INDEX

Name Shashank Patel C.J.

Standard Section 5'F' Roll No.

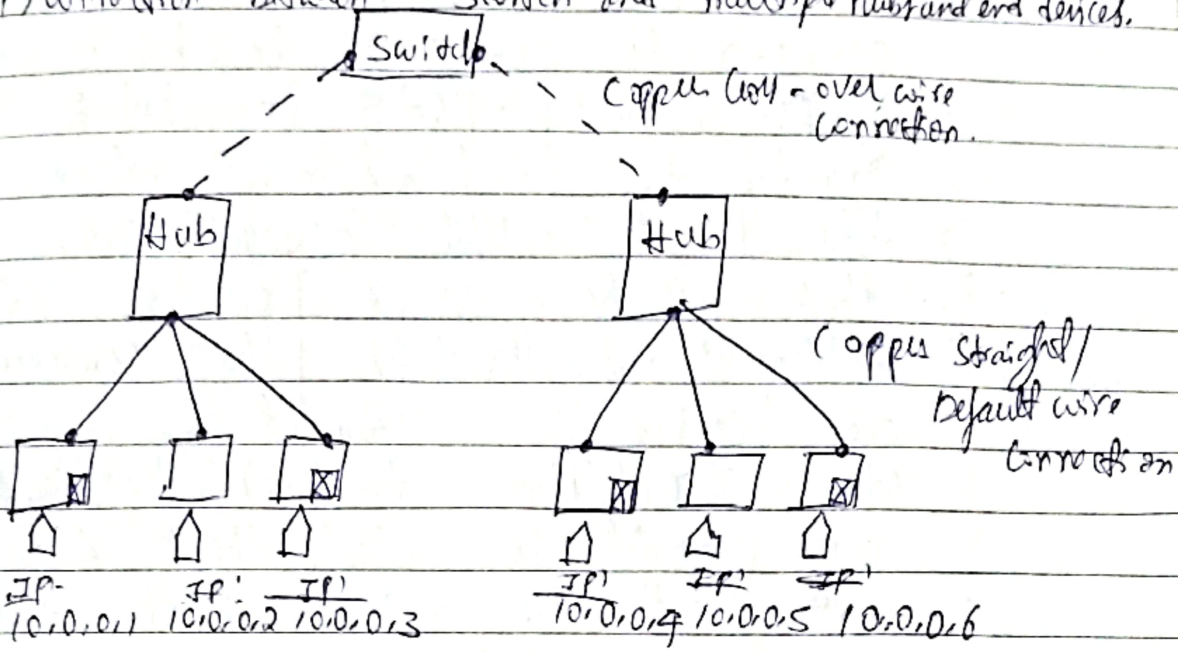
Subject Computer Networks Lab observation book

| SL No. | Date | Title | Page No. | Teacher Sign / Remarks |
|---|---|---|---|---|
| 1 | 27/9/24 | Experimenting with cisco packet tracer. | | 27/9 |
| 2 | 04/10/24 | connection between multiple hubs, Switch and end devices. | | 4/10 |
| 3 | 18/10/24 | multiple router connections. | | 25/10/24 |
| 4 | 25/10/24 | i) Static route<br>ii) default route | | 25/10/24 |
| 5 | 08/11/24 | i) DHCP configuration<br>ii) DNS server configuration | | |
| 6 | 15/11/24 | i) CRC problem<br>ii) Leaky Bucket Algorithm | | 15/11/24 |
| 7 | 22/11/24 | Routing Information protocol(RIP) | | 22/11/24 |
| 8 | 29/11/24 | OSPF - Open Short Path First. | | |
| 9 | 20/12/24 | ARP & TCP & UDP programs. | | 20/12/24 |
| 10 | 20/12/24 | Telnet | | 20/12/24 |
| 11 | 20/12/24 | VLAN | | 20/12/24 |
| 12 | 20/12/24 | WLAN | | 20/12/24 |

Bafna Gold
Date: 04/10/24

# Lab - 1

1) Task 1 : Create a topology involving multiple hubs, and a switch - connecting them to simulate a simple PPV.
i) Connection between Switch and multiple hubs and end devices.




Implementation :

* Selected the Six end devices which are generic ones, which are PC-PT 0 to PC-PT 5 respectively.
* Dividing three of them into two hubs and Connecting them.
* Connecting hub and end devices by Copper Straight through wire connection for both hubs.
* Configuring the end devices by IP-address for 10.0.0.1 to 10.0.0.6 respectively for all the 6 end devices.
* That created a Subnet mask of 255.0.0.0 for all the 6 end devices named as PC PT-0 to PC-PT-5 respectively.
* Connecting two hubs with the one switch with a Copper-Cross-over wire connection.
* The connection is Automated based on the unique port and ethernet connection between the

hubs and Switch, hubs and end devices. the devices are connected.

* A message (Simple PDU) was passed from PC-PT-0 to PC-PT-2 in hub-1 and PC-PT-3 to PC-PT-5 in hub-2 in the connected to Switch and in this network.
* After the successful message sent, we need to check for the source end device and go to Desktop option, then Selecting the command prompt and writing a command of ping destination-name where TTL (Time to live) for the end device is used until it sends the 4 datapackets and all are received by the destination PC and returns back.

Observation:-

* It is found that connected network forms a Star topology between all the N-devices in the Network.
* A green signal at the both end is seen which indicates the successful connection.
* The Status was Successful and type of layer was ICMP for both the messages i.e which was sent and the messages are received by the destination end devices.
* The hub broadcast the data to all the connected device in the network and Switch identify the MAC media access control) address or the data header to transmit the data only to the device that requested it, so, here the Switch checks

whether the message is sent between two hubs or not and if sent it sends the message to a given destination hub - end device and else it becomes incompatible and does not send the message from the hubs from which it received and does not operation in the network.

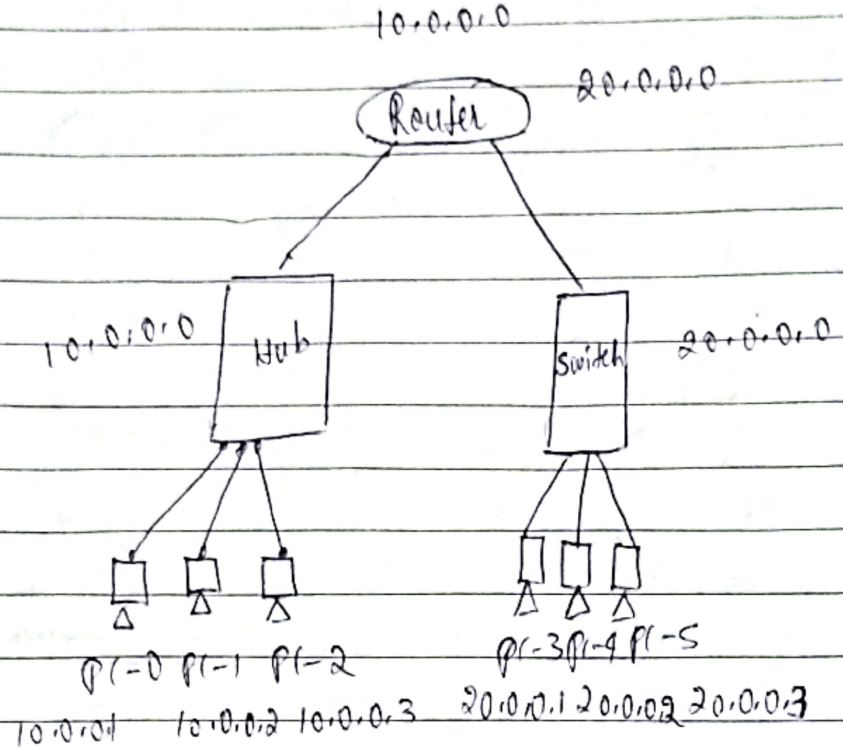
* The command prompt in the Desktop option in the source end device where the ping is done to the destination ip address, when the 4 data packets are sent from the source end device and 32 4 replies are received from the destination and total size of the ip address, time and TTL is shown or displayed. Ping statistics for destination address is displayed with sent, received and lost of all the packets. The time for the ~~round~~ trip is calculated and displayed with minimum, Maximum and Average time in ms for the packets.

* The message sent from the source and when we click on the N-device, we can see the different layers of the Network and where the message is present.

4/10/24

Lab-2 18/10/24

A topology using a router, switch, hub and PC's.




Implementation:

→ The PC's connected to hub have same network address.

→ The PC's connected to Switch have same network address, i.e., for hubs starting from 10, for Switches starting from 20, (10.0.0.0), (20.0.0.0).

Observation:

→ For router give IP address for the port connected to hub starting from 10 whereas for the port connected to Switch from 20.

→ For each PC give gateway address which is same as the IP address of the router's port to which it is connected.

→ Label the ping process is done has mentioned above.

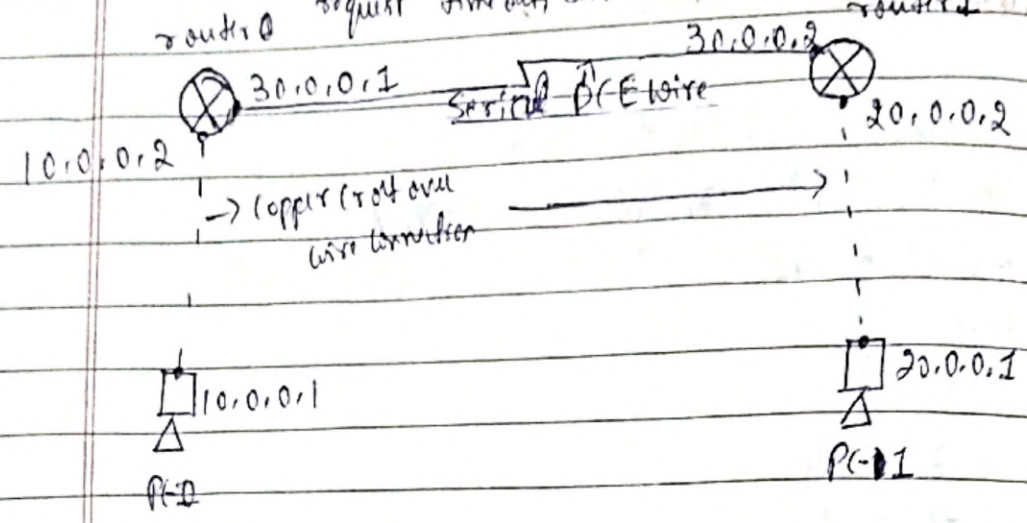
## Router CLI Commands

```
interface fastethernet 0/0
ip address 10.0.0.0 255.0.0.0
no shut down
exit

router>enable
router#config t
router(config)

interface fastethernet 0/0
ip address 20.0.0.0 255.0.0.0
no shutdown
exit
```

Lab-3

12/10/21

* topology using a multiple routers, end devices and connection between multiple networks, Ping response, ping router 0 request time out, destination unreachable.




Implementation:

* Select two end devices which are generic ones which are PC-0 and PC-1
* Configure the Ip address of both end devices with 10.0.0.1 and 20.0.0.1 respectively.
* Select two routers and which are generic ones which are router-0 and router-1.
* Configure the Ip address of both routers after the connecting router-0 to PC-0 by copper cross over same thing between router-1 and PC-1.
* Connection between router-0 and router-1 by Serial DCE Connection.
* The PC-0 and router are connected by the port fastethernet by 0/0 by configuring Ip address of router-0 is 10.0.0.2 and same for router-1 is 20.0.0.2 and the same Ip address are configurable Gateway for PC-0 and PC-1 respectively.
* The router-0 and router-1 are connected by Serial 2/0 connection port and configuring Ip

address as 30.0.0.1 and 30.0.0.2 respectively.

Observation :-

* The message (Simple PDU) is sent between PC-0 and PC-1 but it is blocked failed because of two different networks.
* Now to get successful first open CLI command and type ip route 20.0.0.0 255.0.0.0 30.0.0.2 for router-0 and 10.0.0.0 255.0.0.0 30.0.0.1 for router-1 respectively.
* After establishing connection, send from PC0 and the message is successfully sent to pc-1.
* Thus establishing the common connection between the two networks.

request timed out

ping 20.0.0.1

pinging 20.0.0.1 with 32 bytes of data:

Request timed out
Request timed out
Request timed out
Request timed out

ping statistics for 20.0.0.1:

packets: Sent = 4, Received = 0, Lost = 4 (100% Loss).

destination

ping 20.0.0.1

pinging 20.0.0.1 with 32 bytes of data:

Reply from 20.0.0.1 : bytes = 32
Reply from 20.0.0.1 : bytes = 32
Reply from 20.0.0.1 : bytes = 32
Reply from 20.0.0.1 : bytes = 32

ping (statistics) for 20.0.0.1 :

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

Unreachable:

ping 20.0.0.1

pinging 20.0.0.1 with 32 bytes data

Reply from 20.0.0.1 Destination not reachable

Output

18/10/24

Show ip route

CLI Comm

router# 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.

ip route 20.0.0.0 255.0.0.0 30.0.0.2

ip route 20.0.0.0 255.0.0.0 30.0.0.1

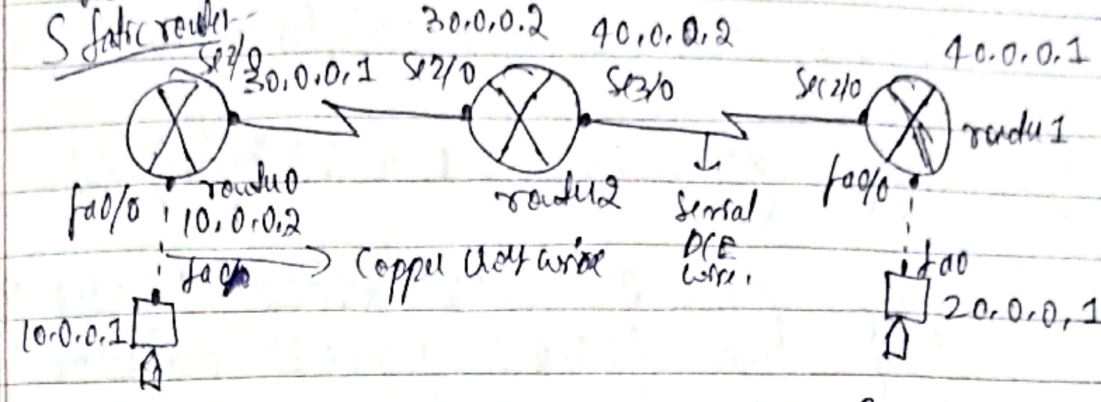
25/10/24

Lab-1

Bafna Gold
Date 25/10/21

3. Configure default route, Static route to the router.

S Static routing




Static routing Implementation:

→ Place two generic end devices PC-PT-PC0 & PC-PT-PC1 & Config their with ip address 10.0.0.1 & 20.0.0.1 respectively.

→ Then place three routers, router PT0, router PT1 & router PT2 & Connect router 0 & router 1 with PC-0 & PC-1 respectively with Copper Cross over wire.

→ Configure router 0 with ip address 10.0.0.2 & router 1 with ip address 20.0.0.2 respectively.

→ Connect router 0 & router 2 with Serial DCE & Similarly router 1 & router 2 with Serial DCE Se2/0 and Se3/0 port.

→ Next, configure the Serial port with 30.0.0.1 & 30.0.0.2 for router 0 & router 2, 30.0.0.1 & 40.0.0.2 for router 1 & router 2 respectively.

→ Add gateway 10.0.0.2 & 20.0.0.2 for PC-0 & PC-1 respectively.

Observation:

* After Successful connection/ ping/ Simulating package from PC-0 to PC-1.

~~sending packet~~

PC> ping 20.0.0.1

pinging 20.0.0.1 with 32 bytes of data:

~~Reply from 20.0.0.1 with 32 bytes of data~~

Reply from 20.0.0.1: bytes=32 time=8ms TTL=125
Reply from 20.0.0.1: bytes=32 time=8ms TTL
Reply from 20.0.0.1: bytes=32 time=8ms TTL
Reply from 20.0.0.1: bytes=32 time=8ms TTL

Ping statistics for 20.0.0.1:
Packets: Sent = 4, Received = 4, Lost = 0 (0% Loss),
Approximate round trip times in milli-seconds:
Minimum = 8ms, Maximum = 8ms, Average = 8ms.

## Configuration of router in CLI

router 0:

router(config)# ip route 40.0.0.0 255.0.0.0 30.0.0.2
router(config)# ip route 20.0.0.0 255.0.0.0 30.0.0.2

router 2:

router(config)# ip route 10.0.0.0 255.0.0.0 30.0.0.1
router(config)# ip route 20.0.0.0 255.0.0.0 40.0.0.2

router 1:

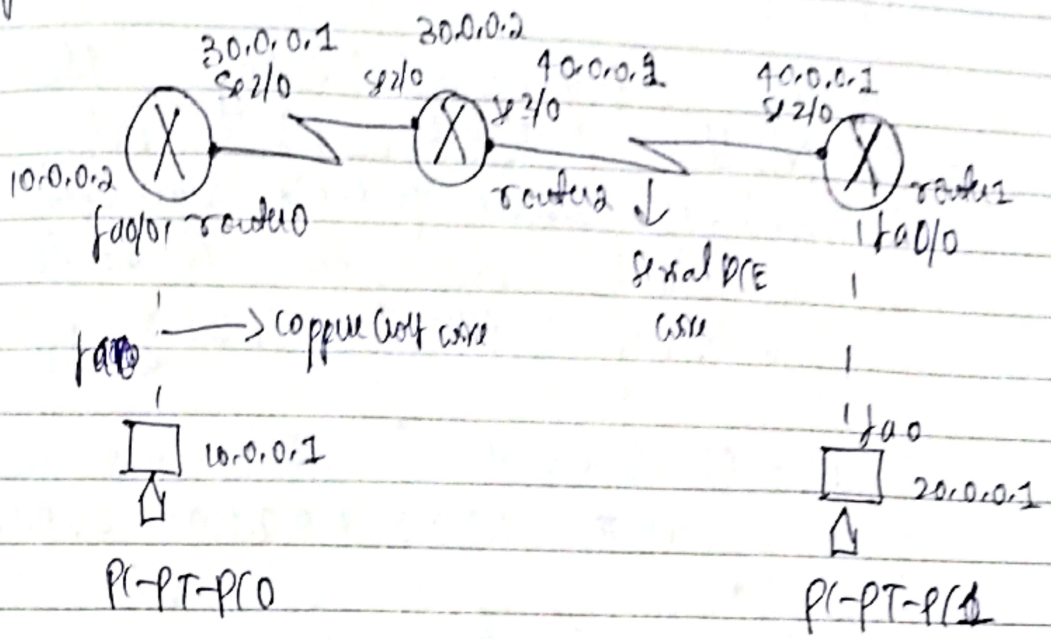
router(config)# ip route 30.0.0.0 255.0.0.0 40.0.0.2
router(config)# ip route 10.0.0.0 255.0.0.0 40.0.0.2

Lab-1

Default router

Bafna Gold
25/10/24

## Default router




Implementation:

→ place two generic end device PC-PT-PC0 & PC-PT-PC1 & config their with ip address of 10.0.0.1 & 20.0.0.1 respectively.

→ Then place three router, router PT0, router PT-1 & router PT-2 & connect router 0 & router 1 with PC-PT-PC0 & PC-PT-PC1 respectively with copper cross over wire.

→ Configure router0 with ip address 10.0.0.2 & router 1 with ip address 20.0.0.2 respectively.

→ connect router 0 & router 2 with Serial DCE & Similarly connect router 1 & router 2 with Serial DCE Se2/0 and Se3/0 ports.

→ Now, Configure the Serial port with 30.0.0.1 & 30.0.0.2 for router 0 & router2, 40.0.0.1 & 40.0.0.2 for router 1 & router 2 respectively.

→ Add gateway 10.0.0.2 & 20.0.0.2 for PC-0 & PC-1 respectively.

Observation:

* After Successful Connection/ping/Simulating package

from pc-0 to pc-1.

* The preference between static & default route is the default route is made between N-device and the pc's (end devices) and static route is made between the N- devices.

Configuration of routers in CLI

router 0: default route

ip route 0.0.0.0 0.0.0.0 30.0.0.2

router 1: default route

ip route 0.0.0.0 0.0.0.0 40.0.0.2

router 2: static route

ip route 10.0.0.0 255.0.0.0 30.0.0.1
ip route 20.0.0.0 255.0.0.0 40.0.0.1

Sending message

PC> ping 20.0.0.1

pinging 20.0.0.1 with 32 bytes of data:
reply from 20.0.0.1: bytes = 32 time = 1ms TTL = 125
reply from 20.0.0.1: bytes = 32 time = 1ms TTL = 125
reply from 20.0.0.1: bytes = 32 time = 1ms TTL = 125
reply from 20.0.0.1: bytes = 32 time = 1ms TTL = 125
ping statistics for 20.0.0.1
packets: sent = 4, received = 4, lost = 0 (0% Lost),
Approximate round trip times in milli-seconds:
minimum = 1ms, maximum = 11ms, average = 1ms

# To verify the default connection the output should be as follows:-

Gateway of last resort is 30.0.0.2 to network 0.0.0.0

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

C 30.0.0.0/8 is directly connected, Serial 2/0

S* 0.0.0.0/0 [1/0] via 30.0.0.2

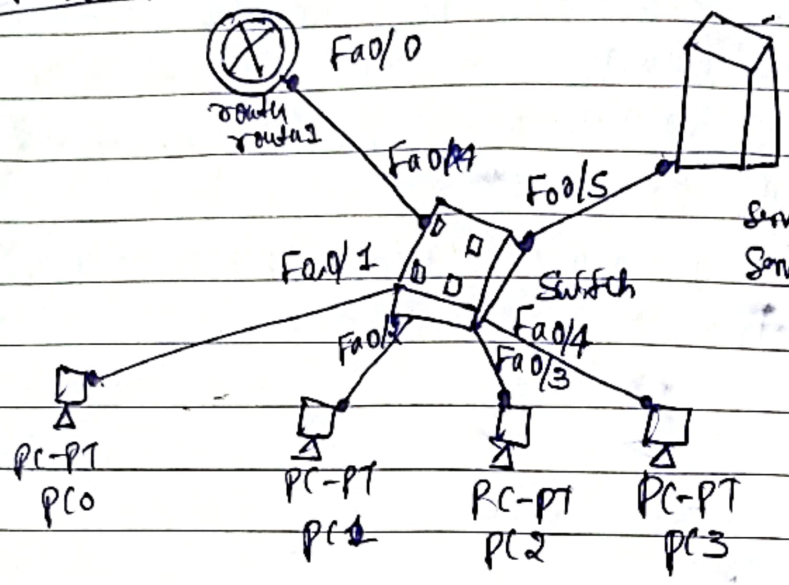
28/10/20

# Lab-5

8/11/24

Q) How to Configure DHCP within a LAN in a packet Trace topology.

Topology:




Implementation:

- Configure router interface with ip 10.0.0.1 and Subnet mask 255.0.0.0
- Click on gateway in Server → config, then assign gateway at 10.0.0.1
- Then click on FastEthernet and assign ip address and Subnet mask, 10.0.0.2 and Subnet mask 255.0.0.0 for server respectively.
- Click on DHCP
  just give default gateway at 10.0.0.1
  DNS Server, just give our server ip address i.e., 10.0.0.2
- Then, just edit start ip address, I am going to give 10.0.0.10 and Subnet mask of 255.0.0.0.
- Maximum no. of user, just give how many ip address you want in this pool. I am going to give it as 500.

- Assign TFTP Server ip address, just give our Server ip address 10.0.0.2
- and click Save
- Now, click on any of the PC → Then Click on Desktop → IP configuration and choose DHCP wait for some time, if your DHCP request failed then try few more times. This is how you should get.

Result:

PC0

Physical | Config | Desktop

IP Configuration

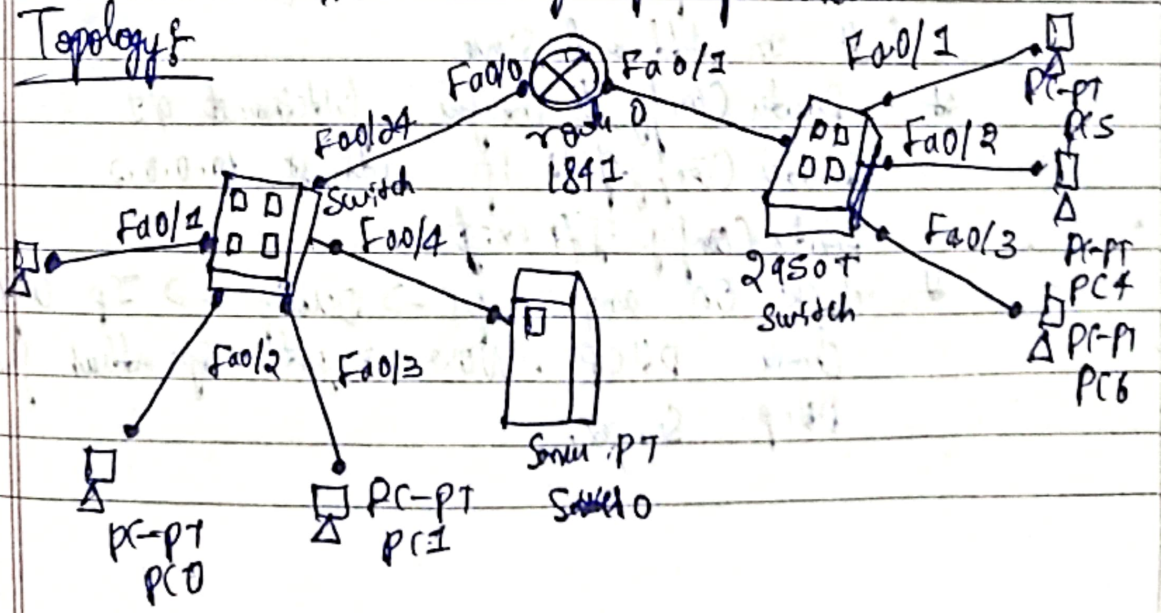
(•) DHCP — DHCP request successful

( ) Static

| | |
|---|---|
| IP Address | 10.0.0.14 |
| Subnet Mask | 255.0.0.0 |
| Default Gateway | 10.0.0.1 |
| DNS Server | 10.0.0.2 |

Q) How to get IP from DHCP that is present in some other network using IP Helper Address.

Topology:



Implementation:

- configure router interface fastethernet 0/0 and fastethernet 0/1 with ip address.
- click on Server → Config → then just give the gateway ip address i.e., 10.0.0.1
- Then, click on fastethernet align ip address 10.0.0.2 & Subnet mask 255.0.0.0, DHCP will automatically align to network for default pool. we don't have to create pool for 10 network again, just we need to give ip for DNS, gateway and TFTP, then we may configure starting ip address or leave it & save.
- Now, click on pc in a LAN with Server and check whether DHCP working fine in that network, click on any pc → Desktop → Ip configuration → choose DHCP, then you will get ip from DHCP Server for that pc.
- Default gateway → 10.0.0.1
  DNS Server → 10.0.0.2
  Start Ip address → 10.0.0.10
  Subnet Mask → 255.0.0.0
  Maximum number of users → 100
  TFTP Server 10.0.0.2
  Click on Add and Save
- Router (config)# interface fastethernet 0/1
  Router (config-if)# ip address 10.0.0.2
  Router (config-if)# exit
- click on any pc → Desktop → Ip Configuration → choose DHCP. Now it got ip address from DHCP Server.

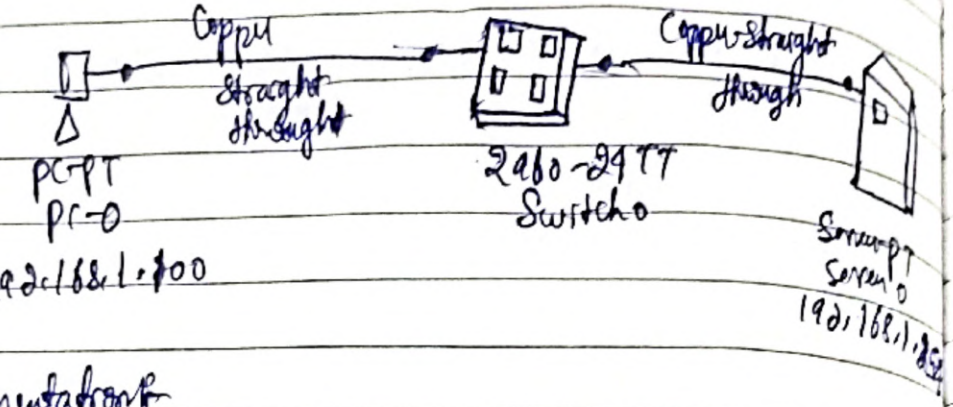
~~Add & Save~~
~~For help ipaddress for Router.~~
Now, it is automatically assigned by DHCP Server.

# LAB-5

08/11/24

Q) How to demonstrate WEB Server and DNS using packet tracer

Topology:




Implementation:

- Click on end devices and then click on a generic PC and place it in the logical workspace, click on switches & place it, click on generic Server and place it in the logical workspace.
- Assign Ip Address to PC - 192.168.1.200
  Server - 192.168.1.254
- Click on Interface → fast ethernet on the left hand side, make sure that the port status is on.
- To set up PC, double click on it, Go to the Config tab, Go to fastethernet on the left hand side.
  Make sure the port is turned on under static ip address and enter 192.168.1.100.
- We will now ping from the pc to the server to make sure that we have a connection.

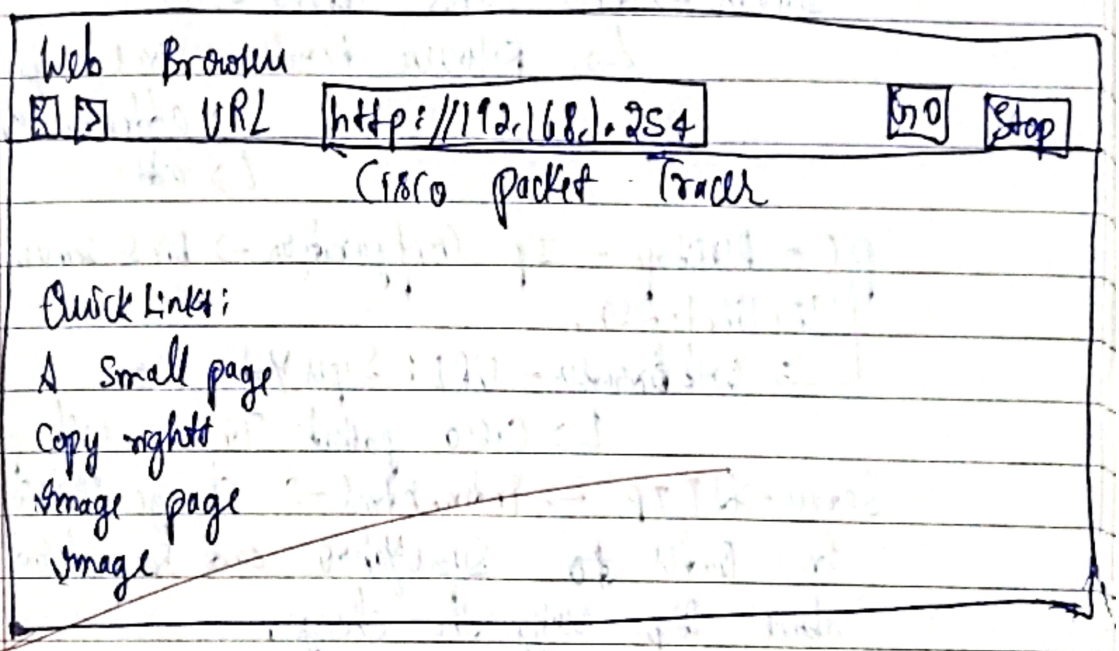
Observation:

PC> ping 192.168.1.254
pinging 192.168.1.254 with 32 bytes of data

Reply from 192.168.1.254 : bytes=32 time=1ms TTL=128
Reply from 192.168.1.254 : bytes=32 time=1ms TTL=128
Reply from 192.168.1.254 : bytes=32 time=1ms TTL=128
Reply from 192.168.1.254 : bytes=32 time=1ms TTL=128

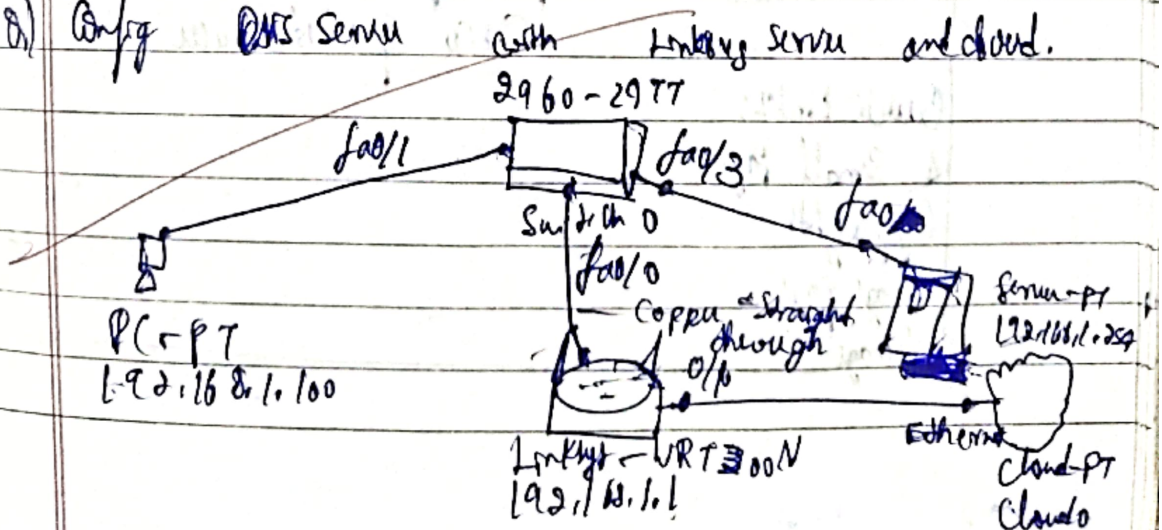
* Double Click on the Server, go to the config tab.

* In the virtual browser that appears type in the ip address of the Server and click go.
* You should now be able to see the Server that we saw earlier.




* You are now connected to the FTP Server.

8) Config DNS Server with Linksys server and cloud.



→ Create a LAN using a generic PC, Server, 2960-24TT Switch, WRT300N Wireless device and cloud, using Copper Straight through.

Config PC → 192.168.1.100 → fastethernet 0/1
Server → 192.168.1.254 → fastethernet 0/3
WRT300N → 192.168.1.1 → fastethernet 0/2

→ Switch to WRT300N → Ethernet 1
WRT300N to cloud → Ethernet 2

→ Server → Gateway → 192.168.1.1
Webserver → http → Cisco packet Tracer → html code
PC → Gateway → 192.168.1.1
Server → dns → dns Service → ON
↳ Resource Records → Name - SuperYahoo.com
→ address - 192.168.1.254
↳ add

PC → Desktop → Ip Configuration → DNS Server → 192.168.1.254.
↳ Web Browser - URL : SuperYahoo.com
↳ Cisco packet Tracer web page display

Server → HTTP → index.html → Change "Cisco Packet Tracer" in order to SuperYahoo and Save. Now using above step verify the change.

Output:

Web Browser
URL http://SuperYahoo.com Go Stop
Cisco packet Tracer
Quick Links:
A Small Page
Copyright
Image page
Image

# Lab-6

Bafna Gold

15/11/24

Q) Write a Program for Error Detection using CRC-CCITT (16 bit).

```
def crc(input_msg, poly, mode):
    input_len = len(input_msg)
    poly_len = len(poly)
    padded_input = list(input_msg)

    if mode == 1:
        padded_input += ['0'] * (poly_len - 1)

    padded_input = list(padded_input)

    for i in range(input_len):
        if padded_input[i] == '1':
            for j in range(poly_len):
                padded_input[i+j] = '0' if
                    padded_input[i+j] ==
                        poly[j] else '1'

    output = list(input_msg)
    output += padded_input[input_len:]
    return ''.join(output)

def check_crc(received, poly):
    received_len = len(received)
    poly_len = len(poly)
    received = list(received)
    for i in range(received_len - poly_len + 1):
        if received[i] == '1':
            for j in range(poly_len):
                received[i+j] = '0'
```

```
            if received[i+j] == poly[j]:
                received[i+j] = '0'
        for i in range(received_len - poly_len + 1, received_len):
            if received[i] == '1':
                return False
    return True

def main():
    poly = "1000100000010000"
    ip = input("Enter the input message in binary: ")
    transmitted_message = CRC(ip, poly, 1)
    print("The transmitted message is: ", transmitted_message)
    recv = input("Enter the received message in binary: ")
    if checkCRC(recv, poly):
        print("No error in data")
    else:
        print("Error in data transmission has occurred").

if __name__ == "__main__":
    main()
```

Output: Successful call

Enter the input message in binary : 11111
The transmitted message is: 1111110001111010000
Enter the received message in binary: 1111110001111010000
No error in data.

Output 2: failure/error case

Enter the input message in binary: 11111

The transmitted message is: 1111111101011110111110

Error in data transmission has occurred.

# Lab-6

15/11/24

a) Leaky Bucket Problem:

```
def main()

    Storage = 0
    no_of_queues = 4
    bucket_size = 10
    input_pkt_size = 4
    output_pkt_size = 1

    for i in range(no_of_queues)
        size_left = bucket_size - Storage
        if input_pkt_size <= size_left:
            Storage += input_pkt_size
        else:
            print("Packet loss = {input_pkt_size}")

    print("Buffer size = {Storage} out of bucket size = {bucket_size}")

    Storage -= output_pkt_size

if __name__ == "__main__":
    main()
```

15/11/24

Output:

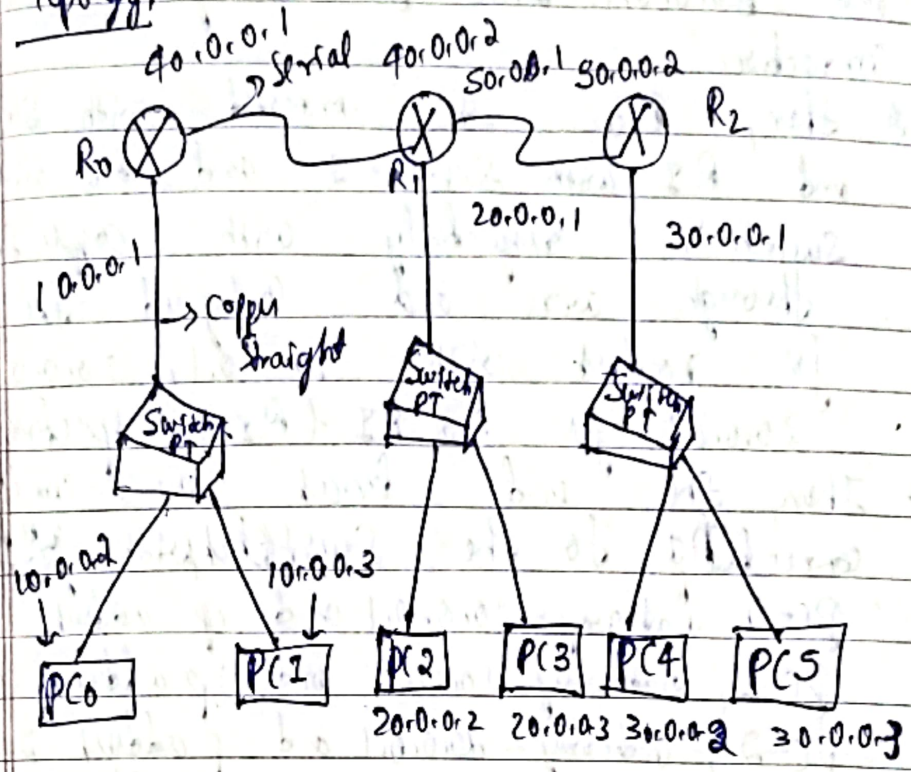
Buffer Size = 4 out of bucket size = 10
Buffer size = 7 out of bucket size = 10
Buffer size = 10 out of bucket size = 10
Packet loss = 4
Buffer_Size = 9 out of bucket size = 10

Lab-7

22/11/24

# Routing Information Protocol (RIP).

Topology:




Implementation:

* Select 6 Generic end devices which were, PC-0, to PC-5 respectively.
* Select the Generic three routers and three Generic Switches.
* Configured the end devices using the topology.
* Connected the two end devices for three Switches each respectively by Copper Straight through wire.
* Connected the three Generic Switches to the three Generic routers using the Copper Straight through wire.
* The routers are R0, R1 and R2 respectively and Switches are Switch-PT-0, Switch-PT-1, Switch-PT-2 respectively.

* Connected the routers R0 and R1 and R1 and R2 with Serial DCE wire and configured with 40.0.0.1 and 40.0.0.2 for R0 and R1 and 50.0.0.1 and 50.0.0.2 for R1 and R2 connection
* Then, R0 is connected with Switch-0 and R1 with Switch-1 and R2 with Switch-2 respectively with copper straight through wire and configured the connection for routers with 10.0.0.1, 20.0.0.1 and 30.0.0.1 for R0, R1 & R2 respectively.
* Then, the end devices are configured by connecting to the switches with IP address,
PC-0, Gateway - 10.0.0.1 and ip address 10.0.0.2
PC-1, Gateway - 10.0.0.1 and ip address 10.0.0.3
PC-2, Gateway - 20.0.0.1 and ip address 20.0.0.2
PC-3, Gateway - 20.0.0.1 and ip address 20.0.0.3
PC-4, Gateway - 30.0.0.1 and ip address 30.0.0.2
PC-5, Gateway - 30.0.0.1 and ip address 30.0.0.3 respectively.
* So, after connecting all the end devices switches and routers and configuring them with the ip addresses.

* Now Click on Router 0
(2) Go to CLI
(3) Type Enable → Router Enable
(4) Type Config terminal
(5) # router rip
(6) (for R0) → # network 10.0.0.0
# network 40.0.0.0
# exit

Router# Show ip route.

repeat this #route rip for, all routers R1 & R2.

③ For R2 → #network 20.0.0.0
#network 40.0.0.0
#network 50.0.0.0
#exit

Router# Show ip route
#route rip.

④ For R2 → #network 30.0.0.0
#network 50.0.0.0
#exit

Router# Show ip route.

⑤ For R2 Show ip route, (router 2).

R 10.0.0.0/8 [120/2] via 50.0.0.1, 00:00:25, Serial 2/0
R 20.0.0.0/8 [120/1] via 50.0.0.1, 00:00:25 Serial 2/0
C 30.0.0.0/8 is directly connected, FastEthernet 0/0
R 40.0.0.0/8 [120/1] via 50.0.0.1, 00:00:25, Serial 2/0
C 50.0.0.0/8 is directly connected, Serial 2/0

like this Router 0 and Router 1, save Five connections.

⑥ Send Simple PDU (message) from pc-0 to pc-4 and observe the observations.

Observations

* After sending the packets from pc-0 to pc-4, the packet is received at pc-4 and the Acknowledgement is sent to the pc-0.
* The status of the message is successful using this procedure.
* We can do, this procedure using ping also.

* Click on PC0
* Go to Desktop
* Click on Command prompt and type ping commands.
* PC> ping 30.0.0.2

pinging 30.0.0.2 with 32 bytes of data:

Request timed out
Reply from 30.0.0.2 : bytes=32 time=2ms TTL=125
Reply from 30.0.0.2 : bytes=32 time=5ms TTL=125
Reply from 30.0.0.2 : bytes=32 time=12ms TTL=125
Ping statistics for 30.0.0.2:
Packets: Sent=4, received=3, Lost=1(25% Loss),
Approximate round trip times in milli-seconds:
Minimum = 2ms, Maximum = 12ms, Average = 6ms

(A) For R0 show ip route (route 0)

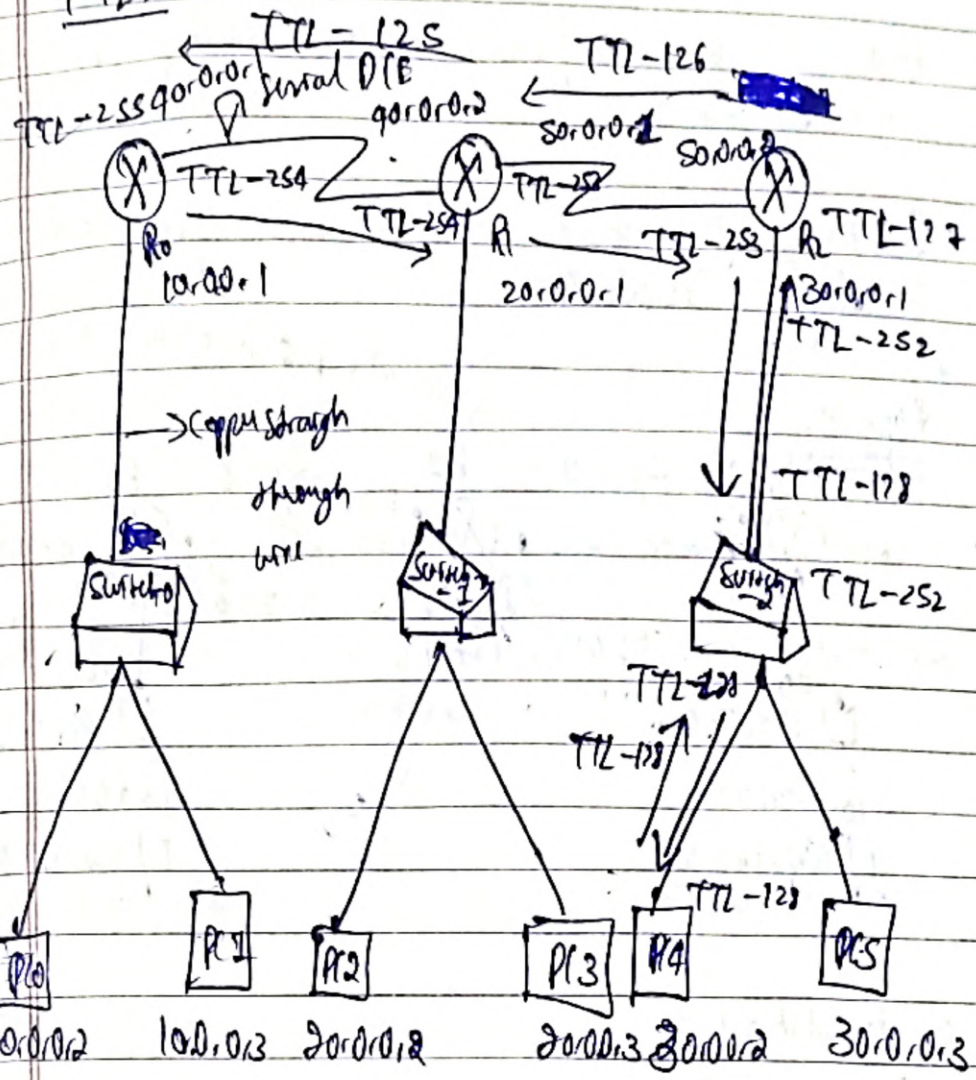
C 10.0.0.0/8 is directly connected FastEthernet 0/0
R 20.0.0.0/8 [120/1] via 40.0.0.2, 00:00:19, Serial 2/0
R 30.0.0.0/8 [120/2] via 40.0.0.2, 00:00:19, Serial 2/0
C 40.0.0.0/8 is directly connected, Serial 2/0
R 50.0.0.0/8 [120/1] via 40.0.0.2, 00:00:19, Serial 2/0

(A) For R1 show ip route (route 1)

R 10.0.0.0/8 [120/1] via 40.0.0.1, 00:00:19, Serial 2/0
C 20.0.0.0/8 is directly connected, FastEthernet 0/0
R 30.0.0.0/8 [120/1] via 50.0.0.2, 00:00:22, Serial 3/0
C 40.0.0.0/8 is directly connected, Serial 2/0
C 50.0.0.0/8 is directly connected, Serial 3/0

Bafna Gold
Date Page
22/11/24

TTL:




Checking TTL:

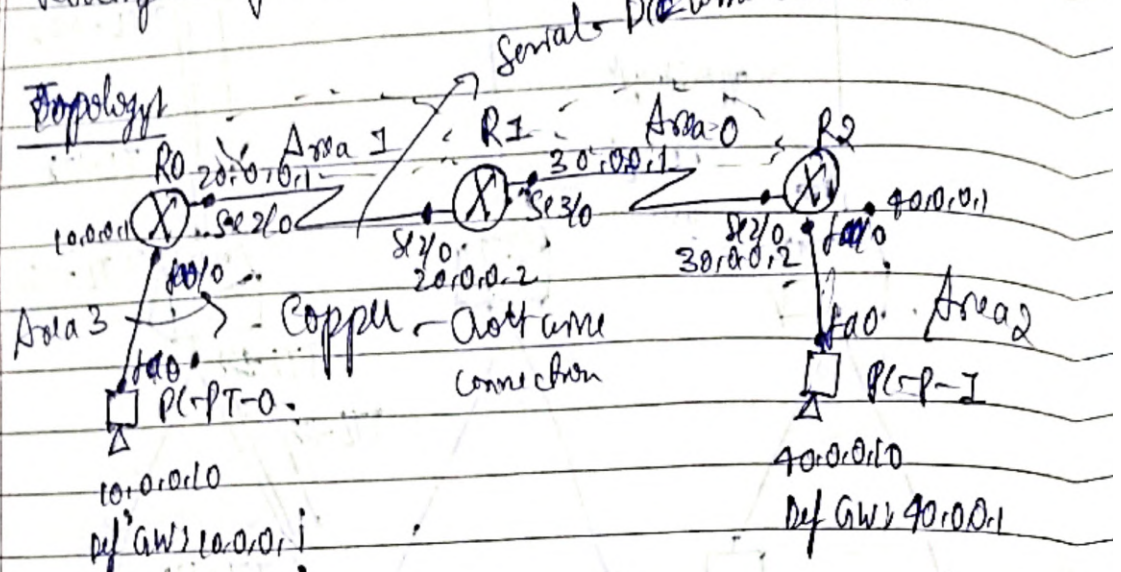
→ For the same topology as above.
→ Go to Simulation mode.
→ Click on PDU and send from 10.0.0.2 to 30.0.0.2
→ Auto / Capture/play it until it succeeds.
→ Click on the first PDU and observe the TTL.
→ It started at 255 before encountering a router
→ When it reaches a router incoming is 255 outgoing will be 254
→ The same way for the other two routers and reaches the destination it is 252 to 128.4

Lab-8 29/11/24

How to configure OSPF Routing Protocol and Connect Areas

OSPF - Open Shortest Path First, It is a Open Standard Routing Protocol.

Topology:




Implementation:

* Select 3 Generic routers and 2 Generic PC's which are respectively R0, R1, R2 and PC-0 and PC-1.
* Making Connection between Router R0 and PC-0 and Router R2 and PC-1 with Copper - Cross connection.
* Connect Router R0, Router R1 with Serial DCE Connection and Router R1 and Router R2 with Same Serial - DCE cable connection.
* Configure the PC-0 and PC-1 and Laptop with IP address 10.0.0.10 and Def. Gateway at 10.0.0.1 for PC-0 and IP address at 40.0.0.10 and Default gateway at 40.0.0.1 for PC-1

* Configuring ip address for all interface for routers.

In Router R1,

```
R1(config)# interface fastethernet 0/0
R1(config)# ip address 10.0.0.1 255.0.0.0
R1(config)# no shutdown
R1(config)# exit

R1(config)# interface serial 1/0
R1(config)# ip address 20.0.0.1 255.0.0.0
R1(config)# encapsulation ppp
R1(config)# clock rate 64000
R1(config)# no shutdown
R1(config)# exit
```

In Router R2,

```
R2(config)# interface serial 1/0
R2(config)# ip address 20.0.0.2 255.0.0.0
R2(config)# encapsulation ppp
R2(config)# no shutdown
R2(config)# exit

R2(config)# interface serial 1/1
R2(config)# ip address 30.0.0.1 255.0.0.0
R2(config)# encapsulation ppp
R2(config)# clock rate 64000
R2(config)# no shutdown
R2(config)# exit
```

In Router R3,

```
R3(config)# interface serial 1/0
R3(config)# ip address 30.0.0.2 255.0.0.0
```

R3(config-if)# encapsulation PPP
R3(config-if)# no shutdown
R3(config-if)# exit

R3(config)# interface fastethernet 2/0
R3(config-if)# ip address 40.0.0.1 255.0.0.0
R3(config-if)# no shutdown
R3(config-if)# exit

Step 3: Now, Enable ip routing by configuring ospf routing protocol in all routers.

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
R2(config)# 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
R2(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 10.0.0.0 0.255.255.255 area 2
R3(config-router)# exit

You have to configure router-id when we configure OSPF. It is used to identify the router.

Step 4: Now check routing table of R1.

Router# show ip route
Codes: C - connected, S - static, 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
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 Lo 0.0.0/8 is directly connected, FastEthernet 2/0
C 20.0.0/8 is directly connected, Serial 1/0
O IA 40.0.0.0/8 [110/129] via 20.0.0.2, 00:04:23, Serial 1/0
O IA 30.0.0.0/8 [110/128] via 20.0.0.2, 00:07:29, Serial 1/0

Here, R2 knows Area 0 network 20.0.0.0 connected to R2 from R1. So, R1 learns networks through this network.

R3(config)# router OSPF 1, Here, 1 is process ID, it can be

1 - 65535, Id should be ospf process.

There must be one interface up to top ospf process up. So we have to configure loopback address to router. Id as a virtual interface never goes down once we configured.

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

R2(config)# interface loopback 0
R2(config-if)# ip add 172.16.1.253 255.255.0.0
R2(config-if)# no shutdown

R3(config)# interface loopback 0
R3(config-if)# ip add 172.16.1.254 255.255.0.0
R3(config-if)# no shutdown

Step 5: Now, checking routing table of R3

R3# show ip route

Gateway of last resort is not set

O IA 20.0.0.0/8 [110/138] via 30.0.0.1, 00:18:58, Se0/0
C 40.0.0.0/8 is directly connected, FastEthernet0/0
C 30.0.0.0/8 is directly connected, Serial 1/0.

Here, R3 doesn't know about the area 3
So we have to Create virtual link
between R1 and R2.

Step 6: Create virtual link between R1, R2, by this we create a virtual link to connect area 3 to area 0.

On router R1,

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

from loading to full, Loading Done.

On router R2,

received invalid packet: mismatch area 10, from backbone area must be virtual-link but not found from 20.0.0.1, Serial 1/0a,

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

from Loading to full, Loading Done.

Step 7: R2 and R3 get updated about Area 3.
Now, check routing table of R3.

```
R3#show ip route
```

Gateway of last resort is not set.

```
O IA    20.0.0.0/8 [110/128] via 30.0.0.1, 00:01:56, Serial 1/0
C       40.0.0.0/8 is directly connected, FastEthernet 0/0
O IA    10.0.0.0/8 [110/128] via 30.0.0.1, 00:01:56
C       30.0.0.0/8 is directly connected, Serial 1/04
```

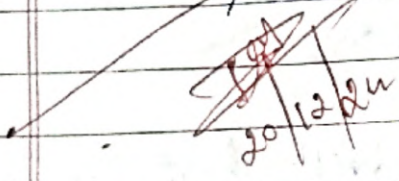
Step 8: check connectivity between host 10.0.0.10 to 40.0.0.10

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=128
Reply from 40.0.0.10; bytes=32 time=8ms TTL=128
Reply from 40.0.0.10; bytes=32 time=6ms TTL=128
Reply from 40.0.0.10; bytes=32 time=8ms TTL=128

ping statistics for 40.0.0.10:
Approximate packets: Sent=4, Received=4, Lost=0 (0% Loss),
round trip times in milli-seconds:
Minimum=6ms, Maximum=9ms, Average=7ms.



# Lab-9

20/12/24

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

ClientTCP.py

```
from socket import*
ServerName = '127.0.0.1'
Serverport = 12000
ClientSocket = socket(AF_INET, SOCK_STREAM)
ClientSocket.connect((ServerName, ServerPort))
Sentence = input("\n Enter file name:")

ClientSocket.send(Sentence.encode())
fileContents = ClientSocket.recv(1024).decode()
print("\n From Server:\n")
print(fileContents)
ClientSocket.close()
```

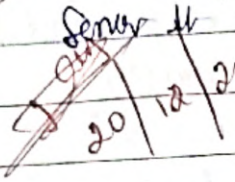
ServerTCP.py

```
from socket import*
ServerName = "127.0.0.1"
Serverport = 12000
ServerSocket = socket(AF_INET, SOCK_STREAM)
ServerSocket.bind(ServerName, ServerPort))
ServerSocket.listen()
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("In Send contents of It Sentence")
file.close()
connectionSocket.close()
```

Output

The server is ready to receive
Send contents of ServerTCP.py
The server is ready to receive



# Lab-9

Bafna Gold
Date: Page:
20/12/24

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

## Client UDP.Py

```
from socket import *
ServerName = "127.0.0.1"
ServerPort = 12000
ClientSocket = socket(AF_INET, SOCK_DGRAM)
Sentence = input("\n Enter file name: ")
ClientSocket.sendto(bytes(Sentence, "utf-8"), (ServerName,
                    ServerPort))
fileContents, serverAddress = ClientSocket.recvfrom(2048)
print("\n Reply 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)
ServeSocket.sendto(bytes(con,"utf-8"),
        clientAddress)
print("\n Sent Contents of", end=" ")
print(Sentence)
# for i in Sentence:
    # print(str(i), end="")
file.close()
```

Output:

The Server is ready to receive

Sent Contents of ServerUDP.py

The server is ready to receive.

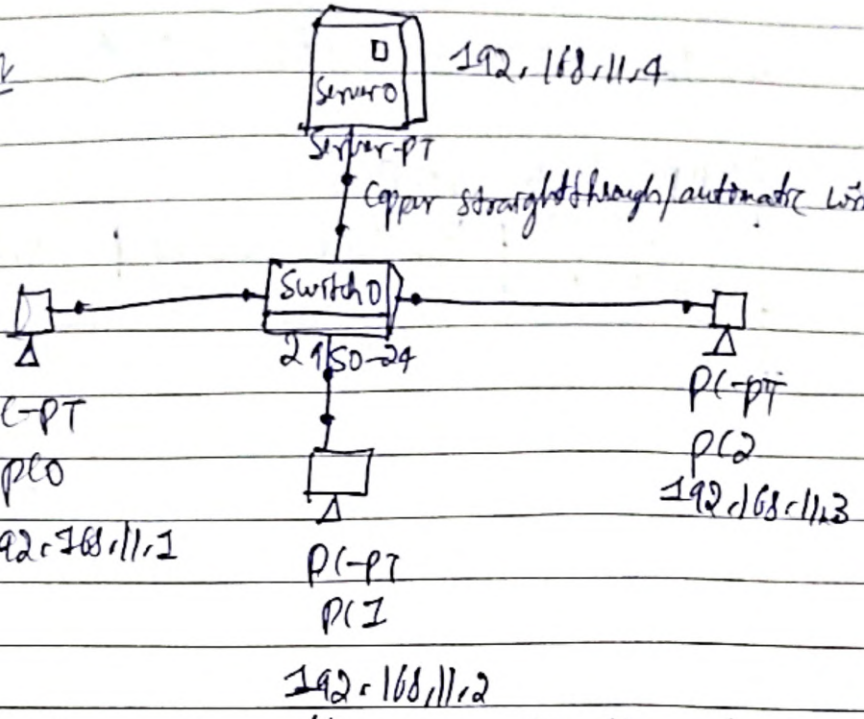
20/12/22

# Lab-9

Bafna Gold
Date Page
20/11/24

③ To construct simple LAN and understand the concept and operation of Address Resolution protocol.

Topology




Step1: Assign ip address to all PCs and Server.

Step2: Go to simulation panel, click on Inspect and right click on PC0.

Step3: Notice that there are no entries in the ARP table

Step4: Repeat the same for Server

Step5: Click on PC0 and go to command prompt
Type arp -a
(Initially there are no ARP entries

Step6: Try pinging from PC0 to Server
PC> ping 192.168.11.4
Pinging 192.168.11.4 with 32 bytes of data.

Step 7: Two packets are created ICMP and Step.

Step 8: Hover over the packets to check the type packets.

Step 9: Click on ARP packet

| ARP table for PC0 | | |
|---|---|---|
| Ipaddress | Hardware Address | Interface |
| — | — | — |

Step 10: Click on capture button to start the simulation ping request from PC0 to Switch will happen.
Again click on capture
Click on capture twice to see the ping acceptance from Server0 to PC0

Step 11: Click on capture to see the ICMP packet movements.

PC> arp -a
No ARP entries found

PC> ping 192.168.11.4

Pinging 192.168.11.4 with 32 bytes of data:

Reply from 192.168.11.4: bytes=32 time=1ms TTL=128
Reply from 192.168.11.4: bytes=32 time=0ms TTL=128
Reply from 192.168.11.4: bytes=32 time=0ms TTL=128
Reply from 192.168.11.4: bytes=32 time=0ms TTL=128

Ping statistics for 192.168.11.4:
    Packets: Sent=4, Received=4, Lost=0 (0% Loss),
Approximate round trip times in milli-seconds:
    Minimum=0ms, Maximum=1ms, Average=0ms.

After pinging ARP table for PC0

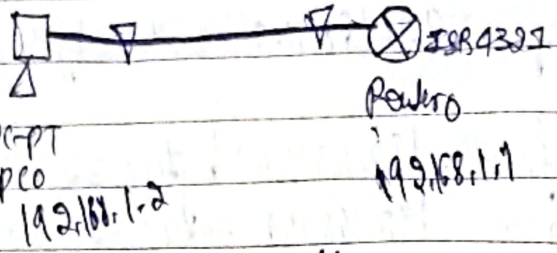
ARP table for PC0

| IP address | Hardware address | Interface |
|---|---|---|
| 192.168.11.4 | 0001.96AE.A042 | FastEthernet0 |

# Lab-9

## TELNET Protocol

Step 1: Create the topology.




Step 2: Configure the IP address and gateway for PC0

- Configuring PC0 IP address as 192.168.1.2 and Subnet Mask as 255.255.255.0 and default gateway as 192.168.1.1.
- Connecting PC0 and router0 by
- establishing connection between them.

Step 3: Configure the router in CLI.

Configuring the router0,

```
router# conf t
router(Config)# hostname R1
(Config)# enable secret cp
(R1(config)# int fa0/0
R1(Config-if)# ip add 192.168.1.1 255.255.255.0
R1(config-if)# no shut
R1(config-if)# line vty 0 5
R1(config-line)# login
% login disabled on line 132, until 'password' is set.
% login disabled on line 133, until 'password' is set.
% login disabled on line 134, until 'password' is set.
```

% login disabled on line 135, until 'password' is set
% login disabled on line 136, until 'password' is set
% login disabled on line 137, until 'password' is set.
R1(config-line)# password dp
R1(config-line)# exit
R1(config)# exit
R1#
% SYS-5-CONFIG_I: Configured from console by console.
R1# wr
Building configuration...
[OK]
R1#
R1#

Step 4: Go to PC-0 command prompt and ping the 192.168.1.1 (router) and telnet and enter the password.

PC> ping 192.168.1.1

pinging 192.168.1.1 with 32 bytes of data:

Reply from 192.168.1.1: bytes=32 time=0ms TTL=255
Reply from 192.168.1.1: bytes=32 time=0ms TTL=255
Reply from 192.168.1.1: bytes=32 time=0ms TTL=255
Reply from 192.168.1.1: bytes=32 time=0ms TTL=255

20/12/24

ping Statistics for 192.168.1.1:
Packets: Sent = 4, Received = 4, Lost = 0 (0% Loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms.

Telnet:

PC> telnet 192.168.1.1
Trying 192.168.1.1 ---- Open

User Acct verification

Password:
R1>en
Password:
R1#
R1#

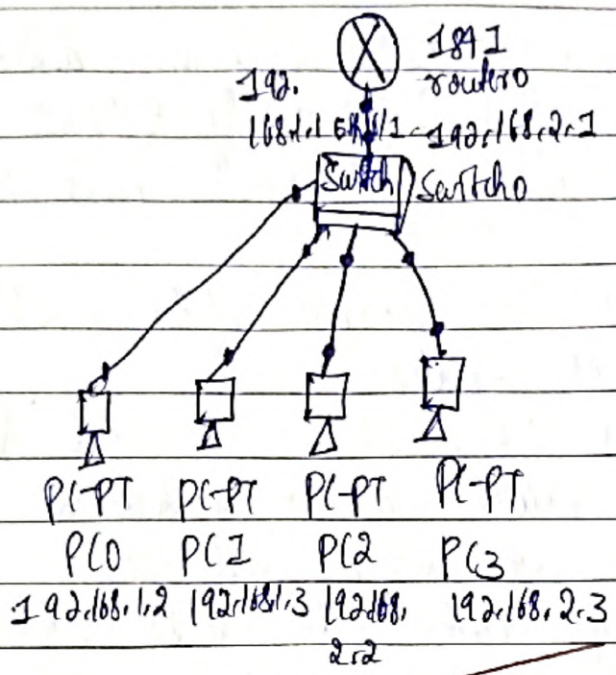
Bafna Gold
Date Page
20/12/24

# Lab-9

## Virtual LAN

To create a new VLAN, we use class C type address.

Topology:
Choose the 1841 router




Implementation:

- Select one 1841 router and Switch-PT-Generic and 4 - end devices (PC's).
- Config the PC0, PC1, PC2 & PC3 with 192.168.1.2, 192.168.1.3, 192.168.2.2 & 192.168.2.3 respectively.
- In the switch, go to physical and change the port to add the router using Ethernet port, and on the switch
- Here, connect the switch and router and switch and end devices with automatic wire connection.
- Gateway for PC-0 & PC-1 is 192.168.1.1 & PC-2 & PC-3 with 192.168.2.1 respectively.
- Now, in the switch, go to config tab and select VLAN Database.

* In the VLAN Database, Give any VLAN Number and Name and Number Say 2 here, Say Add,
* Select the Interface i.e, 6/1 Over the Switch from router and make it the trunk,
* VLAN trunking allow Switches to forward frames from different VLANs over a single link called trunk,
* This is done by adding an additional header information called tag to the Ethernet frame. The process of adding this small header is called VLAN tagging.
* Look into the interface of the Switches with the 2 NEWVLAN Systems.
* This makes the Switch understand NEWVLAN.
* Next the router it to Understand the NEWVLAN. Do this for fastethernet 2/1 and 3/1.
* Config tab of router Select VLAN Database enter the number and name of the vlan created Goto CLI,

Observation

① router0, GotoCLI,
Router(Vlan)#exit
Apply completed.
Exiting....
Router#config t
Router(config)# interface fastEthernet 0/0.1
Router(config-subif)# encapsulation dot1q 2
Router(config-subif)# ip address 192.168.2.1 255.255.255.0
Router(config-subif)# no shut
Router(config-subif)#exit
Router(config)#exit.

Bafna Gold
Date: Page:

(5) ping message from PC0 to PC2 and check the result, the message is received.

PC> ping 192.168.2.2

pinging 192.168.2.2 with 32 bytes of data:

Reply from 192.168.2.2: bytes=32 time=2ms TTL=127
Reply from 192.168.2.2: bytes=32 time=1ms TTL=127
Reply from 192.168.2.2: bytes=32 time=4ms TTL=127
Reply from 192.168.2.2: bytes=32 time=1ms TTL=127

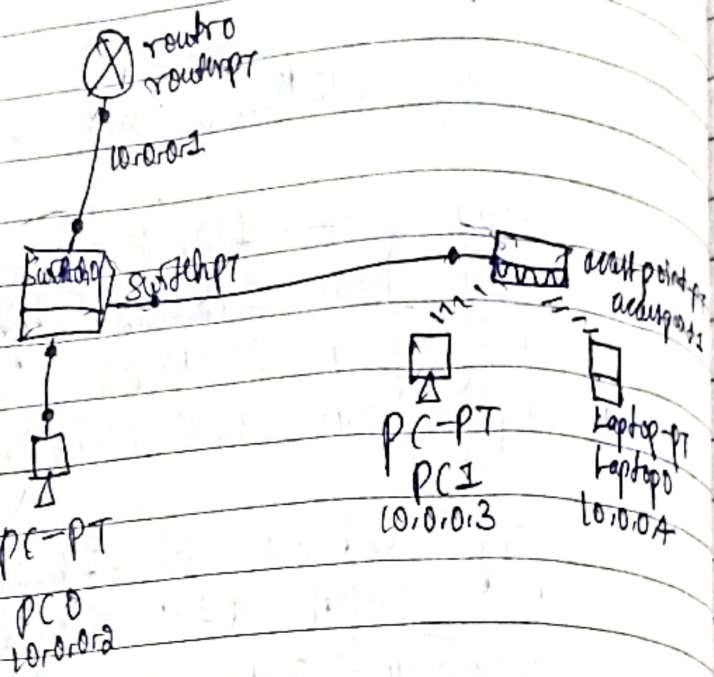
ping Statistics for 192.168.2.2:
Packets: Sent = 4, Received = 4, Lost = 0 (0% Lost),
Approximate round trip times in milli-seconds:
Minimum = 1ms, Maximum = 4ms, Average = 2ms.

20/12/24

# Lab-9

## Wireless LAN

Topology:




Implementation:

- Select generic router and switch and accesspoint-PT, two - 2 PCs and one laptop.
- Configure the PC-0 and router-0 with 10.0.0.2 and 10.0.0.1 respectively. The Gateway for PC-0 is 10.0.0.1.
- Connect Switch and PC-0 and Switch and router with automatic core connection.
- Connect switch and accesspoint-PT with automatic core connection.
- Now, Configure Accesspoint-1 - port-1 → SSID Name - any name (WLAN here) and select WEP and give any 10 digit hex key - 1234567890 here.
- Now, configure the PC-1 with wireless standard. Switch off the device, Drag the existing PT-HOST-NM-1AM to the component listed in the LHS. Drag WMP300N wireless interface to the empty port.

Switch on the device. On the config tab, a new wireless interface would have been added. Now configure SSID, WEP, WEP key, Ip address and Gateway (as normally done) to the device and Ip address as 10.0.0.3.

* repeat the same procedure for Laptop-0 and assign/config Ip address as 10.0.0.4.
* Now, the wireless connection is established from access point 1 to the PC-1 and Laptop-0.

Observations

* ping the message from every device to every other device and the successful ping was done.
* pinging from PC-0 to PC-1.

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=25ms TTL=128
Reply from 10.0.0.3 : bytes=32 time=14ms TTL=128
Reply from 10.0.0.3 : bytes=32 time=6ms TTL=128
Reply from 10.0.0.3 : bytes=32 time=9ms TTL=128

Ping statistics for 10.0.0.3:
Packets: Sent = 4, Received = 4, Lost = 0 (0% Loss).
Approximate round trip times in milli-seconds:
Minimum = 6ms, Maximum = 25ms, Average = 13ms.