VISVESVARAYA TECHNOLOGICAL UNIVERSITY

"JnanaSangama", Belgaum -590014, Karnataka.

LAB REPORT

on

COMPUTER NETWORKS

Submitted by

ADITYA S HUDDAR(1BM21CS007)

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

JUN2023to SEPTEMBER-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 "LAB COURSE **COMPUTER NETWORKS**" carried out by **ADITYA S HUDDAR(1BM21CS007),** 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.

Dr.Nandhini Vineeth
Assistant Professor

Department of CSE
BMSCE, Bengaluru
`

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

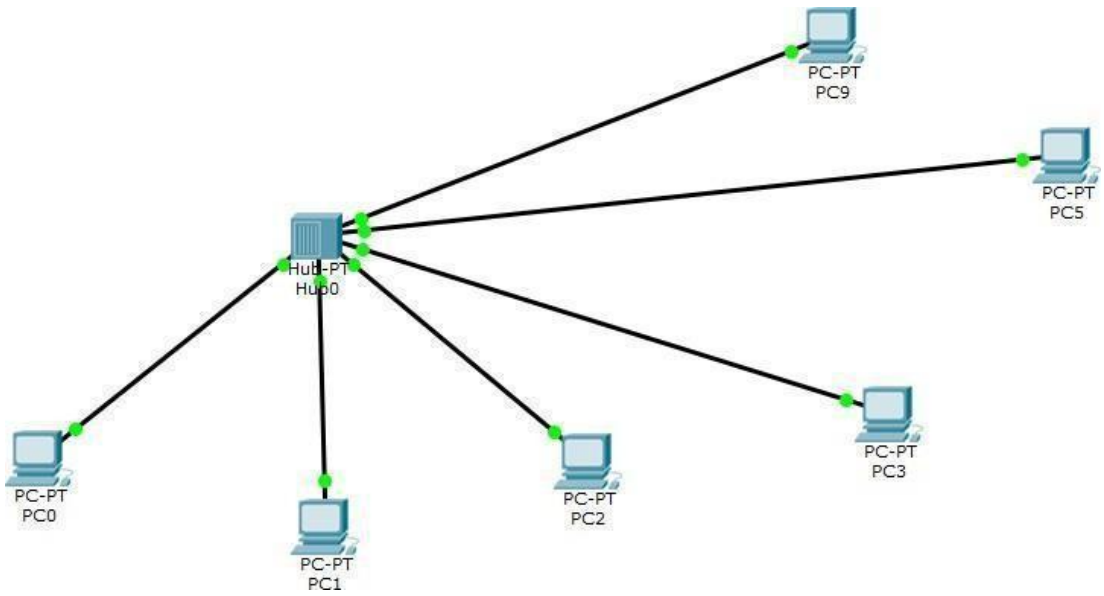
Sl. No.	Date	Experiment Title	Page No.
1.		**Create a topology and simulate sending a simple PDU from source to destination using hub and switch as connecting devices and demonstrate ping message.**	
2		Configure IP address to routers in packet tracer. Explore the following messages: ping responses, destination unreachable, request timed out, reply	
3		Configure default route, static route to the Router	
4		Configure DHCP within a LAN and outside LAN. ,.	
5		Configure RIP routing Protocol in Routers	
6		Configure OSPF routing protocol	
7		Demonstrate the TTL/ Life of a Packet	
8		Configure Web Server, DNS within a LAN.	
9		To construct simple LAN and understand the concept and operation of Address Resolution Protocol (ARP)	
10		To understand the operation of TELNET by accessing the router in server room from a PC in IT office.	
11		To construct a WLAN and make the nodes communicate wirelessly	
12		To construct a VLAN and make the PC's communicate among a VLAN	
		Cycle - 2	
13		Write a program for error detecting code using CRC-CCITT (16-bits).	
14		Write a program for congestion control using Leaky bucket algorithm.	
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.	
16/ 17		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. Tool Exploration -Wireshark	

CYCLE - 1

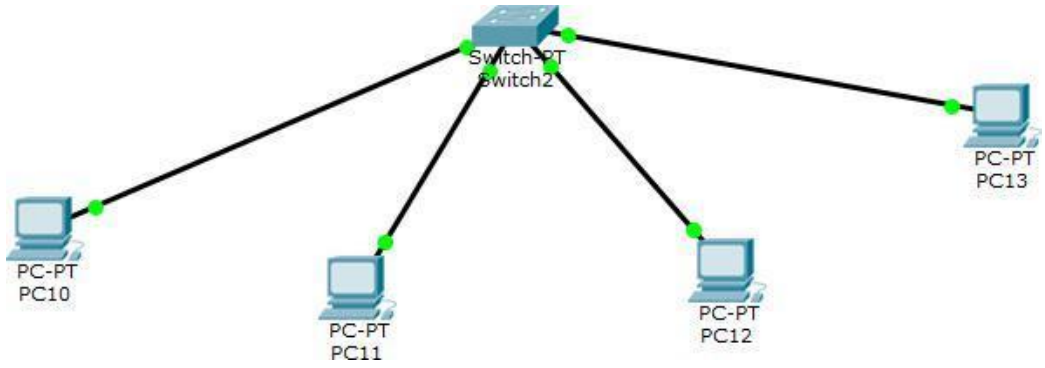
Experiment No-1

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

1. PC and Hub

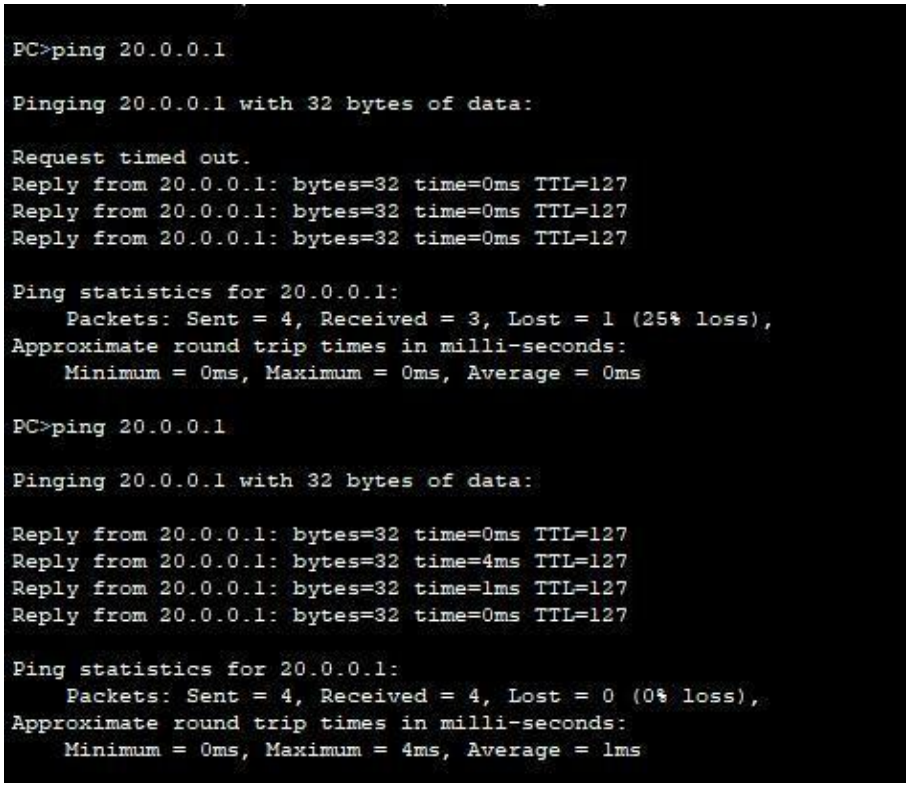


2. Pc and Switch



Procedure:

- Put all the devices(PCs, Hubs and Switches) needed for the experiment on the screen by looking at the topology.
- Choose the correct wire and make the Connection as shown in the topology
- Give ip address to all the devices
- Ping from one pc to all other pc in the network to make sure that the connection is correct

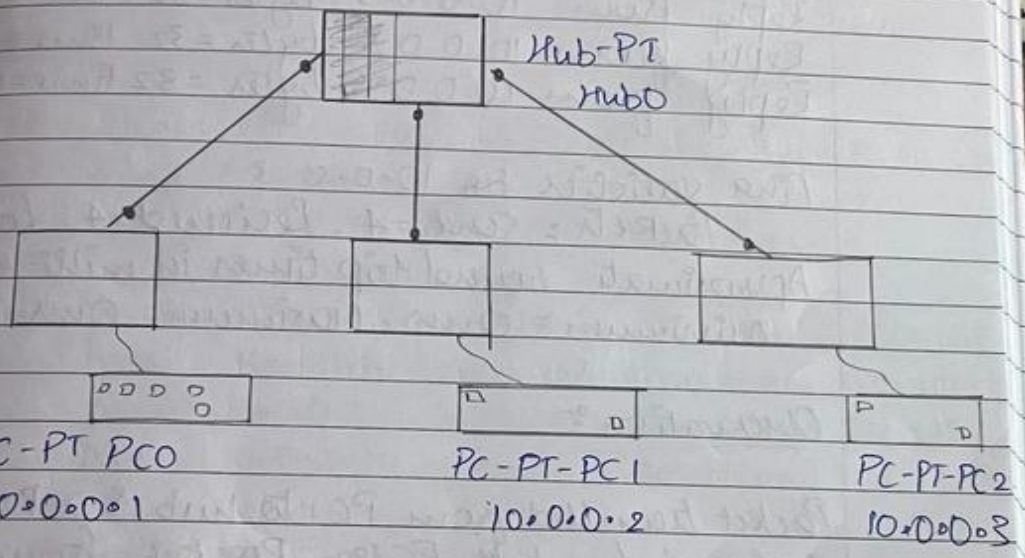


Page No.:
Date:

06/6/25 **Experiment -1 :-**

Aim : Create a topology & simulate sending a simple PDU from source to destination using Hub & switch as connecting devices & demonstrate ping message.

Topology1:



Procedure :-

- Select end devices & opt on the PC and place it in the workspace given.
- Set the IP address for all the PC's selected by clicking on the PC.
- Connect PC & the hub by copper-straight wire which is in the connections mode.
- Start the Stimulation by selecting the sender PC & giving the IP address of the destination PC.
- Select the command prompt in the PC & write ping + IP address of destination PC to get the Output.

Date:

Procedure:-

Select ~~ten~~ PC's from the end devices and set their respective IP addresses. Connect PC's to the switch by connecting wire (copper straight). Start the realtime & simulation & note the observations.

Result:-

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

Ping statistics for 10.0.0.3
Packets: Sent = 4, Recieved = 4, Lost = 0 (0% Loss)
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms

Observation:-

Packet travelled from source to destination without being broadcasted to all the end devices.

Result :-

Packet Tracer PC Command Line 1.0
PC > ping 10.0.0.3
Pinging 10.0.0.3 with 32 bytes of data.

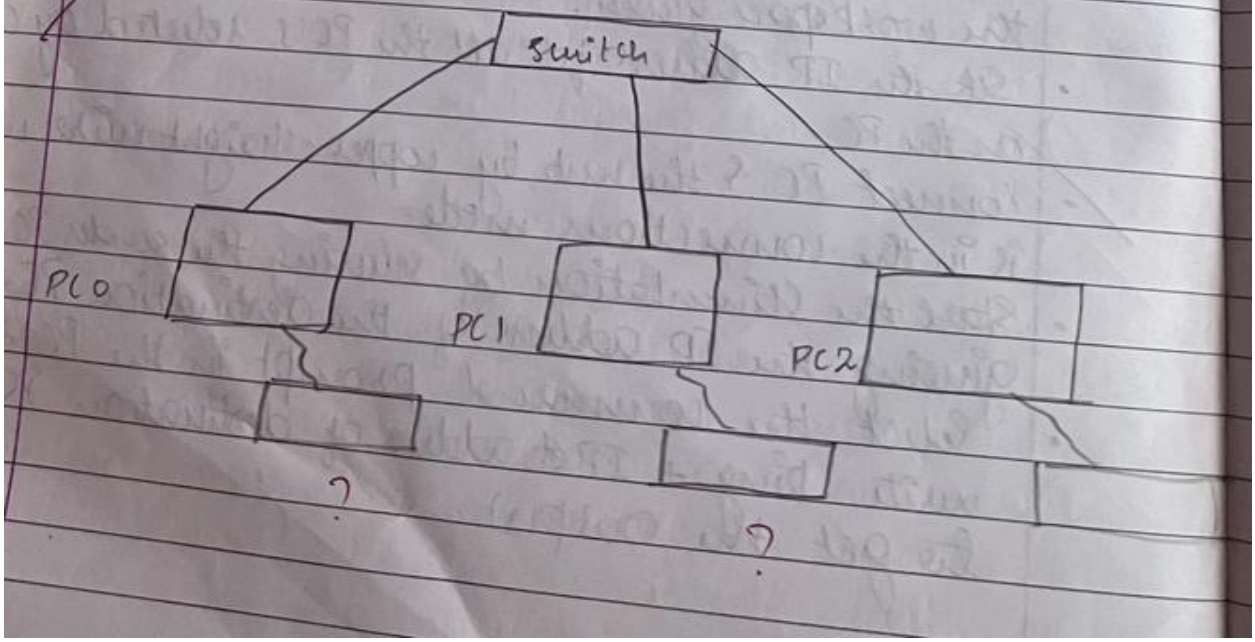
Reply from 10.0.0.3 bytes=32 time=0ms TTL=128
Reply from 10.0.0.3 bytes=32 time=0ms TTL=128
Reply from 10.0.0.3 bytes=32 time=0ms TTL=128
Reply from 10.0.0.3 bytes=32 time=0ms TTL=128

Ping statistics for 10.0.0.3:
Packets: Sent=4, Recieved=4, Lost=0 (0% Loss)
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms

Observation :-

Packet travelled from PC1 to hub & acknowledgement recieved by both PC's. Packet travelled from reciever to sender PC via a hub.

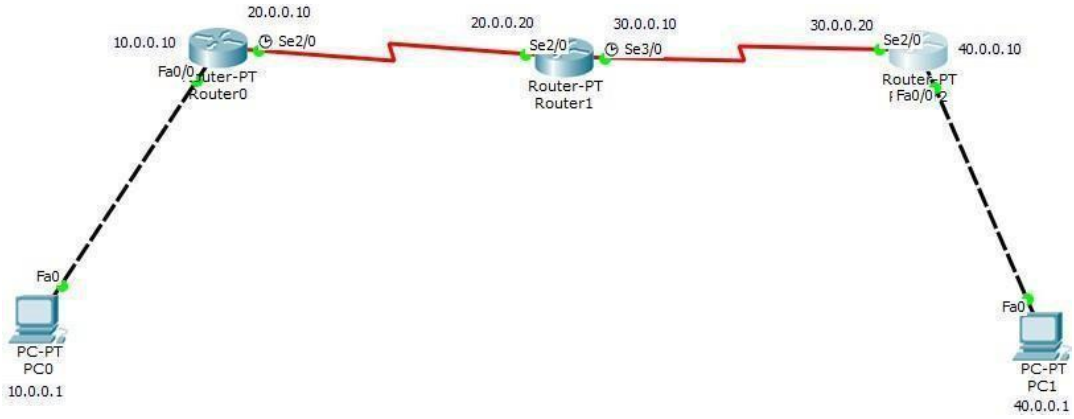
Topology 2 :-



Experiment No-2

Aim : Configuring IP address to Routers in Packet Tracer. Explore the following messages: Ping Responses, Destination unreachable, Request timed out, Reply

Topology:



Procedure:

1. connect PC-0 with Router-0 using copper cross-over cable - fastethernet0/0
2. connect Router-0 to Router-1 using Serial DCE with the connection named as serial2/0, then connect Router1 to Router2 using serial DCE named serial3/0
3. connect Router2 to PC1 using copper cross-over cable - fastethernet1/0
4. set the IP addresses, subnet mask (255.0.0.0 for all PCs and routers) and gateways accordingly.

   a. PC0: IP address = 10.0.0.1 gateway = 10.0.0.10

   b. Router0: gateway1 = 10.0.0.10 gateway2 = 20.0.0.10

   c. Router1: gateway1 = 20.0.0.20 gateway2 = 30.0.0.10

   d. Router2: gateway1 = 30.0.0.20 gateway2 = 40.0.0.10

   e. PC1: IP address = 40.0.0.1 gateway = 40.0.0.10

5. for Router0, the first gateway is set to IP address of 10.0.0.10 which is as same as the gateway of PC0 then set up the connection between the

i. Router0 and the PC0 using the CLI.

ii. Router0 and Router

Output:

```
Packet Tracer PC Command Line 1.0

PC>ping 10.0.0.10

Pinging 10.0.0.10 with 32 bytes of data:

Reply from 10.0.0.10: bytes=32 time=1ms TTL=255

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

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

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

Ping statistics for 10.0.0.10:

    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

Approximate round trip times in milli-seconds:

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

22/6/15

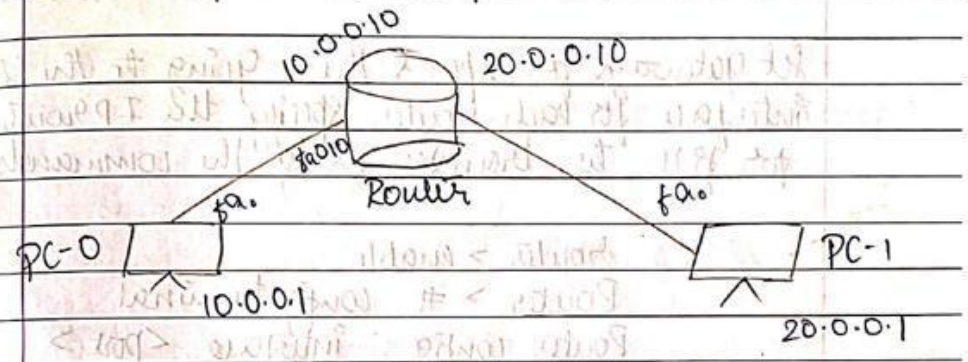
Page No.:
Date:
YOUVA

Experiment -2

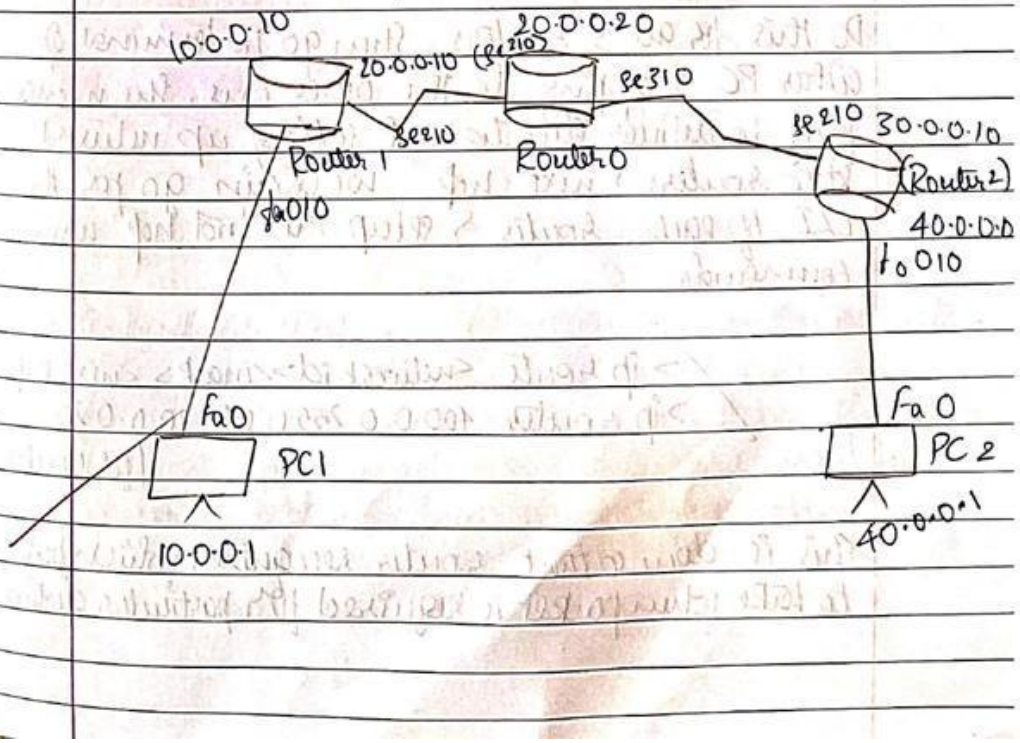
Aim :
Configure IP address to routers in packet tracer explore the following messages ping responses destination unreachable, request timed out, reply.

Topology:

Experimental setup:



3 Router Topology:



Date: YOUVA

Procedure :-

Take 2 PC & place them as shown in the topology, 2 different IP address (10.0.0.1 & 40.0.0.1) as they belong to 2 different networks.
Place 2 routers belonging to their 2 networks (10.0.0.10 & 40.0.0.10) being their gateways & place the 3rd router in between to connect the networks

Set gateways for 2 pcs & then going to the CLI interface for each router specify the IP route for file to transfer using the commands.

```
Router > enable
Router > # config terminal
Router config : interface <port>
Router config-if : IP address <ip> <subnet mask>
Router (config): no shut
```

Do this for all 3 routers. Then go to terminal of either PC & by ping to the other one, the message fails to deliver due to not setting up network static routers & next hop. We again go for the CLI of each router & setup the "next hop" using the commands.

```
> ip route <network-id> <mask> <next hop>
> ip route 400.0.0 255.0.00 20.0.0.20
                                (for router1)
```

This is done so that router recognizes which pathway to take when packet is required for particular destination.

Page No.:
Date:
YOUVA

Result:-

(i) Ping 40.0.0.1

Pinging 40.0.0.1 with 32 bytes of data
Reply from 40.0.0.10 : Destination host unreachable
Reply from 40.0.0.10 : Destination host unreachable
Reply from 40.0.0.10 : Destination host unreachable
Reply from 40.0.0.10 : Destination host unreachable
Reply from 40.0.0.10 : Destination host unreachable

Ping statistics
Packets sent = 4 : Recieved = 10 : lost = 4 (100% Loss)

(ii) ping 40.0.0.1

pinging 40.0.0.1 with 32 bytes of data

Request timed out
Reply from 40.0.0.10 bytes=32 time=2ms TTL
Reply from 40.0.0.10 bytes=32 time=2ms TTL
Reply from 40.0.0.10 bytes=32 time=2ms TTL

Ping statistics.
Packets sent = 4 : Recieved = 1. Loss = 1 (25% Loss)

Observation:-

The router connects LAN to the internet. It connects "different networks" with different IDs.
Packets are forwarded to the destination through network hopping. serial ports are used to connect 2 routers the connecting cables.

8/10

13/7/23

Experiment No-3

Aim : Configuring default route to the Router.

Topology:

Procedure:

- Do the connections as shown in the topology diagram.
- Assign an IP address to all the PCs.
- For router-to-router configuration do:
    - **(config)#ip route 0.0.0.0 0.0.0.0 {Next-hop-Address}**

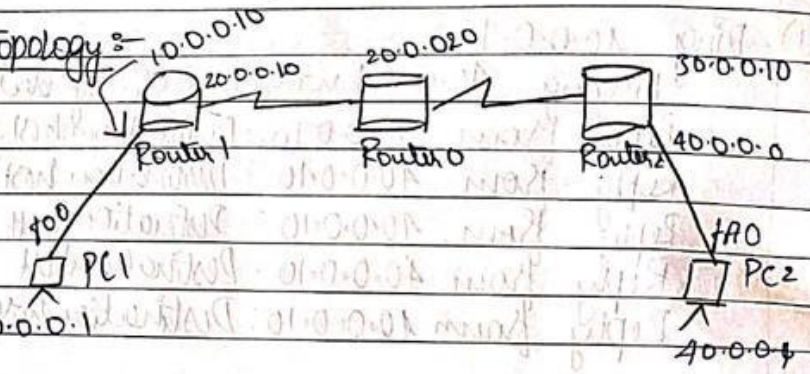
Output:

```
PC>ping 10.0.0.10

Pinging 10.0.0.10 with 32 bytes of data:

Reply from 10.0.0.10: bytes=32 time=50ms TTL=255

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

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

bytes=32 time=0ms TTL=255 Ping statistics for 10.0.0.10:

      Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),

Approximate round trip times in milli-seconds:

      Minimum = 0ms, Maximum = 50ms, Average = 12ms
```

Question? Experiment-3 :-

Default IP address :-

Topology :-



Procedure :-

- Select PC's & router & configure them with suitable IP.
- Make connections to all the devices using suitable connection links.
- For the routers to link between PC's & other routers. Use CLI mode & start typing the commands.
- 000 > Enable
  - > config T
    - > Interface fastethernet 0/0
      - > IP address 10.0.0.10 255.0.0.0
      - > No shut

Repeat these steps similarly to all the router in order to make default path.

> config t

ip route destination subnetmask device

> If router 0.0.0.0 0.0.0.0 20.0.0.10 similarly perform this to all the router

> Show IP router enter

10.0.0.0/8 is directly connected to fastethernet 0/0

20.0.0.0/8 is directly connected serial 2/0

0.0.0.0/0 [1/0] via 20.0.0.10

similarly all the router are connected.

Page No.:
Date:
YOUVA

Observation :-

In the previous one, we have given the IP address to all the router with destination subnet, & intermediate IP address of these particular devices, but here in this experiment we use a default IP address i.e. 0.0.0.0 & subnet mask 0.0.0.0 so that it can create a pass through channel to all the packets that are sent will be transfered by the intermediate device. This is generally used in large no. of device connections.

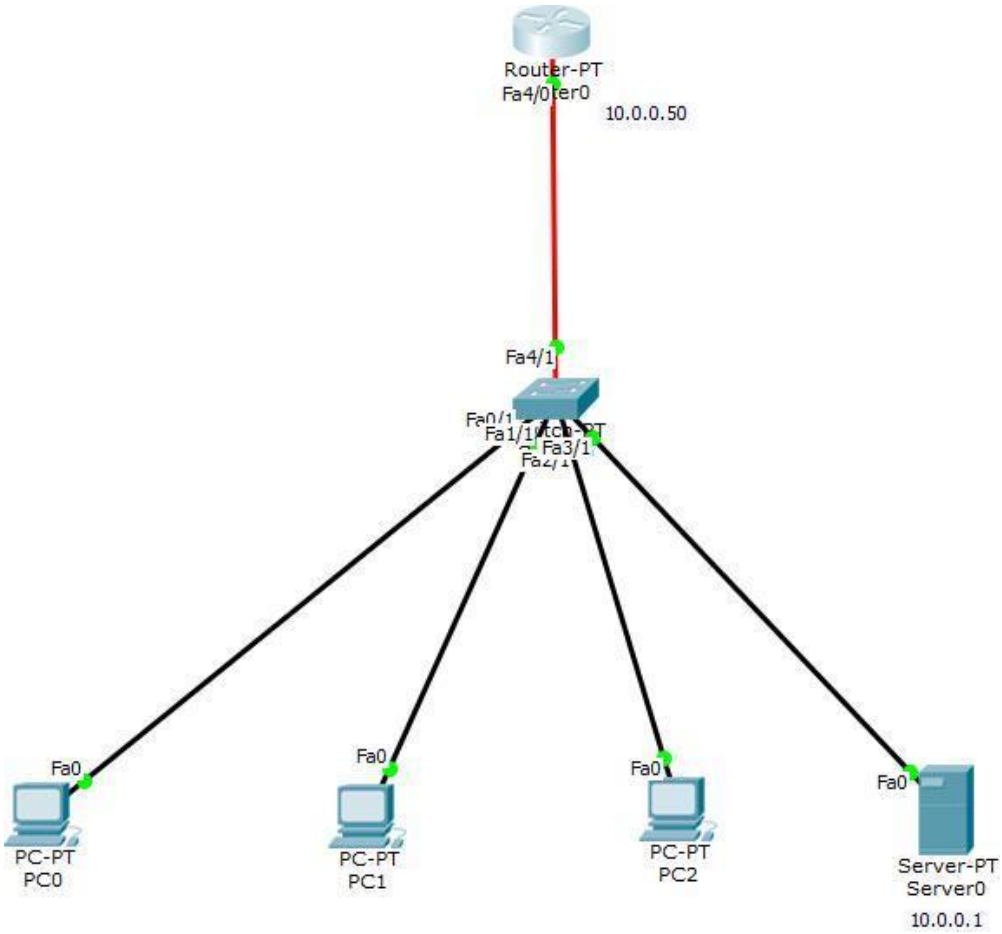
9/0

20/7/23

Experiment No-4

Aim : Configuring DHCP within a LAN in a packet Tracer

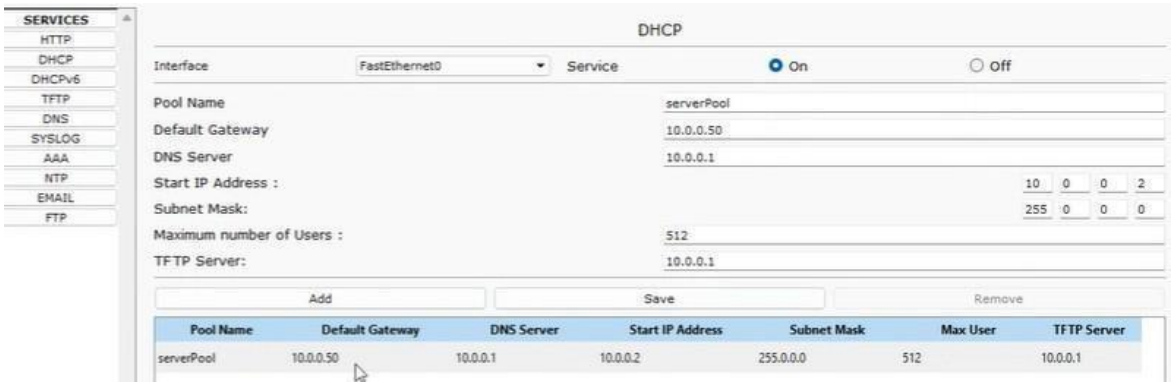
Topology:



Procedure:

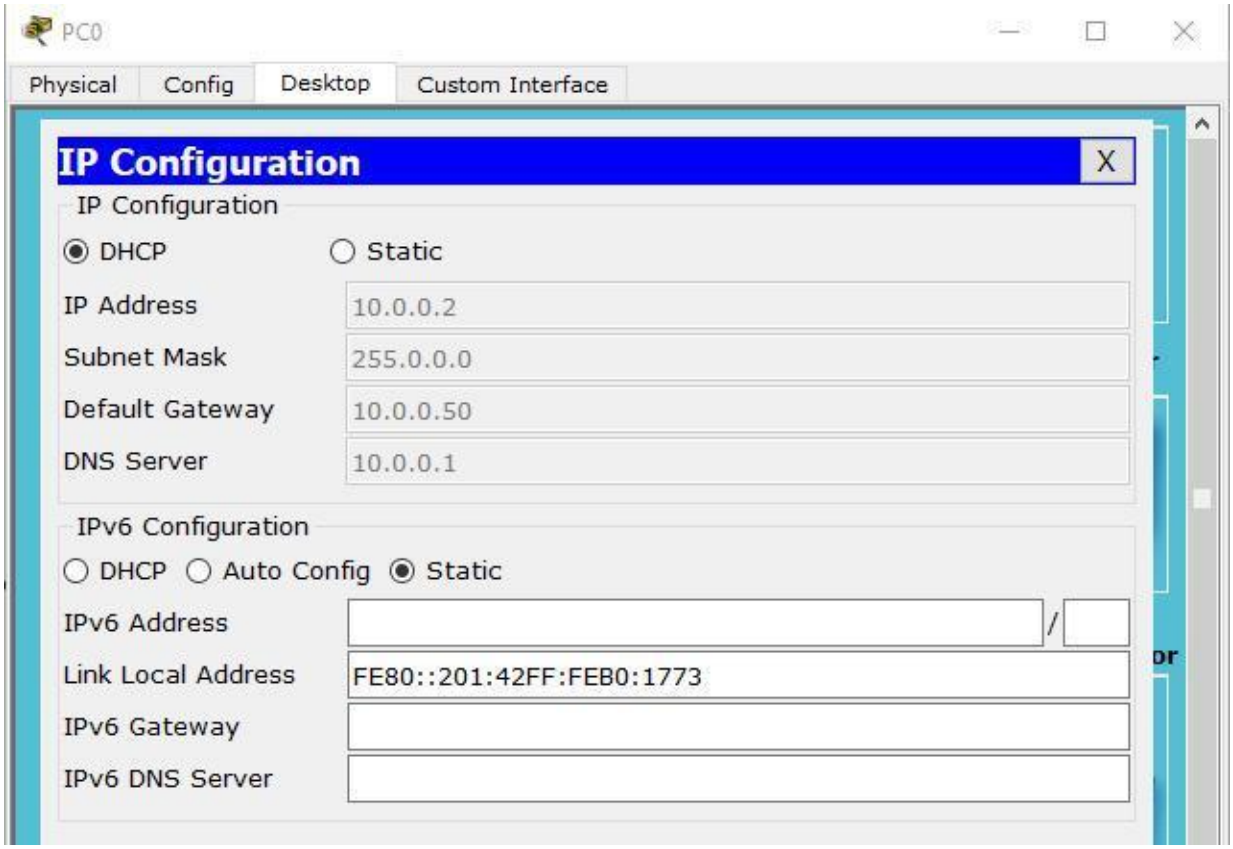
- Do the connection as shown in the topology diagram.

- For DHCP settings go to server and do the following



- For the PCs Go to ip configuration>Select DHCP.

=



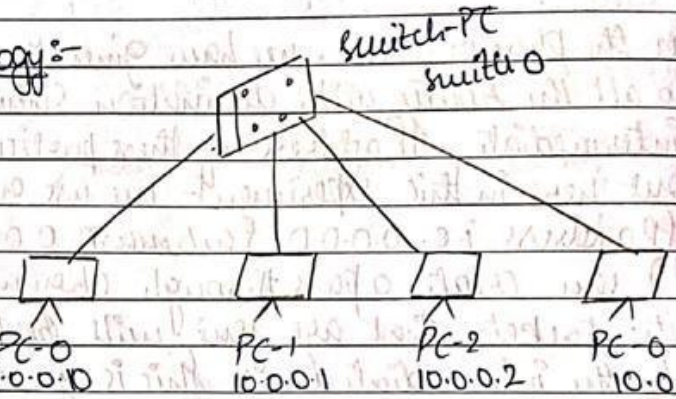
Output:

```
Packet Tracer PC Command Line 1.0

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=0ms TTL=128

Reply from 10.0.0.4: bytes=32 time=0ms TTL=128

Reply from 10.0.0.4: bytes=32 time=0ms 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 = 0ms, Average = 0ms
```

Experiment-4:-

Lab-A 4(A):-

Topology:-



Procedure:-

> Connect 3PC's & 1 server to a switch using copper straightthrough cable.
> Click on server & go to services tab select DHCP & turn on DHCP service.
> Set the IP address & the start IP address as 10.0.0.2 & click on s-button.
> Before this, set the IP address of server in config Tab under fastethernet to 10.0.0.1
> Next click on PC0 & go to desktop tab, here click on IP configuration. Select DHCP bar. It will request for an IP address & sucessfully get the DHCP request also set the IP address
> Repeat this steps for other 2pc's
> To send a packet across PC's, go to PC's command prompt & type ping destination IP 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.

Page No.:
Date:
YOUVA

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
Packets sent = 4 Recieved = 4 Lost = 0 (0% Loss)
Approximate round trip times in ms.
Minimum = 0ms, Maximum = 2ms, Average = 0ms

Observations :-

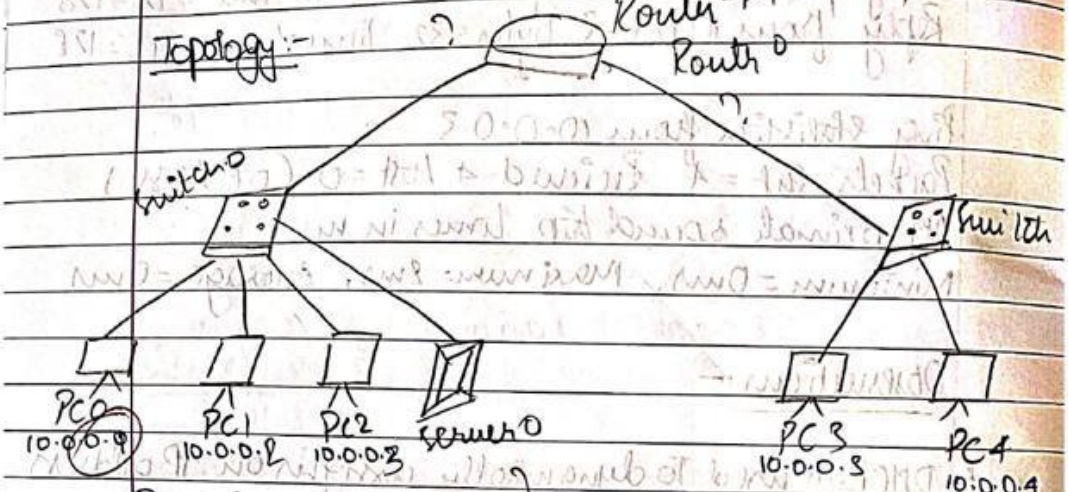
* DHCP is used to dynamically assigns an IP address to any device or node.
* It is a client-serial protocol in which servers manage a pool of unique IP address & also about client configuration parameters.
* DHCP enabled clients sends a request to DHCP server when they want to connect to a network.
* The DHCP server responds to the client request by providing IP configuration information from address pools, previously, specified by a network administrator.

Date:

4[B]:-

Aim:- Configure DHCP within a LAN & outside LAN.

Topology:-



Procedure:-

> Add a router, a switch & 2PC's do 4(A) program network & connect the router to both switches.

> Set the server IP address of server & with the help of server set the first 3PC's IP address through DHCP.

> Click on server.

> Go to desktop → IP configuration

> Add IP address, subnet mask & gateway

IP address 10.0.0.1
Subnet mask 255.0.0.0
Gateway 10.0.0.20

> Configure the router
Click on router go to CLI
enable.

Page No.:
Date:
YOUVA

Router # config t
Router (config)# fastethernet 0/0
Router (config) # ip address 20.00.20 255.0.0.0
Router (config-if)# no shut
Router (config-if) # exit
Router (config)# interface fastethernet 1/0
Router (config)# ip address 20.0.0.20 255.0.0.0
Router (config-if)# no shut
Router (config-if)# exit

Routing Table
Router → show ip route
10.0.0.0/1 is directly connected. fastethernet 0/0

> Go to server [DHCP server configuration]
- Select servers then go to DHCP
- Set service on
- Set start IP address pool (e.g. 20.0.0.0) then save

> Then configure the PC's
- Select a PC then desktop - go to IP configuration select DHCP.
- Repeat the same procedure for all other PC's

9/10

Observations:

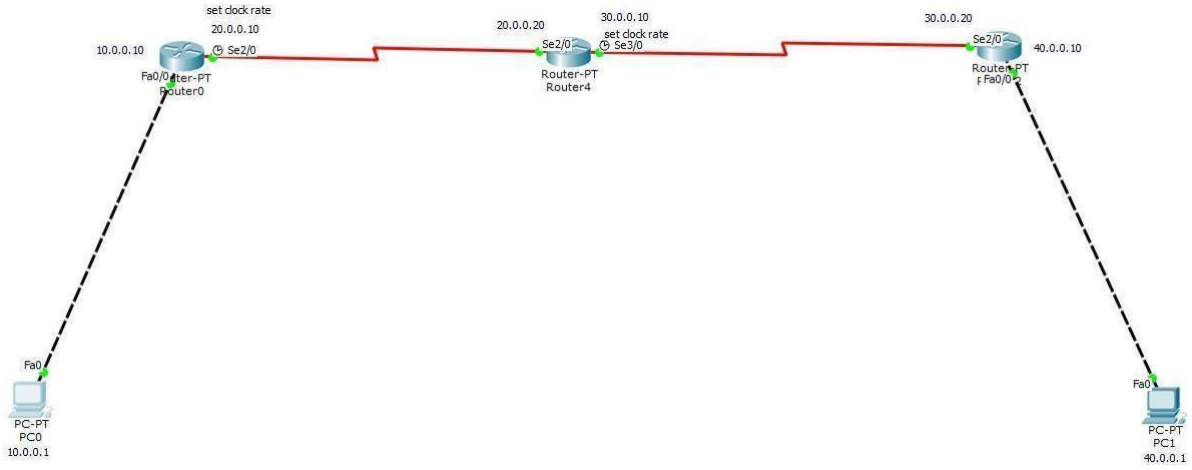
22/1/22

- DHCP is used to design IP address dynamically to different devices.
- To assign continuous IP address we create a server pool where we assign the starting IP address & a default gateway number & For PC's under different switches we create a different server pool again & start

Experiment No-5

Aim : Configuring RIP Routing Protocol in Routers.

Topology:



Procedure:

Router enable Router#config t

Router (config)#interface fastethernet0/0

Router (config-if)# ip address 10.0.0.10

255.0.0.0 Router (config-if)#no shut

Router (config-if)#exit

Router (config)#interface serial2/0

Router (config-if)#ip address 20.0.0.10

255.0.0.0 Router (config-if)#encapsulation ppp

Router (config-if)#clock rate 6400 Unknown clock rate

Router (config-if)#clock rate 64000

Router (config-if)#no shut

Routeí (config) #nteíface seíial2/0 Routeí

(config-if) #p addíess 20.0.0.20 255.0.0.0

#
Routeí (config-if) encapsulation ppp

Routeí (config-if)#no shut

Routeí (config) #nteíface seíial 3/0

Routeí (config-if)#p addíess 30.0.0.10

255.0.0.0 Routeí #config-if) encapsulation ppp

Routeí (config-if)#clock íate 64000 Routeí (config-if)no shut

Output:

```
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=12ms TTL=125

Reply from 40.0.0.1: bytes=32 time=6ms TTL=125

Reply from 40.0.0.1: bytes=32 time=14ms 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 = 6ms, Maximum = 14ms,
Average = 10ms
```

Experiment No-6:

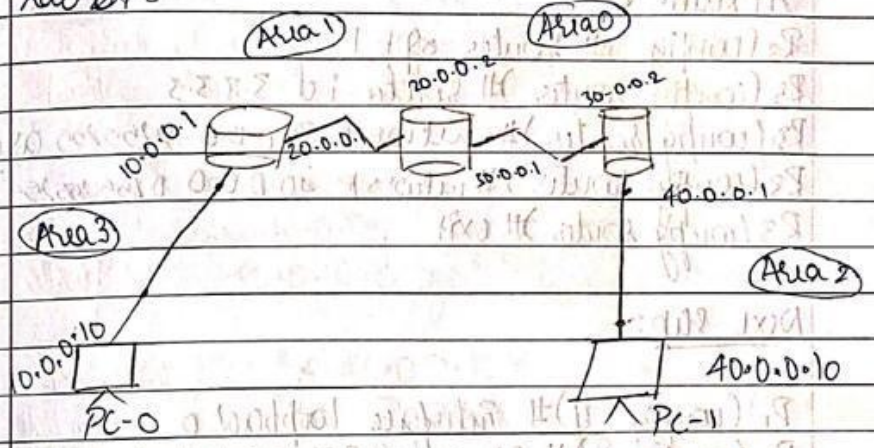
Aim : Configure OSPF routing protocol

Topology:

Page No.:
Date:

Experiment 8 :- OSPF -?

27/7/23 Lab 8 :-



Procedure :-

(i) Configure the PC's with IP address & gateway.
(ii) Configure each of the routers according to IP addr given.
(iii) Encapsulation ppp & clockrate need to be set as done in RIP protocol experiment.

In Router R1,
R1(config)# router ospf 1
R1(config-router)# router-id 1.1.1.1
R1(config-router)# network 10.0.0.0 0.255.255.255 area 3
R1(config-router)# network 20.0.0.0 0.255.255.255 area 1
R1(config-router)# exit.

In Router R2,
R2(config)# router ospf 1
R1(config-router)# router-id 2.2.2.2
R1(config-router)# network 20.0.0.0 0.255.255.255 area 1
R1(config-router)# network 30.0.0.0 0.255.255.255 area 0
R1(config-router)# exit

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
R3(config-router)# exit
```

Next step:-

```
R1(config-if)# interface loopback 0
R1(config-if)# ip address 172.16.1.252 255.255.0.0
R1(config-if)# no shutdown
```

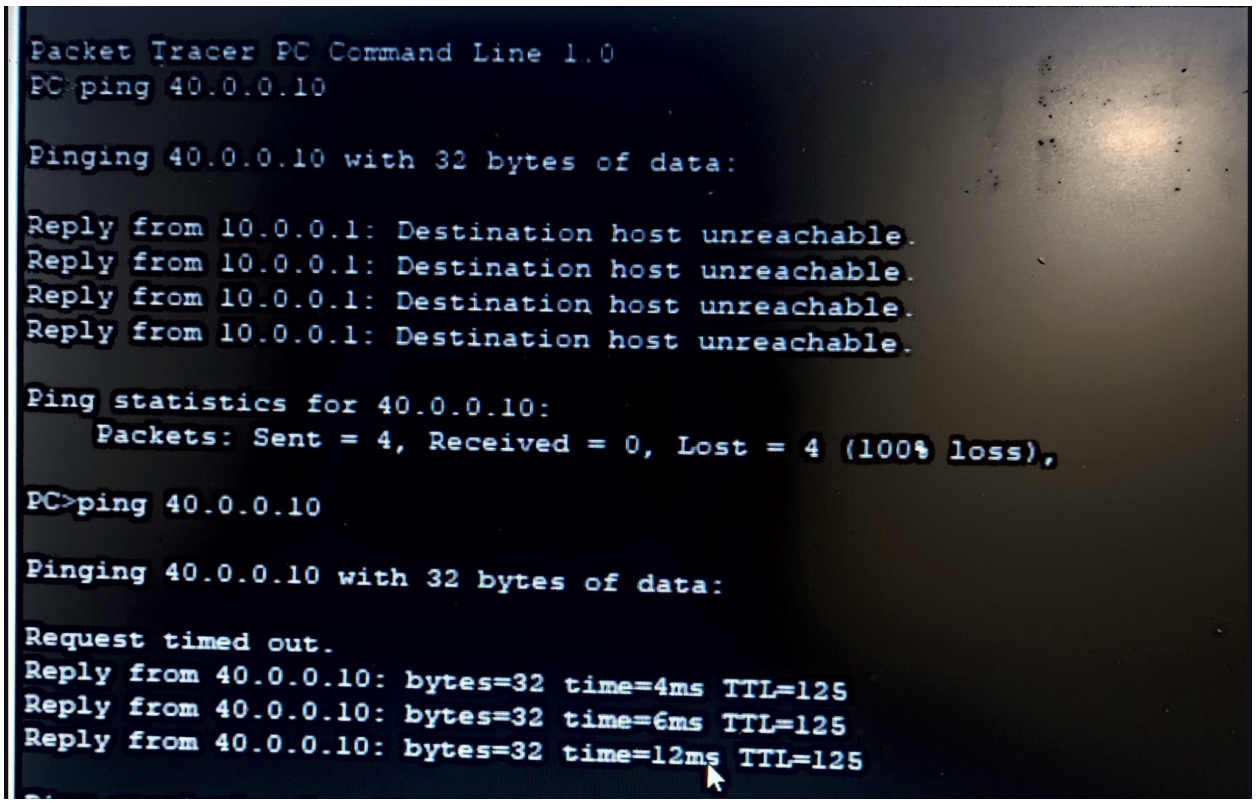
```
R2(config-if)# interface loopback 0
R2(config-if)# ip address 172.16.1.253 255.255.0.0.0
R2(config-if)# no shutdown
```

```
R3(config-if)# interface loopback 0
R3(config-if)# ip address 172.16.1.254 255.255.0.0.0
R3(config-if)# no shutdown
```

Next step:-

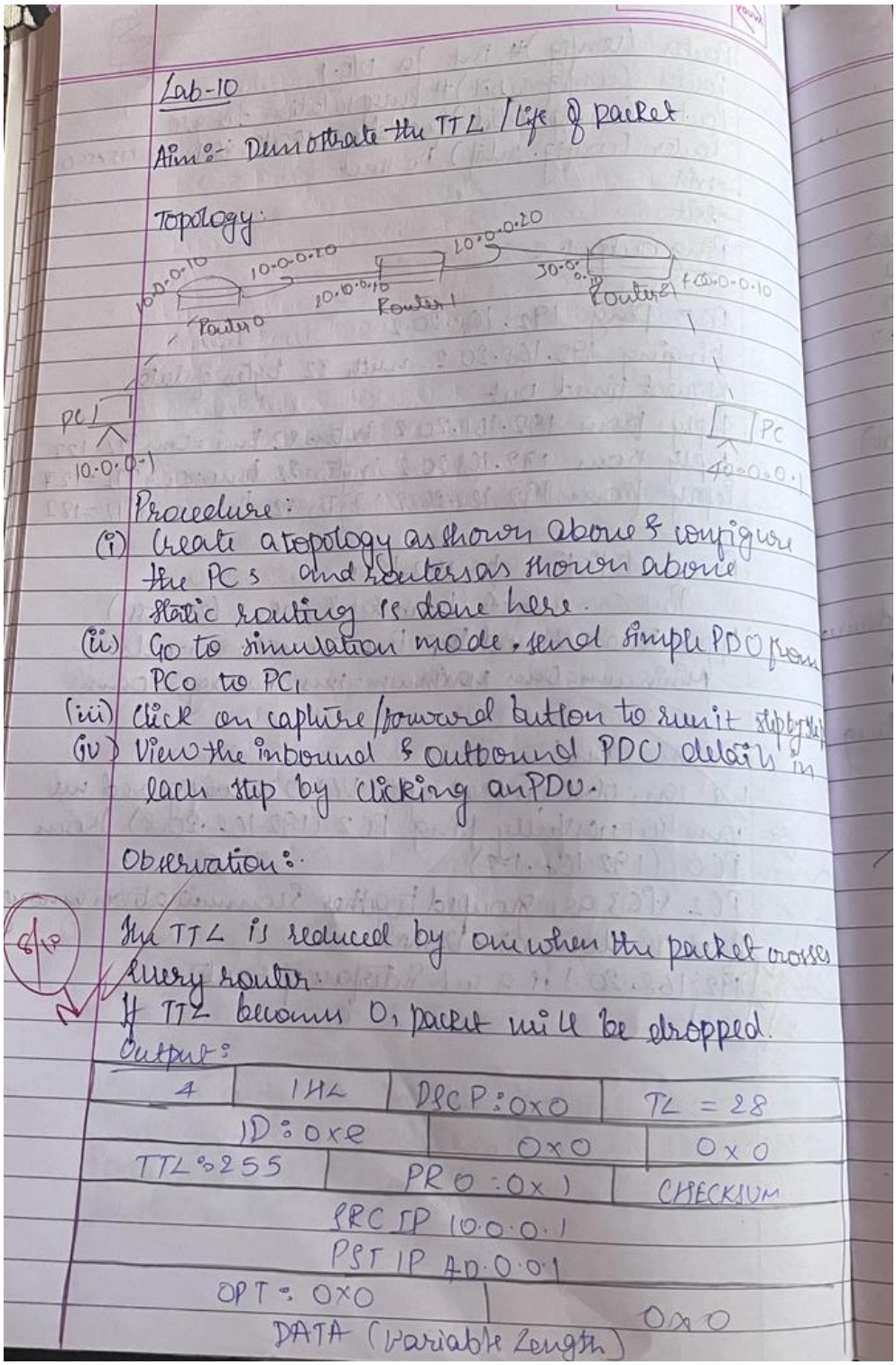
```
R1(config)# router ospf 1
R1(config-router)# area 1 virtual-link 2.2.2.2
R1(config-router)#
```

```
R2(config)# router ospf 1
R2(config-router)# area 1 virtual-link 1.1.1.1
R2(config-router)# exit
```


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

Experiment 7:Demonstrate the TTL/ Life of a Packet:

Demonstrate the TTL/ Life of a Packet:



Lab-10

Aim:- Demonstrate the TTL / Life of packet

Topology:

Procedure:

(i) Create a topology as shown above & configure the PCs and routers as shown above Static routing is done here.

(ii) Go to simulation mode. send simple PDU from PC0 to PC1

(iii) Click on capture/forward button to emit step by step

(iv) View the inbound & outbound PDU details in each step by clicking an PDU.

Observation:

8/10

The TTL is reduced by one when the packet crosses every router.

If TTL becomes 0, packet will be dropped.

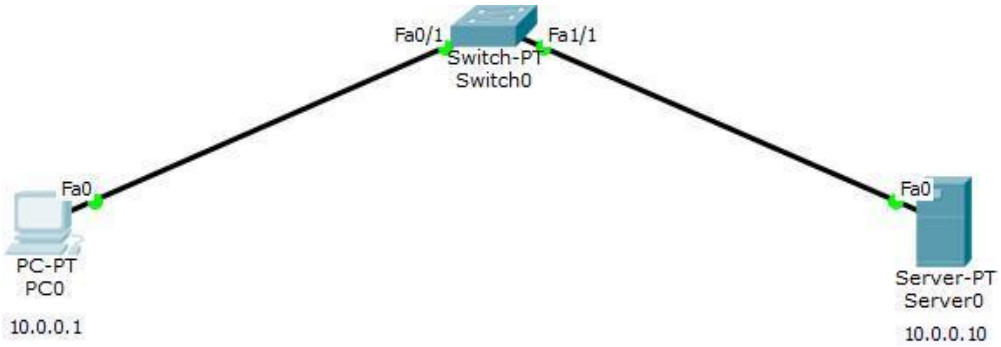
Output:

4	IHL	DSCP:0x0	TL = 28
ID:0xe		0x0	0x0
TTL:255	PRO:0x1		CHECKSUM
SRC IP 10.0.0.1			
PST IP 40.0.0.1			
OPT:0x0		0x0	
DATA (Variable Length)			

Experiment No-8

Aim: Demonstration of WEB server and DNS using Packet Tracer.

Topology:



Procedure:

- set up IP address for PC0 and server

- select PC, choose Desktop tab, choose Web Browser and enter 10.0.0.10 IP address, which displays the home page
- select server, choose Services tab, select HTTP and switch it on. CLick the edit button for ìndex.html and edit the file.

- switch the DNS on, and add a domain name - bmsce with the address 10.0.0.10
- search for the domain name in the web browser of the PC.

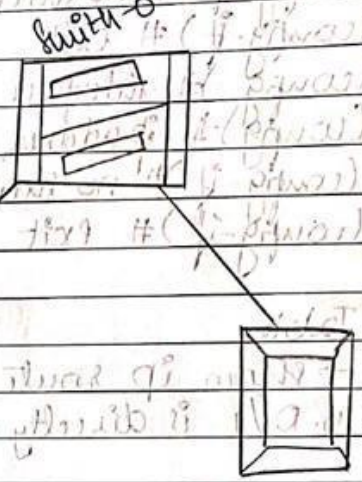
Output:

Page No.:
Date:

20/7/23 Lab-5 :-

Aim :- Configure Web server, DNS within a LAN.

Topology :-



Procedure :-

- connect a switch, PC & a server to form a LAN.
- Set PC's IP address by clicking on it & go to config tab in fastethernet option set IP address as 10.0.0.1 & subnet mask.
- Set server's IP address as 10.0.0.2 & subnet mask
- Go to PC's desktop & click on web browser, In the URL type 10.0.0.2, you will get default display
- To make a CV here, we need to make changes in server services.
- Go to server → services → HTTP → index.html Create CV & click on save.
- Again go to PC → Desktop → webbrowser & type 10.0.0.2 you will see CV or content that is changed.
- Next, goto server → services → DNS & switch on the services. Now add a domain name & type the IP address 10.0.0.2. Press add & save it.
- Again go back to PC → desktop → webbrowser & type the given domainname. We will see our CV that we change

Experiment No-9

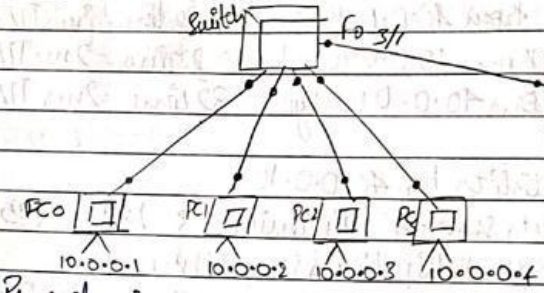
To construct simple LAN and understand the concept and operation of Address Resolution Protocol (ARP)

Lab 8

3/8/23

Aim:- To construct simple LAN & understand the concept and operation of Address Resolution Protocol (ARP)

ARP:-



Procedure:-

i) Create a topology of 4 PC's & a server
(ii) Set IP address to all of them.
(iii) Connect the PC's through a switch.
(iv) Use the inspect tool to click on a PC to see the ARP table
(v) Command in CLI for the same is arp -a
(vi) Initially ARP table will be empty.
-> Use the capture button in the simulation panel to go step by step so that the changes in ARP can be clearly noted.
-> Go to the search button in the panel to get ARP tables of all the end devices.
-> Observe the switch as well the nodes update the ARP table as & when a new communication starts

Ping output:-

PC > arp -a
= PC > ping 10.0.0.2
PC > ping 10.0.0.3
PC > ping 10.0.0.4

Internet address	Physical address	Type
10.0.0.2	00d0.bd15.3918	dynamic
10.0.0.3	00e0.0fb7.b7e8	dynamic
10.0.0.4	00d0.bc57.0a38	dynamic

Observation:-

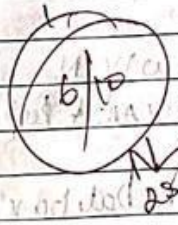
We observe the changes when we capture every time when we do in all the PC's. ARP tables will have values empty at the starting & values will be formed whenever we ping each PC in the ping.

Now in switch(1):-

Switch > show mac address table

Mac Address Table

Vlan	Mac Address	Type	Port
1	0001.9613.7660	Dynamic	Fa 1/1
1	0001.9796.1267	Dynamic	Fa 2/1
1	0090.2673.c004	Dynamic	Fa 0/1
1	00e0.a349.9164	Dynamic	Fa 3/1



Experiment10:To understand the operation of TELNET by accessing the router in server room from a PC in IT office.

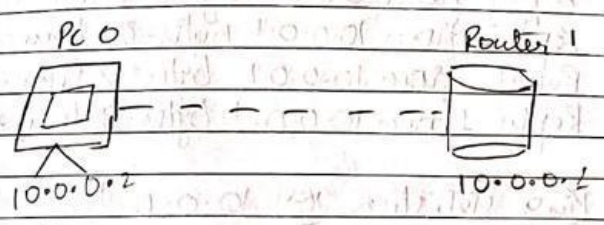
Page No.:
Date:

Experiment-12 :-

Aim:

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

Topology:



Procedure:

(i) Configure topology as above, IP address of PC & gateway & router configuration as normal.

In Router (1)

Router > enable

- config t
- host name R1
- enable secret P1
- interface fastethernet 0/0
- ip address 10.0.0.1 255.0.0.0
- no shut
- login

~~9.~~ login disable on line 132, until password is set

~~1.~~ login disabled on line 133, until password is set

1. login disabled on line 132, until password is set
1. login disabled on line 135, until password is set
1. login disabled on line 136, until password is set
1. login disabled on line 137, until password is set

R1 (config-line)# password P0

R1 (config-line)# exit

R1 #w R

Page No.:
Date:

Ping output:

In PC0

PC > ping 10.0.0.1

Pinging 10.0.0.1 with 32 bytes of data

Reply from 10.0.0.1 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)

Packet sent=4, Recieved=4, Lost=0 (0% Loss)

Approximate rounded trip time in ms,

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

PC > telnet 10.0.0.1

Trying 10.0.0.1 ---Opm

User access verification

password (typed p0)

R1>enable

password (typed p1)

R1# show ip route

Code;

Gateway of last is not set 10.0.0.0/8 is directly connected fastEthernet 0/0

Observation:-

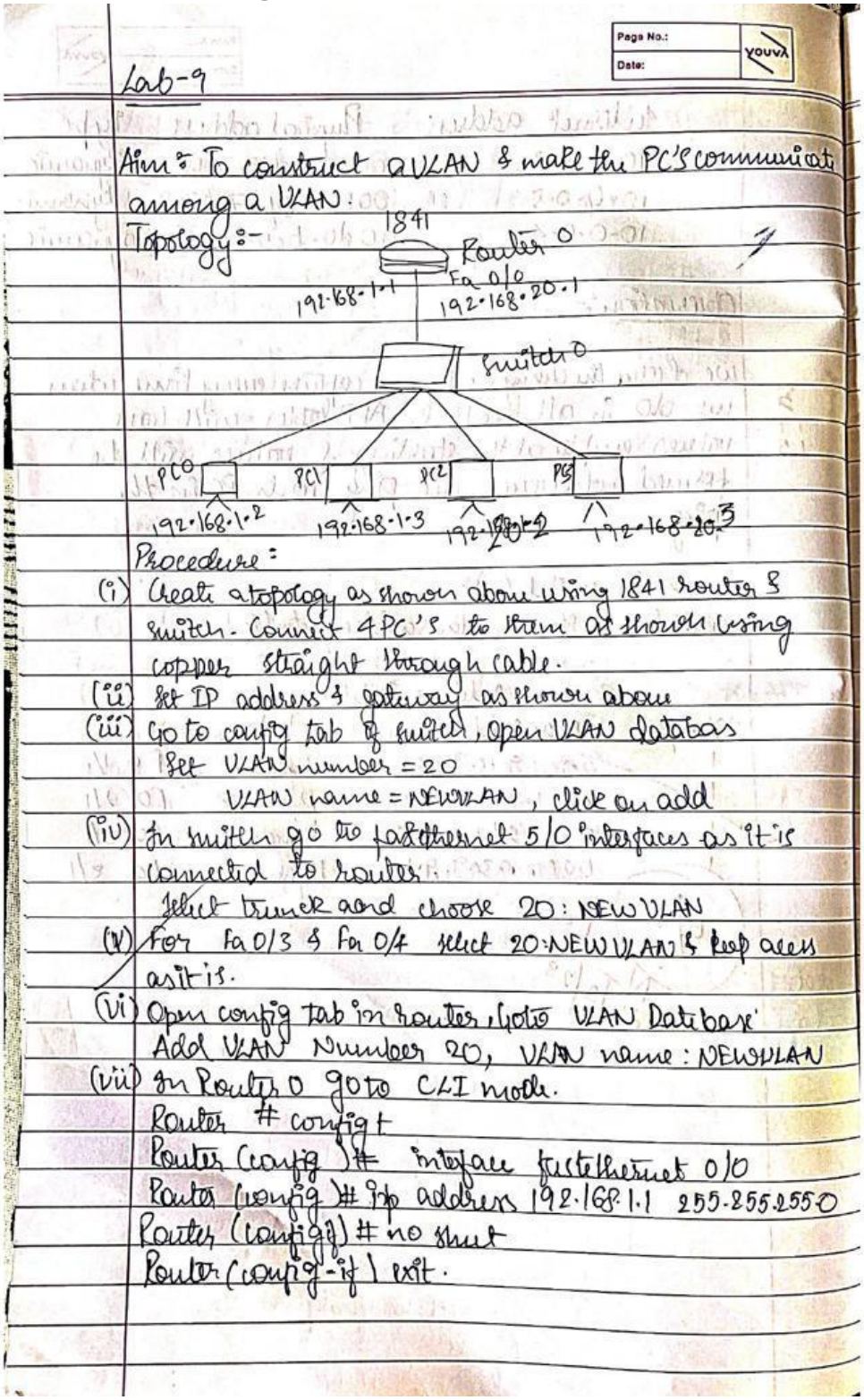
(i) TELNET is used by terminal emulation programs that allow you to

(ii) we logged int 10.0.0.1 through 10.0.0.2 device

(iii) the password when typed is not visible.

8/10

Experiment11:To construct a VLAN and make the PC's communicate among a VLAN



Lab-9

Aim: To construct a VLAN & make the PC's communicate among a VLAN.

Topology:-

Procedure:

(i) Create a topology as shown above using 1841 router & switch. Connect 4 PC's to them as shown using copper straight through cable.

(ii) Set IP address & gateway as shown above

(iii) Go to config tab of switch, open VLAN database
Set VLAN number = 20
VLAN name = NEWVLAN, click on add

(iv) In switch go to fastethernet 5/0 interfaces as it is connected to router.
Select trunk and choose 20: NEW VLAN

(v) For Fa 0/3 & Fa 0/4 select 20:NEW VLAN & keep access as it is.

(vi) Open config tab in router, goto VLAN Database
Add VLAN Number 20, VLAN name: NEWVLAN

(vii) In Router 0 go to CLI mode.
Router # config t
Router (config)# interface fastethernet 0/0
Router (config)# ip address 192.168.1.1 255.255.255.0
Router (config)# no shut
Router (config-if)# exit.

Page No.:
Date:
YOUVA

Router (config)# int fa 0/0.1
Router (config-subif)# encapsulation dot1q 20
Router (config-subif)# ip address 192.168.20.1 255.255.255.0
Router (config-subif)# no shut.
→exit
→exit

Ping output :-

PC> ping 192.168.20.2
pinging 192.168.20.2 with 32 bytes of data
Request timed out.
Reply from 192.168.20.2 : bytes=32 time=0ms TTL=127
Reply from 192.168.20.2 : bytes=32 time=0ms TTL=127
Reply from 192.168.20.2 : bytes=32 time=0ms TTL=127

Ping statistics for 192.168.20.2
Packets: Sent=4, Recived=3 Lost=1 (25% Loss)
Approximate round trip times in milli seconds:
Minimum=0ms maximum=1ms, Average=0ms.

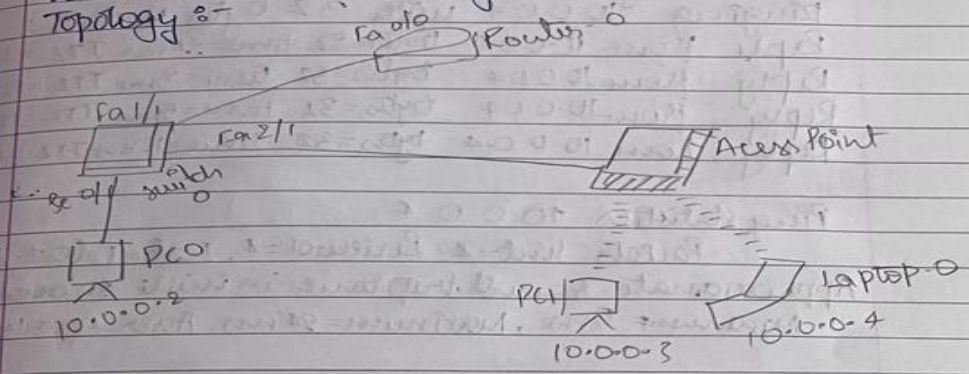
Observation:
We can observe that after VLAN is configured we can successfully ping PC2 (192.168.20.2) from PC0 (192.168.1.2)
PC2 & PC3 are grouped together & communication among them is done via VLAN
192.168.20.1 is a sub interface of router.

Experiment12 :To construct a WLAN and make the nodes communicate Wirelessly

Lab-11

Aim :- To construct a WLAN & make the nodes communicate wirelessly.

Topology :-



Procedure :

(i) Create the topology as shown above.

(ii) Configure PC the access point-PT in config of port 1
→ SSID = WLAN
select WEP & give any 10 digit hexadecimal as password
1234567890

(iii) Set IP address of the PC.

) To configure PC1 & Laptop with wireless standard
Switch off device.

Drag the existing PT-HOST-NM-1AM to LHS & place it on its name mentioned

Drag the WMP300N wireless interface to empty port switch on the device.

) In config Tab of the device a new wireless interface will now become visible.

) Now configure the SSID, WEP, WEP Key, gateway & IP address of PC & Laptop.

WEP Key = 1234567890

Gateway = 10.0.0.1

Date: YOUVA

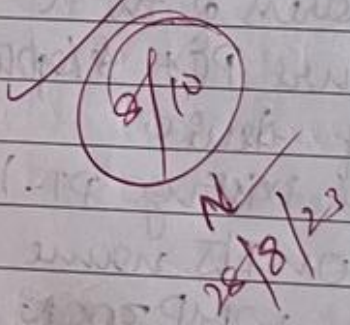
Ping Output :-

In PC0 go to command prompt
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=24ms TTL=127
Reply from 10.0.0.4 bytes=32 time=15ms TTL=127
Reply from 10.0.0.4 bytes=32 time=5ms TTL=127
Reply from 10.0.0.4 bytes=32 time=12ms TTL=127.

Ping statistics 10.0.0.4
Packets: Sent=4, Recieved=4. Lost=0(0% loss)
Approximate round trip times in milli seconds.
Minimum=5ms, Maximum=24ms. Average=1ms

Observation :-

We can ping each & every devices to the other devices so we can observe that wireless connection is done sucessfully. When connection is established there is stripped line connected access points and end devices.



CYCLE - 2

Program 1: Write a program for error-detecting code using CRC-CCITT (16-bits).

Code :

```
#include <iostream>

#include <string.h>

using namespace

std;

int crc(char *ip, char *op, char *poly, int mode)

{

   strcpy(op, ip);

   if (mode) {

      for (int i = 1; i < strlen(poly);

         i++) strcat(op, "0");

   }

   /* Perform XOR on the msg with the selected polynomial */
   for (int i = 0; i < strlen(ip); i++) {

      if (op[i] == '1') {

         for (int j = 0; j < strlen(poly); j++) {

            if (op[i + j] == poly[j])

               op[i + j] =

            '0'; else

               op[i + j] = '1';

         }

      }

   }

   /* check for errors. return 0 if error detected */
```

```
    for (int i = 0; i < strlen(op); i++)
        if (op[i] ==
            '1') return 0;
    return 1;
}
int main(){
    char ip[50], op[50], recv[50];
    char poly[] = "10001000000100001";

    cout << "Enter the input message in binary"<< endl
cin >> ip;
    crc(ip, op, poly, 1);
    cout << "The transmitted message is: " << ip << op + strlen(ip) <<
    endl; cout << "Enter the recevied message in binary" << endl; cin
    >> recv;
    if (crc(recv, op, poly, 0))
        cout << "No error in data" <<
    endl; else
        cout << "Error in data transmission has occurred" << endl;
    return 0;
}
```

Output

```
/tmp/kiPKSgKXwt.o
Enter the input message in binary
11100011100100000
The transmitted message is: 1110001110010000010011100100010001
Enter the recevied message in binary
1110001110010000010011100100010001
No error in data
```

Program2:Write a program for congestion control using Leaky bucket algorithm.

```
#include <bits/stdc++.h>
using namespace std;
int main()
{
	int no_of_queries, storage, output_pkt_size;
	int input_pkt_size, bucket_size, size_left;
	storage = 0;
	no_of_queries = 4;

	// total no. of packets that can
	// be accommodated in the bucket
	bucket_size = 10;

	// no. of packets that enters the bucket at a time
	input_pkt_size = 4;

	// no. of packets that exits the bucket at a time
	output_pkt_size = 1;
	for (int i = 0; i < no_of_queries; i++) // space left
	{
		size_left = bucket_size - storage;
		if (input_pkt_size <= size_left) {
			// update storage
			storage += input_pkt_size;
		}
		else {
			printf("Packet loss = %d\n", input_pkt_size);
		}
		printf("Buffer size= %d out of bucket size= %d\n",
			storage, bucket_size);
		storage -= output_pkt_size;
	}
	return 0;
}
```

Output

```
Buffer size= 4 out of bucket size= 10
Buffer size= 7 out of bucket size= 10
Buffer size= 10 out of bucket size= 10
Packet loss = 4
Buffer size= 9 out of bucket size= 10
```

Program 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

Code :

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()
```

Client :

```
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(
```

Output :

```
Enter file name: hello.cpp

From Server:

#include<iostream>

using namespace std;

int main(){
    int n;
    cin>>n;
    cout<<n<<endl;
    return 0;
}
```

```
Sent contents of hello.cpp
The server is ready to receive
```

Program 4 : 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

Code :

Server :

```
from socket import

*          server
Port = 12000

serverSocket = socket(AF_INET,

SOCK_DGRAM)
serverSocket.bind((gethostname(), serverPort))
print ("The server is ready to receive")
while 1:

    sentence,clientAddress = serverSocket.recvfrom(2048)

    file=open(sentence,"r")

    l=file.read(2048)

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

    print("sent back to client",l)

    file.close()
```

Client :

```
from socket import *

serverName = gethostname()

serverPort = 12000

clientSocket = socket(AF_INET, SOCK_DGRAM)

sentence = input("Enter file name")
clientSocket.sendto(bytes(sentence,"utf-8"),(serverName,
serverPort)) filecontents,serverAddress =
clientSocket.recvfrom(2048) print ('From Server:',
```

filecontents) clientSocket.close()

Output

```
Enter file namehello.cpp
From Server: b'#include<iostream>\n\nusing namespace std;\n\nint main(){\n    in
t n;\n    cin>>n;\n    cout<<n<<endl;\n    return 0;\n}'
```