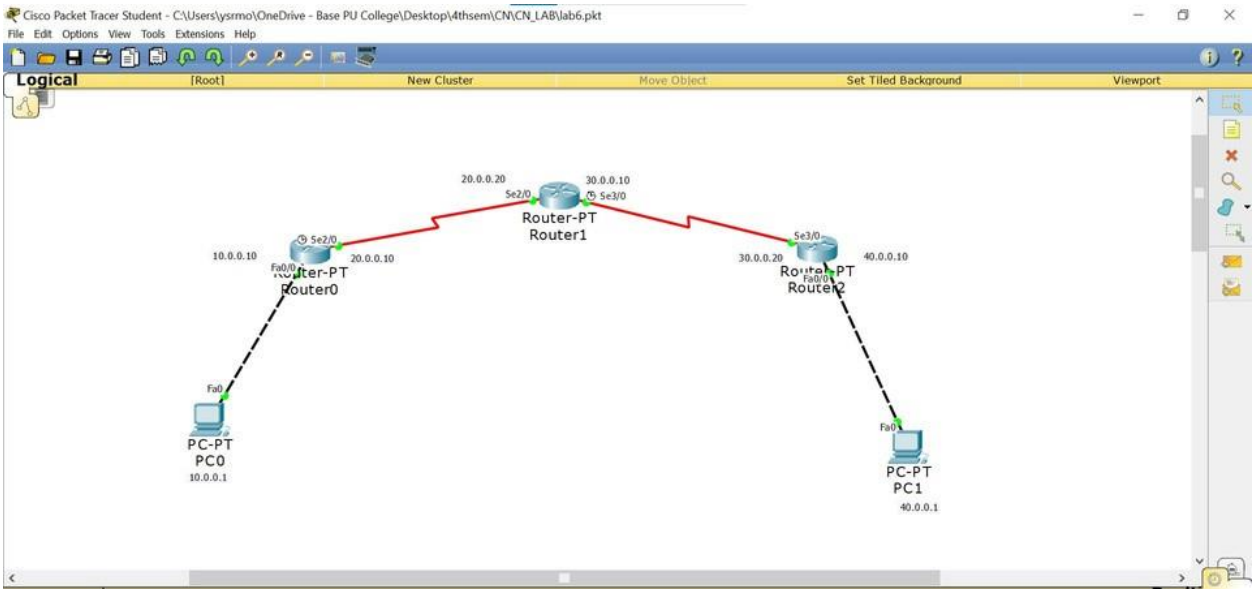
Observation

RIP is the Routing Information Pro... is a distance vector protocol that ~~was~~ uses hop count as its primary metric. RIP defines how router should share information when moving traffic among an interconnected group of local area networks.

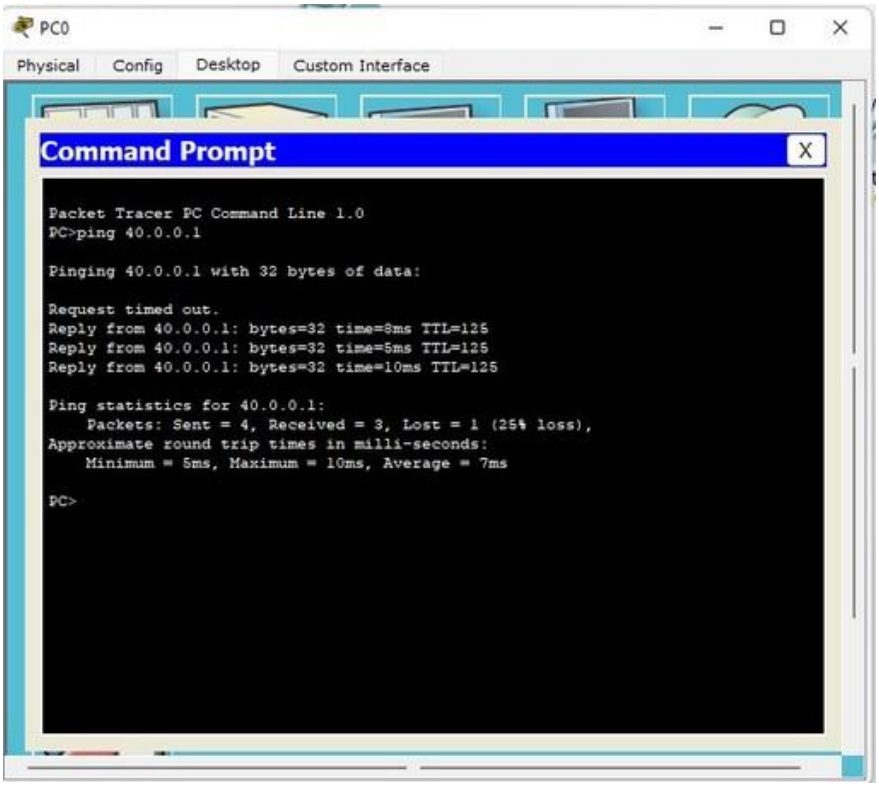
→ The RIP protocol here, used to connect the routers to one other & PC's using RIP protocol & message is pinged successfully.

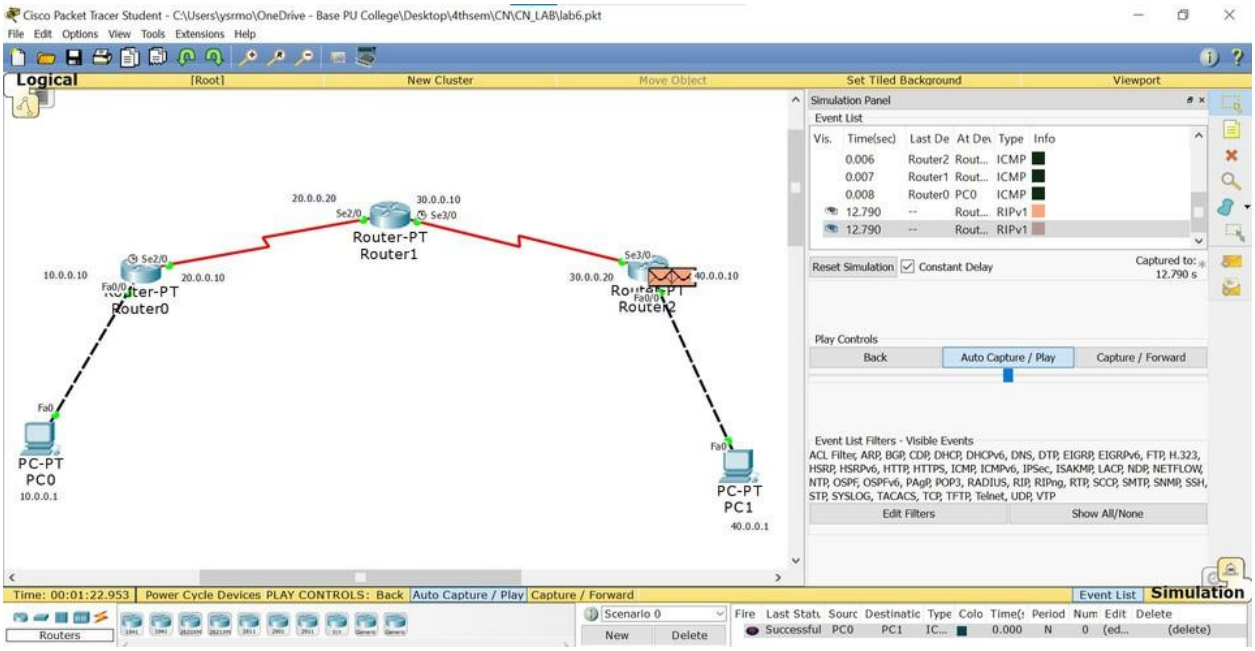
8/8/2023

TOPOLOGY:


Cisco Packet Tracer Student - C:\Users\ysrmo\OneDrive - Base PU College\Desktop\4thsem\CN\CN_LAB\lab6.pkt
File Edit Options View Tools Extensions Help
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
20.0.0.20
30.0.0.10
Se2/0
Se3/0
Router-PT
Router1
10.0.0.10
Fa0/0
Se2/0
20.0.0.10
Router-PT
Router0
30.0.0.20
Se3/0
40.0.0.10
Fa0/0
Router-PT
Router2
Fa0
PC-PT
PC0
10.0.0.1
Fa0
PC-PT
PC1
40.0.0.1

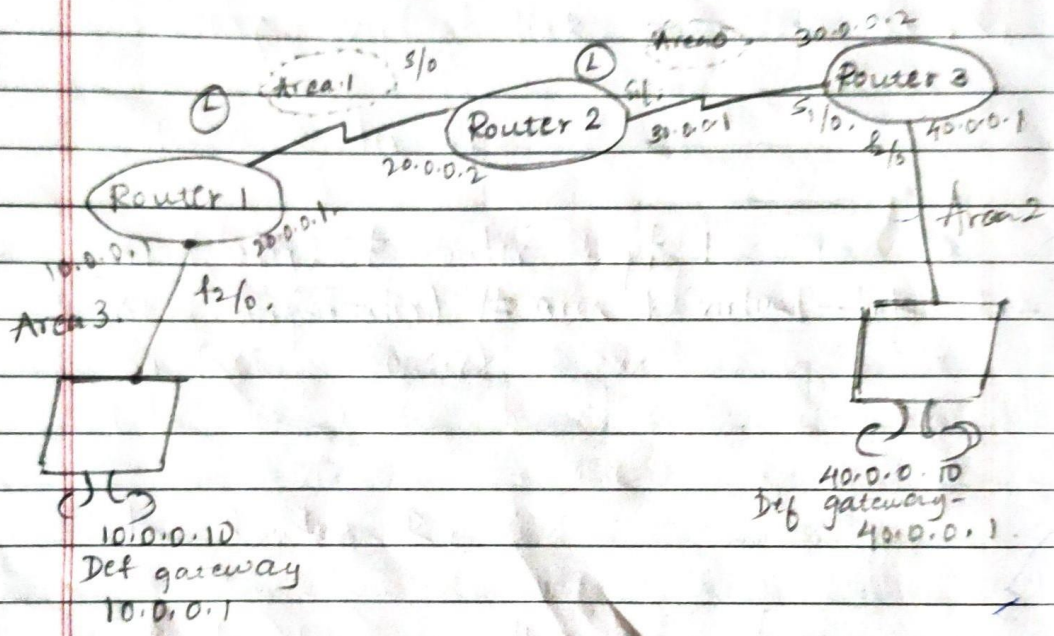
OUTPUT:


PC0
Physical
Config
Desktop
Custom Interface
Command Prompt
X
Packet Tracer PC Command Line 1.0
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=8ms TTL=125
Reply from 40.0.0.1: bytes=32 time=5ms TTL=125
Reply from 40.0.0.1: bytes=32 time=10ms TTL=125
Ping statistics for 40.0.0.1:
Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
Minimum = 5ms, Maximum = 10ms, Average = 7ms
PC>


Cisco Packet Tracer Student - C:\Users\ysrmo\OneDrive - Base PU College\Desktop\4thsem\CN\CN_LAB\lab6.pkt
File Edit Options View Tools Extensions Help
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
Router-PT
Router1
Router-PT
Router0
Router-PT
Router2
PC-PT
PC0
10.0.0.1
PC-PT
PC1
40.0.0.1
20.0.0.20
30.0.0.10
Se2/0
Se3/0
10.0.0.10
20.0.0.10
30.0.0.20
40.0.0.10
Fa0
Fa0/0
Fa0/1
Simulation Panel
Event List
Vis. Time(sec) Last De At De Type Info
0.006 Router2 Rout... ICMP
0.007 Router1 Rout... ICMP
0.008 Router0 PC0 ICMP
12.790 -- Rout... RIPv1
12.790 -- Rout... RIPv1
Reset Simulation
Constant Delay
Captured to: 12.790 s
Play Controls
Back
Auto Capture / Play
Capture / Forward
Event List Filters - Visible Events
ACL Filter, ARP, BGP, CDP, DHCP, DHCPv6, DNS, DTP, EIGRP, EIGRPv6, FTP, H.323, HSRP, HSRPv6, HTTP, HTTPS, ICMP, ICMPv6, IPSec, ISAKMP, LACP, NDP, NETFLOW, NTP, OSPF, OSPFv6, PAgP, POP3, RADIUS, RIP, RIPng, RTP, SCCP, SMTP, SNMP, SSH, STP, SYSLOG, TACACS, TCP, TFTP, Telnet, UDP, VTP
Edit Filters
Show All/None
Time: 00:01:22.953
Power Cycle Devices PLAY CONTROLS: Back
Auto Capture / Play
Capture / Forward
Event List
Simulation
Routers
Scenario 0
New
Delete
Fire Last Statu Sourc Destinatic Type Colo Time(s Period Num Edit Delete
Successful PC0 PC1 IC... 0.000 N 0 (ed... (delete)

Aim: To understand the working of OSPF

Topology:



Procedure

Configure your PCs with IP address and gateway, configure each of routers according to IP address given in topology. Encapsulation ppt & clock rate need to be set as done in RIP Protocol experiment.

WEEK 7

Configure OSPF routing

protocol. OBSERVATION:

In Router R_1

R_1 (config) # router ospf 1
R_1 (config-router) # router-id 1.1.1.1
R_1 (config-router) # network 10.0.0.0 255.255.255 area 3
R_1 (config-router) # network 20.0.0.0 255.255.255 area 1
R_1 (config-router) # exit

In Router R2

R_2 (config) # router ospf 1
R_2 (config-router) # router-id 2.2.2.2
network 20. 255.255.255 area 1
30 255.255.255 area 0
exit

In Router R3

R_3 (config) # router ospf 1
R_3 (config-router) # network 30.0.0.0 0.255.255.255 area 1
R3 # network 40.0.0.0 0.255.255.255 area 0
R_3 (config-router) # exit

Loopback (In Router 1's serial connection)
R_1 (config-if) # interface loopback 0
ip address 172.16.1.253 255.255.0.0
no shutdown

In Router -2 any or serial connection
R_2 (config-if) # interface loopback 0
ip address 172.16.1.253 255.255.0.0
no shutdown

7. Creating virtual link between R_1, R_2

In Router R_1

```
R1 (config) # router ospf 1
R1 (config-router) # area 1 virtual-link 2.2.2.2
R1 (config-router) # exit
```

In Router R_2

```
R2 (config)          # router ospf 2
   (config-router) # area 2 virtual-link 1.1.1.1
                   # exit
```

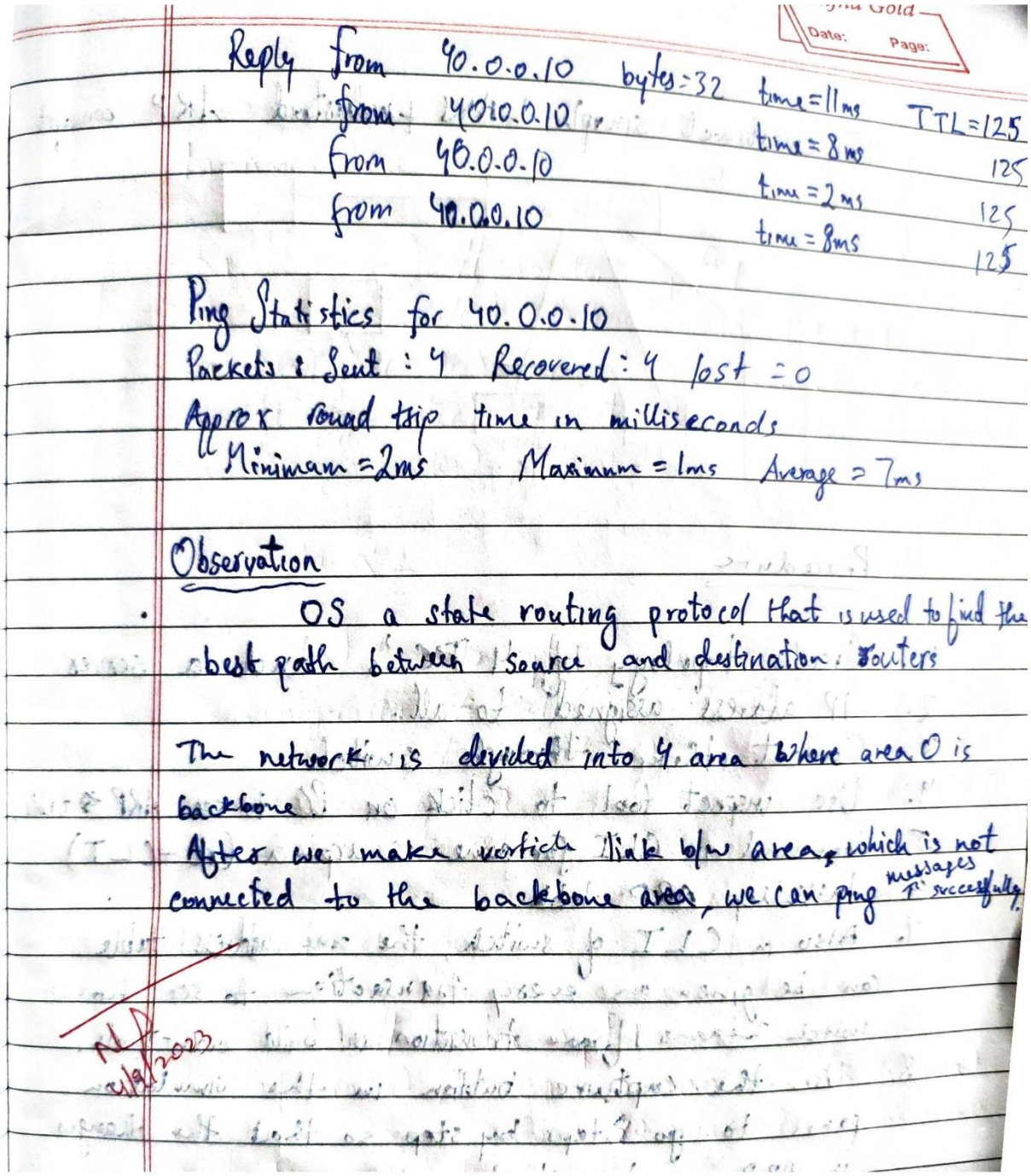
8. (For every router, we show ip-route command)

Now, check connecti

Output

Ping PC > Ping 40.0.0.10

Pinging 40.0.0.10 with 32 bytes of data



Reply from 40.0.0.10 bytes=32 time=11ms TTL=125
from 40.0.0.10 time=8ms 125
from 40.0.0.10 time=2ms 125
from 40.0.0.10 time=8ms 125

Ping Statistics for 40.0.0.10
Packets : Sent : 4 Recovered : 4 lost = 0
Approx round trip time in milliseconds
Minimum = 2ms Maximum = 1ms Average = 7ms

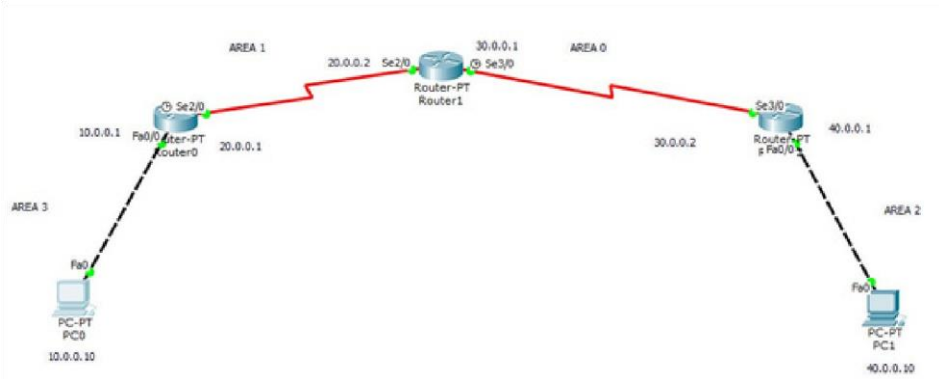
Observation

OS a state routing protocol that is used to find the best path between source and destination routers

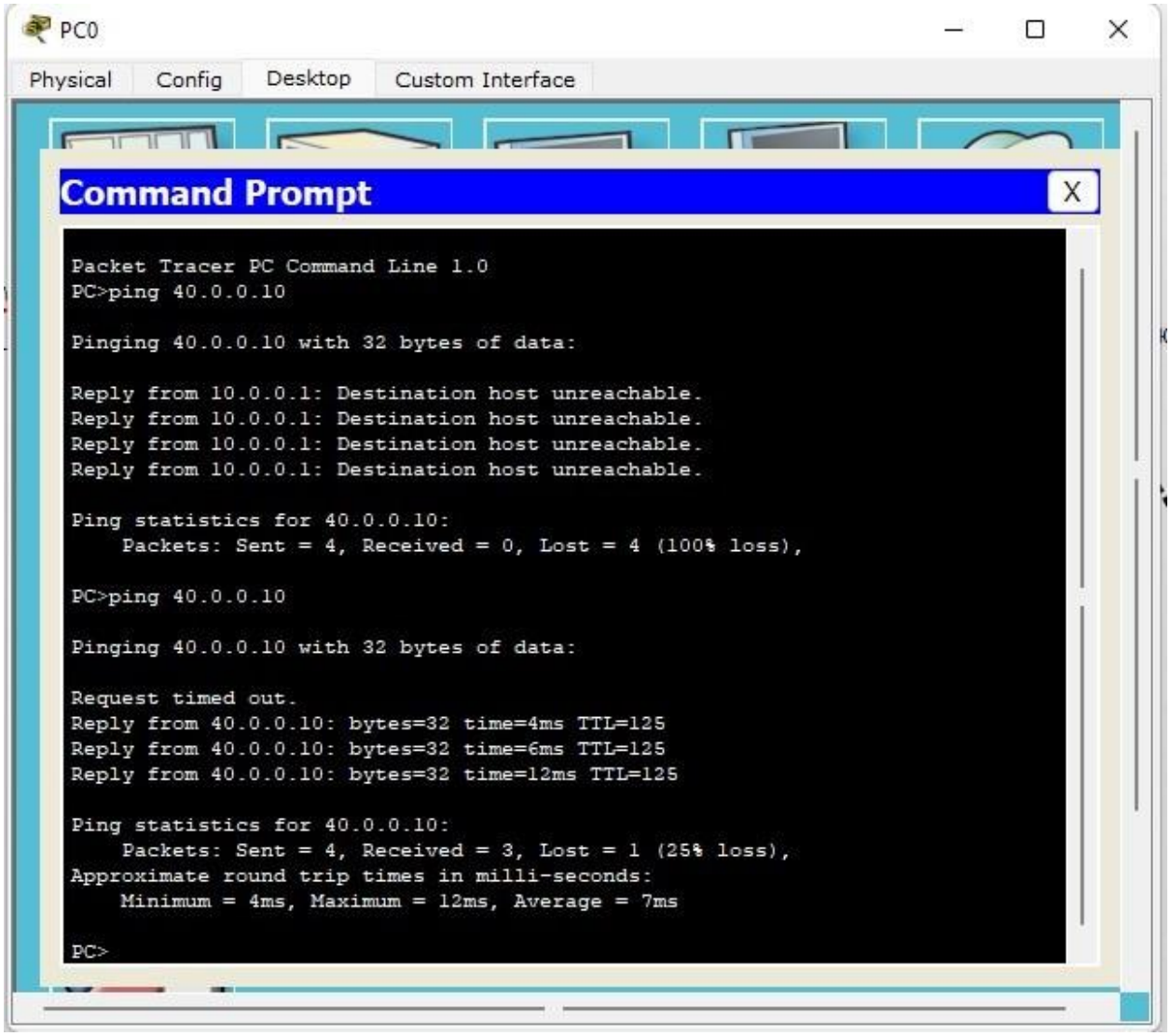
The network is devided into 4 area where area 0 is backbone.

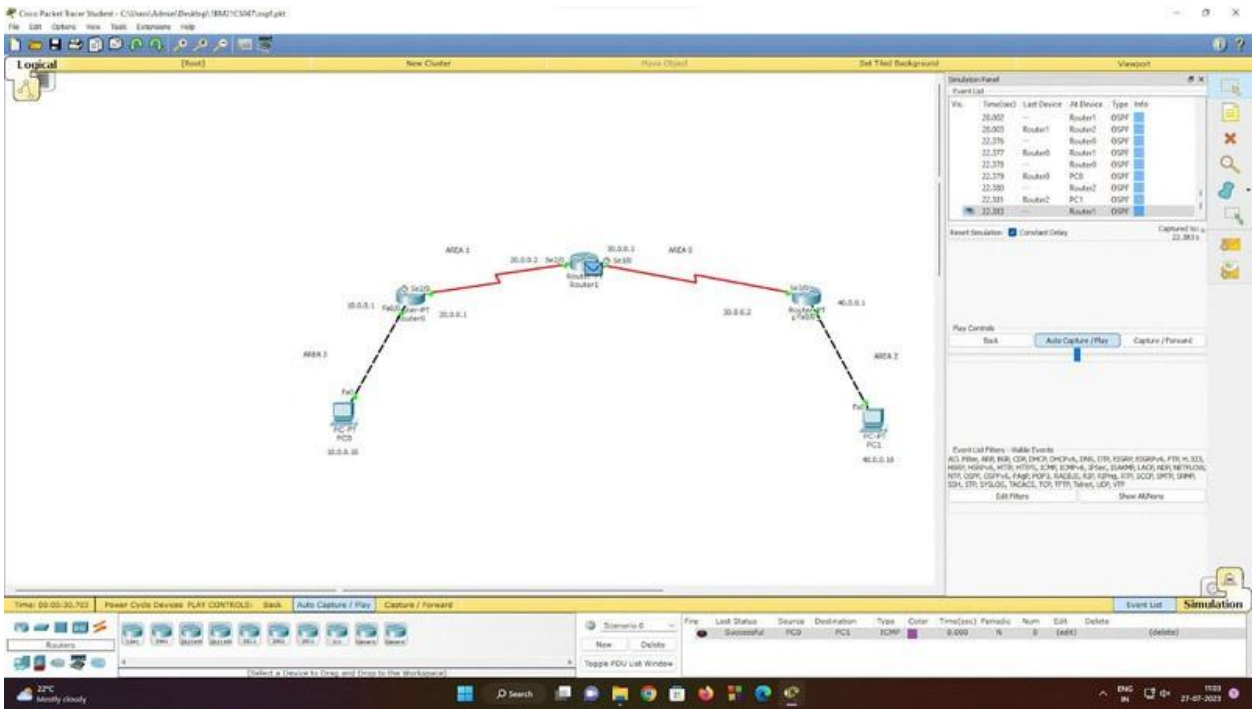
After we make vertical link b/w area, which is not connected to the backbone area, we can ping messages successfully.

TOPOLOGY:


AREA 1
20.0.0.2
Se2/0
30.0.0.1
Se3/0
AREA 0
Router-PT
Router1
Se2/0
10.0.0.1
Fa0/0
Router0
20.0.0.1
Se3/0
30.0.0.2
40.0.0.1
AREA 3
AREA 2
Fa0
PC-PT
PC0
10.0.0.10
Fa0
PC-PT
PC1
40.0.0.10

OUTPUT:

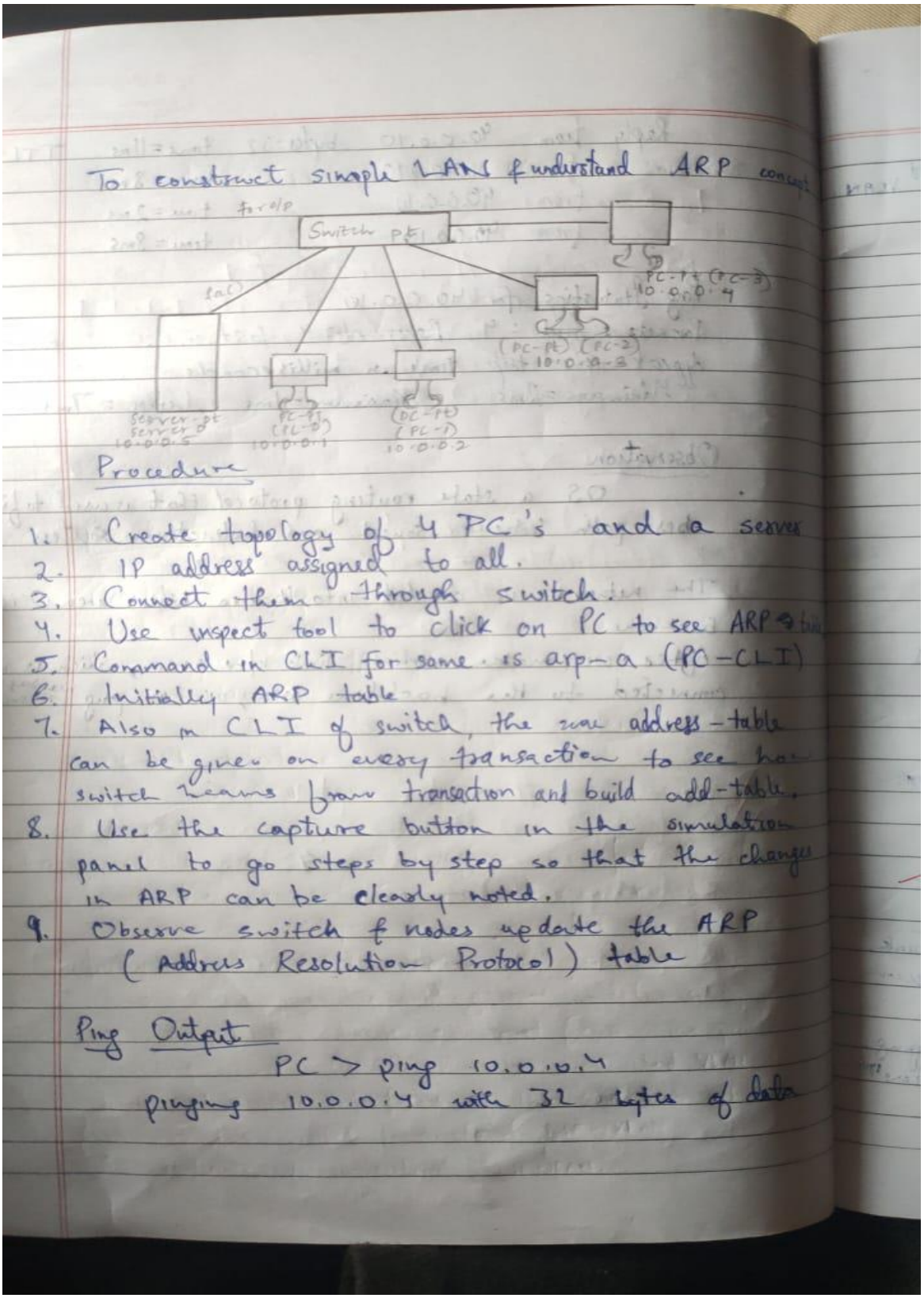

PC0
Physical
Config
Desktop
Custom Interface
Command Prompt
X
Packet Tracer PC Command Line 1.0
PC>ping 40.0.0.10
Pinging 40.0.0.10 with 32 bytes of data:
Reply from 10.0.0.1: Destination host unreachable.
Reply from 10.0.0.1: Destination host unreachable.
Reply from 10.0.0.1: Destination host unreachable.
Reply from 10.0.0.1: Destination host unreachable.
Ping statistics for 40.0.0.10:
Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
PC>ping 40.0.0.10
Pinging 40.0.0.10 with 32 bytes of data:
Request timed out.
Reply from 40.0.0.10: bytes=32 time=4ms TTL=125
Reply from 40.0.0.10: bytes=32 time=6ms TTL=125
Reply from 40.0.0.10: bytes=32 time=12ms TTL=125
Ping statistics for 40.0.0.10:
Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
Minimum = 4ms, Maximum = 12ms, Average = 7ms
PC>


Logical
AREA 1
AREA 0
AREA 3
AREA 2
Router1
Router0
Router2
PC0
PC1
Simulation
Event List

WEEK 8

To construct a simple LAN and understand the concept and operation of Address Resolution Protocol (ARP).

OBSERVATION:



Bafna Gold
Date: Page:

Reply from 10.0.0.4 bytes=32 time=0ms TTL=125
10.0.0.4
10.0.0.4
10.0.0.4 TTL=125

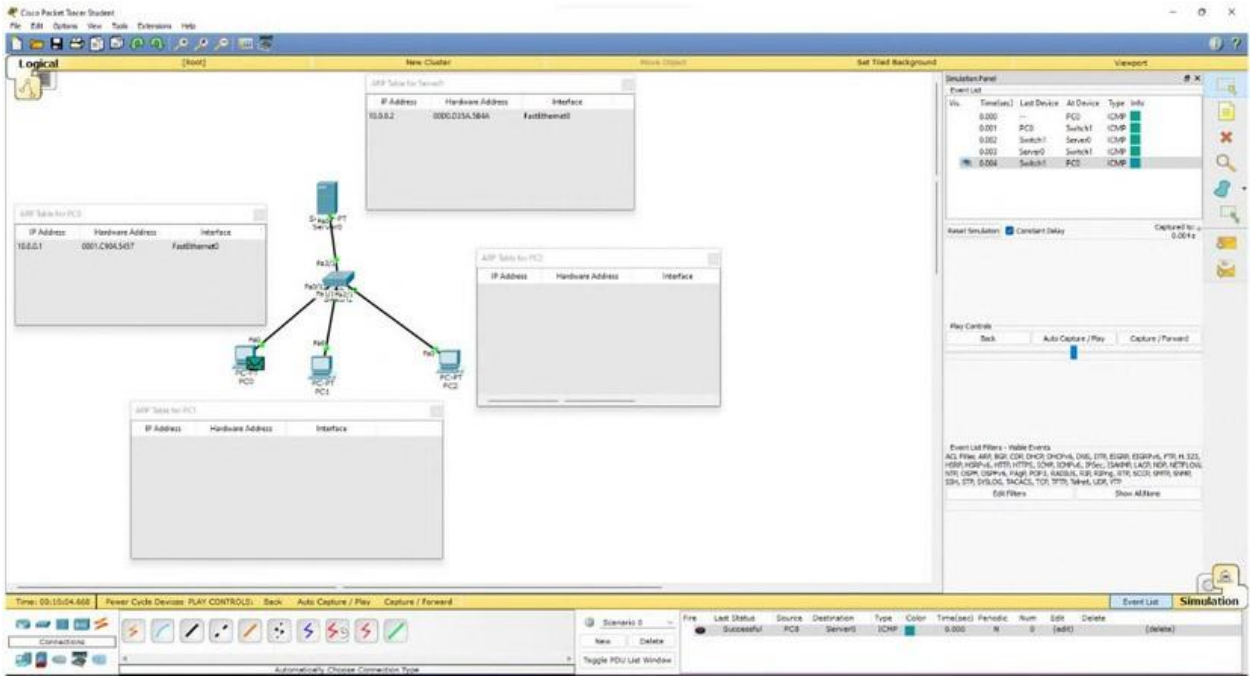
Ping statistics for 10.0.0.4
Packets : sent=4, recieved=4, lost = 0
Approx round times in milliseconds
minimum = 0ms, max = 0ms, Avg = 0ms

Observations

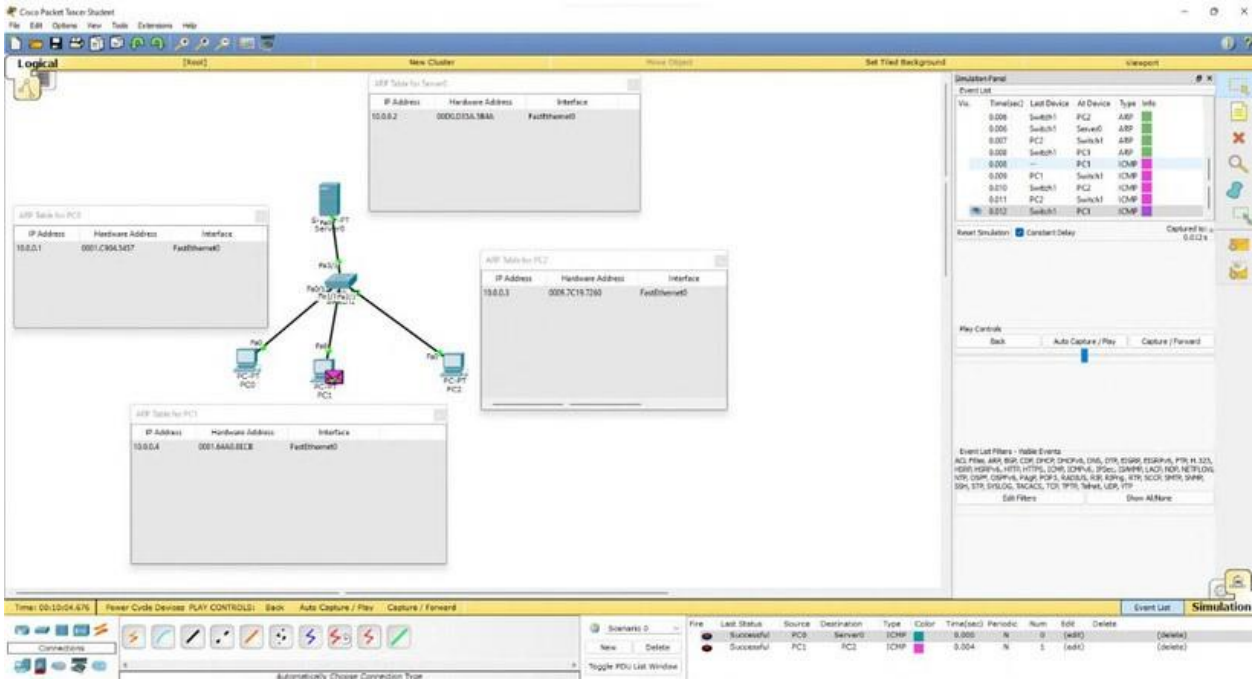
- When we ping 1 PC & server, the address of server is known to PC & vice versa.
- When we ping between other 2 PC, simultaneously the address of each other are unknown.
Everytime a host request a MAC address in order to send a packet to another host in LAN, it checks its ARP cache to see if IP to MAC address already exists, if it doesn't exist, ARP is performed.

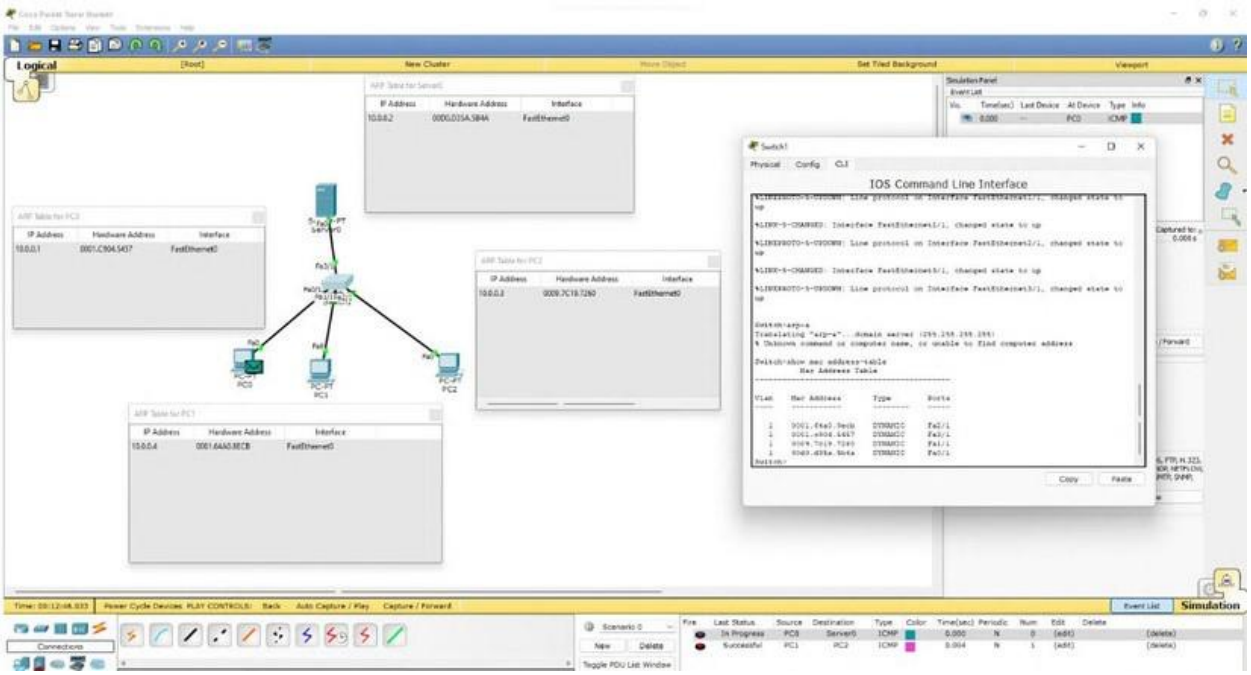
4/12/2023

TOPOLOGY:



OUTPUT:




IOS Command Line Interface

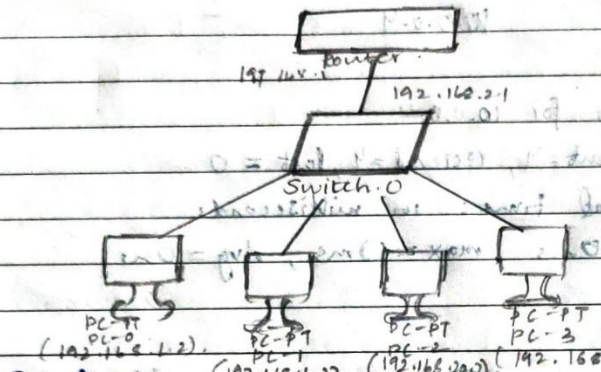
WEEK 9

To construct a VLAN and make a pc communicate among

VLAN. OBSERVATION:

Construct VLAN and make PC communicate among VLAN

Aim: To construct VLAN to make PC communicate in VLAN.



Procedure

1. Create topology
2. Add extra part to switch as its needed
3. Use copper st. through wire, set IP address & gateway
4. In switch go to config tab and select VLAN database
5. Give VLAN no, say 2 here include any name say add
6. Select add, select interface i.e fast ethernet (near the switch from router) & make it trunk
7. VLAN trunking allows switches to forward from diff VLANs over a single link called Trunk
8. This is done by adding an additional header into called tag to the ethernet frame. The process of adding this small header is called VLAN tagging
9. Look into the interfaces of the switches with NEWLAN IP
10. This makes switch understand NEWVLAN
11. Next router is to understand NEWLAN.

12. Config tab of router, select VLAN DATABASE enter the number and name of VLAN created.

13. In CLI of router

Router (VLAN) # exit

Apply completed

Existing

Router # config t

(config) # interface fastEthernet 0/0.1

Router (config-subif) #

( ) # encapsulation of 192

ip address 192.168.21 255.255.0

no shut

(config-subif) # exit

Result (in PC0)

PC > ping 192.168.20.3

pinging from 192.168.20.3 bytes=32 time=1ms TTL=128

192.168.20.3 time 0ms TTL=128

Pinging from bytes=32 time=0ms

192.168.20.3 time 0ms TTL=128

Ping statistics for 192.168.20.3

Packets sent = 4, recieved = 4, lost = 0.

Approximate round trip times in milliseconds

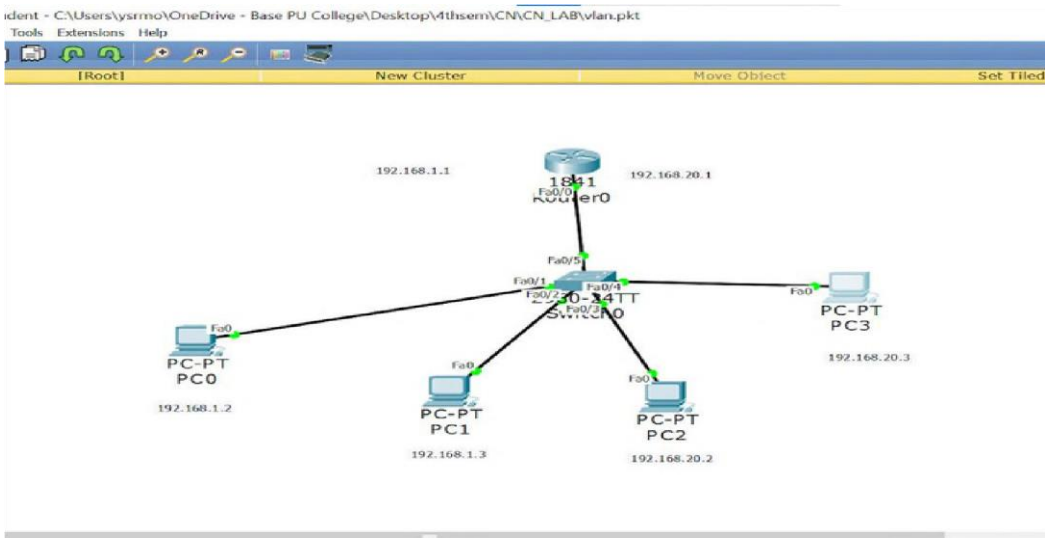
minimum = 0ms, max = 1ms, average = 0ms

Observation:

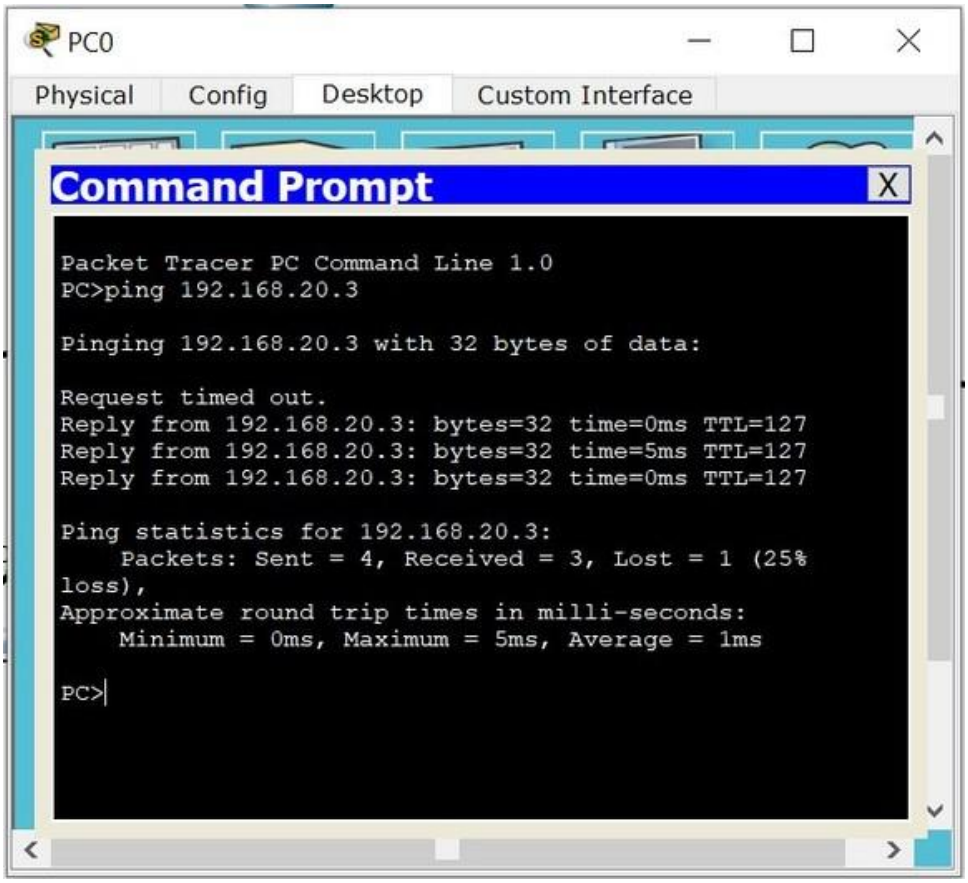
(i) VLAN - Virtual local area network is any broadcast domain that is partitioned and isolated in a conjuncted network at the data link layer.

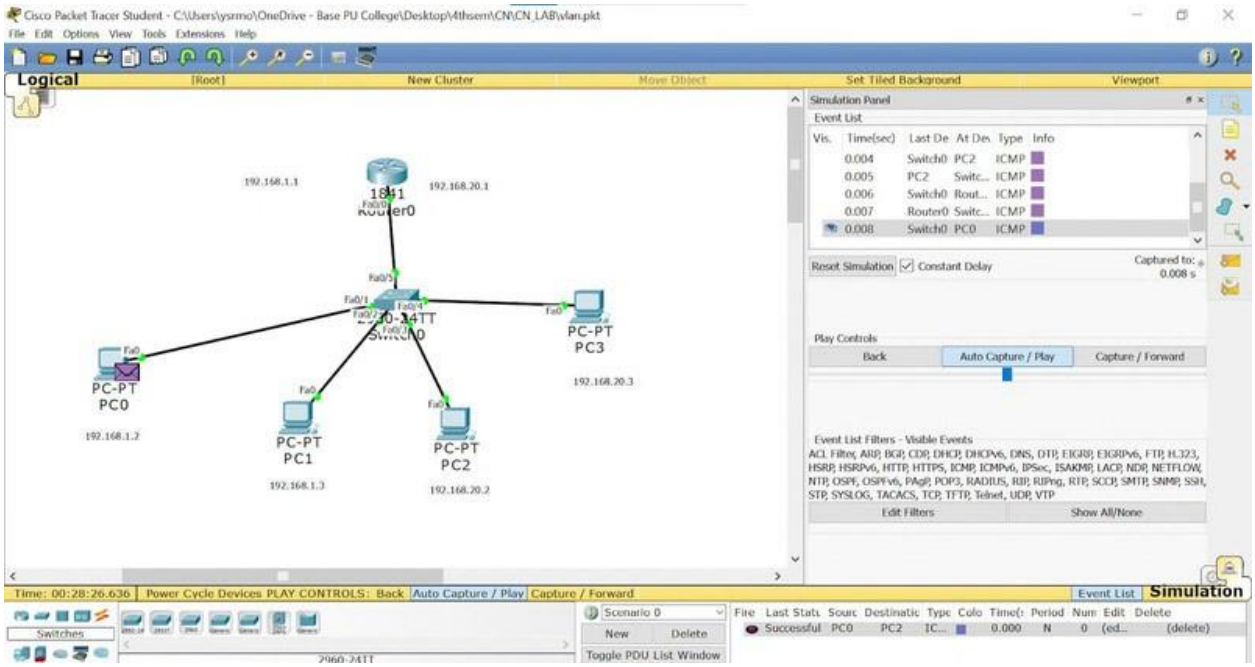
(ii) It is virtualized connection that connects multiple devices & network nodes from diff. LAN's into one logical network.

TOPOLOGY:



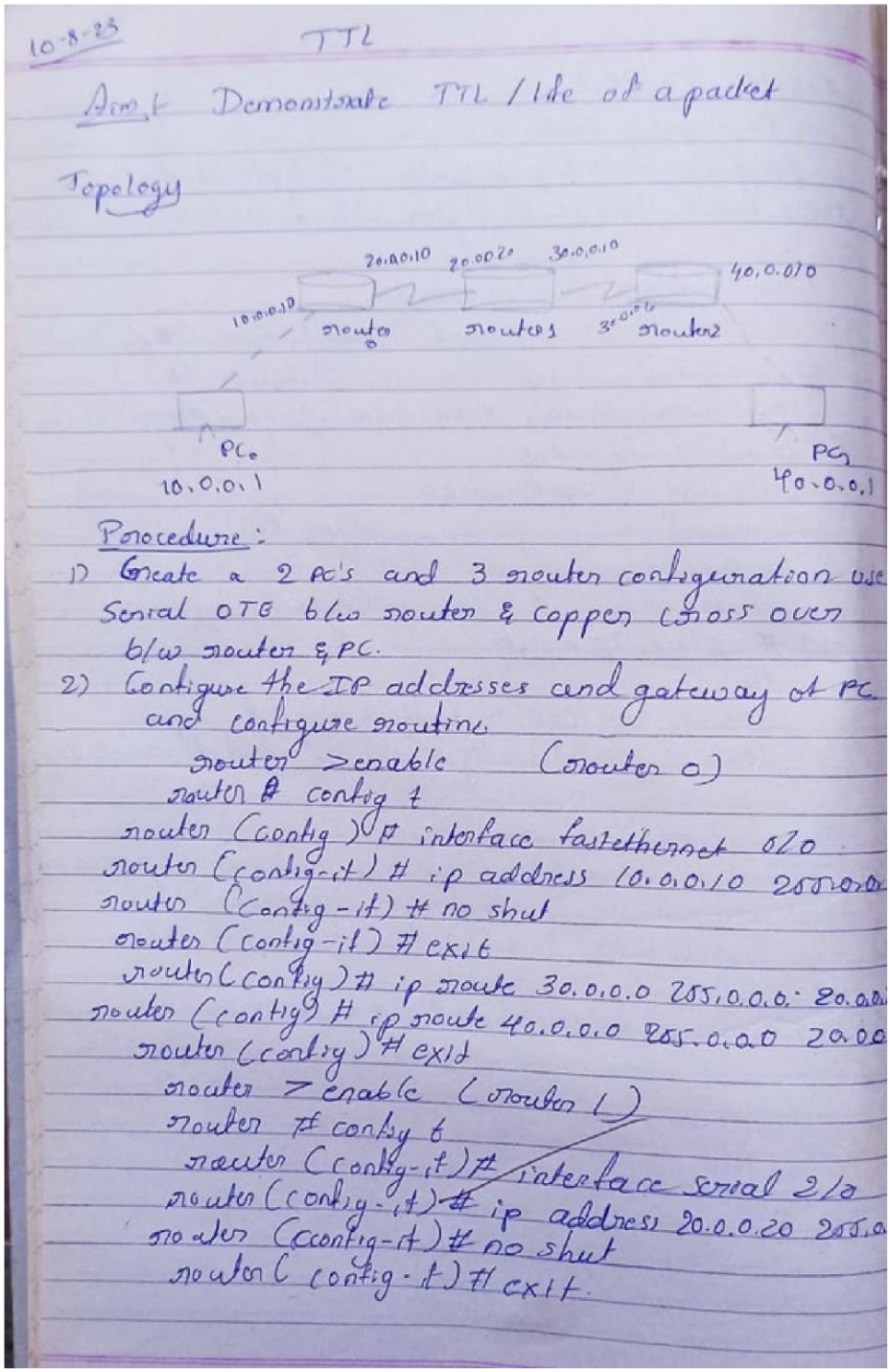
OUTPUT:




Cisco Packet Tracer Student - C:\Users\ysrmo\OneDrive - Base PU College\Desktop\4thsem\CN\CN_LAB\vlan.pkt
File Edit Options View Tools Extensions Help
Logical
[Root]
New Cluster
Move Object
Set Tiled Background
Viewport
192.168.1.1
192.168.20.1
Router0
Fa0/1
Fa0/4
PC-PT
PC0
192.168.1.2
PC-PT
PC1
192.168.1.3
PC-PT
PC2
192.168.20.2
PC-PT
PC3
192.168.20.3
Simulation Panel
Event List
Vis. Time(sec) Last De At De Type Info
0.004 Switch0 PC2 ICMP
0.005 PC2 Switc... ICMP
0.006 Switch0 Rout... ICMP
0.007 Router0 Switc... ICMP
0.008 Switch0 PC0 ICMP
Reset Simulation
Constant Delay
Captured to:
0.008 s
Play Controls
Back
Auto Capture / Play
Capture / Forward
Event List Filters - Visible Events
ACL Filter, ARP, BGP, CDP, DHCP, DHCPv6, DNS, DTP, EIGRP, EIGRPv6, FTP, H.323, HSRP, HSRPv6, HTTP, HTTPS, ICMP, ICMPv6, IPSec, ISAKMP, LACP, NDP, NETFLOW, NTP, OSPF, OSPFv6, PAgP, POP3, RADIUS, RIP, RIPng, RTP, SCCP, SMTP, SNMP, SSH, STP, SYSLOG, TACACS, TCP, TFTP, Telnet, UDP, VTP
Edit Filters
Show All/None
Time: 00:28:26.636
Power Cycle Devices PLAY CONTROLS: Back
Auto Capture / Play
Capture / Forward
Event List
Simulation
Switches
Scenario 0
New
Delete
Toggle PDU List Window
Fire Last Stat Sourc Destinatic Type Colo Time(Period Num Edit Delete
Successful PC0 PC2 IC... 0.000 N 0 (ed... (delete)

Demonstrate the TTL/ Life of a

Packet. OBSERVATION:



10-8-23 TTL

Aim: Demonstrate TTL / life of a packet

Topology

Procedure:

1) Create a 2 PC's and 3 router configuration use Serial DTE b/w router & copper cross over b/w router & PC.
2) Configure the IP addresses and gateway of PC and configure routing

router > enable (router 0)
router # config t
router (config) # interface fastethernet 0/0
router (config-if) # ip address 10.0.0.10 255.0.0
router (config-if) # no shut
router (config-if) # exit
router (config) # ip route 30.0.0.0 255.0.0.0 20.0.0
router (config) # ip route 40.0.0.0 255.0.0.0 20.0.0
router (config) # exit
router > enable (router 1)
router # config t
router (config-if) # interface serial 2/0
router (config-if) # ip address 20.0.0.20 255.0
router (config-if) # no shut
router (config-if) # exit

router (config-if)# interface serial 3/0
router (config-if)# ip address 30.0.0.10 255.0.0.0
router (config-if)# no shut
router (config-if)# exit
router (config)# ip route 10.0.0.0 255.0.0.0 20.0.0.10
router (config)# ip route 40.0.0.0 255.0.0.0 30.0.0.20
router (config)# exit
router # config t
router (config)# interface serial 2/0
router (config-if)# ip address 30.0.0.20 255.0.0.0
router (config-if)# no shut
router (config-if)# exit
router (config)# interface fastethernet 0/0
router (config-if)# ip address 40.0.0.10 & 255.0.0.0
router (config-if)# no shut
router (config)# ip route 10.0.0.0 255.0.0.0 30.0.0.10
router (config)# ip route 20.0.0.0 255.0.0.0 30.0.0.2

iii) Select Simulation mode, select simple PDU and select & source & destination PC's
iv) use capture button to send PDU, from PC to router to PC
v) Click on PDU every transfer to see inbound & out PDU details observe the different in TTLs

Result

PDU information at service PC0,
outbound PDU details
TTL : 255
PDU information at device : router 0
inbound PDU details
TTL : 255
outbound PDU details
TTL : ~~255~~ 254

PDU info at device : router 1
inbound PDU details
TTL : 254
outbound PDU details
TTL : 253
PDU info at device : router 2
inbound PDU details
TTL : 253
Outbound PDU details at device : PC1
Inbound PDU details
TTL : 252
An example : for inbound details of router 0
PDU format

Ethernet II

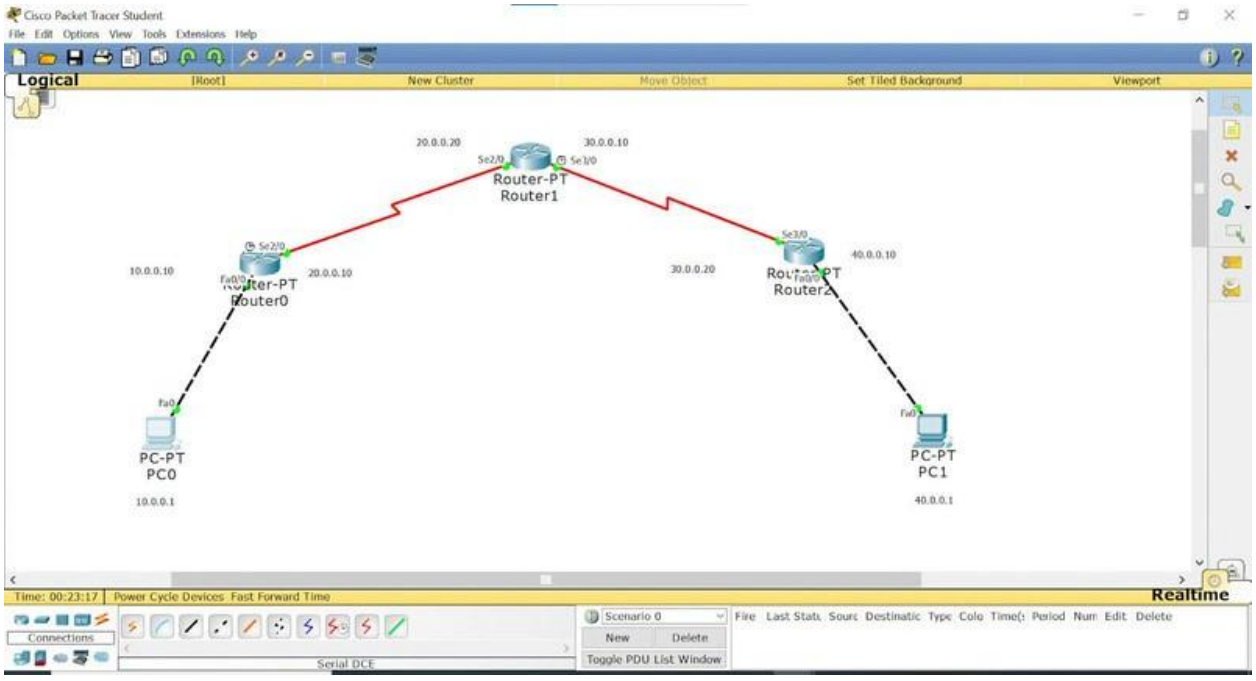
Preamble	DEST MAC	SRC MAC
10101 -- 1011	D0E0 F489. -	0009.7C05 E079
Type	DATA	FCS:
0X800	(variable length	0XS

observation

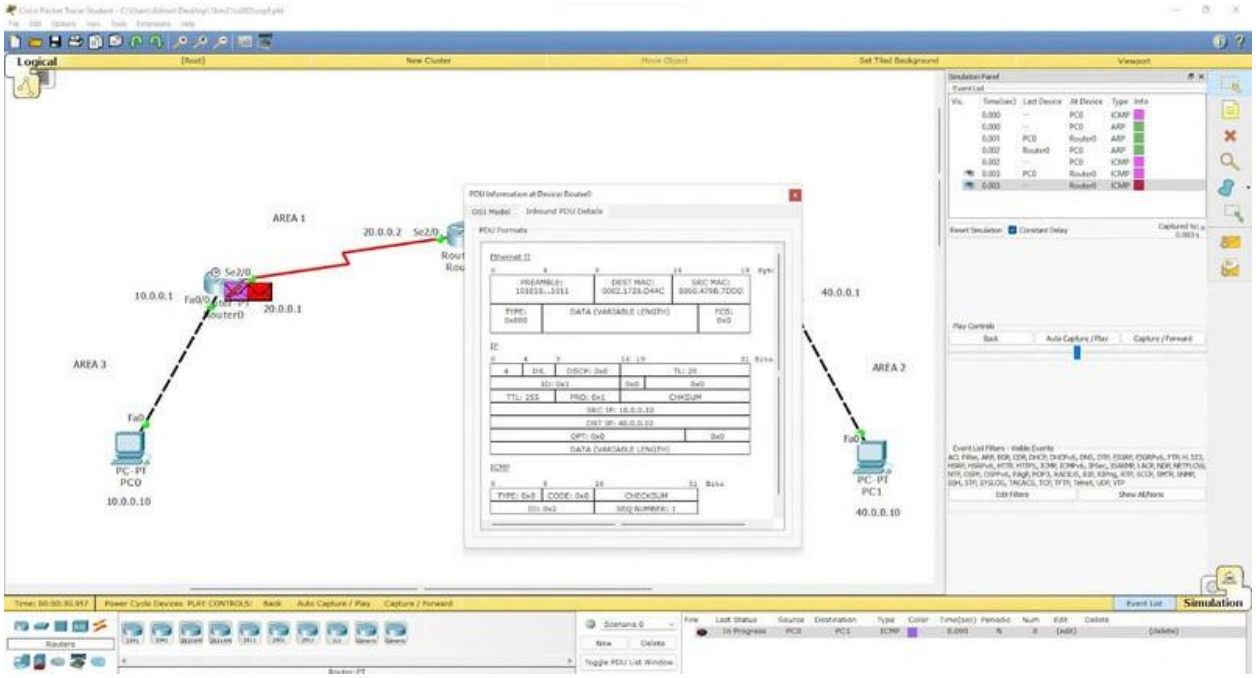
1) The TTL is reduced by 1 in every router
Time to live (TTL) is a mechanism which limits the life form or lifetime data in a computer or network. It is a computer 255 is set as max TTL.

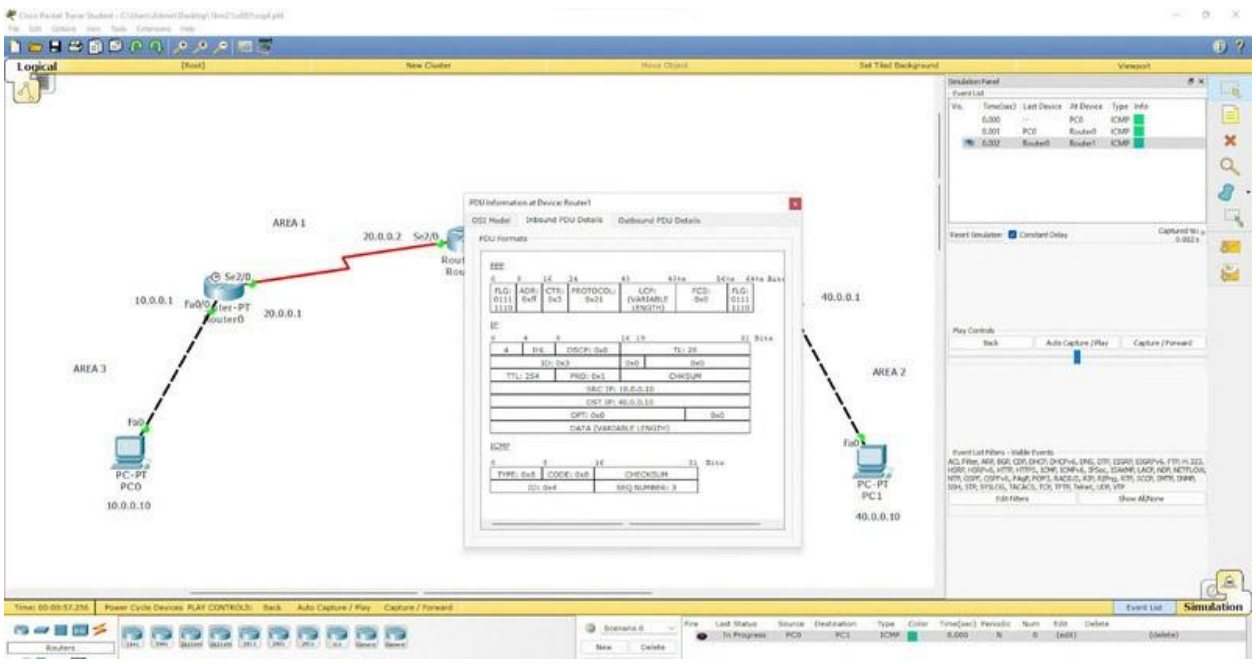
NLJ
29/8/2023

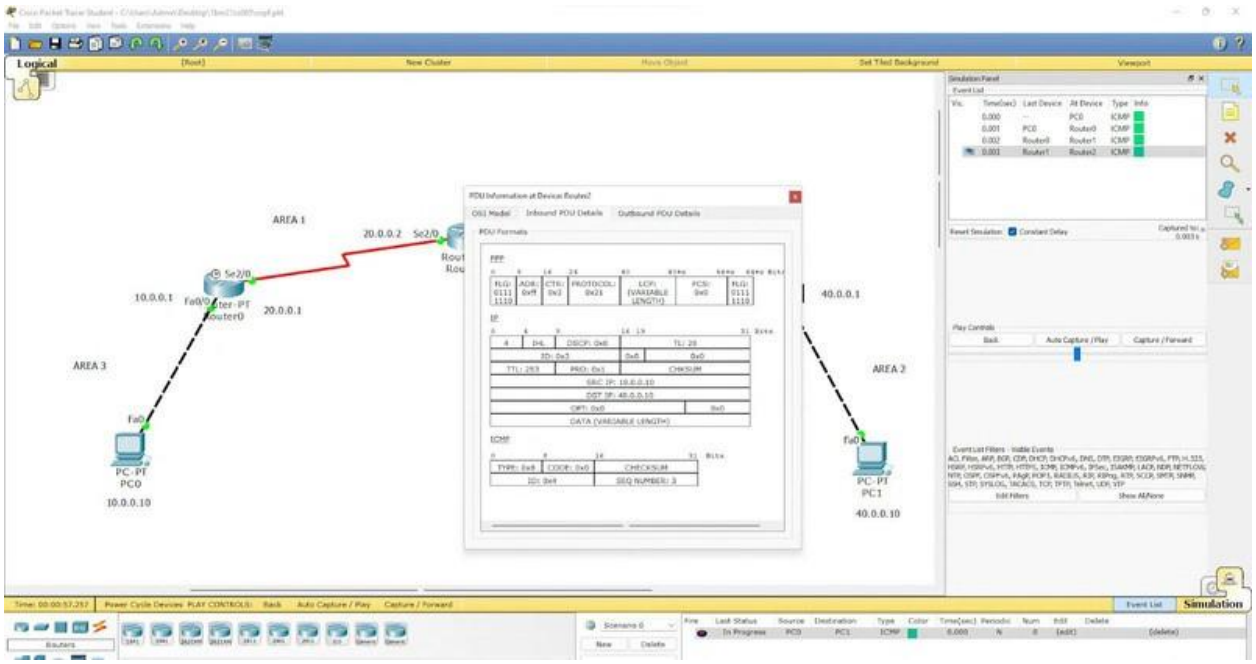
TOPOLOGY:

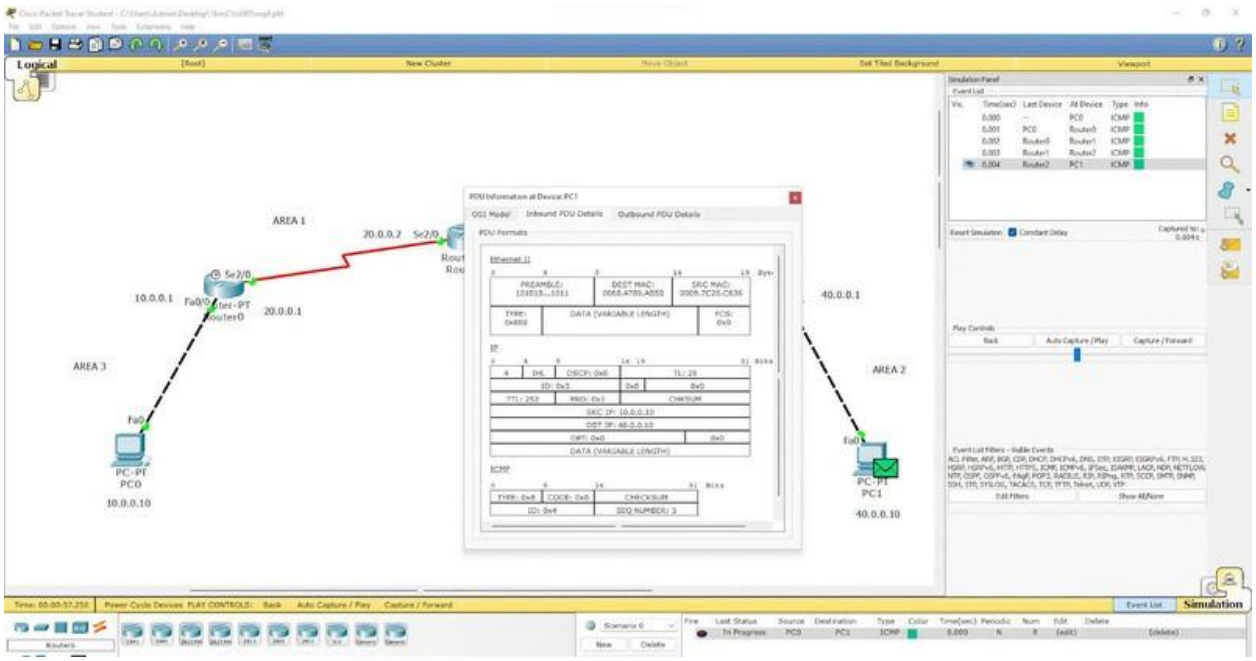


OUTPUT:




Logical
AREA 1
AREA 2
AREA 3
20.0.0.2
Se2/0
10.0.0.1
Fa0/0
20.0.0.1
40.0.0.1
Fa0
PC-PT
PC0
10.0.0.10
PC-PT
PC1
40.0.0.10


Logical
AREA 1
AREA 2
AREA 3
20.0.0.2
Se2/0
10.0.0.1
Fa0/0
20.0.0.1
40.0.0.1
Fa0
PC-PT
PC0
10.0.0.10
PC-PT
PC1
40.0.0.10

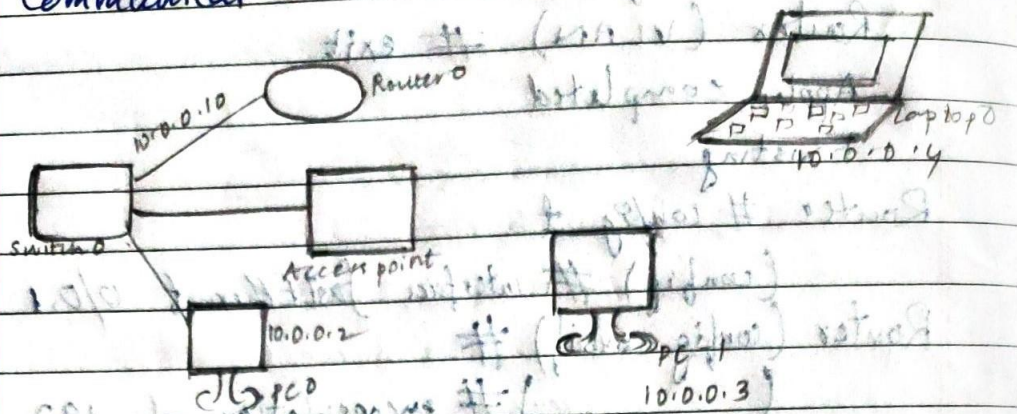

Logical
AREA 1
20.0.0.2
Se2/0
10.0.0.1
Fa0/0
20.0.0.1
AREA 3
AREA 2
40.0.0.1
Fa0
PC-PT
PC0
10.0.0.10
Fa0
PC-PT
PC1
40.0.0.10
PDU Information at Device: PC1
OSI Model
Inbound PDU Details
Outbound PDU Details
PDU Formats
Ethernet II
IP
ICMP
Simulation Panel
Event List
Simulation

WEEK 11

To construct a WLAN and make the nodes communicate wirelessly

OBSERVATION:

To construct WLAN & make nodes communicate

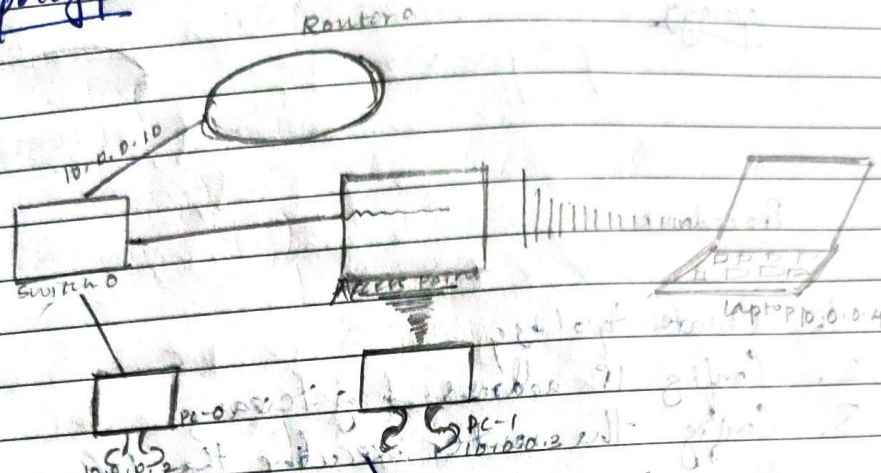


Procedure

1. Connect topology
2. Configure PC3 & Router1
3. Config Access Point1 - port1 → SSID Name - anyname (WLAN)
4. Select WEP and give any digit here key - 1234567890 here
5. Configure PC4 & laptop with wireless standards.
6. Switch off device. Drag existing PT-host-NM-1 to the component listed in the LHS. Drag WMP300N wireless interface to empty port. Switch on device
7. In config tab, new wireless interface would have been added. Now configure SSID, WEP, WEP key, IP address & gateway.
8. Ping from each device to every other device and see result.

Result

Topology



In PC (10.0.0.2)

PC > ping 10.0.0.3

pinging 10.0.0.3 with 32 bytes of data

reply from 10.0.0.3 bytes=32 time=1ms TTL=120

reply from 10.0.0.3 time=13ms TTL=120

reply from 10.0.0.3 bytes=32 time=6 ms TTL=120

reply from 10.0.0.3 time=0 ms TTL=120

Ping statistics for 10.0.0.3

Packets Sent=4, recieved=4, lost=0

Approx round trip in millisecond

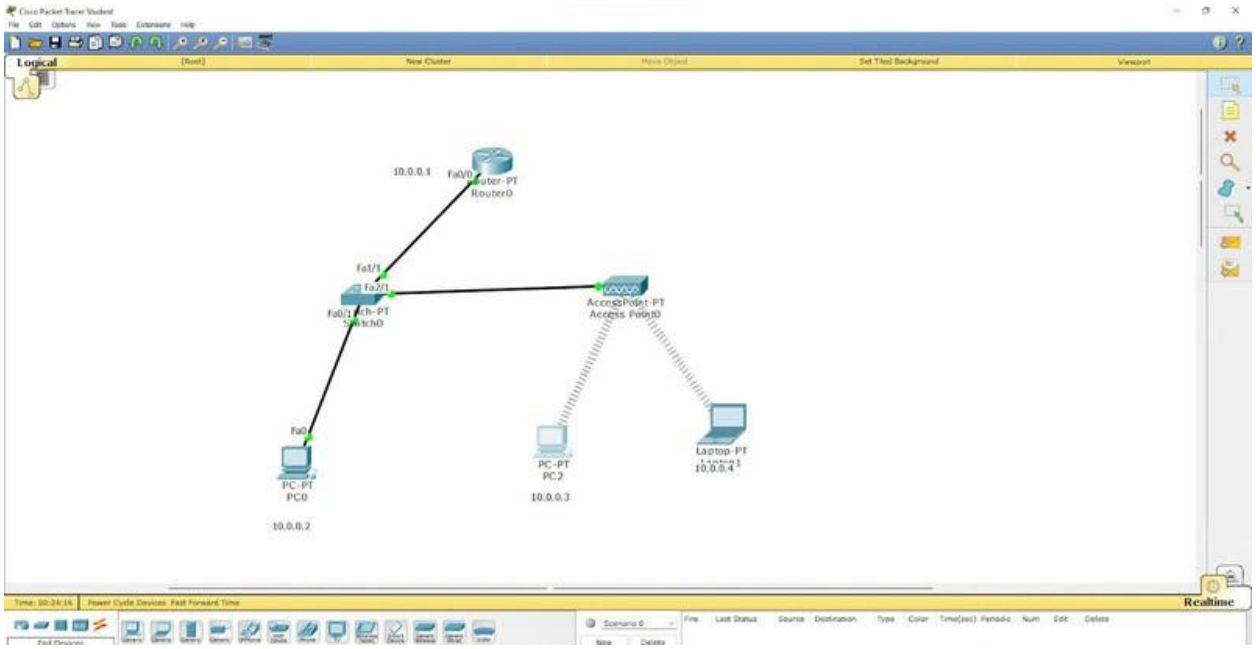
Min = 6ms, max = 21ms Averay = 12 ms.

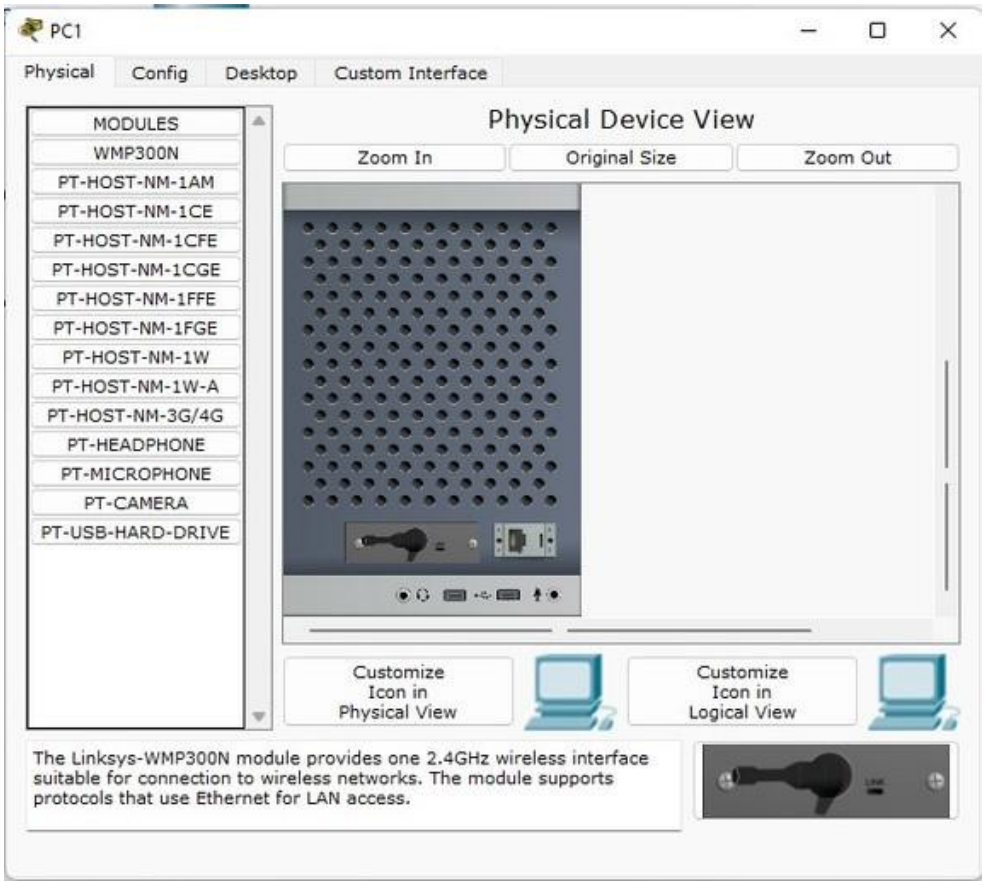
4/9/22

Observation

1. Wireless local area Network is a group of located computers or other devices that form network based on radio transmission rather than wire connections.
2. After WLAN is setup, the wired connection appear in the topology from the access point.

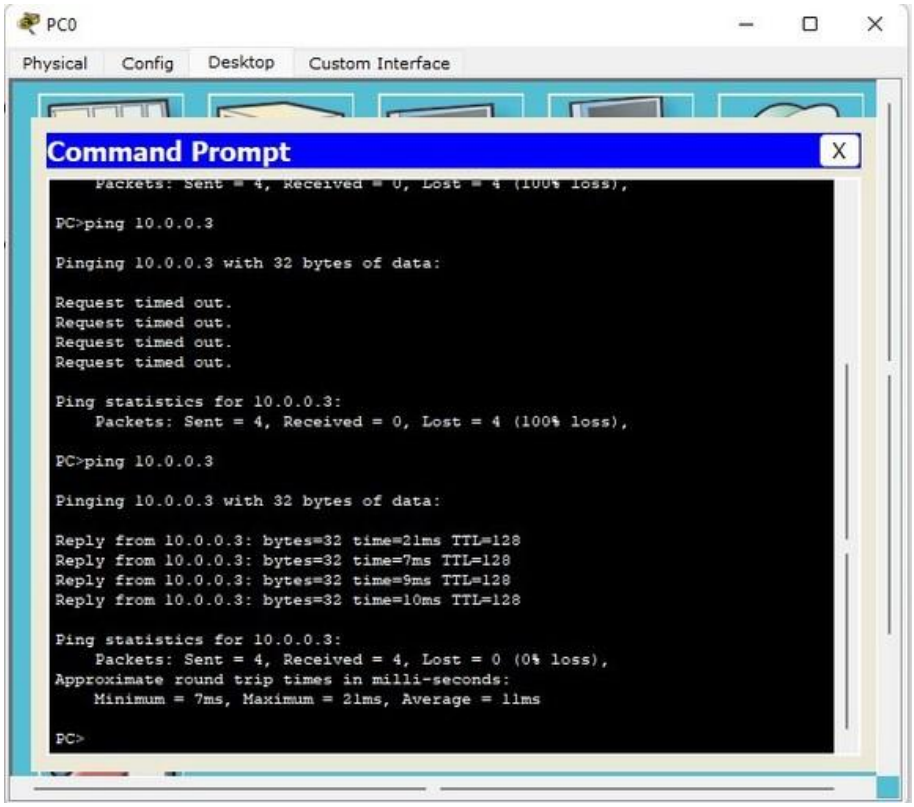
TOPOLOGY:


Logical
10.0.0.1
Router-PT
Router0
Fa1/1
Fa2/1
Switch-PT
Switch0
Access Point-PT
Access Point0
Fa0
PC-PT
PC0
10.0.0.2
PC-PT
PC2
10.0.0.3
Laptop-PT
10.0.0.4
Realtime


PC1
Physical
Config
Desktop
Custom Interface
MODULES
WMP300N
PT-HOST-NM-1AM
PT-HOST-NM-1CE
PT-HOST-NM-1CFE
PT-HOST-NM-1CGE
PT-HOST-NM-1FFE
PT-HOST-NM-1FGE
PT-HOST-NM-1W
PT-HOST-NM-1W-A
PT-HOST-NM-3G/4G
PT-HEADPHONE
PT-MICROPHONE
PT-CAMERA
PT-USB-HARD-DRIVE
Physical Device View
Zoom In
Original Size
Zoom Out
Customize Icon in Physical View
Customize Icon in Logical View
The Linksys-WMP300N module provides one 2.4GHz wireless interface suitable for connection to wireless networks. The module supports protocols that use Ethernet for LAN access.


Laptop0
Physical
Config
Desktop
Custom Interface
MODULES
WPC300N
PT-LAPTOP-NM-1AM
PT-LAPTOP-NM-1CE
PT-LAPTOP-NM-1CFE
PT-LAPTOP-NM-1CGE
PT-LAPTOP-NM-1FFE
PT-LAPTOP-NM-1FGE
PT-LAPTOP-NM-1W
PT-LAPTOP-NM-1W-A
PT-LAPTOP-NM-3G/4G
PT-HEADPHONE
PT-MICROPHONE
PT-CAMERA
PT-USB-HARD-DRIVE
Physical Device View
Zoom In
Original Size
Zoom Out
Customize Icon in Physical View
Customize Icon in Logical View
The Linksys-WPC300N module provides one 2.4GHz wireless interface suitable for connection to wireless networks. The module supports protocols that use Ethernet for LAN access.

OUTPUT:


PC0
Physical
Config
Desktop
Custom Interface
Command Prompt
X
Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
PC>ping 10.0.0.3
Pinging 10.0.0.3 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.
Ping statistics for 10.0.0.3:
Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
PC>ping 10.0.0.3
Pinging 10.0.0.3 with 32 bytes of data:
Reply from 10.0.0.3: bytes=32 time=21ms TTL=128
Reply from 10.0.0.3: bytes=32 time=7ms TTL=128
Reply from 10.0.0.3: bytes=32 time=9ms TTL=128
Reply from 10.0.0.3: bytes=32 time=10ms TTL=128
Ping statistics for 10.0.0.3:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 7ms, Maximum = 21ms, Average = 11ms
PC>

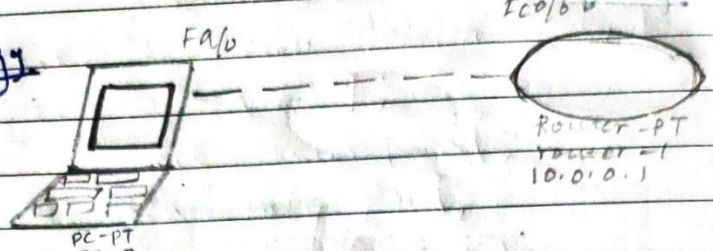
WEEK 12

To understand the operation of TELNET by accessing the router in server room from a PC in IT office.

OBSERVATION:

To understand operation of TELNET by accessing routers in servers room a PC in IT office.

Topology



Procedure

1. Create topology
2. Config IP address & gateway
3. Config the router executing the following
   1. Enable
   2. Config t
   Host name S
   4. enable secret P
   5. interface fast ethernet 0/0
   IP address 10.0.0.1 255.0.0.0
   7. no shut
   time vty 05
   login
   10. password Po
   V. Exit 1. Exit
   12. ^z

Ping message to router
password for web verify is Po
password for enable is P
Accessing router from PC
show IP router

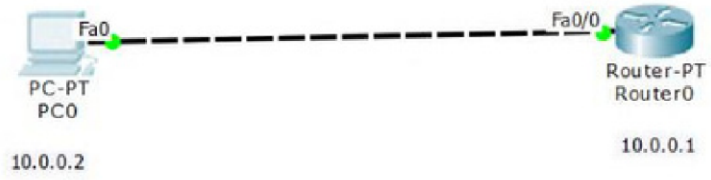
Bafna Gold
Date: Page:

Ping Statistics for 10.0.0.1
Packets sent = 4 recieved = 4 lost = 0
Approx round trip in milliseconds
min = 0ms, max = 0ms, Average = 0ms
PC > telnet 10.0.0.1
Typing 10.0.0.1 ... open
User access verification
password P0
P1 > enable
password P1
n# show ip route
L 10.0.0.0/8 is directly connected to fa 0/0.

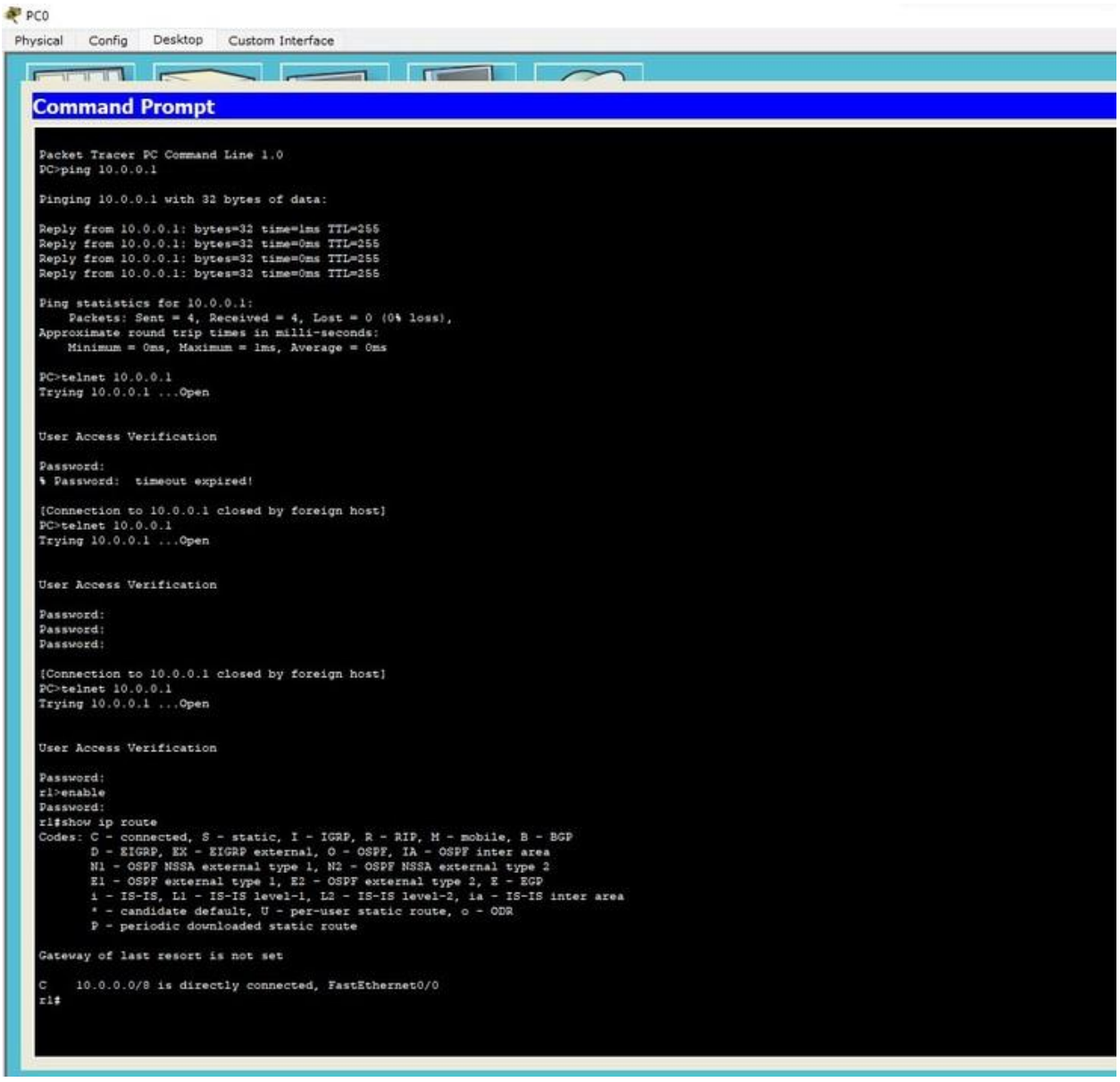
Observation

1. Telnet stands for tele type network; it is a type of protocol that enables one comp to connect to local comp.
2. It is used as standard TCP/IP pro for virtual terminal service provided by ISO.
3. During TELNET, whatever is being performed the remote comp will be displaced by local comp. TELNET operates on search principle.

TOPOLOGY:



OUTPUT:


PC0
Physical
Config
Desktop
Custom Interface
Command Prompt
Packet Tracer PC Command Line 1.0
PC>ping 10.0.0.1
Pinging 10.0.0.1 with 32 bytes of data:
Reply from 10.0.0.1: bytes=32 time=1ms TTL=255
Reply from 10.0.0.1: bytes=32 time=0ms TTL=255
Reply from 10.0.0.1: bytes=32 time=0ms TTL=255
Reply from 10.0.0.1: bytes=32 time=0ms TTL=255
Ping statistics for 10.0.0.1:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 1ms, Average = 0ms
PC>telnet 10.0.0.1
Trying 10.0.0.1 ...Open
User Access Verification
Password:
% Password: timeout expired!
[Connection to 10.0.0.1 closed by foreign host]
PC>telnet 10.0.0.1
Trying 10.0.0.1 ...Open
User Access Verification
Password:
Password:
Password:
[Connection to 10.0.0.1 closed by foreign host]
PC>telnet 10.0.0.1
Trying 10.0.0.1 ...Open
User Access Verification
Password:
r1>enable
Password:
r1#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
* - candidate default, U - per-user static route, o - ODR
P - periodic downloaded static route
Gateway of last resort is not set
C 10.0.0.0/8 is directly connected, FastEthernet0/0
r1#

WEEK 13

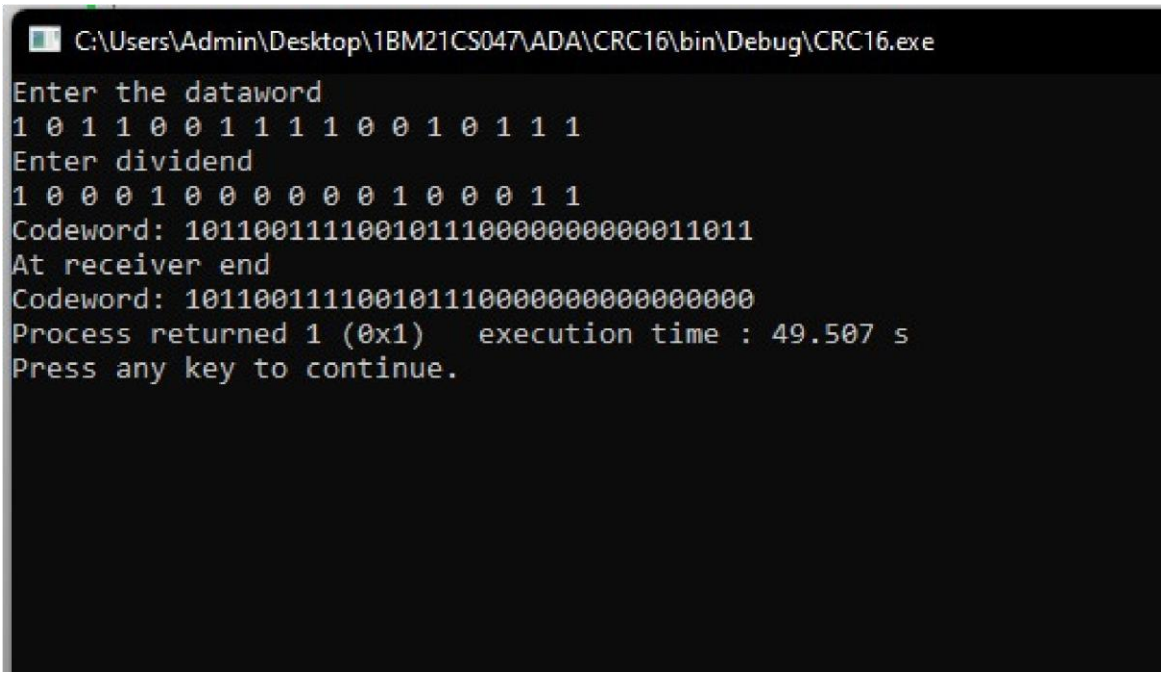
Write a program for error detecting code using CRC- CCITT (16-bits).

CODE:

```
#include<stdio.h> int
arr[17];

void xor(int x[], int y[])
{ int
k=0;
for(int i=1;i<16;i++)
{ if(x[i]==y[i])
arr[k++]=0;
else arr[i]=1;
}
}

void main()
{
int dd[17],div[33],ze[17],i,k;

printf("Enter the dataword \n");
for(i=0;i<17;i++) scanf("%d",&div[i]);

for(i=i;i<33;i++) div[i]=0;

for(i=0;i<17;i++) ze[i]=0;
printf("Enter dividend \n");
   for(i=0;i<17;i++)
   scanf("%d",&dd[i]);

   i=0; k=0;
   for(i=i;i<17;i++)
   arr[k++]=div[i];
   while(i<33)
```

```
    { if(arr[0]==0)
    xor(arr,ze); else
    xor(arr,dd);
    arr[16]=div[i++];

    }
    k=0; for(i=17;i<33;i++)
    div[i]=arr[k++];
    printf("Codeword: ");
    for(i=0;i<33;i++)
    printf("%d",div[i]);

    for(i=0;i<17;i++) arr[i]=0;

    printf("\nAt receiver end \n");

    k=0; for(i=i;i<17;i++)
    arr[k++]=div[i];
    while(i<33)
    {
if(arr[0]==0)
            xor(arr,ze);
else
 xor(arr,dd); arr[16]=div[i++];

}
k=0; for(i=17;i<33;i++)
div[i]=arr[k++];

          printf("Codeword: ");
             for(i=0;i<33;i++)
            printf("%d",div[i]);
}
```

OUTPUT:



OBSERVATION:

Cycle - 2

17-8-23

CRC implementation

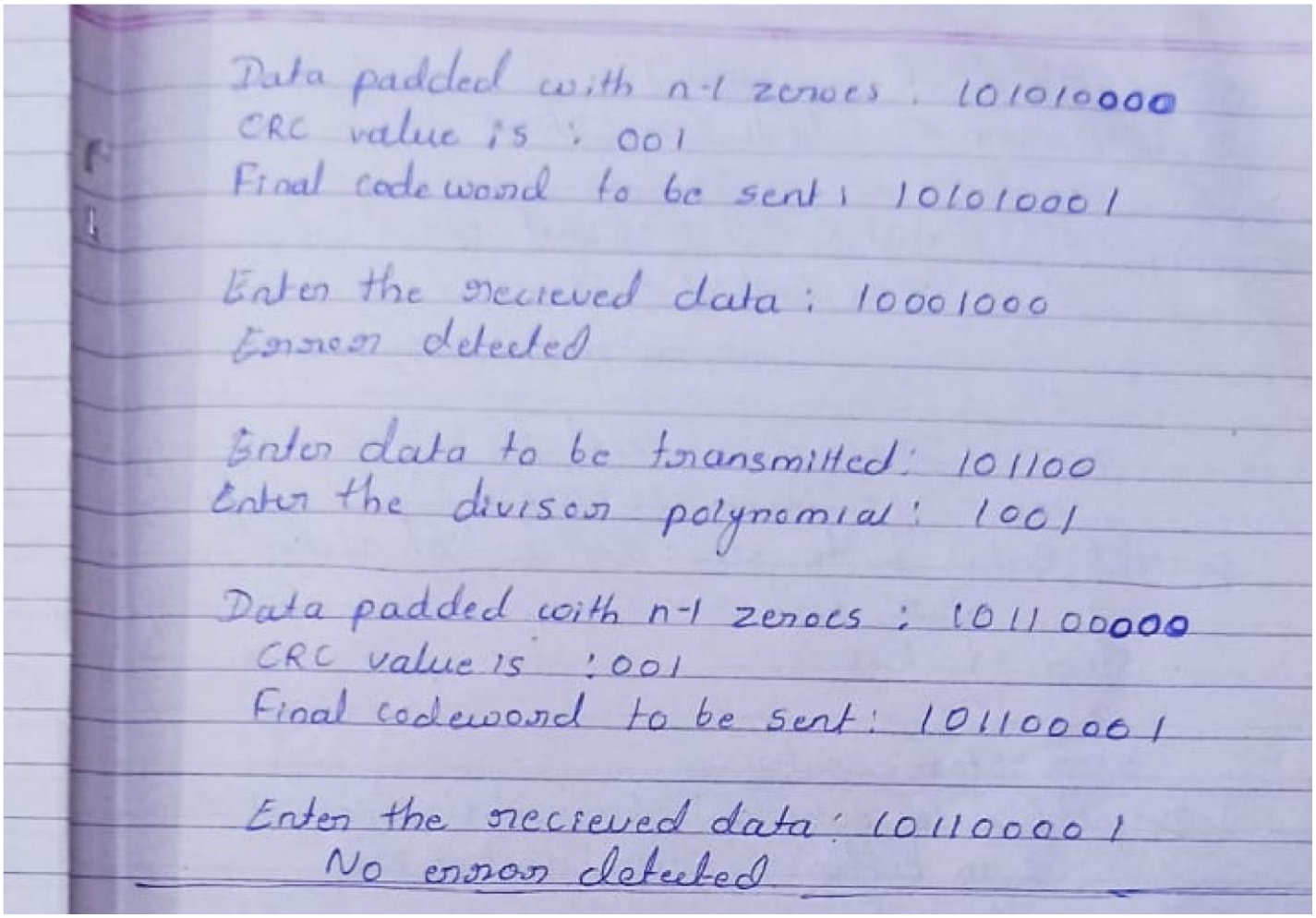
Write a program for error detecting code using CRC-CCITT

```
#include <stdio.h>
#include <string.h>
#define N strlen(poly)

char data[30];
char check_value[30];
char poly[10];
int data_length, i, j;

void XOR
{
    for (j=1; j<N; j++)
        checkvalue[j] = ((Check value[j] == poly[j])?
                        '0' : '1');
}

void receiver()
{
    printf("Enter the recieved data: ");
    scanf("%s", data);
    printf("Data recieved : %s", data);
    crc();
    for (i=0; (i<N-1) && (check_value[i] != '1');
        i++)
    if (N-1)
        printf("\n Error detected \n");
    else
        printf("\n No error detected \n");
}

void crc()
{
    for (i=0; i<N; i++)
        check_value[i] = data[i];
    do
```

```
    if (checkvalue[0]=='1')
        XOR();
    for(j=0; j<N-1 ; j++)
        check_value[j]=check_value[j+1];
    check_value[j]=data[i++];
    }
    while(i<=data_length+N+1);
    }
int main()
{
    printf("\nEnter data to be transmitted:");
    scanf("%s",data);
    printf("\nEnter the divisor polynomial:");
    scanf("%s",poly);
    data_length=strlen(data);
    for(i=data_length;i<datalength+N-1;i++)
        data[i]='0';
    printf("\nData padded with n-1 zeroes:%s",
            data);
    crc();

    printf("\nCRC value is %s",check_value);
    for(i=data_length;i<data_length+N-1;i++)
        data[i]=check_value[i-data_length];
    printf("\nFinal datawword to be sent:%s",
                                    data);
    receiver();
    return 0;
}
```

Output

```
Enter data to be transmitted: 101010
Enter the divisor polynomial: 1011
```



WEEK 14

Write a program for congestion control using Leaky bucket algorithm.

CODE:

```
#include <stdio.h>
#include <stdlib.h> // Include this for the rand() function int main()
{
int buckets, outlets, k = 1, num, remaining; printf("Enter
Bucket size and outstream size\n"); scanf("%d %d",
&buckets, &outlets); remaining = buckets;
while (k)
{
num = rand() % 1000; // Generate a random number between 0 and 999 if (num <
remaining)
{
remaining = remaining - num;
printf("Packet of %d bytes accepted\n", num); // Added missing variable
}
```

```
else
{
printf("Packet of %d bytes is discarded\n", num);
}
if (buckets - remaining > outlets)
{
remaining += outlets; // Fixed the calculation
}
else
remaining = buckets;
printf("Remaining bytes: %d \n", remaining);
printf("If you want to stop input, press 0, otherwise, press 1\n"); scanf("%d", &k);
}
while (remaining < buckets) // Fixed the condition
{
if (buckets - remaining > outlets)
{
remaining += outlets; // Fixed the calculation
}
else
remaining = buckets;
printf("Remaining bytes: %d \n", remaining);
}
return 0; // Added a return statement to indicate successful completion
}
```

OUTPUT:

```
PS D:\VS Code> cd "d:\VS Code\OS\" ; if ($?) { gcc bucket.c -o bucket } ; if ($?) { .\bucket }
Enter Bucket size and outstream size
2000
100
Packet of 41 bytes accepted
Remaining bytes: 2000
If you want to stop input, press 0, otherwise, press 1
1
Packet of 467 bytes accepted
Remaining bytes: 1633
If you want to stop input, press 0, otherwise, press 1
1
Packet of 334 bytes accepted
Remaining bytes: 1399
If you want to stop input, press 0, otherwise, press 1
1
Packet of 500 bytes accepted
Remaining bytes: 999
If you want to stop input, press 0, otherwise, press 1
1
Packet of 169 bytes accepted
Remaining bytes: 930
If you want to stop input, press 0, otherwise, press 1
1
Packet of 724 bytes accepted
Remaining bytes: 306
If you want to stop input, press 0, otherwise, press 1
1
Packet of 478 bytes is discarded
Remaining bytes: 406
If you want to stop input, press 0, otherwise, press 1
1
Packet of 358 bytes accepted
Remaining bytes: 148
If you want to stop input, press 0, otherwise, press 1
1
Packet of 962 bytes is discarded
Remaining bytes: 248
If you want to stop input, press 0, otherwise, press 1
0
Remaining bytes: 348
Remaining bytes: 448
Remaining bytes: 548
Remaining bytes: 648
Remaining bytes: 748
```

```
Remaining bytes: 348
Remaining bytes: 448
Remaining bytes: 548
Remaining bytes: 648
Remaining bytes: 748
Remaining bytes: 848
Remaining bytes: 948
Remaining bytes: 1048
Remaining bytes: 1148
Remaining bytes: 1248
Remaining bytes: 1348
Remaining bytes: 1448
Remaining bytes: 1548
Remaining bytes: 1648
Remaining bytes: 1748
Remaining bytes: 1848
Remaining bytes: 1948
Remaining bytes: 2000
PS D:\VS Code\OS>
```

OBSERVATION:

* Write a program for congestion control using Leaky Bucket algorithm

```
#include<stdio.h>
int main()
{
  int incoming, outgoing, buck_size, n, store=0;
  printf("Enter bucket size:");
  scanf("%d", &buck_size);
  printf("Enter outgoing size:");
  scanf("%d", &outgoing);
  printf("Enter no. of inputs:");
  scanf("%d", &n);

  while(n!=0)
  {
    printf("Enter the Incoming packet size:");
```

```
scanf("%d", &incoming);
if (incoming <= (buck_size - store))
{
    store += incoming;
    printf("Bucket buffer size %d out of %d\n",
            store, buck_size);
}
else {
    printf("Dropped %d no. of packets\n",
        incoming - (buck_size - store));
    printf("Bucket buffer size %d out of %d\n",
            store, buck_size);
    store = buck_size;
}
store = store - outgoing;
printf("After outgoing %d packets left out of
    %d in buffer\n", store, buck_size);
        n--;
    }
}
```

output:

```
Enter bucket size: 5000
Enter outgoing rate: 2000
Enter number of inputs: 2
Enter the incoming packet size: 3000
Bucket buffer size 3000 out of 5000
After outgoing 1000 packets left out of 5000 in buffer
Enter the incoming packet size: 1000
Bucket buffer size 2000 out of 5000
After outgoing 0 packets left out of 5000
                                    in buffer.
```

29/8/2023

WEEK 15

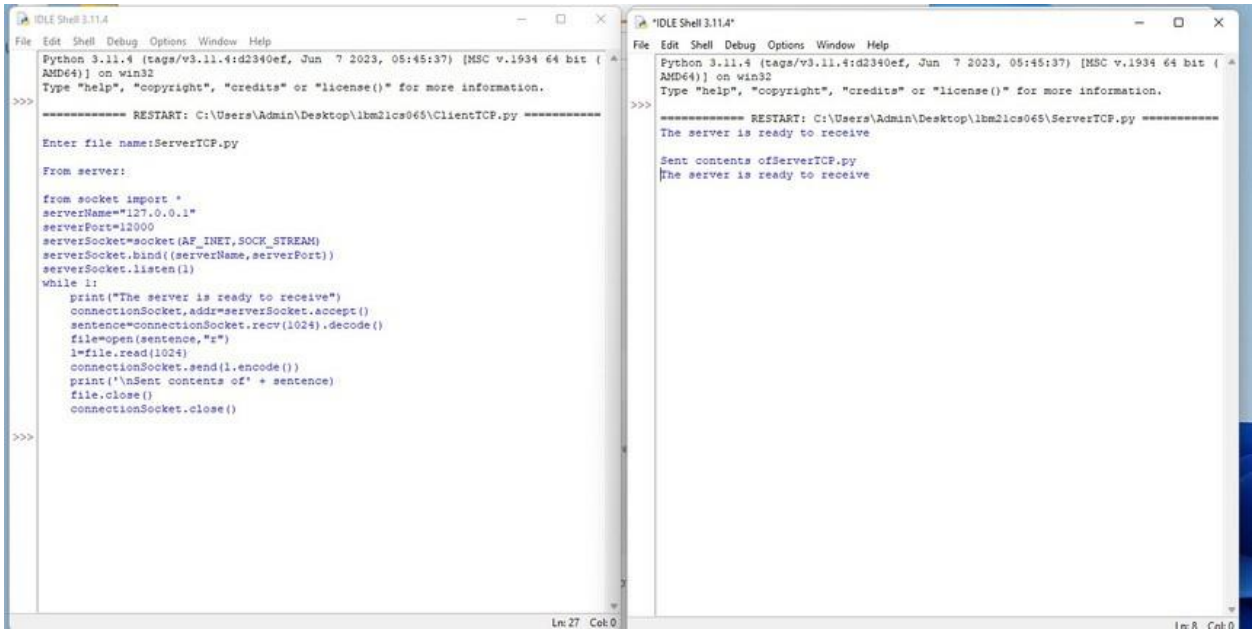
Using TCP/IP sockets, write a client-server program to make client sending the file name and the server to send back the contents of the requested file if present.

CODE:

ClientTCP.py from socket import * serverName = “127.0.0.1” serverPort = 12000 clientSocket = socket(AF_INET, SOCK_STREAM)
clientSocket.connect((serverName,serverPort))
sentence = input(“\nEnter file name: “)
clientSocket.send(sentence.encode()) filecontents = clientSocket.recv(1024).decode() print (“\nFrom Server:\n”) print(filecontents) clientSocket.close()

ServerTCP.py from socket import *
serverName=”127.0.0.1” serverPort = 12000 serverSocket = socket(AF_INET,SOCK_STREAM)
serverSocket.bind((serverName,serverPort))
serverSocket.listen(1) while 1: print (“The server is ready to receive”) connectionSocket, addr = serverSocket.accept() sentence = connectionSocket.recv(1024).decode()
file=open(sentence,”r”) l=file.read(1024)
connectionSocket.send(l.encode())
print (“\nSent contents of “ + sentence) file.close()
connectionSocket.close()

OUTPUT:



OBSERVATION:

24-8-23

Exp-3

- Using TCP/IP sockets, write a client-server program to make client sending the file name and the server to send back the contents of the requested file if present.

ClindTCP.py

```
from socket import *
serverName = '127.0.0.1'
serverPort = 12000
clientSocket = socket(AF_INET, SOCK_STREAM)
clientSocket.connect((serverName, serverPort))
    sentence = input("\n Enter file name :")

clientSocket.send(sentence.encode())
filecontents = clientSocket.recv(1024).decode()
print("\n From Server: \n")
print(filecontents)
clientSocket.close()
```

ServerTCP.py

```
from socket import *
serverName = "127.0.0.1"
serverPort = 12000
serverSocket = socket(AF_INET, SOCK_STREAM)
serverSocket.bind((serverName, serverPort))
serverSocket.listen(1)
  while 1:
        print("The server is ready to recieve")
    connectionSocket, addr = serverSocket.accept()
      sentence = connectionSocket.recv(1024).decode()
```

```
file = open (sentence, "r")
l = file.read (1024)

connectionSocket.send (l.encode())
print ("\n Sent contents of' + sentence)
file.close()
connectionSocket.close()
```

Output

Server is ready to recieve

In client

Enter file name : ServerTCP.py

found at server details will come

WEEK 16

Using UDP sockets, write a client-server program to make the client send the file name and the server to send back the contents of the requested file if present.

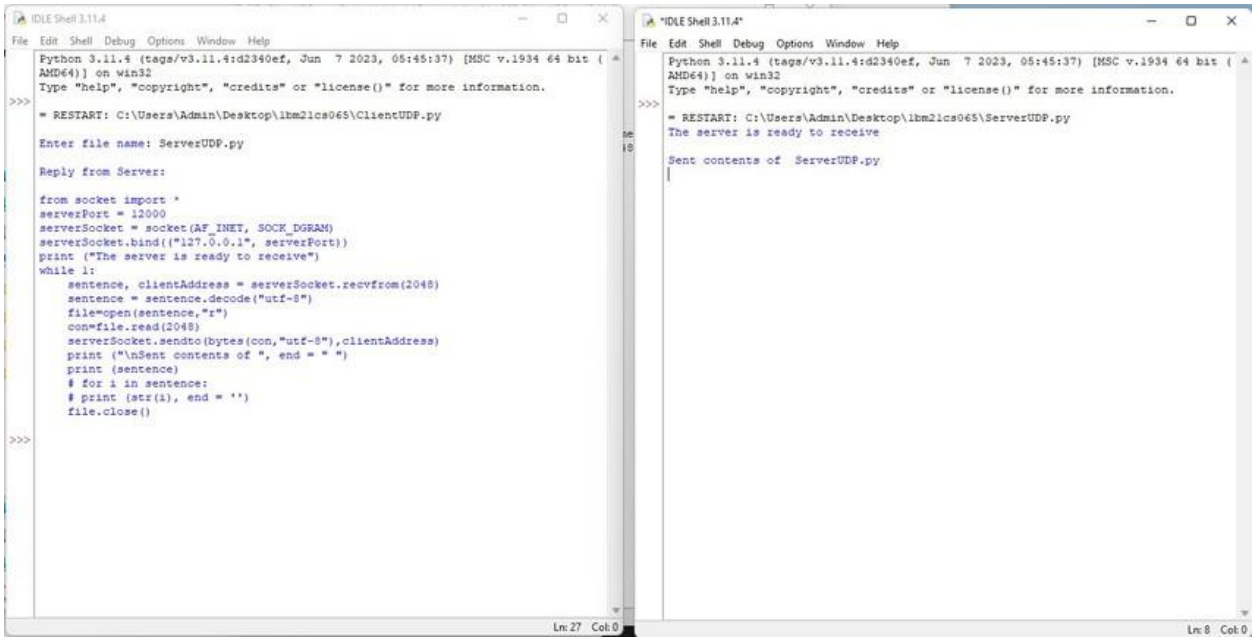
CODE:

ClientUDP.py

from socket import * serverName = “127.0.0.1”
serverPort = 12000 clientSocket = socket(AF_INET,
SOCK_DGRAM) sentence = input(“\nEnter file name:
“)
clientSocket.sendto(bytes(sentence,”utf-8”),(serverName,
serverPort)) filecontents,serverAddress =
clientSocket.recvfrom(2048) print (“\nReply from Server:\n”) print
(filecontents.decode(“utf-8”)) # for i in filecontents: # print(str(i),
end = “ ”) clientSocket.close() clientSocket.close() ServerUDP.py

from socket import * serverPort = 12000 serverSocket =
socket(AF_INET, SOCK_DGRAM) serverSocket.bind((“127.0.0.1”,
serverPort)) print (“The server is ready to receive”) while 1:
sentence, clientAddress = serverSocket.recvfrom(2048)
sentence = sentence.decode(“utf-8”) file=open(sentence,”r”)
con=file.read(2048)
serverSocket.sendto(bytes(con,”utf-8”),clientAddress)
print (“\nSent contents of “, end = “ “) print
(sentence) # for i in sentence: # print (str(i),
end = “ “) file.close()

OUTPUT:



OBSERVATION:

* Using UDP sockets, write a client-server program to make client sending the file name and the server to send back the contents of the requested file if present.

Client UDP.py

```
from socket import *
serverName = "127.0.0.1"
serverPort = 12000
clientSocket = socket(AF_INET, SOCK_DGRAM)

sentence = input("\nEnter file name :")

clientSocket.sendto(bytes(sentence, "utf-8"), (serverName,
                                    serverPort))

fileContents, serverAddress = clientSocket.recvfrom(2048)
print("\nReply from Server:\n")
print(fileContents.decode("utf-8"))
```

29/8/2023

```
# for i in filecontents:
    # print (str(i), end = '')
clientSocket.close()
clientSocket.close()
```

ServerUDP.py

```
from socket import *
serverPort= 12000
serverSocket= socket(AF_INET, SOCK_DGRAM)
serverSocket.bind(("127.0.0.1", serverPort))
print("The server is ready to receive")
while 1:
    sentence, clientAddress = serverSocket.recvfrom(2048)
    sentence = sentence.decode("utf-8")
    file= open(sentence,"r")
    con = file.read(2048)

    serverSocket.sendto(bytes(con,"utf-8"), clientAddress)

    print('\n Sent contents of', end = ' ')
    print(sentence)
    # for i in sentence:
        # print(str(i), end = '')
    file.close()
```

output

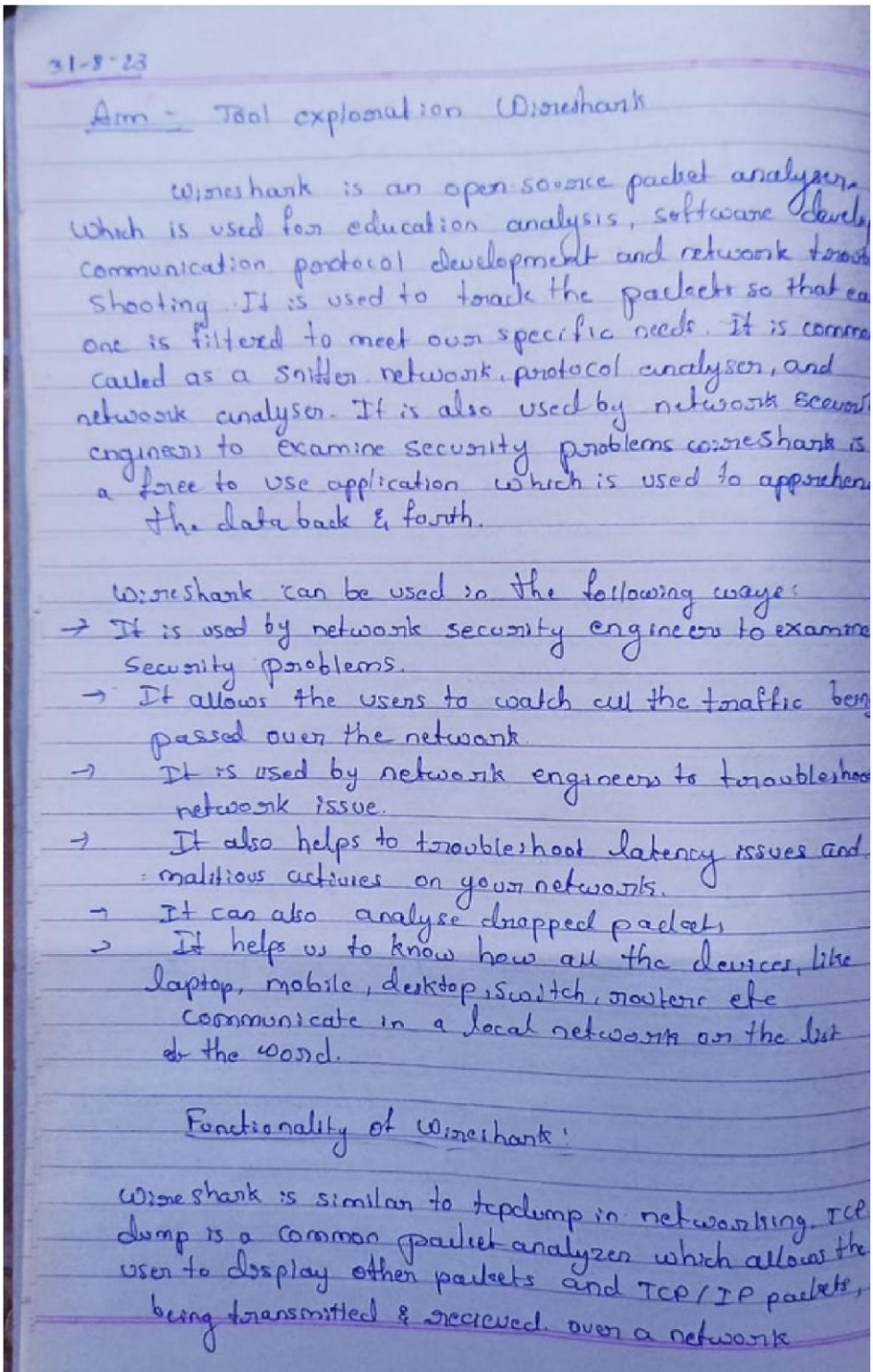
```
Enter file name :ServerUDP.py

Reply from Server:
whole serverUDP contents

The server is ready to receive

Sent contents of ServerUDP.py
The server is ready to receive
```

WEEK 17

Tool Exploration -

Wireshark

OBSERVATION:



31-8-23

Aim :- Tool exploration Wireshark

Wireshark is an open source packet analyzer. Which is used for education analysis, software devel, communication protocol development and network trouble Shooting. It is used to track the packets so that ea one is filtered to meet our specific needs. It is comme called as a Sniffer network, protocol analyser, and network analyser. It is also used by network securi engineers to examine security problems. Wireshark is a free to use application which is used to apprehen the data back & forth.

Wireshark can be used in the following ways:

→ It is used by network security engineers to examine Security problems.
→ It allows the users to watch all the traffic being passed over the network.
→ It is used by network engineers to troubleshoo network issue.
→ It also helps to troubleshoot latency issues and malitious activites on your networks.
→ It can also analyse dropped packets.
→ It helps us to know how all the devices, like laptop, mobile, desktop, switch, routers etc communicate in a local network or the list of the world.

Functionality of Wireshark:

Wireshark is similar to tcpdump in networking. TCP dump is a common packet analyzer which allows the user to display other packets and TCP/IP packets, being transmitted & received over a network

attached to the computer. It has a graphic end and some sorting & filtering functions. Wireshark users can see all the traffic passing through the network wireshark can also monitor the unicast traffic which is not sent to networks MAC address interface. But, the switch does not pass all the traffic to port. Hence, the promiscious mode is not sufficient to see all the traffic. The various network taps or port mirroring is used to extend capture at any point. Port mirroring is a method to monitor network traffic. When it is enabled, the switch sends the copies of all the network packets present at one port to another port.

NP
11/9/2023