

VISVESVARAYA TECHNOLOGICAL UNIVERSITY

“JnanaSangama”, Belgaum -590014, Karnataka.



LAB REPORT

on

COMPUTER NETWORKS

Submitted by

CHAITANYA RAVINDRA(1BM21CS041)

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 **CHAITANYA RAVINDRA (1BM21CS041)**, who is 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
Department of CSE
BMSCE, Bengaluru

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

Index

Sl. No.	Date	Experiment Title	Page No.
1	15/6/23	Create a topology and simulate sending a simple PDU from source to destination using hub and switch as connecting devices and demonstrate ping message.	5
2	27/7/23	Configure IP address to routers in packet tracer. Explore the following messages: ping responses, destination unreachable, request timed out, reply	12
3	27/7/23	Configure default route, static route to the Router	22
4	27/7/23	Configure DHCP within a LAN and outside LAN	25
5	27/7/23	Configure Web Server, DNS within a LAN	32
6	27/7/23	Configure RIP routing Protocol in Routers	35
7	27/7/23	Configure OSPF routing protocol	39
8	1/9/23	To construct simple LAN and understand the concept and operation of Address Resolution Protocol (ARP)	42
9	1/9/23	To construct a VLAN and make the PC's communicate among a VLAN	45
10	1/9/23	To construct a WLAN and make the nodes communicate wirelessly	48
11	1/9/23	Demonstrate the TTL/ Life of a Packet	52

12	1/9/23	To understand the operation of TELNET by accessing the router in server room from a PC in IT office.	55
		CYCLE 2	
13	1/9/23	Write a program for error detecting code using CRCCCITT (16-bits)	59
14	1/9/23	Write a program for congestion control using Leaky bucket algorithm	64
15	1/9/23	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.	67
16	1/9/23	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	75
17	1/9/23	Tool Exploration -Wireshark	80

EXPERIMENT-1

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

Aim: Create a topology ~~fever~~ to simulate a simple PDU from source to destination using switch and hubs as connecting devices.

Topology Hub to PC's

Hub - PT - Hub 0

PC - PT - PC 0 PC - PT - PC 1 PC - PT - PC 2

Procedure

- 1) Select 3 PC's and a hub
- 2) Connect 3 PC's to hub using copper straight through wire
- 3) Set IP address of PC's as 10.0.0.1, 10.0.0.2, 10.0.0.3 respectively
- 4) Now select source and destination PC and send a PDU packet for ex:- from PC-0 to PC-2
- 5) The PDU packet will go from PC to PC-1 and PC-2 but will be accepted at PC-2 only

Observation:-

PDV packet is received at PC-2 as it is selected as destination PDV sent to PC-1 is discarded.

Output:-

Reply from 10.0.0.2 : byte = 32 time = 5ms TTL = 120
 Reply from 10.0.0.2 : byte = 32 time = 0ms TTL = 120
 Reply from 10.0.0.2 : byte = 32 time = 6ms TTL = 120
 Reply from 10.0.0.2 : byte = 32 time = 0ms TTL = 120

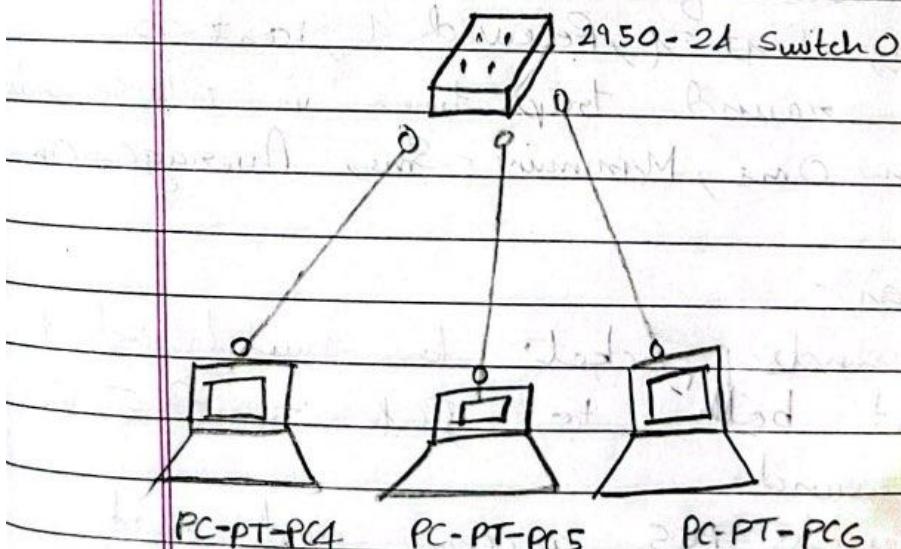
Ping Statistics for 10.0.0.2 :-

Packets : sent = 4 , Received = 4 , Lost = 0

Approximate round trip times in millisecond

Minimum = 0ms , Maximum = 3ms , Average = 0m

Topology Switch to PC's



- 1) Select three PC's and switch
- 2) Connect them using copper straight through wire
- 3) Set IP address as 10.0.0.4, 10.0.0.5, 10.0.0.6 respectively for the PC's
- 4) Now select source and destination PC for the instance PC-3 to PC5 and send a packet.

Output

Reply from 10.0.0.6 : Bytes=32 time=0ms TTL=120
 Reply from 10.0.0.6 : Bytes=32 time=0ms TTL=120
 Reply from 10.0.0.6 : Bytes=32 time=0ms TTL=120
 Reply from 10.0.0.6 : Bytes=32 time=0ms TTL=120

Ping statistics for 10.0.0.6

Package: Sent = 4, Received = 4, lost = 0

Approx round trip times in milli-second

Minimum = 0ms, Maximum = 3ms, Average = 0ms

Observation:-

PC3 sends packets to switch and it sends it both to PC4 to PC5 in first round

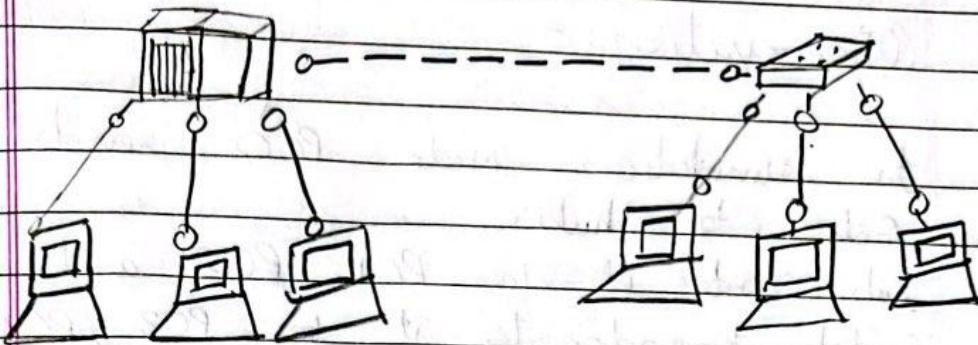
PC4 rejects PC5 accepts and send acknowledgement packet to both PC3 & PC5

PC5 discards the PC3 and accepts it

Now, when PC3 sends packets it

Topology

PC, hub and switch



Procedure

- 1) Select 6 PC's and switch and hub
- 2) Connect 3 PC's to hub and 3 PC's to switch using copper straight through wire and connect switch and hub using copper cross-over piece.
- 3) Set IP address of all 6 PC's respectively as 10.0.0.1 and so on.
- 4) Now select source and destination and send a PDU packet from PC-0 to PC-4

Output

Reply from 10.0.0.4 : bytes=32 time=0ms TTL=120
 Reply from 10.0.0.4 : bytes=32 time=0ms TTL=120
 Reply from 10.0.0.4 : bytes=32 time=4ms TTL=120
 Reply from 10.0.0.4 : bytes=32 time=0ms TTL=120

Minimum = One, Maximum = ∞ , Average = $\frac{1}{2}$

Observation:-

In simulation mode PC0 sends packet to hub.

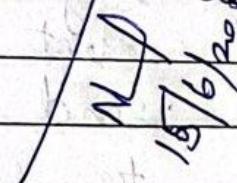
hub sends it to PC1, PC2 and switch broadcasts it to PC3, PC4 and PC5.

PC1, PC2, PC4 and PC5 discard them. PC3 accepts and sends acknowledgement to hub through switch.

Only PC0 accepts others discard. In second round PC0 sends packet to hub. It broadcasts to PC1, PC2, switch.

Switch broadcasts only to PC3 thus it's a smart logical

way to do it.



length

Cisco Packet Tracer Student

File Edit Options New Tools Extensions Help

Logical [Root] New Cluster Move Object Set Tiled Background Viewport

Simulation Panel

Via	SimObject	Last Device	All Device	Type	Info
5.429	Switch1	Hub0	ICMP		
5.430	Hub0	PC0	ICMP		
5.431	Hub0	PC1	ICMP		
5.432	Hub0	PC2	ICMP		
5.433	Hub0	PC3	ICMP		
7.427	-	Switch1	STP		
7.428	Switch1	Hub0	STP		
7.429	Switch1	PC0	STP		
7.430	Switch1	PC1	STP		
7.431	Switch1	PC2	STP		
7.432	Switch1	PC3	STP		

Reset Simulation Constant Delay Captured to a 7.429.s

Play Controls Back Auto Capture / Play Capture / Forward

Event List Filters - Visible Events

ACL, Ping, ARP, BGP, CDP, DHCP, DHCPOFFER, DHCPOFROFER, DHCPOACK, DHCPOFROACK, EIGRP, FR, FRV, HSRP, IGMP, OSPF, OSPFv3, PAgP, POP3, RADIUS, RIPv1, RIPv2, RIPv3, SCDP, SHM, STP, STP-SYLOG, TACACS, TDR, TFTP, Telnet, VTP, VTPv2

Edit Filter Show All None

Time: 00:01:21.954 Power Cycle Devices PLAY CONTROLS: Back Auto Capture / Play | Capture / Forward Event List Simulation

Connections

Automatically Choose Connection Type

Event List

Scenario	Run	Last Status	Source	Destination	Type	Time(s)	Periodic	Num	Edt	Delete
1	Successful	PC0	PC2	ICMP	0.000	N	0	(edit)	[Delete]	
1	Successful	PC0	PC3	ICMP	3.422	N	1	(edit)	[Delete]	
1	Successful	PC0	PC4	ICMP	5.423	N	2	(edit)	[Delete]	

Event List

Simulation

PC0

Terminal Config Desktop Custom Interface

Command Prompt

```

Packet Tracer PC Command Line 1.0
PCping 192.160.1.6
Pinging 192.160.1.6 with 32 bytes of data:
Reply from 192.160.1.6: bytes=32 time<1ms TTL=128

Ping statistics for 192.160.1.6:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

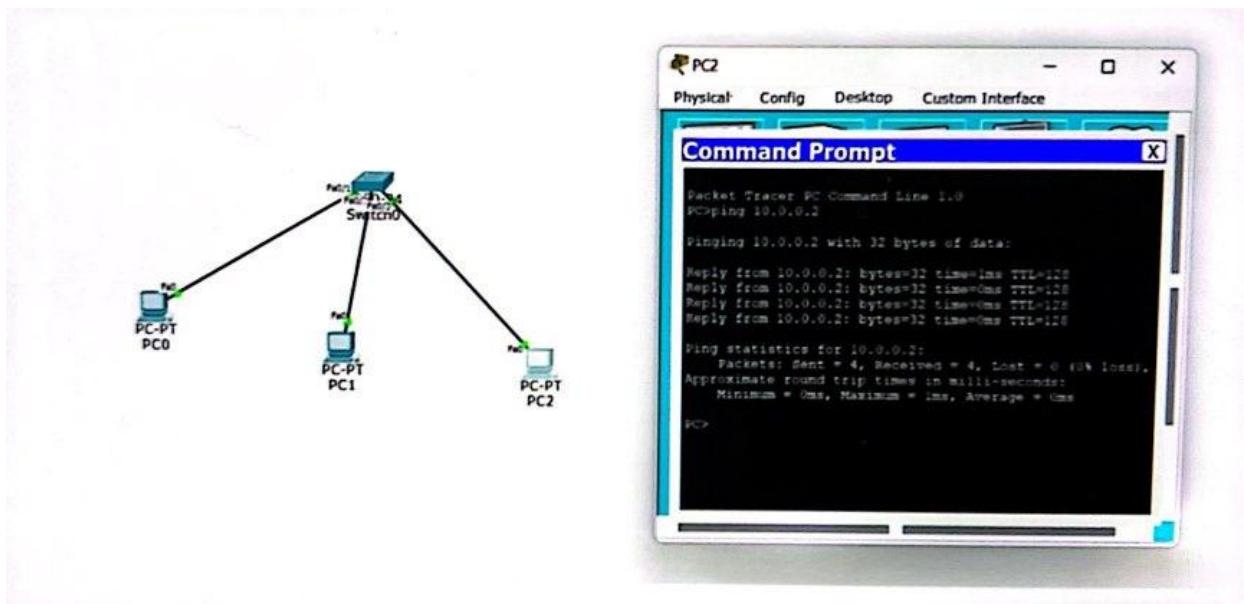
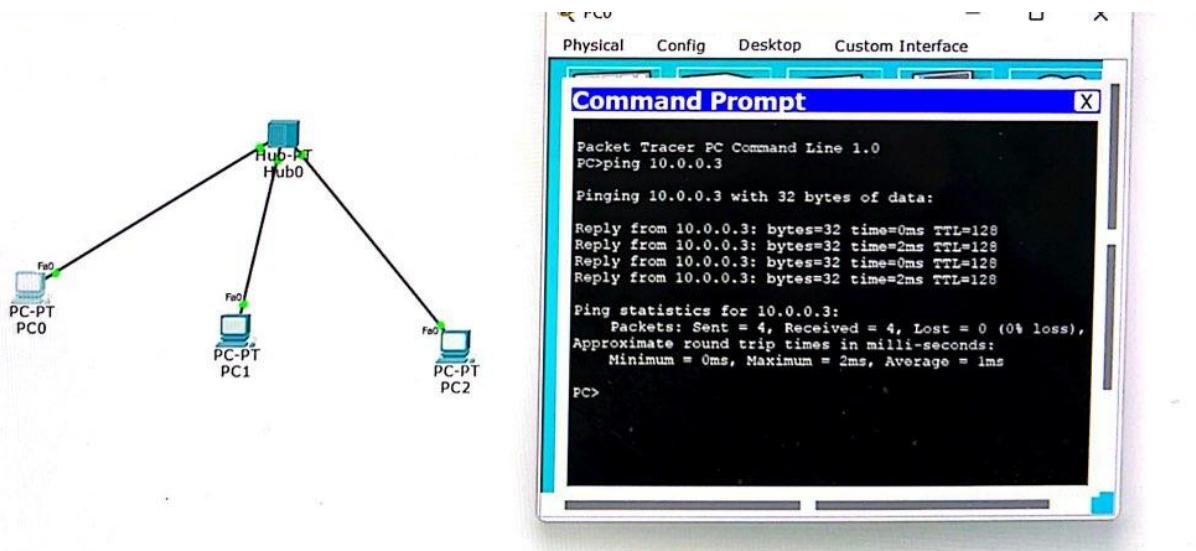
PCping 192.160.1.8
Pinging 192.160.1.8 with 32 bytes of data:
Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.160.1.8:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
PC>192.160.1.2
Invalid Command.

PCping 192.160.1.2
Pinging 192.160.1.2 with 32 bytes of data:
Reply from 192.160.1.2: bytes=32 time<1ms TTL=128

Ping statistics for 192.160.1.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
PC>

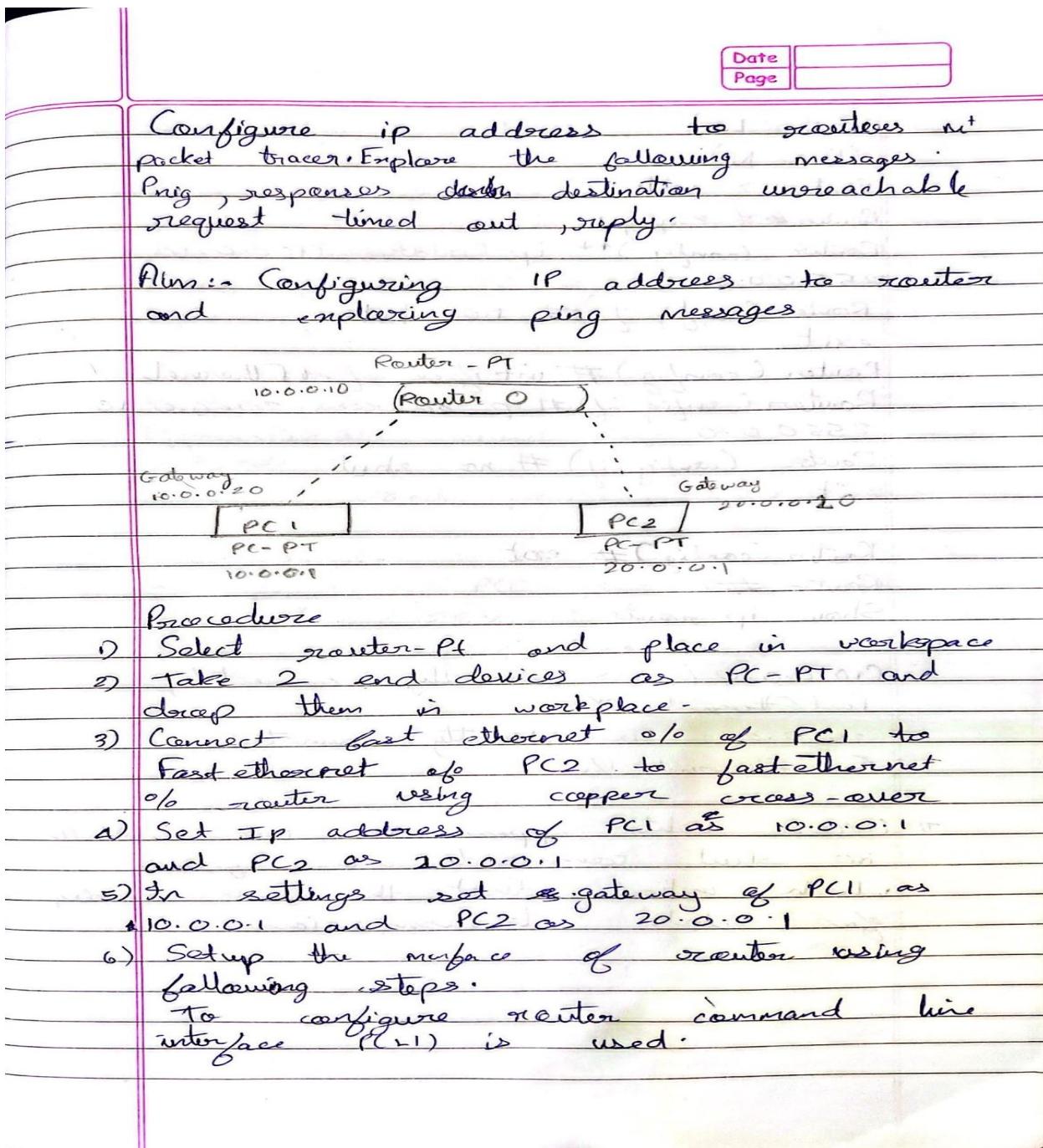
```



EXPERIMENT-2

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

PROGRAM 2.1



Router (LI)

(Press N)

Router > enable

Router # > config +

Router (config) ++ ip address 10.0.0.10

255.0.0.0

Router (config-if) # no shut

exit

Router (config) # interface fastEthernet

Router (config-if) # ip address 20.0.0.10

255.0.0.0

Router (config-if) # no shut

exit

Router (config) # exit

Router #

Show ip route

10.0.0.0/8 is directly connected,
Fast Ethernet 0/0

200.0.0.0/8 is directly connected,
Fast Ethernet 1/0

- 1) Green lights appear on wires when no shut commands are given which indicate that they are ready for data transmission

Date	
Page	

Ping output in PCO:-

P(> 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 = 0 ms TTL = 127

Reply from 20.0.0.1: bytes = 32 time = 0 ms TTL = 127

Ping statistics for 20.0.0.1

Routes:- Sent = 4, received 3: Loss = 1 (25% loss),

Approximate round trip times in milliseconds

minimum = 0 ms, maximum = 1 ms, Average

Observations:-

On 1st ping in PCO for the first two, there is a 25% loss.

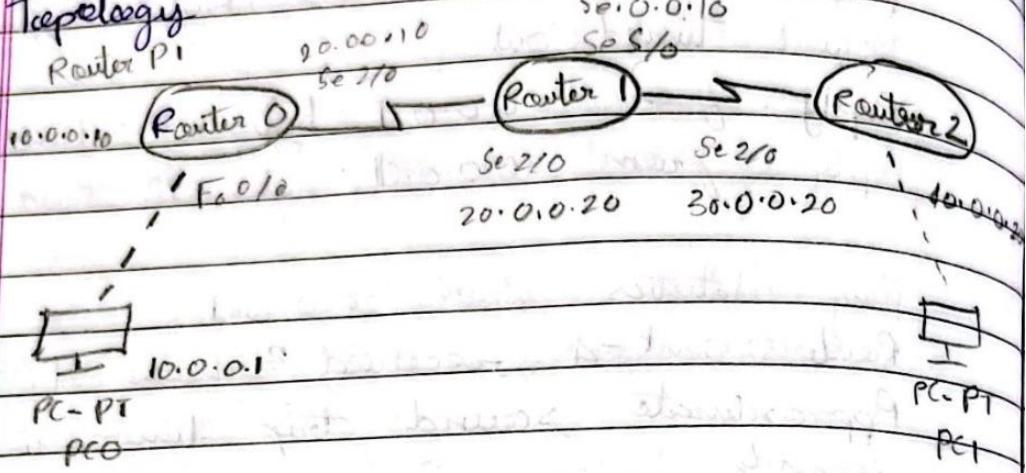
From next ping there are no losses.

N
27/12/2023

2b) Aim configure using 3 router and 2 PCs

Router

Topology



Procedure

- 1) The network is started by selecting ~~end~~ end devices PC0 to PC1 to genuine PC's and placing them in workspace.
- 2) Select 3 router-PT and place them as router 0, router 1, and router workspace.
- 3) PC0 & PC1 are connected to router 0 and router 2 respectively using copper crossover.
- 4) Connect router 0 to router 1, router 1 to router 2.
- 5) Set up IP address of PC0 to 10.0.0.1 and PC1 to 10.0.0.2 as default gateway.

Configure router by opening CLI

In Router 0

Router > enable

Router # config t

Router (config) # interface fastEthernet 0/0

Router (config-if) # ip address 10.0.0.10
255.0.0.0

Router (config-if) # no shut
exit

Router (config) # interface serial 2/0

Router (config-if) # ip address 20.0.0.10
255.0.0.0

Router (config) # ip address 20.0.0.10 no shut
exit
exit

In Router 1

Router > enable

Router # config t

Router (config) # interface serial 2/0

Router (config-if) # ip address 20.0.0.20
to 255.0.0.0

Router (config-if) no shut
exit

Router (config) # interface serial
3/0

Router (config-if) # ip address
30.0.0.20 255.0.0.0

In router 2

Router > enable

Router # config t

Router (config-if) # interface serial 2/0
Router (config-if) # ip address 30.0.0.2
255.0.0.0

Router (config-if) # no shut

exit

Router (config-if) # interface fastEthernet
0/0

Router (config-if) # ip address 40.0.0.10
255.0.0.0

Router (config-if) no shut

exit

Router (config) exit

IP Router table

Router 0

Router # show ip route

C 10.0.0.0/8 > directly connected, FastEthernet 0/0

C 20.0.0.0/8 > directly connected, Serial 2/0

Router 1

Router # show ip route

C 20.0.0.0/8 > directly connected

C 30.0.0.0/8 > directly connected

Ping output in PC0

PC > Ping 10.0.0.1

Pinging 10.0.0.1 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

Ping statistics for 10.0.0.1

• Packets: sent = 4, received = 0, loss = 1 (100% loss)

Observation:

Green lights appear on the wires when no shout is written.

New configure the router which does not have data of other network. Add the network in CLI. In all stations, CLI write config t. then set route

Routers

ip route 30.0.0.0 255.0.0.0 20.0.0.30

ip route 40.0.0.0 254.0.0.0 20.0.0.30

Router 1

ip route 10.0.0.0 255.0.0.0 20.0.0.10

ip route 20.0.0.0 255.0.0.0 30.0.0.20

Router 2

new ip router table
exit

Router 0:

c 10.0.0.0/8 is directly connected, fast

ethernet 0/0

c 20.0.0.0/8 is directly connected, serial

[30.0.0.0/8] via 30.0.0.30

s 40.0.0.0/8 [1/0] via 20.0.0.20

Router 1:

s 10.0.0.0/8 [1/0] via 30.0.0.10

s 20.0.0.0/8 [1/0] via 30.0.0.10

c 30.0.0.0/8 is directly connected, serial

[40.0.0.0/8] via 40.0.0.10

Router 2:

s 10.0.0.0/8 [1/0] via 20.0.0.10

c 20.0.0.0/8 is directly connected serial

[30.0.0.0/8] via 30.0.0.20

s 40.0.0.0/8 [1/0] via 30.0.0.20

Ping messages

PC > Ping 10.0.0.1

Pinging 10.0.0.1 with 32 bytes of data

Reply from 90.0.0.1: bytes = 32 time = 2ms

TTL = 125

Reply from 90.0.0.21: bytes = 32 time = 2ms

TTL = 125

Ping statistics for 90.0.0.1

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

Approximate round trip times in milliseconds:

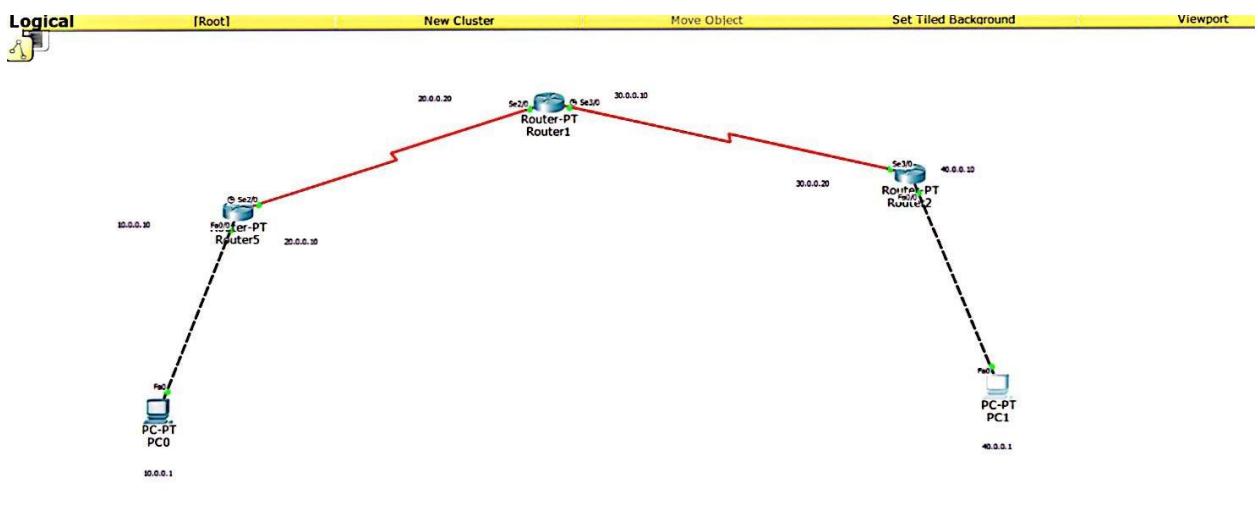
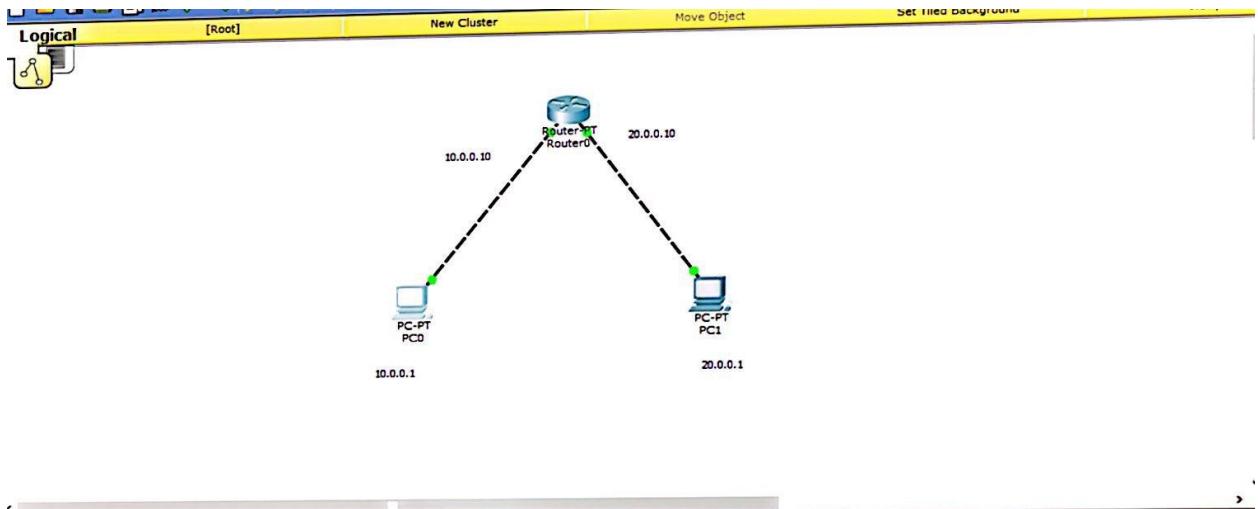
minimum = 2ms, maximum = 2ms, Average = 2ms

Observations :-

In first ping destination host was unreachable as router 0 has no knowledge about the network 30.0.0.0 and 40.0.0.0 and the packets get stuck or lost.

After this host is explicitly known pinging them is 25% loss in first time, the following ones have no loss.

ND
27/7/2023



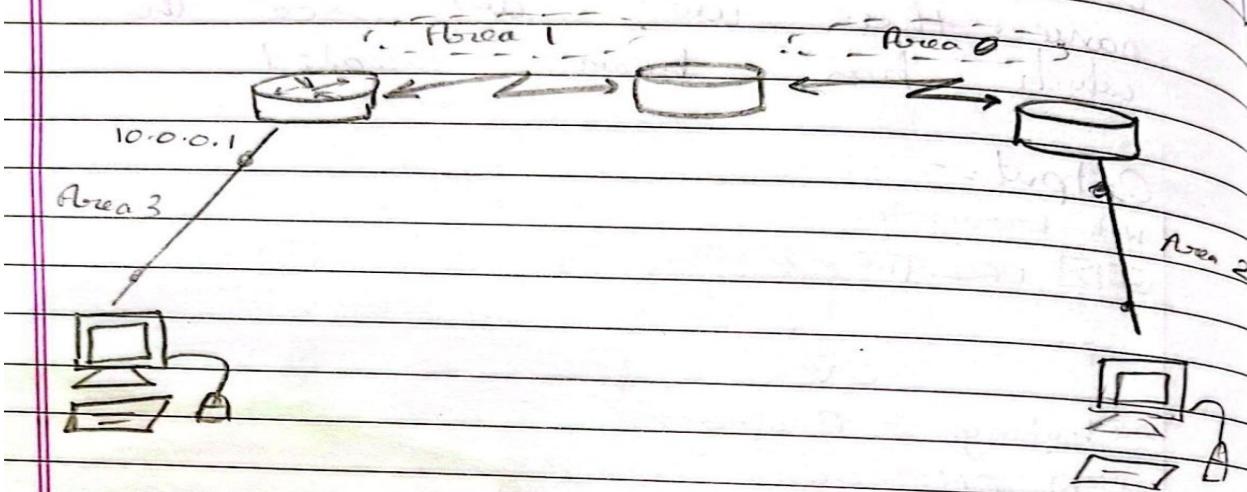
EXPERIMENT-3

Q) Configure default route, static route to the Router

Q1 ab-3

Ans :- Configure default router, static routes to the router

Topology



Procedure

- 1) Configure the PC with IP address and gateway according to the topology
- 2) Configure the router accordingly with the IP addresses given. Encapsulation PPP and clock rate need to be set as done in R1 protocol experiment
- 3) In Router R1
R1 (config)# router
R1 (config-if)# ip route . . .

Reply from 90.0.0.1: bytes = 32 time = 2ms

TTL = 125

Reply from 90.0.0.21: bytes = 32 time = 2ms

TTL = 125

Ping statistics for 90.0.0.1

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

Approximate round trip times in milliseconds:

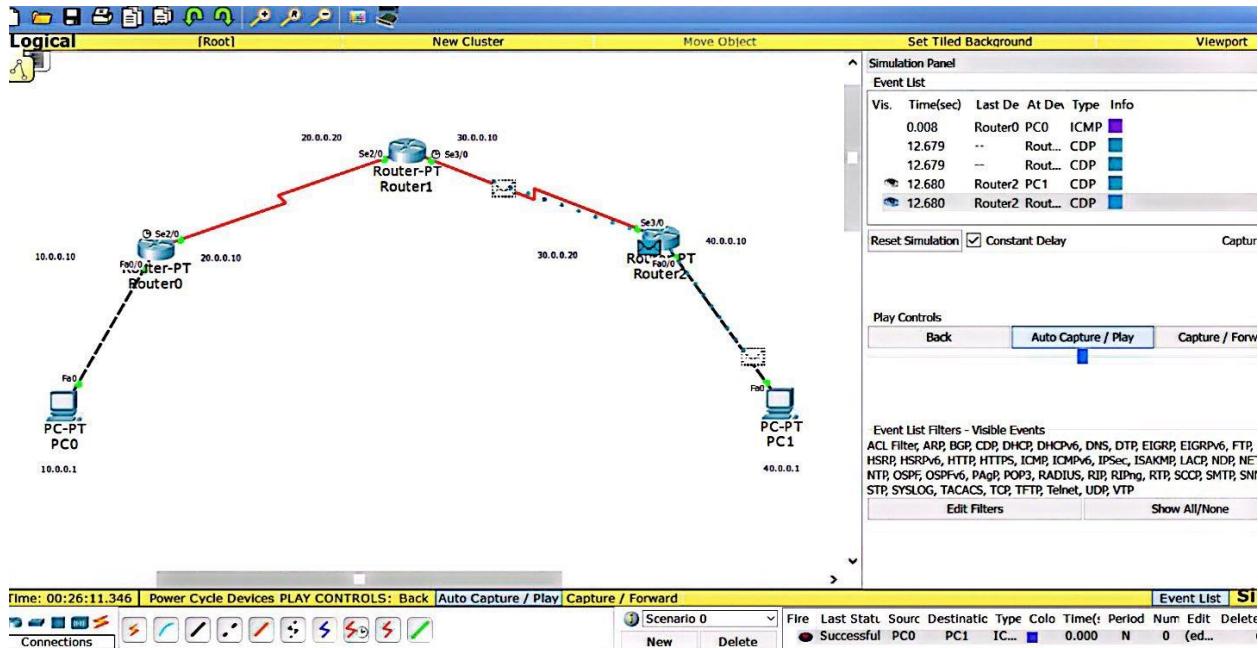
minimum = 2ms, maximum = 2ms, Average = 2ms

Observations :-

In first ping destination host was unreachable as router 0 has no knowledge about the network 30.0.0.0 and 40.0.0.0 and the packets get stuck or lost.

After this host is explicitly known pinging them is 25% loss in first time, the following ones have no loss.

ND
27/7/2023



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 40.0.0.1: bytes=32 time=12ms TTL=125
Reply from 40.0.0.1: bytes=32 time=13ms TTL=125
Reply from 40.0.0.1: bytes=32 time=7ms 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 = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 7ms, Maximum = 13ms, Average = 10ms

PC>

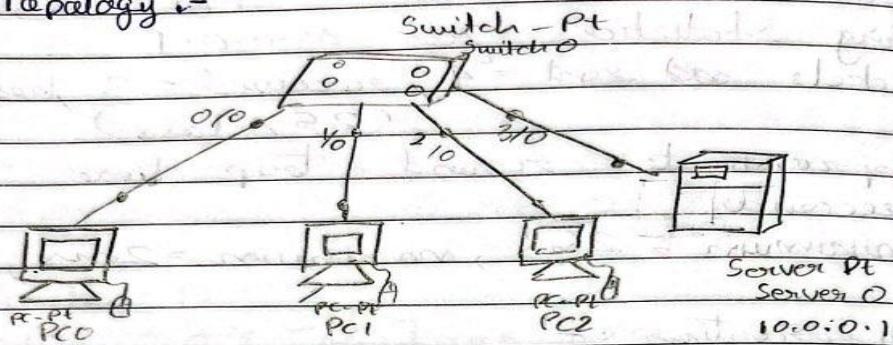
```

Experiment-4

Q) Configure DHCP within a LAN and outside LAN

Aim: Configure DHCP within a LAN and outside LAN

Topology :-



Procedure

- 1) Connect 3 PCs and 1 server to the switch using copper straight through cable.
- 2) Click on server and go to services tab. Select DHCP and turn on the DHCP service.
- 3) Set IP address of the start IP address as 10.0.0.1.
- 4) Now click on PC0 and then go to desktop tab, click on config. Select DHCP here. It will request for an IP address and get DHCP request also set IP address.
- 5) Repeat step for other 2 PCs.
- 6) To send a packet across PCs go to command prompt and type destination address.

Ping Output

Packet tracer PC command line 1.0

PC > Ping 10.0.0.3

Pinging 10.0.0.3 with 32 bytes of data 4

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 from 10.0.0.3

Packet: Sent = 4 Received = 4 Lost = 0 (0% loss)

Approx round trip time in milli-seconds

Minimum = 0ms, Maximum = 1ms, Average = 0ms

Observation :-

DHCP is used to dynamically assign an IP address to any device or node.

It is a client-server protocol in which

server manages a pool of unique IP

addresses & also about client config

parameters. DHCP enabled clients sends a

request to DHCP server.

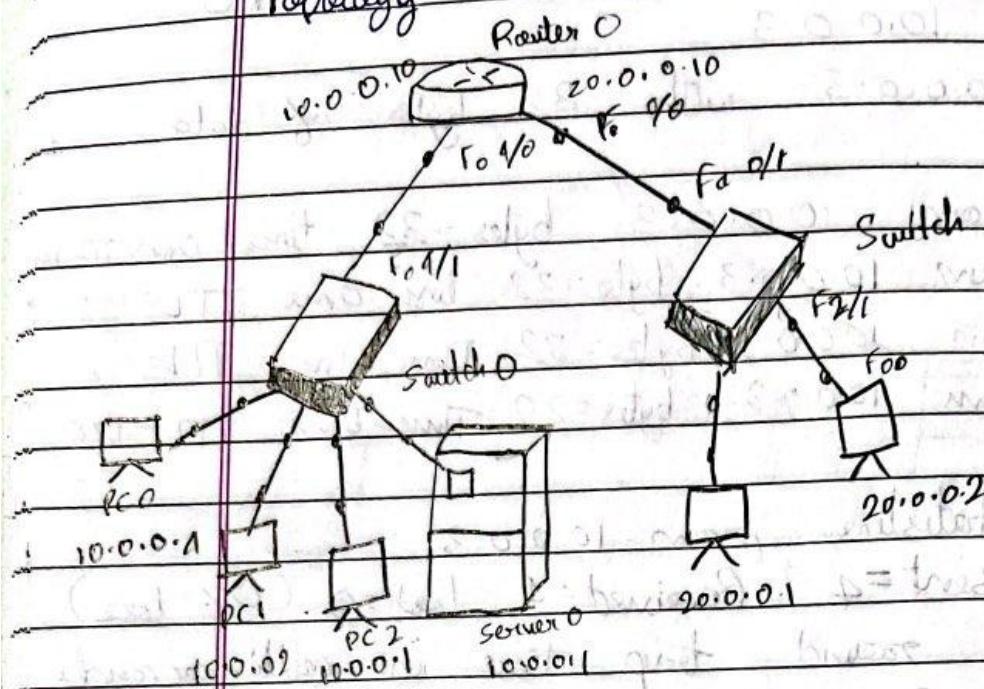
DHCP server responds by providing

config information, previously specified by

network administrator

Outside of LAN

Topology :-



Procedure

- Follow previous steps pertaining to Inside LAN by creating a $10.0.0.0$ network with easier IP addresses $10.0.0.1$ and gateway
- Create another network with 2PC's and a switch and connect 2 networks using a router.
- Config the router to connect the 2 networks using router
 - Config router to connect 2 networks through gateway
-> enable
 - > config +
 - > interface F0/0/0
 - > ip address 10.0.0.20 255.0.0.0

> interface fa 0/0

ip address 20.0.0.20 255.0.0.0

> no shut

> exit

Set another DHCP pool gateway to 20.0.0.10
and Then the in the CLI commands
of router set the server as a
ip address helper

The following are the 2 pools

Pool name	Default gateway	Ans	ServIP	Subnet mask	Waste
Server Pool	10.0.0.10	0.0.0.0	10.0.0.2	255.0.0.0	512
Server Pool	20.0.0.10	0.0.0.0	20.0.0.1	255.0.0.0	512

> config t

> interface fa 0/0

> ip helper address < desired ip address >

> no shut

> exit

Ping from 10.0.0.2 to 20.0.0.2

Result

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

Request 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
Approx time -
Minimum = 0ms, Max = 1ms, Average = 1ms

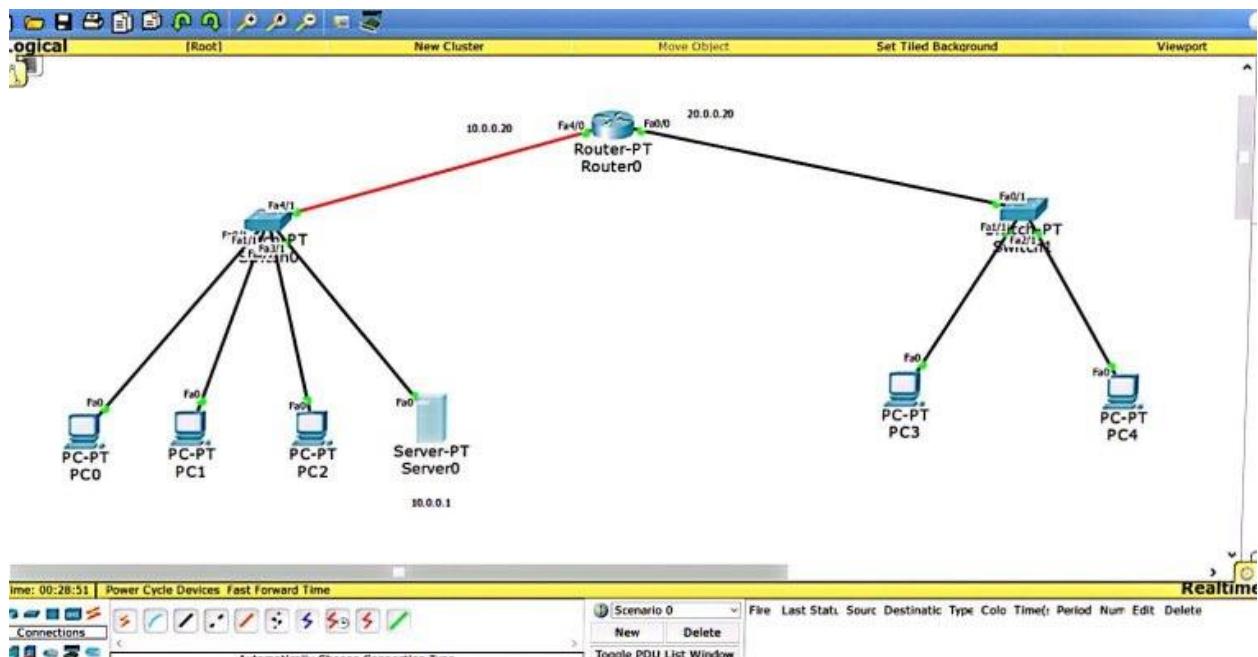
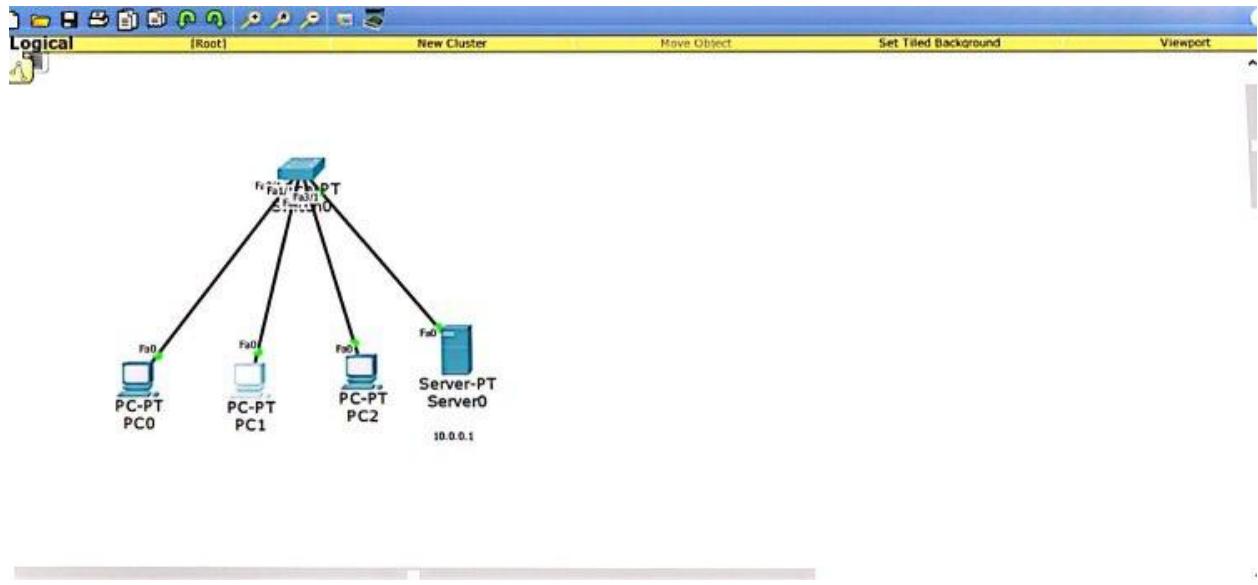
Observation:-

- The DHCP helps manage allocation of IP addresses to end users.
- The device ~~accessing~~ ^{requesting} coating to access a network gets and IP address allocated dynamically to it by the user.

The allocated IP address is taken back when shutdown.

If the requesting device is outside the LAN then device IP address must be assigned to the router as the address helps see that it can automatically configure that device's IP address.

✓ 21/10/23



```
PC>ping 10.0.0.4  
Pinging 10.0.0.4 with 32 bytes of data:  
Reply from 10.0.0.4: bytes=32 time=1ms TTL=128  
Reply from 10.0.0.4: bytes=32 time=0ms TTL=128  
Reply from 10.0.0.4: bytes=32 time=1ms TTL=128  
Reply from 10.0.0.4: bytes=32 time=0ms TTL=128  
  
Ping statistics for 10.0.0.4:  
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),  
Approximate round trip times in milli-seconds:  
    Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

PC>

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>

EXPERIMENT-5

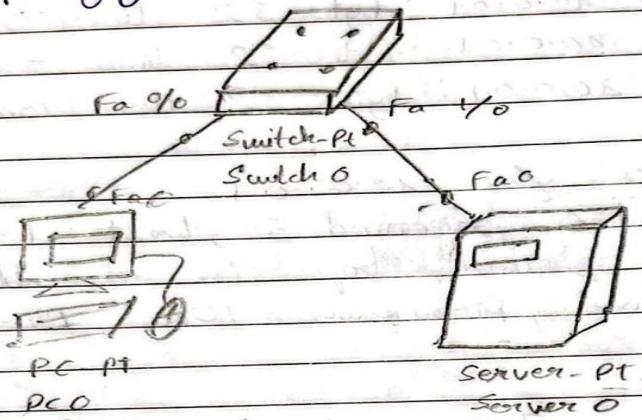
Q) Configure Web Server, DNS within a LAN

Lab - 7

Aim :-

Configure web server, DNS within a LAN

Topology :-



Procedure :-

- Connect a switch, PC and a server to form a LAN.
- Set PC's IP address by clicking on it and go to config tab, in fastEthernet option set IP address as 10.0.0.1 and submit mark.
- Set server IP address as 10.0.0.2 and submit mark.
- Go to PC's desktop and click on web browser and enter URL : 10.0.0.2
- To make a CV and click on save
- Again go to PC → Desktop → web browser and type 10.0.0.2

- Next go to Services → services → DNS and switch on services. Now add a domain name and type IP address as 10.0.0.2. Press add & save it.
- Again go back to PC → desktop - web browser and type the given domain name. Then we can see the CV which has been created.

Output:-

Web browser

[URL

CV

Chaitanya Ravindra

VSN : IBM21CS011

Languages : C/C#/Java

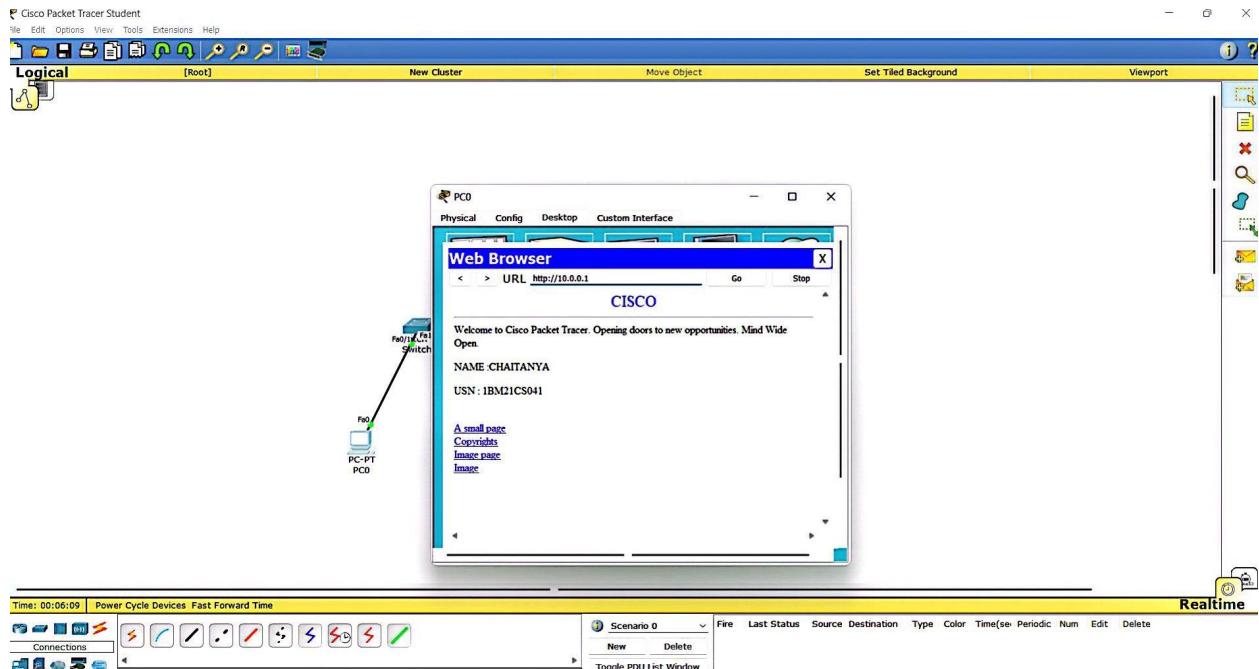
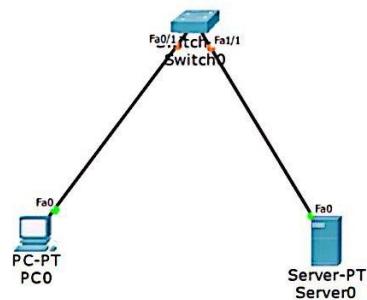
Tunge : 20/07/2023

Observation :-

- If you wanted to go to a certain website you would open web browser and type domain name of that website or else you can also type IP address instead of website IP.

Since we can't remember IP of all websites DNS server will search through its cache to

J



Experiment-6

Q) Configure RIP routing Protocol in Routers

Date _____
Page _____

Lab 6

Aim:
Configure RIP routing protocol in routers

Topology:-

Router 0: 20.0.0.10 (connected to PC-1 at 10.0.0.1)

Router 1: 20.0.0.10 (connected to Router 0 at Se2/0, connected to Router 2 at Se3/0)

Router 2: 10.0.0.2 (connected to Router 1 at Se3/0)

PC-1: 10.0.0.1

PC-2: 10.0.0.2

Procedure

- Create a network using 3 routers and 2 PCs
Connect routers using serial DCF cable
and PC to router using copper crossover cable.
- Set IP and gateway no for both PC's
as 10.0.0.1 - IP 10.0.0.10 - gateway - PC0
pcos 10.0.0.1 - IP 10.0.0.10 - gateway - PC1
respectively
- Go to router CLI mode and execute commands
Step 1 - No
Step 2 - Enable

Step 3: config T
Step 4: fast Ethernet 0/0
Step 5: IP address 10.0.0.10 255.0.0.0
Step 6: No shut

Step 7: Exit

Step 8: Interface se 2/0

Step 9: IP address 20.0.0.10 255.0.0.0
Step 10: Encapsulation PPP

Step 11: clock rate 64000

Step 12: No shut

Step 13:

- Here for router with fastethernet execute only till step 9 and type shut.
- Only for router written to router connection execute all steps , execute step 11 only for router connection which has clock symbol at start. Repeat these steps for all routers.
- Again go to router 0 → CLI mode and type :-

Step 1: config T

Step 2: router rip

Step 3: Network 10.0.0.0

Step 4: Network 20.0.0.0

Step 5: Exit

- Repeat these steps for all routers.
- Now go to each router and type show ip address
- Send message from RCO to

Ping Output :-

Packet toward PC command line 10

PC > ping 10.0.0.1

Pinging 10.0.0.1 with 32 bytes

Request timed out

Reply from 10.0.0.1 : bytes=32 time=3ms TTL=125

Reply from 10.0.0.1 : bytes=32 time=3ms TTL=125

Reply from 10.0.0.1 : bytes=32 time=10ms TTL=125

Ping statistics for 10.0.0.1

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

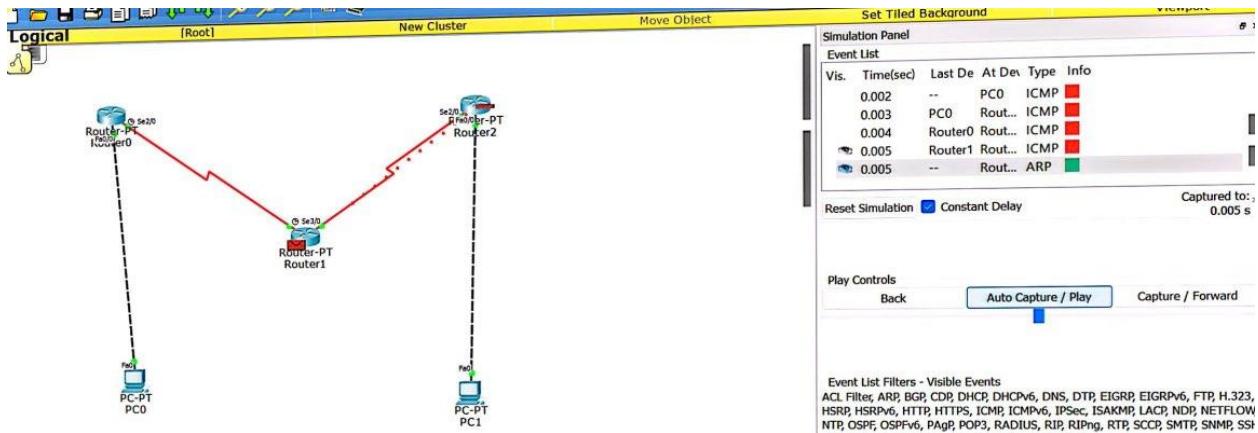
Approximate round trip time in milliseconds

Minimum = 3ms, Maximum = 10ms, Average = 7ms

Observation:-

- Routing information protocol (rip) is a dynamic routing protocol that uses hop count as a routing metric to find the best path b/w source & destination.
- update of routing info are always broadcast.
- Full routing tables are sent ~~to~~ in updates
- ~~Router always trust routing info received from neighbour routers~~

N
27/12/2023



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 40.0.0.1: bytes=32 time=13ms TTL=125
Reply from 40.0.0.1: bytes=32 time=2ms TTL=125
Reply from 40.0.0.1: bytes=32 time=9ms TTL=125
Reply from 40.0.0.1: bytes=32 time=12ms TTL=125
```

Ping statistics for 40.0.0.1:

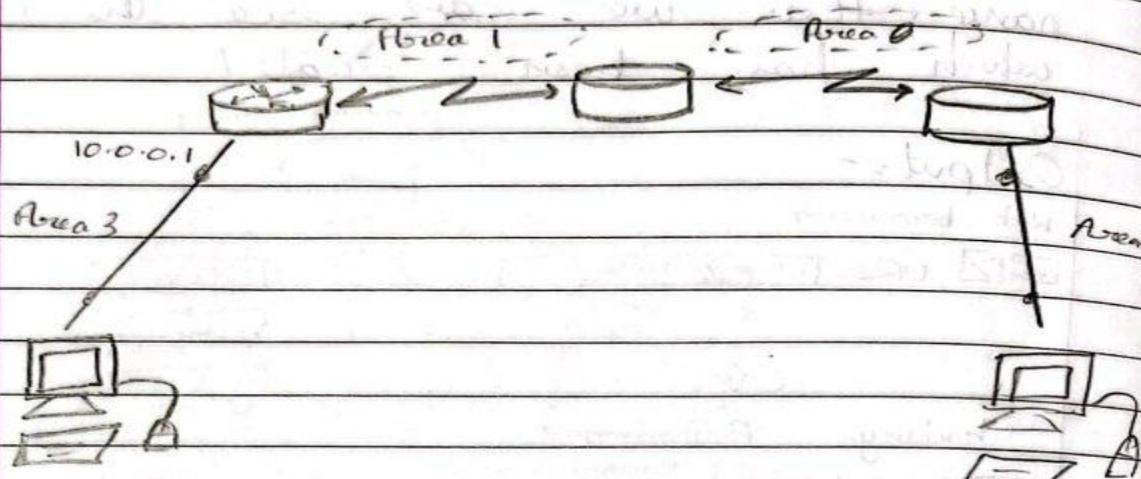
```
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 2ms, Maximum = 13ms, Average = 9ms
```

PC>

Experiment-7

Q) Configure OSPF routing protocol

Topology



Procedure

- 1) Configure the PC with IP address and gateway according to topology
- 2) Configure the router accordingly with the IP addresses given
- 3) Encapsulation PPP and clock rate need to be set as done in RIP protocol experiment
- 4)

In Router R1

```
R1(config)# router
R1(config)# interface loopback 0
R1(config-if)# ip add 172.16.1.2 255.255.255.0
R1(config-if)# no shutdown
```

R2#config-udflistener

In Router R2

R2(config)# router ospf 1

R2(config-router)# router-id 2.2.2.2

R2(config-router)# network 20.0.0.0 0.255.255.255
area 1

R2(config-router)# network 30.0.0.0 0.255.255.255
area 0

In Router R3

R3(config)# router ospf 1

R3(config-router)# router-id 3.3.3.3

R3(config-router)# network 30.0.0.0 0.255.255.255
area 0

R3(config-router)# network 40.0.0.0 0.255.255.255
area 2

R3(config-router)# exit

Set interface loopback

R1(config-if)# interface loopback 0

R1(config-if)# ip address 172.16.1.252
255.255.255.0

R1(config-if)# no shutdown

R2(config-if)# interface loopback 0

R2(config-if)# ip address 172.16.1.252
255.255.255.0

R2(config-if)# no shutdown

R3(config-if)# no shutdown

> In router R1

R1(config)# router ospf 1

R2(config)-router) # area 1 virtual
1.1.1.1

R2(config-router)# exit

i) After this show IP routes & then show all &

Ping output :-

Packet tracer PC command line 10

Ping 10.0.0.10

Pinging 10.0.0.10 with 32 bytes of data
Request timed out

Reply from 10.0.0.10 bytes = 32 time = 1ms

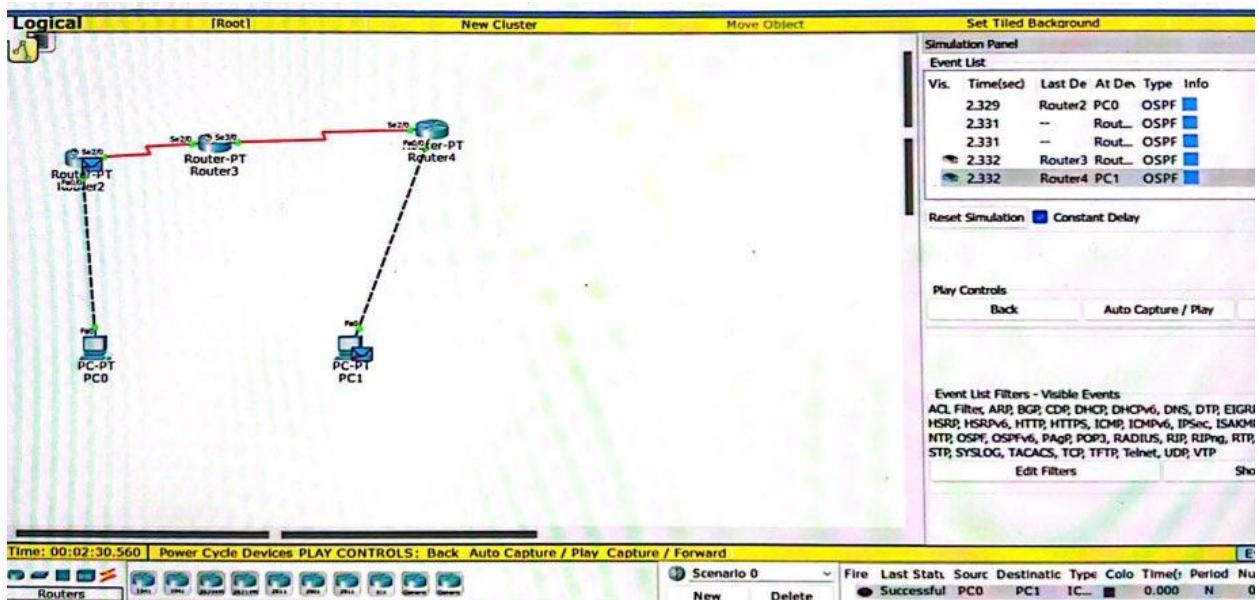
Reply from 10.0.0.10 bytes = 32 time = 1ms

Reply from 10.0.0.10 bytes = 32 time = 8ms

Ping statistics for 10.0.0.10

Packet sent = 4 received = 3 lost = 1

Avg round trip time in milliseconds



Command Prompt

```

Ping statistics for 40.0.0.10:
  Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
  Approximate round trip times in milli-seconds:
    Minimum = 2ms, Maximum = 11ms, Average = 5ms

PC>ping 40.0.0.10

Pinging 40.0.0.10 with 32 bytes of data:

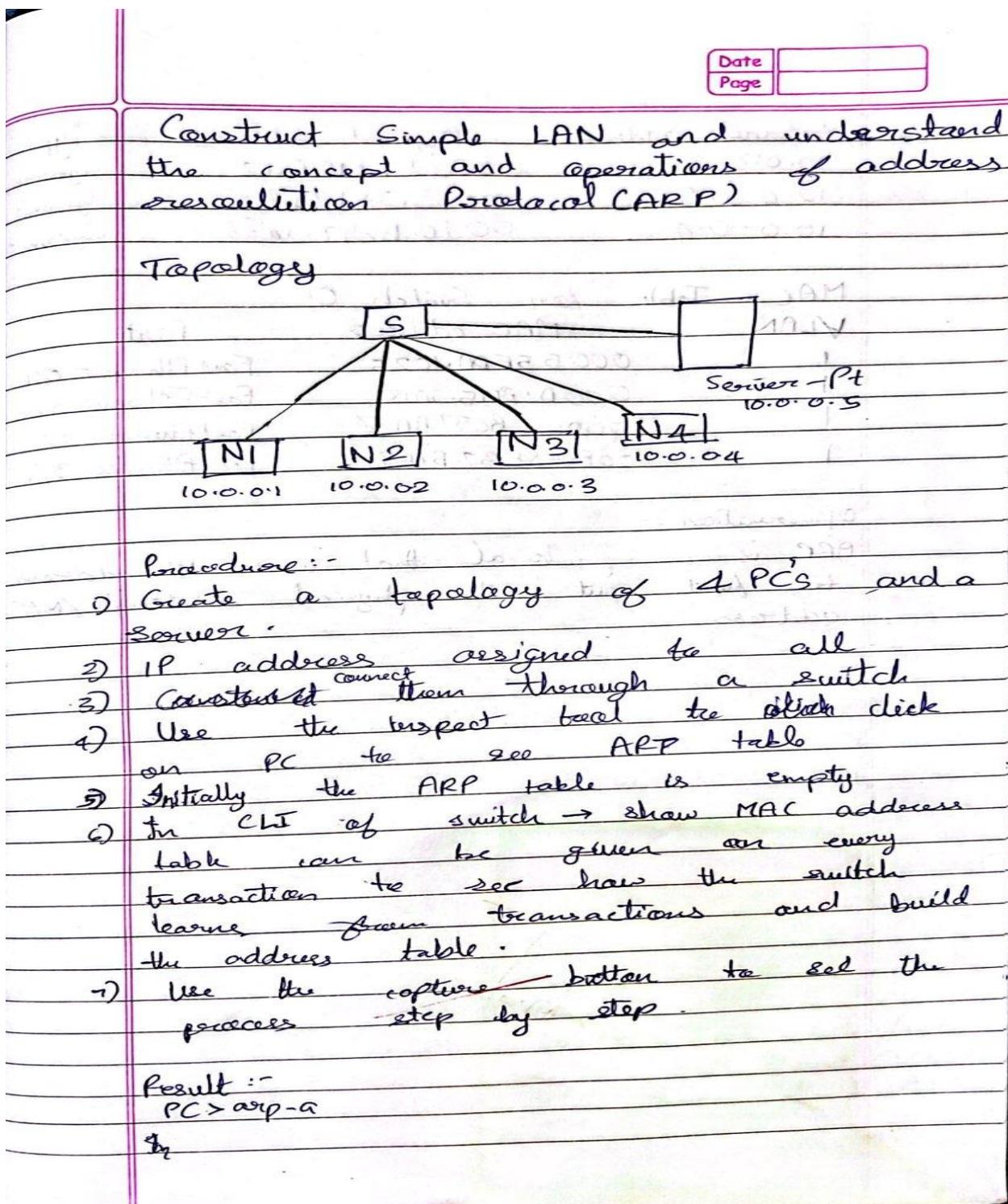
Reply from 40.0.0.10: bytes=32 time=9ms TTL=125
Reply from 40.0.0.10: bytes=32 time=10ms TTL=125
Reply from 40.0.0.10: bytes=32 time=2ms TTL=125
Reply from 40.0.0.10: bytes=32 time=2ms TTL=125

Ping statistics for 40.0.0.10:
  Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
  Approximate round trip times in milli-seconds:
    Minimum = 2ms, Maximum = 10ms, Average = 5ms
PC>

```

EXPERIMENT-8

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



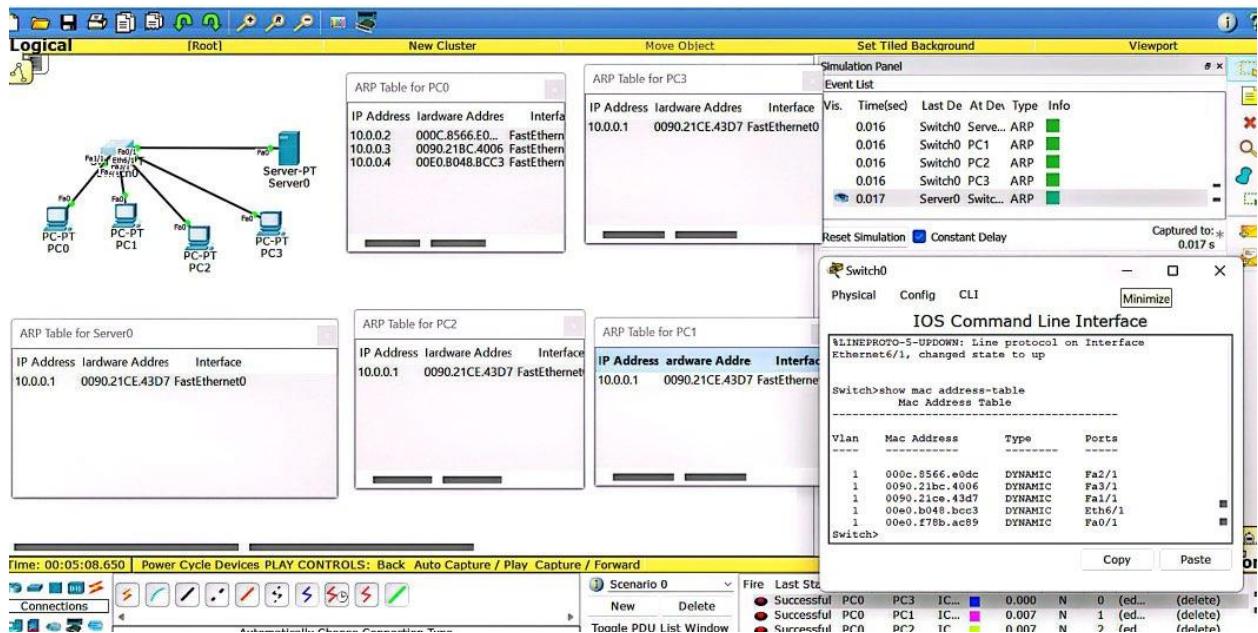
Internet address	Physical address	Type
10.0.0.2	000d.bd15.3a18	dynam.
10.0.0.3	00e0.8t b7-b7e8	dynam.
10.0.0.4	00d0.bc97.aa38	dynam.

MAC Table for Switch 0

VLAN	MAC address	Port
1	0005.5EDA.1C25	Fast Ethernet 0/
1	000d.bd15.3a18	Fast Ethernet 2/
1	00d0.bc97.aa38	Fast Ethernet 1/
1	00e0.8FB7.B1E8	Fast Ethernet 3/

Observation :-

ARP is a protocol that uses IP address to find out the physical addresses / MAC address



EXPERIMENT-9

Q) To construct a VLAN and make the PC's communicate among a VLAN

Date _____
Page _____

LAB-9

Aim :- To construct a VLAN and make a PC communicate among VLAN

Topology

Router R1: IP 192.168.1.1, Interface Fa 0/1
Switch S1: IP 192.168.1.1, Interface Fa 0/5
PC 1: IP 192.168.2.11, Interface Fa 0/1
PC 2: IP 192.168.2.11, Interface Fa 0/2
PC 3: IP 192.168.2.11, Interface Fa 0/3

Procedure :-

- Create a topology as shown choose 1841 router and 2960-24TT switch
- Set IP address of the router and 4 PC's respectively, we use class C type address also set gateway
- In switch go config table and select VLAN database, give a VLAN like 2 and name as VLAN
- Select the interface fastEthernet 0/1 and make it trunk.
- Select switches under 2nd interface which has interface 0/3 and 0/4 which on each of them set VLAN number 2

Go to router \rightarrow config tabs and select VLAN database and enter name VLAN and no 2 created.

1) Write following commands

1 Config T

2 Interface fa 0/0

3 IP address 192.168.1.1 255.255.255.0

4 No shut

5 Exit

6 Config T

7 Interface fa 0/0.1

8 encapsulation dot1q 2

9 ip address 192.168.20.2 255.255.0

10 No shut

11 Exit

12 Ping message from PC to another VLAN PC

Ping Output:-

Packet tracer PC command line 1.0

PC > Ping 192.168.20.3

Pinging 192.168.20.3 with 32 bytes of data
Request timed out

Reply from 192.168.20.3 : bytes=32 time=0ms TTL=11

Reply from 192.168.20.3 : bytes=32 time=5ms TTL=11

Reply from 192.168.20.3 : bytes=32 time=0ms TTL=11

Ping statistics for 192.168.20.3

Packets sent = 1 , Received = 3 , loss = 0%

Approximate round trip times in milliseconds
minimum = 0ms , Maximum = 5ms , Average = 1ms

Observation :-

- 1) We can have one device on one VLAN and another VLAN connected to the same switch. They will only hear other broadcast traffic from ~~so~~ within their VLANs, as if they were connected to two switches.
- 2) ~~There~~ ~~newer~~ VLAN's does not use IP address instead dial with subnet/c class C type address.
- 3) Inter VLAN routing gives a flexible tool to logically subdivide these networks that has potential to enhance security and performance.

11/9/2023

PC0

Physical Config Desktop Custom Interface

Command Prompt

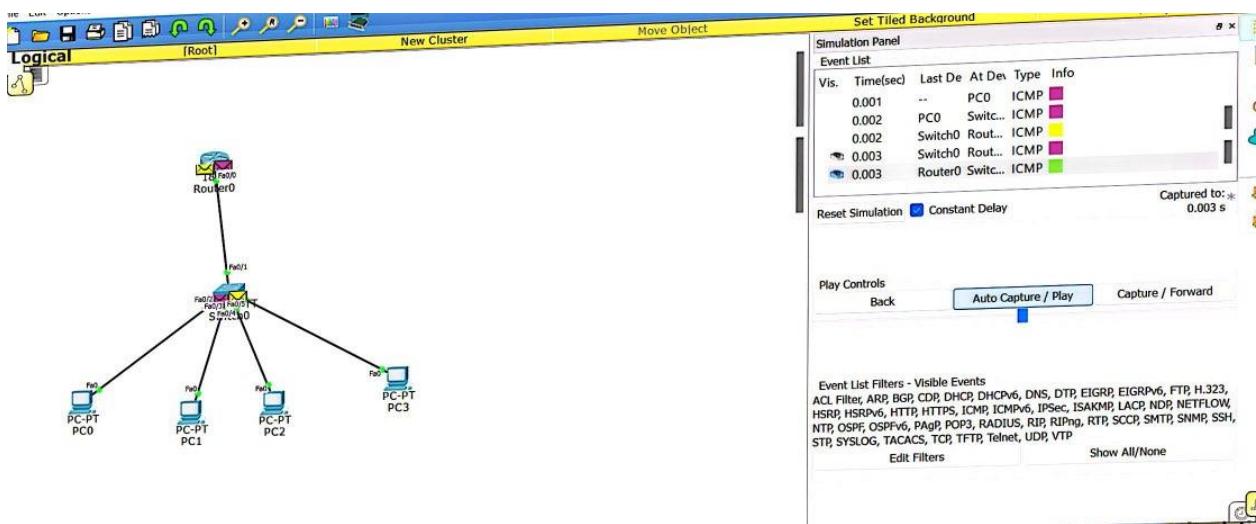
```
Packet Tracer PC Command Line 1.0
PC>ping 192.168.20.3

Pinging 192.168.20.3 with 32 bytes of data:

Reply from 192.168.20.3: bytes=32 time=2ms TTL=127
Reply from 192.168.20.3: bytes=32 time=0ms TTL=127
Reply from 192.168.20.3: bytes=32 time=5ms TTL=127
Reply from 192.168.20.3: bytes=32 time=0ms TTL=127

Ping statistics for 192.168.20.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 5ms, Average = 1ms

PC>
```



EXPERIMENT-10

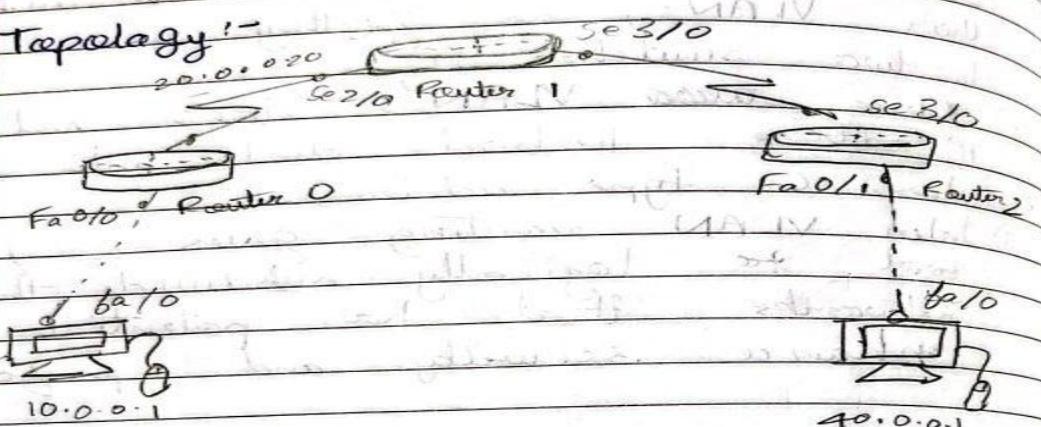
Q) Demonstrate the TTL/ Life of a Packet

LAB-10

Aim :-

Demonstrate the TTL/Life of a packet

Topology :-



Procedure :-

- Create a topology as shown with two PC's and 3 routers
- Set IP addresses and gateway for both PC's
- Configure the routers either statically or routing way.
- In simulation mode send a simple PDU from one PC to another
- Use capture button to capture every transfer.
- Click on the PDU during every transfer
- Click on the PDU during every transfer to see the inbound and outbound PDU details.

Output

IP

0 4 8 16 19 31

4 IHL DSCP TL: 28

IP: 0x6 0x 0x0

TTL: 255 PRO: 0x1 CHKSUM

SRC IP: 10.0.0.1

DST IP: A0.0.0.1

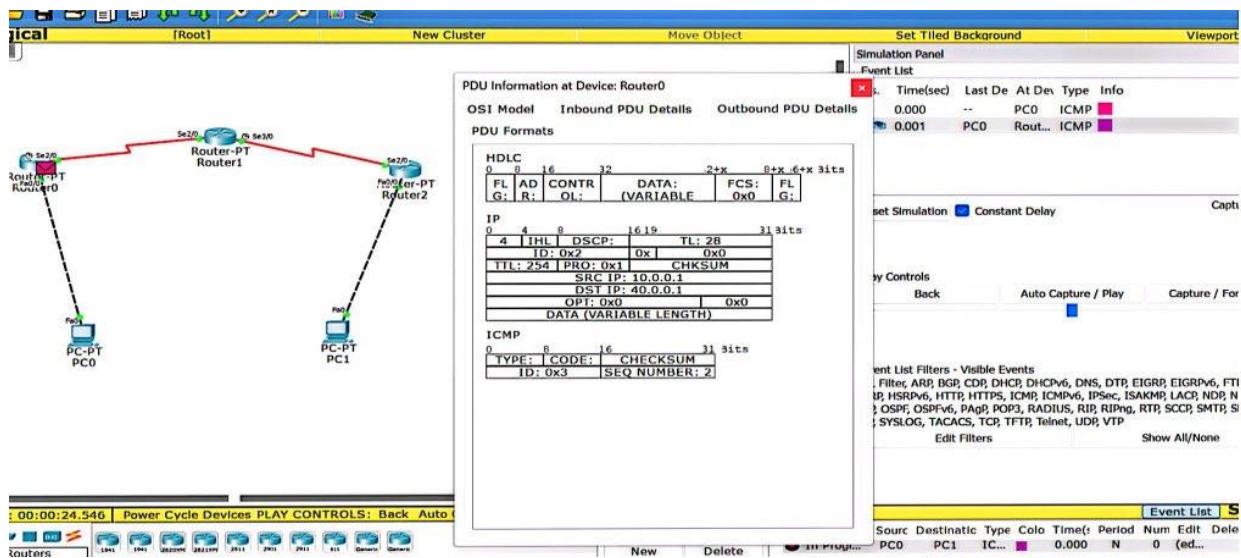
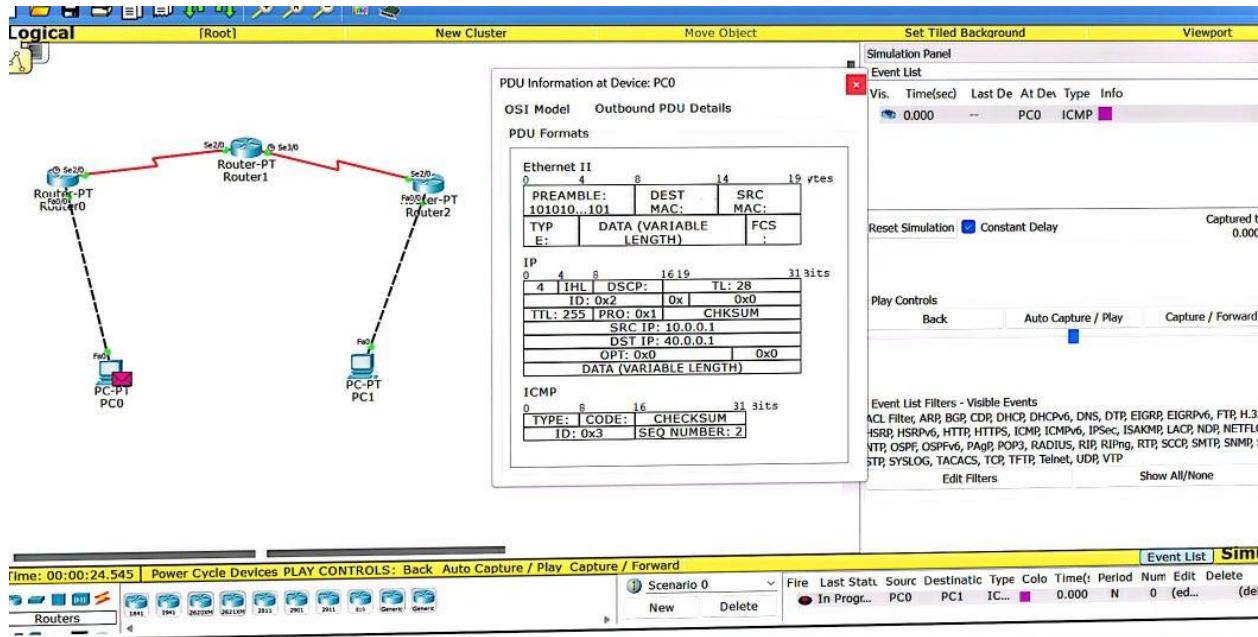
OPT 0x0 0x0

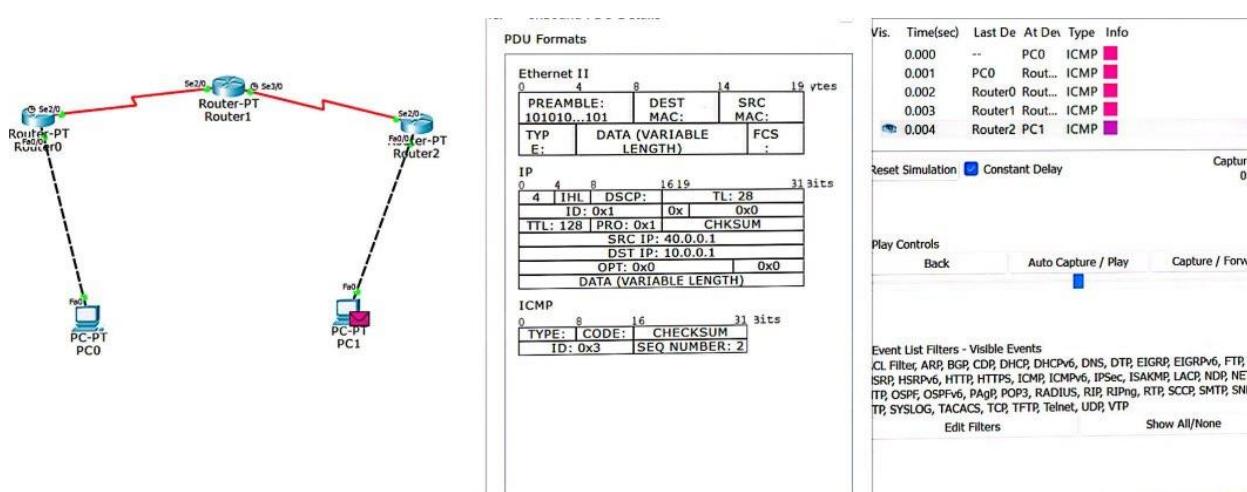
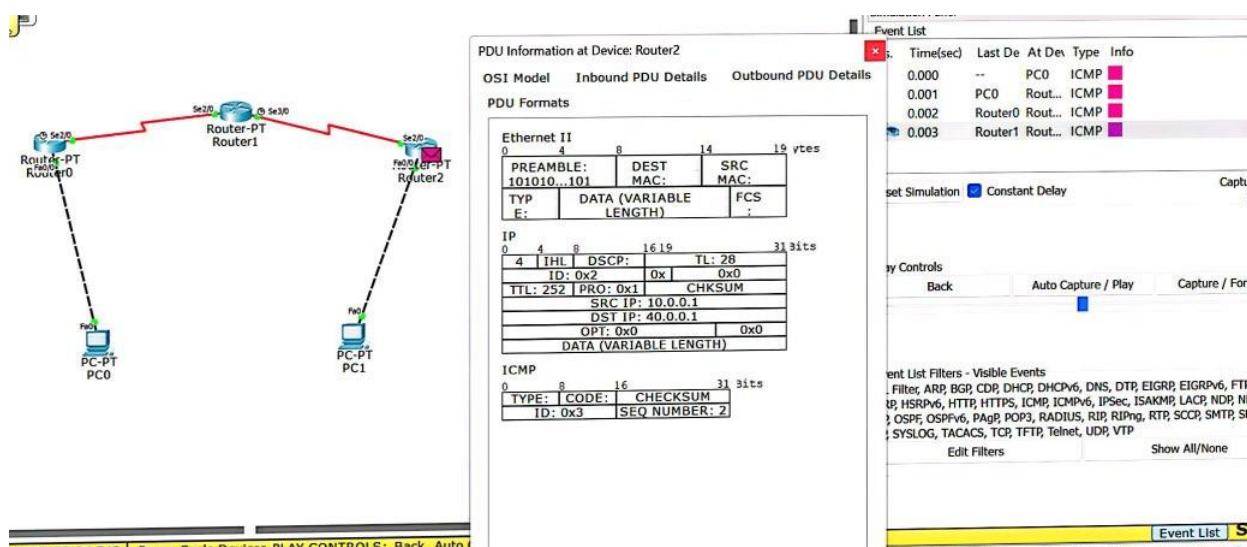
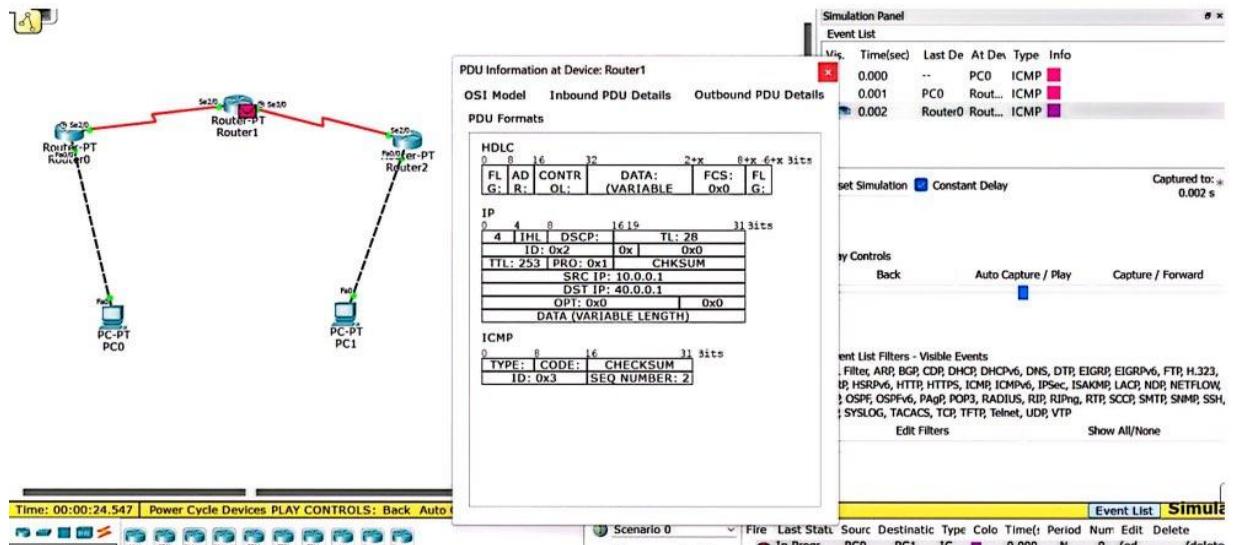
DATA (VARIABLE LENGTH)

Observation :-

- The no of hops the packet before being discarded as TTL
- Dataframe TTL field is set by the sender and reduced by each router along path to destination.
- The router reduces TTL value by one tick while forwarding the packets.
- When TTL value is 0, the router discards and sends an ICMP message

N
1/1/2023





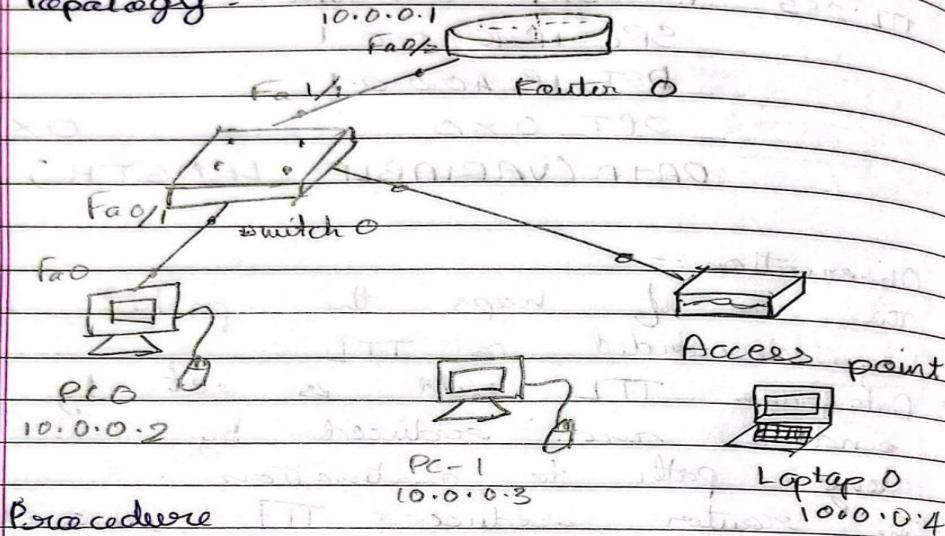
EXPERIMENT-11

Q) To construct a WLAN and make the nodes communicate wirelessly

Lab - 11

Aim:- To construct WLAN and make nodes communicate wirelessly

Topology :-



Procedure

- Construct the above topology
- Configure PC0 and Router 0 as router
- Configure access point 1 - Port 1 → GB10 Name - WLAN
- Select to EP & give any 10 digit hex key - 1234567890
- Config PC1 & Laptop with wireless standards.
- Switch off the device - Drag existing PT-HS - NM- IAM to the component listed in LHS - Drag WMP300.N wireless interface to the empty port - Switch on the device

- In the config tab a new wireless interface would have been added. Now configure SSID, WEP, WEP key, IP address and gateway for the device.
- Ping from every device to every other device.

Ping Output :-

Packet tracer PC command line 10
PC > Ping 10.0.0.3

Pinging 10.0.0.3 with 32 bytes of data
Request timed out

Reply from 10.0.0.3 : bytes = 32 time = 0ms TTL = 127

Reply from 10.0.0.3 : bytes = 82 time = 0ms TTL = 127

Reply from 10.0.0.3 : bytes = 82 time = 0ms TTL = 127

Ping statistics for 10.0.0.3

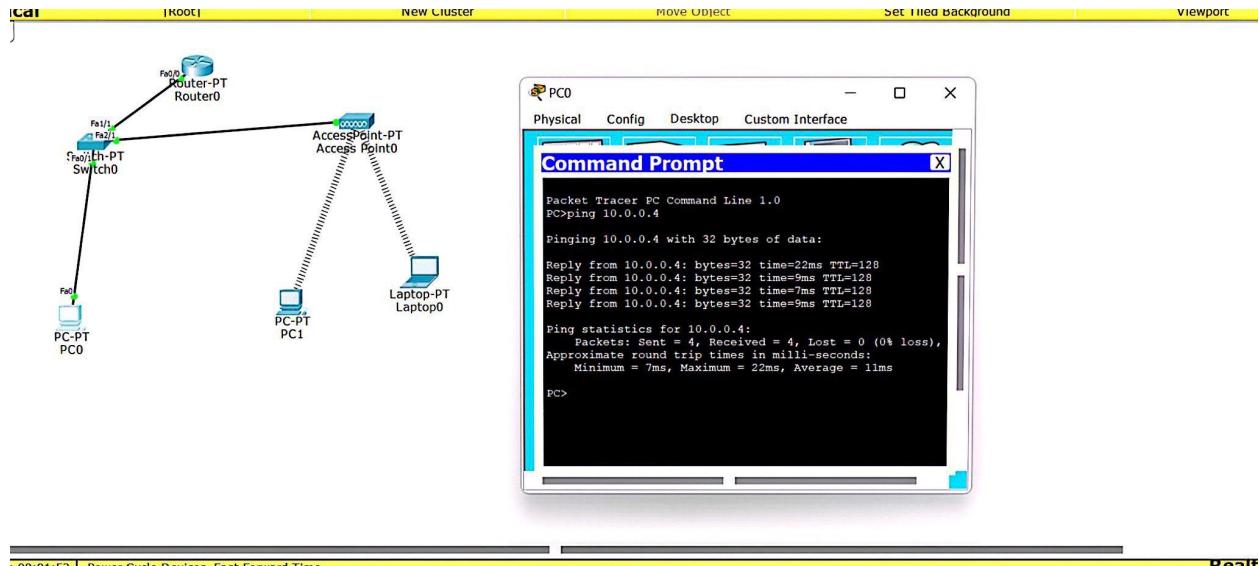
Packet = sent = 4, Received = 3, Lost = 1 (0% loss)

Approximate round trip times in milliseconds:

Minimum = 0ms, Maximum = 1ms, Average = 0ms

Observation :-

- A WLAN is a group of collocated devices that formed a network based on radio transmissions.
- Data sent as packets contains layers with labels and instructions MAC address to endpoints for routing.



EXPERIMENT-12

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

Lab - 12

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

Topology



Procedure

- Create a topology as shown above
 - Configure IP addresses and gateway
 - Configure router by executing following commands
- 1) enable
 - 2) config T
 - 3) host name R1
 - 4) enable secret P1
 - 5) interface fastEthernet 0/0
 - 6) IP address 10.0.0.1 255.0.0.0
 - 7) No Shut
 - 8) line vty 0 5
 - 9) login
 - 10) password po
 - 11) Exit; Exit
 - 12) car

Ring message to router
Password for user Pass verification is
p0

Password for enable is P1

Accessing router CLI from pc
Show IP route

Ping Output

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 = 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

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 milliseconds

Minimum = 0ms, Maximum = 0ms Average = 0ms

PC > telnet 10.0.0.1

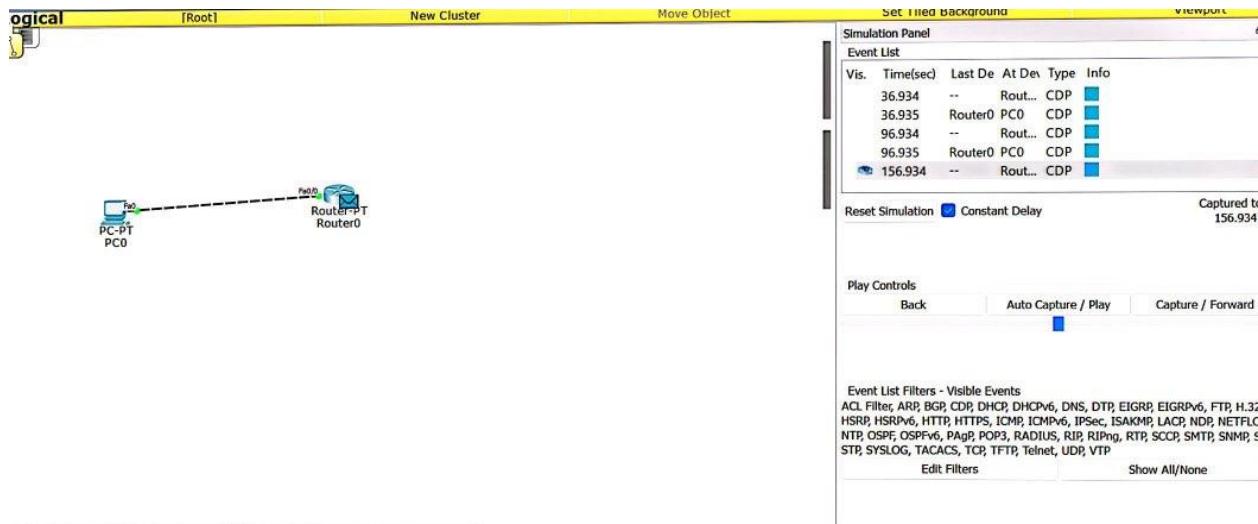
Typing 10.0.0.1 open
User acus verification

Password: p0

P1 > enable

Observation :-

- TELNET stands for teletype Network. It is a type of protocol that enables one computer to connect to the local computer.
- It is used as a standard TCP/IP protocol for virtual terminals provided by ISO.
- During TELNET operation, whatever being performed on the remote computer will be displayed by the local computer.



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=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
Reply from 10.0.0.1: bytes=32 time=1ms 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:
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#

```

EXPERIMENT-13

Q) Write a program for error detecting code using CRCCCITT (16-bits)

Date _____
Page _____

Lab 14

Alm- WAP for error detecting code using CRC

```
#include <stdio.h>
#include <string.h>

void binaryXOR(char*result, const char*a,
const char*b) {
    for(int i=0; i<16; i++) {
        result[i] = (a[i] == b[i]) ? '0' :
        '1';
    }
    result[16] = '\0';
}

void calculateCRC(const char*data, int
length, char*checksum) {
    char crc[17];
    for(int i=0; i<16; i++)
        crc[i] = '0';
    crc[16] = '\0';
    for(int i=0; i<length; i++) {
        for(int j=0; j<8; j++) {
            char msb = data[i];
            for(int k=0; k<16; k++)
                crc[k] = crc[k+1];
            if(msb & 1)
                crc[16] = '1';
            msb = msb >> 1;
        }
    }
    crc[16] = '\0';
}
```

```
if(meb == '1') {  
    char temp[17];  
    binaryXOR(temp, CRC, "10001000000000000000000000000000");  
    CRC_Poly_in_binary  
    strcpy(CRC,temp);  
}
```

```
}  
crc[i5] = (data[i] == '1') ? '1' : '0';
```

```
strcpy(checksum, crc);
```

```
}
```

```
int main() {
```

```
char data[100];
```

```
pf("Enter data in binary: ");
```

```
scanf("%s", data);
```

```
int dataLength = strlen(data);
```

```
char checksum[7];
```

```
calculateCRC(data, dataLength, checksum);
```

```
char receivedChecksum[17];
```

```
pf("Enter received CRC: ");
```

```
scanf("%s", receivedChecksum);
```

```
if(strcmp(receivedChecksum, checksum))
```

```
    pf("Data is error free\n");
```

```
} else {
```

```
    pf("Data contains errors\n");
```

```
}
```

```
return 0;
```

```
}
```

```
Output :-
```

```
Enter dataword: 11001010111001001
```

```
Calculated CRC = 11101001011001
```

CODE-

```
import java.util.Scanner; import java.util.Arrays;

class Program {
    static String Xor(String a, String b) {

        String result = "";
        int n = b.length();
        for (int i = 1; i < n; i++) {

            result=(a.charAt(i) == b.charAt(i))?0:1;

        }

        return result;
    }

    static String Div(String data, String key) { int pick = key.length();
String tmp = data.substring(0, pick);

61

int n = data.length(); while (pick < n) {

    if (tmp.charAt(0) == '1')
        tmp = Xor(data, tmp) + data.charAt(pick);

    else
        tmp = Xor(new String(new char[pick]).replace("\0", "0"), tmp) +
        data.charAt(pick);

    pick += 1;
}

if (tmp.charAt(0) == '1') tmp = Xor(divisor, tmp);

else
    tmp = Xor(new String(new char[pick]).replace("\0", "0"), tmp);

return tmp;
}
```

```

static void Encode(String data, String key) {
int lkey = key.length();
String appended_data = (data + new String(new char[lkey - 1]).replace("\0",
"0")); String remainder = Mod2Div(appended_data, key);
String codeword = data + remainder;
System.out.println("Remainder : " + remainder);
System.out.println("Encoded Data (Data + Remainder) :" + codeword + "\n");

}

public static void main(String[] args) { Scanner s = new Scanner(System.in);

62

System.out.println("enter dataword and key"); String data = s.next();
String key = s.next();

EncodeData(data, key); }

}

```

```

Enter the dataword
1 0 1 1 0 0 1 1 1 1 0 0 1 0 1 1 1
Enter dividend
1 0 0 0 1 0 0 0 0 0 0 1 0 0 1 1
1
Codeword: 101100111100101110000000000011011
At receiver end
Codeword: 101100111100101110000000000000000
Process returned 1 (0x1)    execution time : 53.976 s
Press any key to continue.
|
```


EXPERIMENT-14

Q) Write a program for congestion control using Leaky bucket algorithm

Date	
Page	

LAB - 13

Aim:- WAP for congestion control using bucket algorithm

Procedure:-

```
#include <stdio.h>
#include <stdlib.h>
#define capacity 50
void main() {
    int timeLimit = 10;
    int bucketCapacity = 0, outputRate = 5;
    while (timeLimit < 20) {
        int newPacket;
        pf("Enter new packet size:");
        scanf("%d", &newPacket);
        if (newPacket < capacity) {
            bucketCapacity = bucketCapacity + newPacket;
            pf("bucket Capacity currently: %d", bucketCapacity);
            bucketCapacity = bucketCapacity - outputRate;
            pf("bucket capacity after output: %d", bucketCapacity);
            timeLimit++;
        }
        else if (newPacket > capacity || (newPacket + bucketCapacity) > capacity) {
            pf("New packet cannot be added to bucket");
        }
    }
}
```

~~bucketCapacity = bucketCapacity - outputSize~~

~~pb("after output: %d", bucketCapacity)~~

~~else if (bucketCapacity < 0) {~~

~~bucketCapacity = 0;~~

~~pb("bucketCapacity after output: %d",~~

~~bucketCapacity);~~

~~timeLimit++~~

~~exit(0)~~

~~3~~

~~3~~

~~3 starting time = current time~~

Output

Enter no. of queries, bufferSize, input

output packet size

1

7

4

1

Packet accepted

remaining space = 3

packet is accepted

remaining space = 0

Packet not accepted

remaining space = 1

remaining space = 2

CODE-

```
import java.util.*; class Leakybucket {  
    public static void main(String[] args) {  
        int rem;  
        Scanner sc=new Scanner(System.in); int s= 0;  
  
        System.out.println("enter no of queries,buffer size,input and output packet size "); int  
        q=sc.nextInt();  
  
        int bs=sc.nextInt(); int ip=sc.nextInt();  
  
        int op=sc.nextInt(); for (int i = 0; i < q; i++) {  
  
            rem=bs-s;  
            if (ip <= (rem)) {  
  
                System.out.println("packet is accepted"); s+=ip;  
  
            }  
            else {  
  
                System.out.println("Packet not accepted "); }  
  
            System.out  
  
.println("remaining space="+ (bs-s)); s -= op;  
        }  
        65  
    } }  
}
```

OUTPUT

```
enter no of queries,buffer size,input and output packet size
4 10
6
1
packet is accepted
remaining space=4
Packet not accepted
remaining space=5
packet is accepted
remaining space=0
Packet not accepted
remaining space=1
PS C:\Users\sanja\OneDrive\Documents> █
```

EXPERIMENT-15

Q) 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.

Date	
Page	

Lab 15

Aim:-

Using TCP/IP with sockets, Write a client server program to make client sending the file name and server to send back the contents

Client TCP.py

```
from socket import *
ServerName = '127.0.0.1'
ServerPort = 12000
clientSocket = socket(AF_INET, SOCK_STREAM)
clientSocket.connect((ServerName, ServerPort))
Sentence = input("Enter file name ")
clientSocket.send(Sentence.encode())
filecontents = clientSocket.recv(1024).decode()
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:
    pf("The server is ready to receive")
    ConnectionSocket, addr = ServerSocket.accept()
    Sentence = ConnectionSocket.recv(1024).decode()
    file = open(Sentence, "r")
    fileContent = file.read(1024)
    ConnectionSocket.send(fileContent.encode())
    file.close()
    ConnectionSocket.close()
```

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

Output:-

ServerTCP.py
The server is ready to receive

send contents of serverTCP.py
The server is ready to receive

ClientTCP.py

Enter file name: ServerTCP.py

from socket

from socket import *

ServerName = "127.0.0.1"

ServerPort = 12000

ServerSocket.bind((ServerName, ServerPort))

ServerSocket.listen(1)

while 1:

print("The server is ready to receive")

sentence = ClientConnectionSocket.recv(1024)

file = open("Sentence", "w")

R = file.read(1024)

ClientConnectionSocket.send(l.encode())

print("Sent contents of " + sentence)

file.close()

ClientConnectionSocket.close()

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)

69

    file.close()
    connectionSocket.close()
```

```
Enter file name:ServerTCP.py
=====
RESTART: C:/Users/sanja/OneDrive/Documents/ClientTCP.py =====

Enter file name:ServerTCP.py
=====
RESTART: C:/Users/sanja/OneDrive/Documents/ClientTCP.py =====

Enter file name:
=====
RESTART: C:/Users/sanja/OneDrive/Documents/ClientTCP.py =====

Enter file name:ServerTCP.py

From Server:

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()
```

```
-----
The server is ready to receive
The server is ready to receive
The server is ready to receive
=====
RESTART: C:/Users/sanja/OneDrive/Documents/ServerTCP.py =====

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

EXPERIMENT-16

Q) 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

Date _____
Page _____

LAB-16

Using UDP sockets, write a client server program to make 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("Enter file name: ")
ClientSocket.sendto(sentence.encode(), (ServerName, ServerPort))
filecontents, ServerAddress = ClientSocket.recvfrom(2048)
print(filecontents.decode())
# for i in filecontents:
#     print(str(i), end=' ')
ClientSocket.close()
ClientSocket.close()
```

Server UDP.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")
pesan
```

while 1:

```
sentence, clientAddress = ServerSocket.recvfrom(1024)
sentence = sentence.decode("utf-8")
file = open(sentence, "r")
content = file.read(2048)
ServerSocket.sendto(content, clientAddress)
print("In Send Content of ", end="")
print(sentence)
```

Output

ServerUDP.py

The server is ready to receive

Client-UDP.py

Enter file name : Server.py

Reply from server

content of Server.py

ND
1/9/2023

CODE-

ClientUDP.py

```
from socket import * serverName = "127.0.0.1"

72

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()
```

```
IDLE Shell 3.9.13
File Edit Shell Debug Options Window Help
Python 3.9.13 (tags/v3.9.13:6de2ca5, May 17 2022, 16:36:42) [MSC v.1929 64 b
it (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
----- RESTART: C:/Users/sanja/OneDrive/Documents/ClientUDP.py -----
-----
Enter file name:
----- RESTART: C:/Users/sanja/OneDrive/Documents/ClientUDP.py -----
-----
Enter file name:ServerUDP.py
Reply from Server:
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()

>>>
```

```
File Edit Shell Debug Options Window Help
Python 3.9.13 (tags/v3.9.13:6de2ca5, May 17 2022, 16:36:42) [MSC v.1929 64 b
it (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/sanja/OneDrive/Documents/ServerUDP.py =====
=====
The server is ready to receive
nSent contents of ServerUDP.py
```

EXPERIMENT-17

Q) Tool Exploration - Wireshark

Date _____
Page _____

LAB-17

Tool Exploration - Wireshark

Wireshark is an open source packet analyzer which is used for education analysis, software development, communication protocol development and network troubleshooting.

It is used to track packets so that each one is filtered to meet our specific needs. It is also used by network analysts.

Wireshark is a tool for applications used to apprehend data back and forth. It is also called as a packet sniffer computer application, put through website card into a selective mode, i.e. to accept all packets.

Uses:-

- 1) It is used by network security engineers to examine security problem.
- 2) It is used by network engineers to troubleshoot network uses.
- 3) It is used to analyze dropped packets.
- 4) It helps to troubleshoot many multiple activities on the network.

It helps to all over the mobile phones, router, computer, laptop, as the rest of the world.

Functionality of broadcast

It is similar to a TCP development in networking. It has a sending and listening functions. It also monitors the traffic which is not sent to networks. MAC address interface the narrowing is a method to monitor.

switch sends copies of all network packets present at one port to another port.

N.
17/9/2023

