KALASALINGAM
ACADEMY OF RESEARCH AND EDUCATION
(DEEMED TO BE UNIVERSITY)
Under sec. 3 of UGC Act 1956. Accredited by NAAC with "A++" Grade
Anand Nagar, Krishnankoil, Srivilliputtur (Via), Virudhunagar (Dt) - 626126, Tamil Nadu | info@kalasalingam.ac.in | www.kalasalingam.ac.in

SCHOOL OF COMPUTING

Department of Computer Science and Engineering

Computer Networks

(212CSE3302)

Student Name : …….

Register Number : ……

Slot / Section : ……..

KALASALINGAM
ACADEMY OF RESEARCH AND EDUCATION
(DEEMED TO BE UNIVERSITY)
Under sec. 3 of UGC Act 1956. Accredited by NAAC with "A++" Grade
Anand Nagar, Krishnankoil, Srivilliputtur (Via), Virudhunagar (Dt) - 626126, Tamil Nadu | info@kalasalingam.ac.in | www.kalasalingam.ac.in

SCHOOL OF COMPUTING

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

Bonafide record of work done by

of **III Year / VI Semester** in **212CSE3302 / Computer Network** during

Even Semester in the Academic Year **2024-2025.**

Staff In-charge

Submitted to the End Semester Practical Examination held at Kalasalingam

Academy of Research and Education, Krishnankoil on ----------

REGISTER NUMBER

INTERNAL EXAMINER **EXTERNAL EXAMINE**

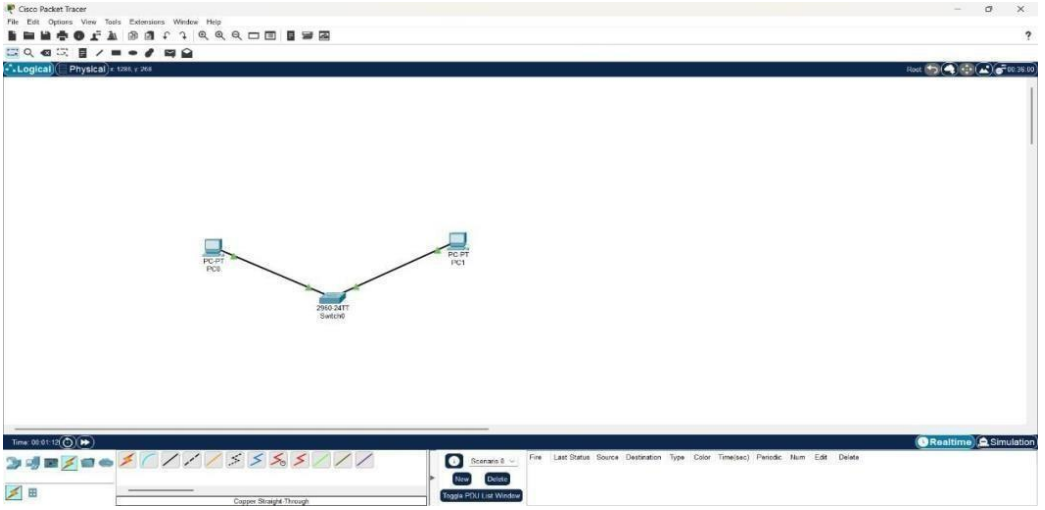
Table of contents

Sl. No	Date	Name of the experiment	Marks	Signature
1		Study of Network Tool-Packet tracer, Wireshark		
2		Study of Network Devices in detail – Hub, Switch, Router		
3		Study of different types of network cables and practically implement the crossover wired and straight through cable using crimping Tool		
4		Topologies- Ring, Star, tree, Mesh, Hybrid		
5		To configure the Intra VLAN using packet tracer		
6		To configure the Inter VLAN using packet tracer		
7		Checking Layer 2 functionality using packet tracer.		
8		Exploring link state and distance vector using routing protocols		
9		DHCP Configuration		
10		Capture and analyse TCP and IP protocols		
11		Analysis of TCP 3-way handshake		
12		Analysis of HTTP protocol		
13		ICMP packet analysis using Wireshark		
14		DNS packet analysis		
15		FTP server configuration		
16		Email server configuration		

Ex.No:	**1a**
Name of the Experi ment	**Study of Network Tool–Packet Tracer**
Date	12-12-2024

1. **Device Requirements:**
    1. Switch
    2. PC0
    3. PC1
    4. Wire

2. **Network Diagram:**



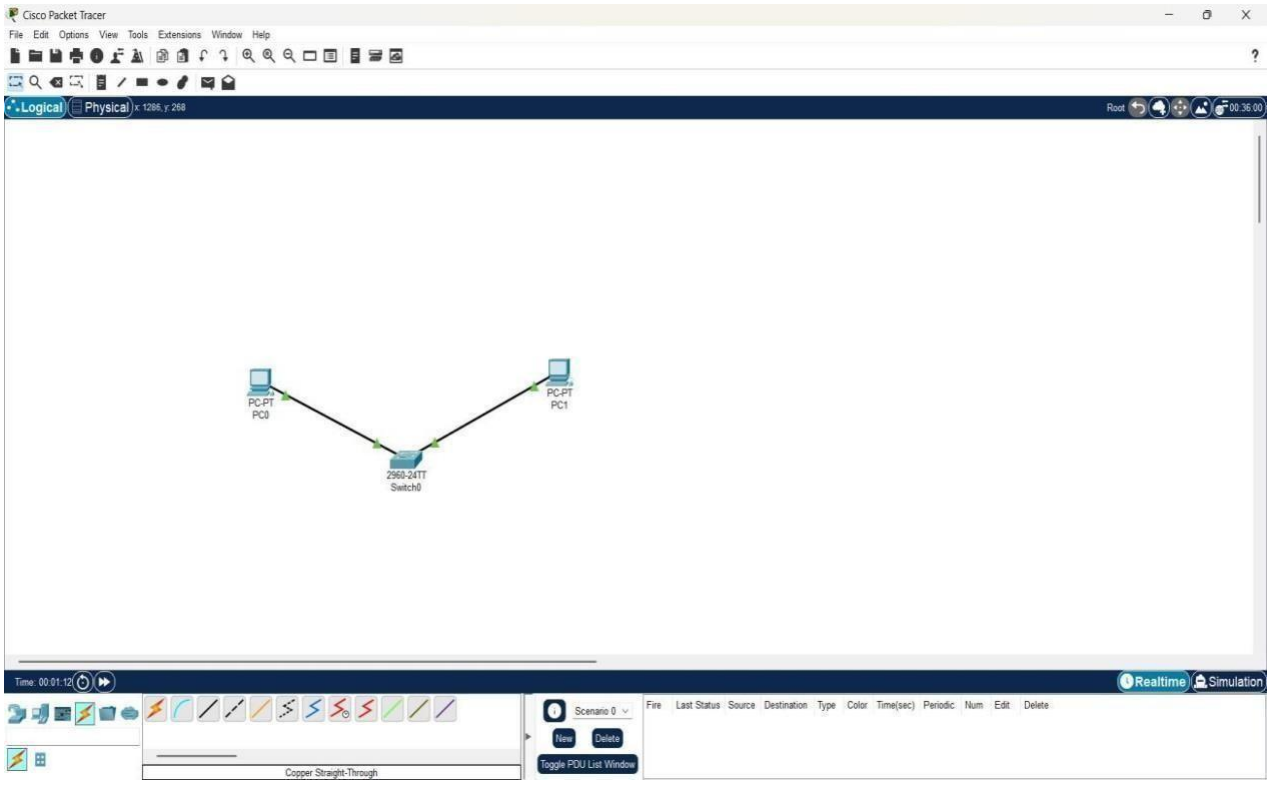
3. **Configuration details:**

DeviceName	InterfaceName	IPAddress	Subnetmask
PC0	Fa0	172.16.108.25	255.255.0.0
PC1	Fa0	172.16.108.26	255.255.0.0
Switch	Fa0		

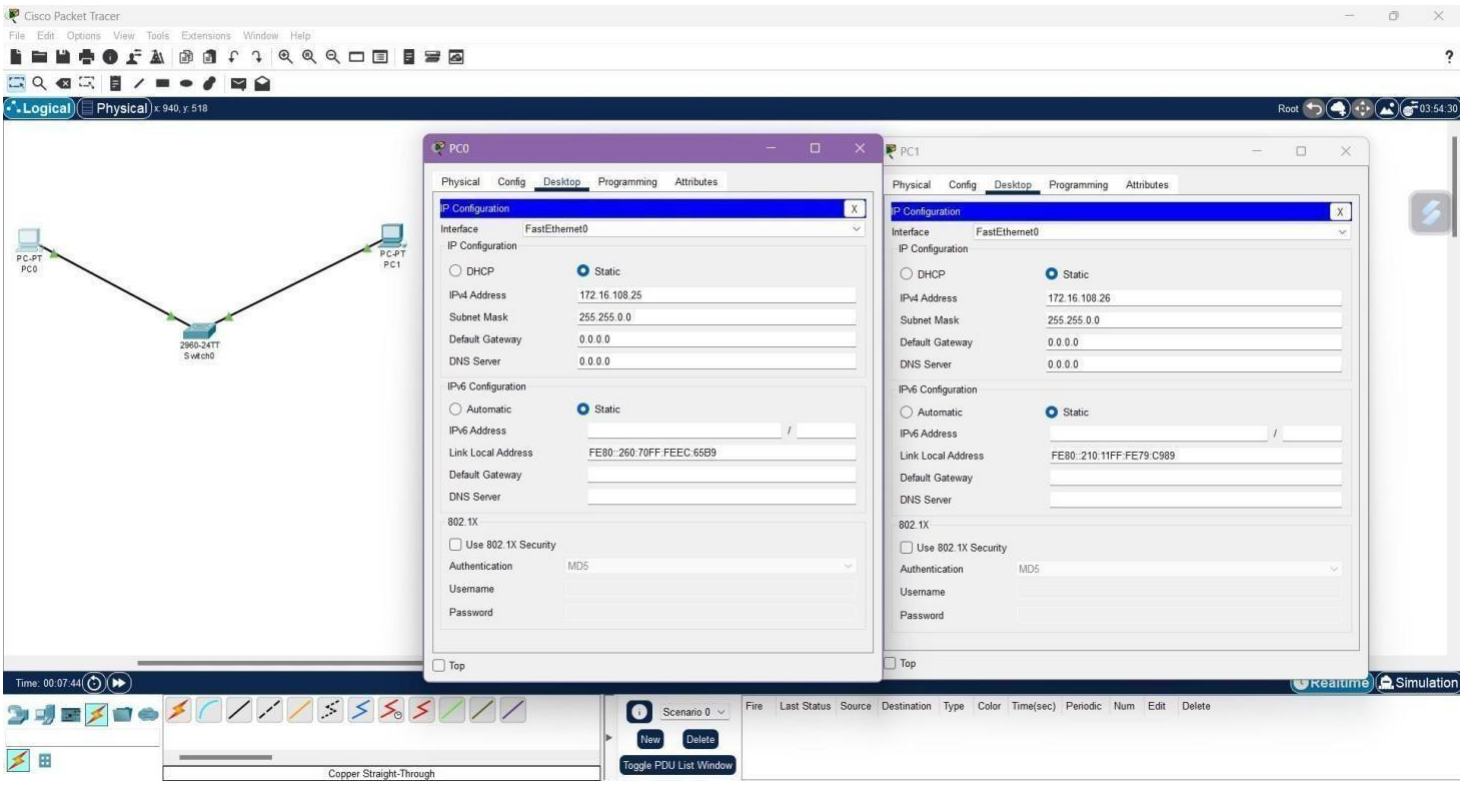
4. Commands used in each of the diagram:

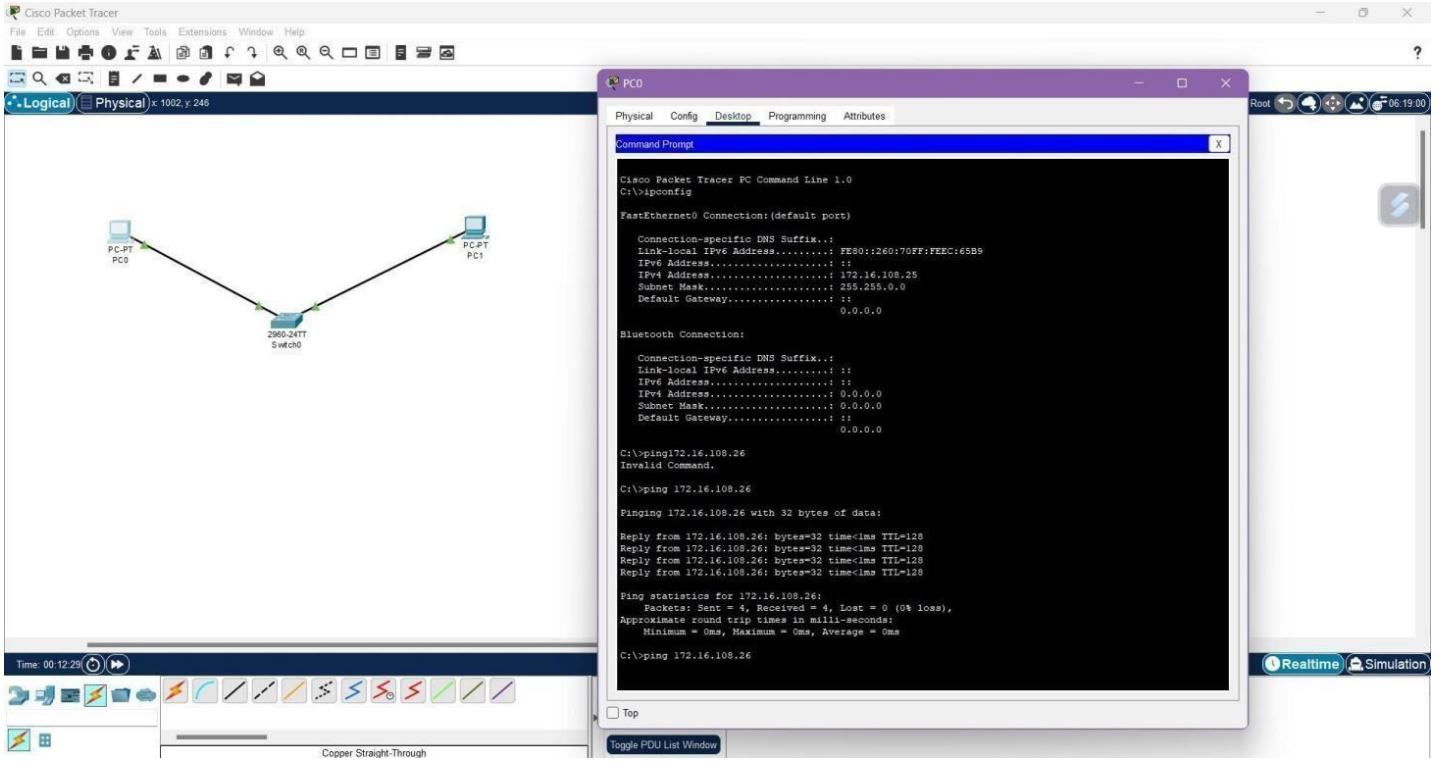
1. Ipconfig
2. Ping

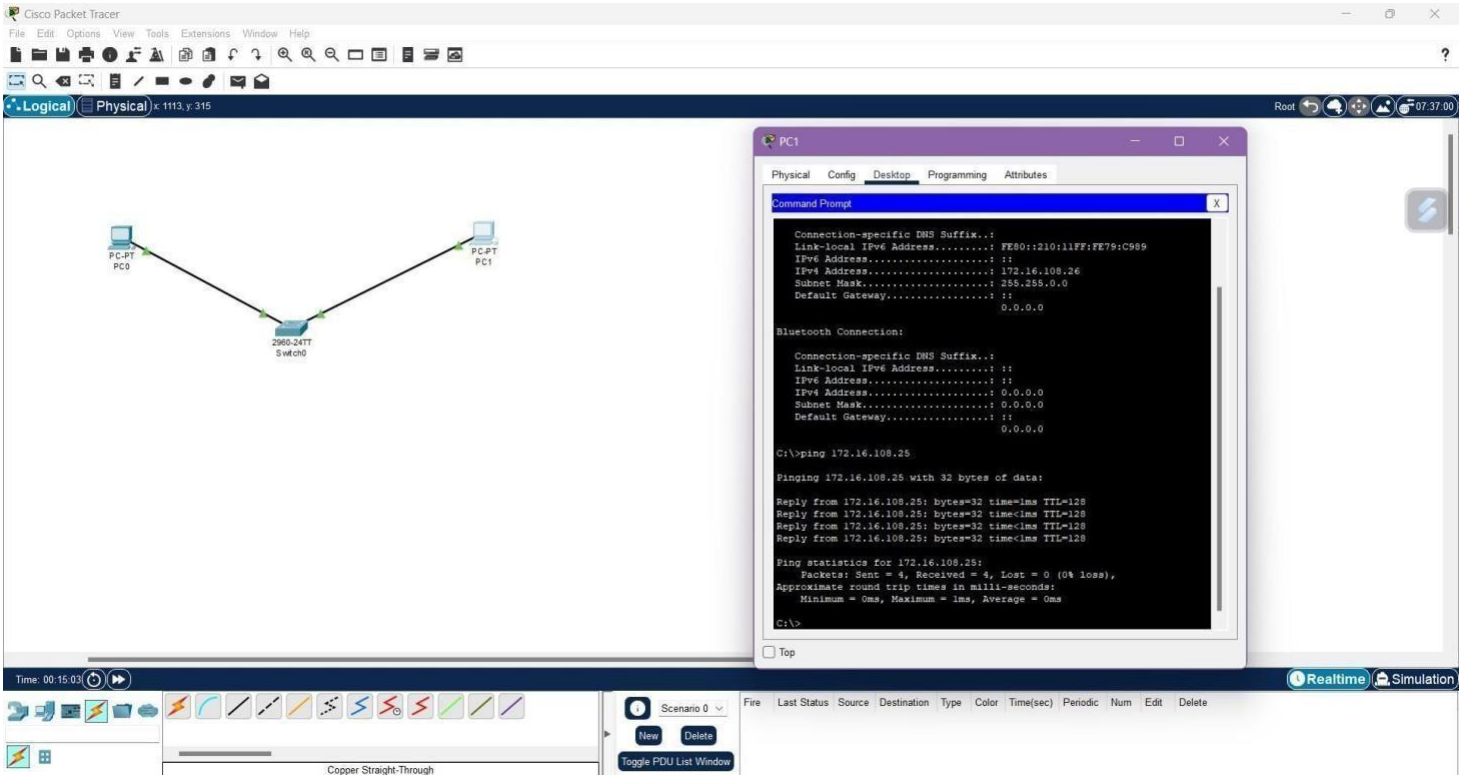
5. OutputDiagram:



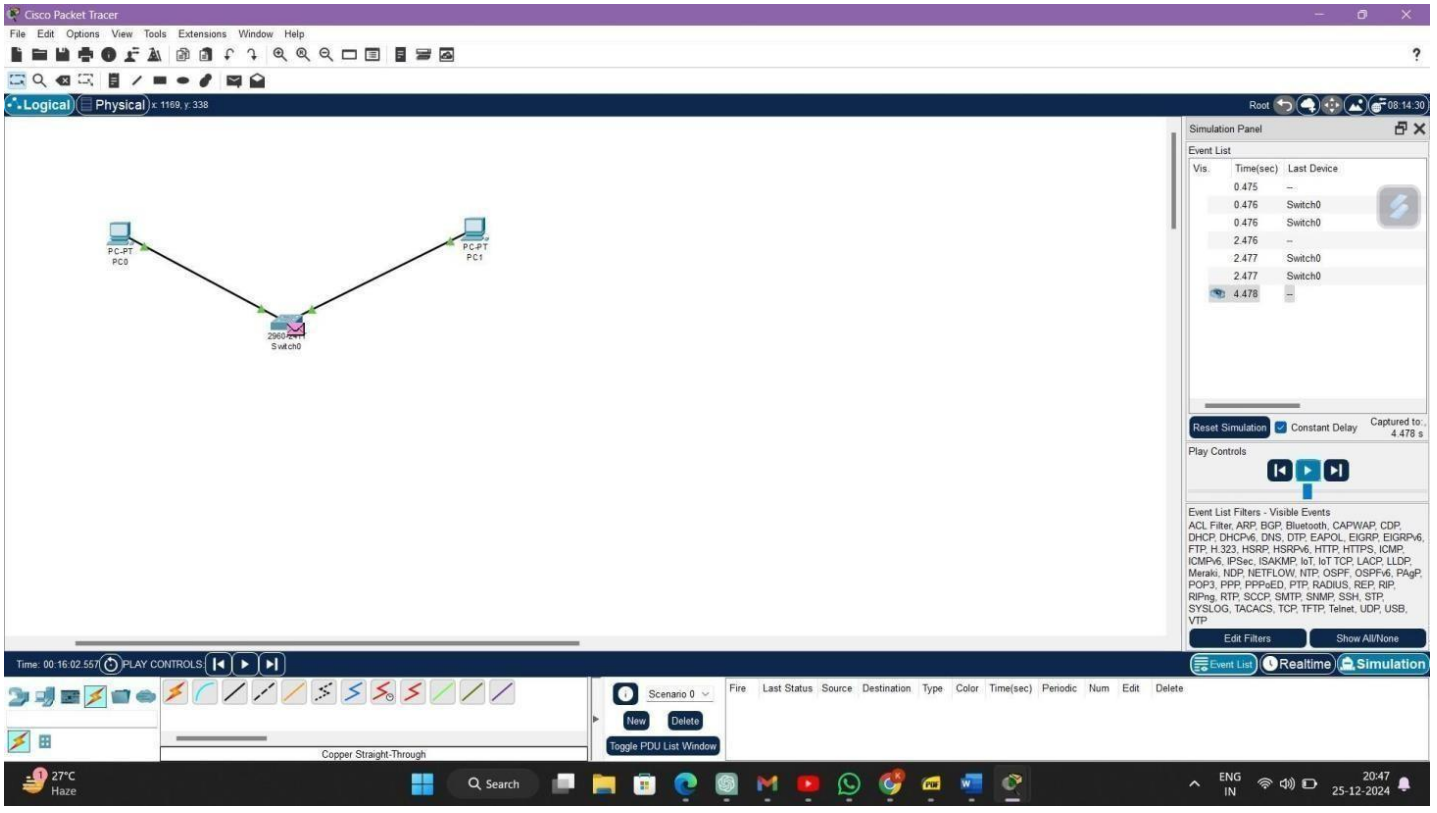
NetworkDiagram


Cisco Packet Tracer
File Edit Options View Tools Extensions Window Help
Logical Physical x: 940, y: 518
Root 03:54:30
PC-PT PC0
PC-PT PC1
2960-24TT Switch0
PC0
Physical Config Desktop Programming Attributes
IP Configuration
Interface FastEthernet0
DHCP Static
IPv4 Address 172.16.108.25
Subnet Mask 255.255.0.0
Default Gateway 0.0.0.0
DNS Server 0.0.0.0
IPv6 Configuration
Automatic Static
IPv6 Address
Link Local Address FE80::260:70FF:FEEC:65B9
Default Gateway
DNS Server
802.1X
Use 802.1X Security
Authentication MD5
Username
Password
Top
PC1
Physical Config Desktop Programming Attributes
IP Configuration
Interface FastEthernet0
DHCP Static
IPv4 Address 172.16.108.26
Subnet Mask 255.255.0.0
Default Gateway 0.0.0.0
DNS Server 0.0.0.0
IPv6 Configuration
Automatic Static
IPv6 Address
Link Local Address FE80::210:11FF:FE79:C989
Default Gateway
DNS Server
802.1X
Use 802.1X Security
Authentication MD5
Username
Password
Top
Time: 00:07:44
Realtime Simulation
Scenario 0
New Delete
Toggle PDU List Window
Fire Last Status Source Destination Type Color Time(sec) Periodic Num Edit Delete
Copper Straight-Through


Cisco Packet Tracer
File Edit Options View Tools Extensions Window Help
Logical Physical x: 1002, y: 246
Root 06:19:00
PC-PT PC0
PC-PT PC1
2960-24TT Switch0
PC0
Physical Config Desktop Programming Attributes
Command Prompt
Cisco Packet Tracer PC Command Line 1.0
C:\>ipconfig
FastEthernet0 Connection:(default port)
Connection-specific DNS Suffix..:
Link-local IPv6 Address.....: FE80::260:70FF:FEEC:65B9
IPv6 Address.....: ::
IPv4 Address.....: 172.16.108.25
Subnet Mask.....: 255.255.0.0
Default Gateway.....: ::
0.0.0.0
Bluetooth Connection:
Connection-specific DNS Suffix..:
Link-local IPv6 Address.....: ::
IPv6 Address.....: ::
IPv4 Address.....: 0.0.0.0
Subnet Mask.....: 0.0.0.0
Default Gateway.....: ::
0.0.0.0
C:\>ping172.16.108.26
Invalid Command.
C:\>ping 172.16.108.26
Pinging 172.16.108.26 with 32 bytes of data:
Reply from 172.16.108.26: bytes=32 time<1ms TTL=128
Reply from 172.16.108.26: bytes=32 time<1ms TTL=128
Reply from 172.16.108.26: bytes=32 time<1ms TTL=128
Reply from 172.16.108.26: bytes=32 time<1ms TTL=128
Ping statistics for 172.16.108.26:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\>ping 172.16.108.26
Top
Time: 00:12:29
Realtime Simulation
Toggle PDU List Window
Copper Straight-Through



AssigningIPAddress



CONCLUSION:

The study concludes that Packet Tracer is arobustanduser- friendly tool fornet work simulation and education. It plays a critical role in preparing individuals for industry certifications and realworld network management challenges. Its capabilities, combined with its accessibility, make it a cornerstone in the toolkit of network engineers and educators.

Rubrics for Experiment Assessment:

Rubrics	Good	Normal	Poor	Marks
Creation of Topology(4)	Created the topology, Identify the proper devices and making the connections **(4)**	Created the topology, Identify the proper devices, making the connections But missing some features **(3)**	Created wrong topology,Failed to Identify the proper devices and making connections **(1)**	
Verify the connectivity (4)	Verified the connectivity in all the levels **(4)**	Verified the connectivity at some levels (only some nodes) **(2)**	Verified the connectivity is not done. **(1)**	
Timely Completion (2)	Completed the lab before the allotted time **(2)**	Completed the lab after the deadline **(1)**	Did not submitted before grading**(0)**	
Total				

Result: Thus the Study of Network Tool-Packet Tracer has been done successfully.

Ex.No:	**1b**
Name of the Experi ment	**Study of Networking Commands**
Date	12-12-2024

Objectives:

To analyze the network basic commands.

Introduction:

In networking there are various commands that can be used to check the connectivity of the networkingdevicesanditisalsorequiredattimeoftroubleshootingofdevices.Wewillbediscussing few of the networking commands such as color help, ipconfig ,ipconfig/all ,nslookup ,tracert commands.

Requirements:

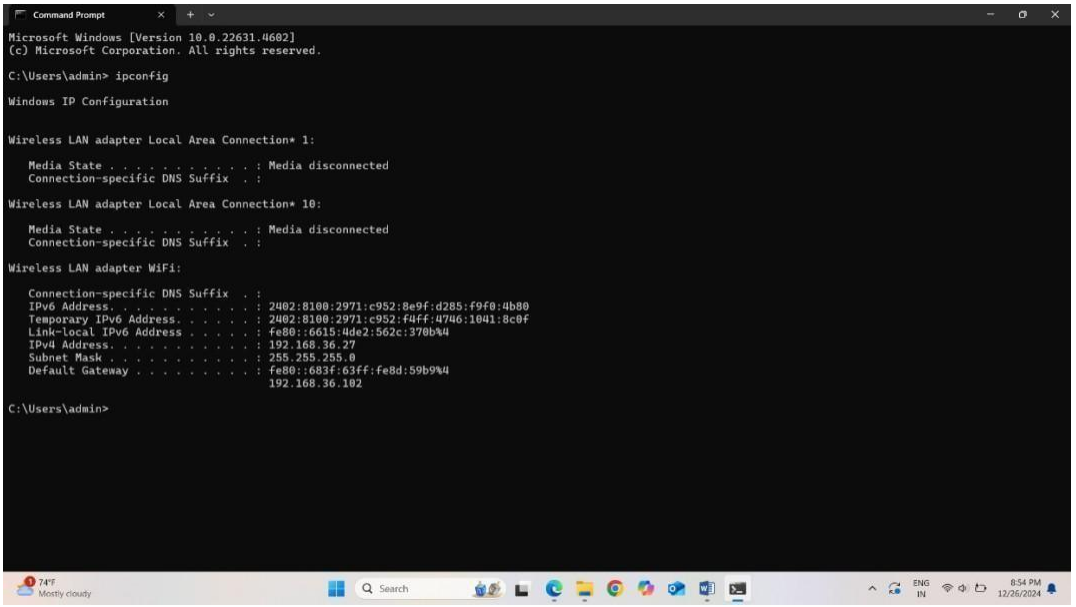
1. End Device(CommandPrompt)
2. Ethernet & InternetServices
3. Commands Commands Execution:

1. **ipconfig:**

This networking commands is used to the IP configuration details. This command provides you the details like IPv4 address ,Subnet Mask or Default Gateway.

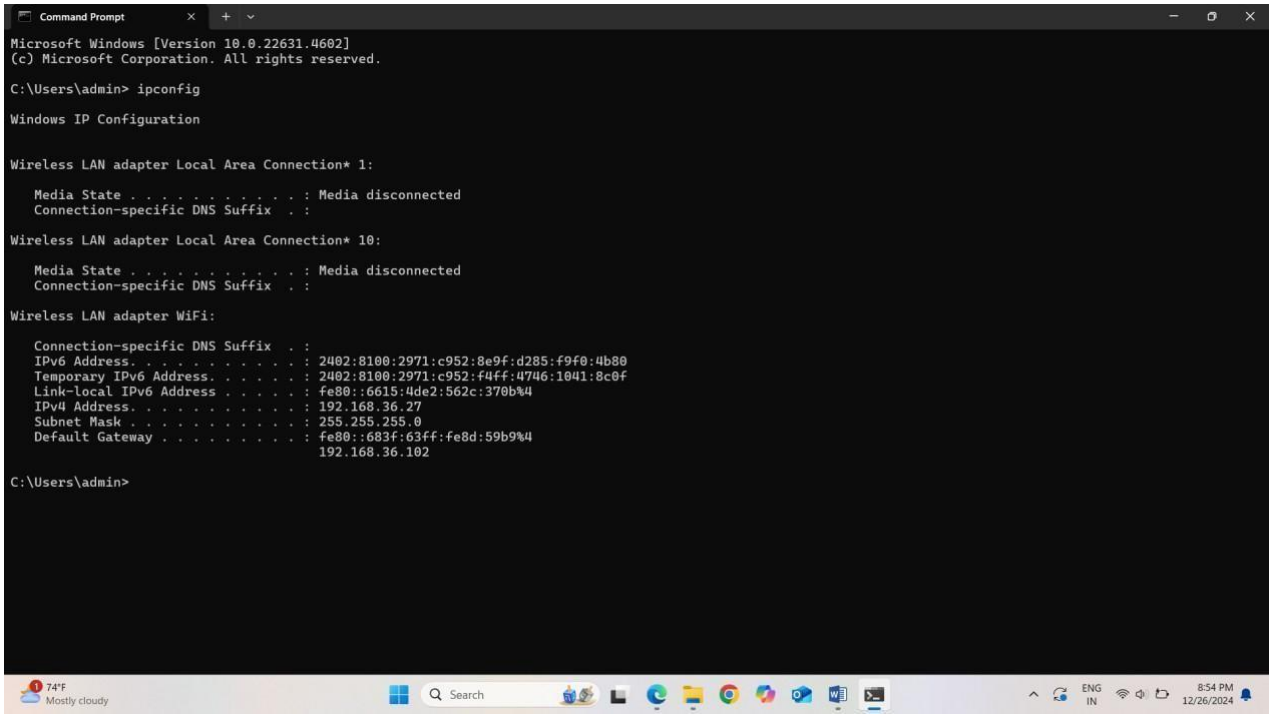
C:\Users\KARE>ipconfigOutput:

2. **ipconfig/all:**



This command can be understood as the updated version of the ipconfig command. This command tellsusthephysicaladdressofourdevice.ItellsusvariousdetailsofourcomputersuchasIPv4,IPv6 default Gateway ,subnet mask ,also it tells to which devices our device is connected ,configuration details of the devices to which are devices are connected.

C:\Users\KARE>ipconfig/allOutput:



3. **hostname:**

The hostname command displays the hostname of the system. The hostname command is much easier to use than going into the system settings to search for it.

C:\Users\KARE>hostnameOutput:

4. **systeminfo:**

This Command is used to display all the necessary information about our System such as configuration, version, hostname, processor details network card details etc.
C:\Users\KARE>systeminfoOutput:

```
Host Name:                 DESKTOP-EHROOQO
OS Name:                   Microsoft Windows 11 Home Single Language
OS Version:                10.0.22631 N/A Build 22631
OS Manufacturer:           Microsoft Corporation
OS Configuration:          Standalone Workstation
OS Build Type:             Multiprocessor Free
Registered Owner:          admin
Registered Organization:
Product ID:                00356-24569-74649-AAOEM
Original Install Date:     10/17/2024, 6:54:27 PM
System Boot Time:          12/26/2024, 8:20:44 PM
System Manufacturer:       LENOVO
System Model:              81WB
System Type:               x64-based PC
Processor(s):              1 Processor(s) Installed.
                           [01]: Intel64 Family 6 Model 142 Stepping 12 GenuineIntel ~2093 Mhz
BIOS Version:              LENOVO DXCN39WW, 10/13/2021
Windows Directory:         C:\WINDOWS
System Directory:          C:\WINDOWS\system32
Boot Device:               \Device\HarddiskVolume4
System Locale:             en-gb;English (United Kingdom)
Input Locale:              00004009
Time Zone:                 (UTC+05:30) Chennai, Kolkata, Mumbai, New Delhi
Total Physical Memory:     8,026 MB
Available Physical Memory: 3,305 MB
Virtual Memory: Max Size:  8,538 MB
Virtual Memory: Available: 3,568 MB
Virtual Memory: In Use:    4,970 MB
Page File Location(s):     C:\pagefile.sys
Domain:                    WORKGROUP
Logon Server:              \\DESKTOP-EHROOQO
Hotfix(s):                 4 Hotfix(s) Installed.
                           [01]: KB5045935
                           [02]: KB5027397
                           [03]: KB5048685
                           [04]: KB5046729
Network Card(s):           1 NIC(s) Installed.
                           [01]: Intel(R) Wireless-AC 9560
                                 Connection Name: WiFi
                                 DHCP Enabled:    Yes
```

5. **nslookup:**

This command is used to transform the given searched words into their corresponding IP addresses.

C:\Users\KARE>nslookup

C:\Users\KARE>nslookupDestinationHostname/DestinationIPAddressOutput:

```
C:\Users\admin> nslookup
Default Server:  UnKnown
Address:  192.168.36.102

> www.google.com
Server:  UnKnown
Address:  192.168.36.102

Non-authoritative answer:
Name:    www.google.com
Addresses:  2404:6800:4009:81f::2004
          142.250.183.100
```

6. **ping:**

Ping command is used to get to know if the particular site can be reached by the ping command. The ping command checks this by sending the packets of data to the destination address and if the data returns to us in the given time frame then it means that the particular website can be reached .We can do this by writing the ping and we write the IP address of the site we want to search.

C:\Users\KARE>pingIPAddress(or) C:\Users\KARE>pinghostname

C:\Users\KARE>ping -t IPAddress / Hostname Output:

```
Microsoft Windows [Version 10.0.22631.4602]
(c) Microsoft Corporation. All rights reserved.

C:\Users\admin> ping

Usage: ping [-t] [-a] [-n count] [-l size] [-f] [-i TTL] [-v TOS]
            [-r count] [-s count] [[-j host-list] | [-k host-list]]
            [-w timeout] [-R] [-S srcaddr] [-c compartment] [-p]
            [-4] [-6] target_name

Options:
    -t             Ping the specified host until stopped.
                   To see statistics and continue - type Control-Break;
                   To stop - type Control-C.
    -a             Resolve addresses to hostnames.
    -n count       Number of echo requests to send.
    -l size        Send buffer size.
    -f             Set Don't Fragment flag in packet (IPv4-only).
    -i TTL         Time To Live.
    -v TOS         Type Of Service (IPv4-only. This setting has been deprecated
                   and has no effect on the type of service field in the IP
                   Header).
    -r count       Record route for count hops (IPv4-only).
    -s count       Timestamp for count hops (IPv4-only).
    -j host-list   Loose source route along host-list (IPv4-only).
    -k host-list   Strict source route along host-list (IPv4-only).
    -w timeout     Timeout in milliseconds to wait for each reply.
    -R             Use routing header to test reverse route also (IPv6-only).
                   Per RFC 5095 the use of this routing header has been
                   deprecated. Some systems may drop echo requests if
                   this header is used.
    -S srcaddr     Source address to use.
    -c compartment Routing compartment identifier.
    -p             Ping a Hyper-V Network Virtualization provider address.
    -4             Force using IPv4.
    -6             Force using IPv6.
```

7. **tracert:**

This command can be understood as traceroute, which tells that our computer reaches or hits which server for reaching the particular route.

C:\Users\KARE>tracert IPAddress (or)

C:\Users\KARE>tracerthostname

```
C:\Users\admin> tracert

Usage: tracert [-d] [-h maximum_hops] [-j host-list] [-w timeout]
               [-R] [-S srcaddr] [-4] [-6] target_name

Options:
    -d                 Do not resolve addresses to hostnames.
    -h maximum_hops    Maximum number of hops to search for target.
    -j host-list       Loose source route along host-list (IPv4-only).
    -w timeout         Wait timeout milliseconds for each reply.
    -R                 Trace round-trip path (IPv6-only).
    -S srcaddr         Source address to use (IPv6-only).
    -4                 Force using IPv4.
    -6                 Force using IPv6.
```

8. **pathping:**

pathping is similar to tracert, except it is more informative and takes a lot longer to execute. After sending out packets from you to a given destination, it analyzes the route taken and computes packet loss on a per-hop basis.

C:\Users\KARE>pathping IPAddress (or)C:\Users\KARE>pathping hostname Output:

```
C:\Users\admin> pathping

Usage: pathping [-g host-list] [-h maximum_hops] [-i address] [-n]
                [-p period] [-q num_queries] [-w timeout]
                [-4] [-6] target_name

Options:
    -g host-list     Loose source route along host-list.
    -h maximum_hops  Maximum number of hops to search for target.
    -i address       Use the specified source address.
    -n               Do not resolve addresses to hostnames.
    -p period        Wait period milliseconds between pings.
    -q num_queries   Number of queries per hop.
    -w timeout       Wait timeout milliseconds for each reply.
    -4               Force using IPv4.
    -6               Force using IPv6.
```

9. **netstat:**

It is a command line tool that is identify and display the connections and ports connected to our computer when we write netstat command on CLI(Command Line Interface). It tells us active connections with our computer and it tells us local address ,foreign address and the state of the device. In local address first 8 digits specify the local address of our computer and and last 5 digits tells the port number to which our computer is connected . In netstat command there are various subcommands such as netstat -n, netstat -a,netstat -b, netstat -f.

C:\Users\KARE>netstatOutput:

```
C:\Users\admin> netstat

Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    192.168.36.27:63237    20.189.173.14:https    ESTABLISHED
  TCP    [2402:8100:2971:c952:f4ff:4746:1041:8c0f]:63172  [2603:1040:a06:6::1]:https  ESTABLISHED
  TCP    [2402:8100:2971:c952:f4ff:4746:1041:8c0f]:63173  [2603:1040:a06:6::1]:https  ESTABLISHED
  TCP    [2402:8100:2971:c952:f4ff:4746:1041:8c0f]:63174  whatsapp-cdn6-shv-02-maa2:5222  ESTABLISHED
  TCP    [2402:8100:2971:c952:f4ff:4746:1041:8c0f]:63224  bom07s31-in-x0e:https  TIME_WAIT
  TCP    [2402:8100:2971:c952:f4ff:4746:1041:8c0f]:63235  g2600-140f-0006-0000-0000-0000-1729-ba4b:https  ESTABLISHED
  TCP    [2402:8100:2971:c952:f4ff:4746:1041:8c0f]:63238  [2620:1ec:bdf::254]:https  CLOSE_WAIT
  TCP    [2402:8100:2971:c952:f4ff:4746:1041:8c0f]:63239  [2603:1030:13:201::254]:https  ESTABLISHED
  TCP    [2402:8100:2971:c952:f4ff:4746:1041:8c0f]:63240  [2606:2800:247:b713:6f8:1d37:ecd5:e137]:https  ESTABLISHED
  TCP    [2402:8100:2971:c952:f4ff:4746:1041:8c0f]:63241  [2606:2800:247:57cb:4371:48bc:8b00:14c3]:http  ESTABLISHED
  TCP    [2402:8100:2971:c952:f4ff:4746:1041:8c0f]:63242  [64:ff9b::cc4f:c5de]:https  ESTABLISHED
  TCP    [2402:8100:2971:c952:f4ff:4746:1041:8c0f]:63243  bom07s29-in-x0e:https  ESTABLISHED
```

10. **getmac:**

Getmac is a Windows command used to display the Media Access Control (MAC) addresses for each network adapter in the computer.

```
C:\Users\admin> getmac

Physical Address    Transport Name
=================== ============================================================
48-68-4A-A5-B5-C7   \Device\Tcpip_{14695CA0-7F65-44C8-9915-54F02070C137}
```

C:\Users\KARE>getmacOutput:

11. **ARP:**

The arp command displays and modifies the Internet-to-adapter address translation tables used by the Address in Networks and communication management. The arp command displays the current ARP entry for the host specified by the HostName variable. The host can be specified by name or number, using Internet dotted decimal notation.

C:\Users\KARE>arp-aOutput:

```
C:\Users\admin> ARP

Displays and modifies the IP-to-Physical address translation tables used by
address resolution protocol (ARP).

ARP -s inet_addr eth_addr [if_addr]
ARP -d inet_addr [if_addr]
ARP -a [inet_addr] [-N if_addr] [-v]

  -a            Displays current ARP entries by interrogating the current
                protocol data.  If inet_addr is specified, the IP and Physical
                addresses for only the specified computer are displayed.  If
                more than one network interface uses ARP, entries for each ARP
                table are displayed.
  -g            Same as -a.
  -v            Displays current ARP entries in verbose mode.  All invalid
                entries and entries on the loop-back interface will be shown.
  inet_addr     Specifies an internet address.
  -N if_addr    Displays the ARP entries for the network interface specified
                by if_addr.
  -d            Deletes the host specified by inet_addr. inet_addr may be
                wildcarded with * to delete all hosts.
  -s            Adds the host and associates the Internet address inet_addr
                with the Physical address eth_addr.  The Physical address is
                given as 6 hexadecimal bytes separated by hyphens. The entry
                is permanent.
  eth_addr      Specifies a physical address.
  if_addr       If present, this specifies the Internet address of the
                interface whose address translation table should be modified.
                If not present, the first applicable interface will be used.
Example:
  > arp -s 157.55.85.212   00-aa-00-62-c6-09  .... Adds a static entry.
  > arp -a                                    .... Displays the arp table.
```

12. **route:**

The `route` command allows you to make manual entries into the network routing tables. The `route` command distinguishes between routes to hosts and routes to networks by interpreting the network address of the Destination variable, which can be specified either by symbolic name or numeric address. The `route` command resolves all symbolic names into addresses, using either the `/etc/hosts` file or the

Rubrics for Experiment Assessment:

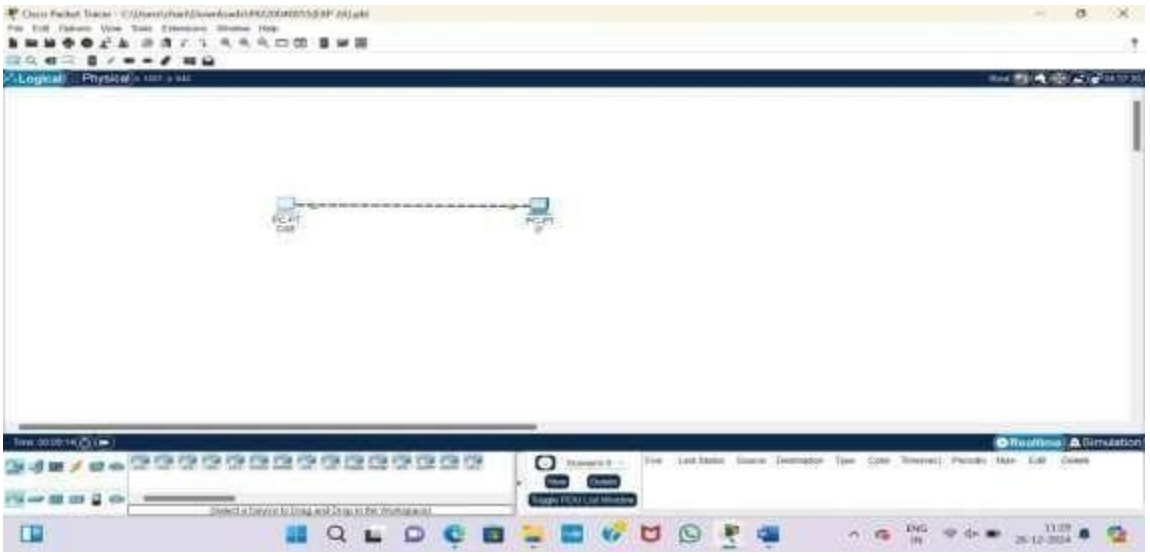
Rubrics	Good	Normal	Poor	Marks
Creation of Topology(4)	Created the topology, Identify the proper devices and making the connections **(4)**	Created the topology, Identify the proper devices, making the connections But missing some features **(3)**	Created wrong topology,Failed to Identify the proper devices and making connections **(1)**	
Verify the connectivity (4)	Verified the connectivity in all the levels **(4)**	Verified the connectivity at some levels (only some nodes) **(2)**	Verified the connectivity is not done. **(1)**	
Timely Completion (2)	Completed the lab before the allotted time **(2)**	Completed the lab after the deadline **(1)**	Did not submitted before grading**(0)**	
Total				

Result: Thus the Study of Networking commands has been done successfully.

Ex. No:	2a
Name of the Experiment	Study of Network Devices a) Building a peer to peer network
Date	19-12-2024

1. **Device Requirements:**
    1. Switch
    2. PC0
    3. PC1
    4. Wires
2. **Network Diagram for your experiment** (draw the diagram either hand drawing/ms paint or any other drawing tools)

3. **Network Diagram** (packet tracer diagram before configuration):

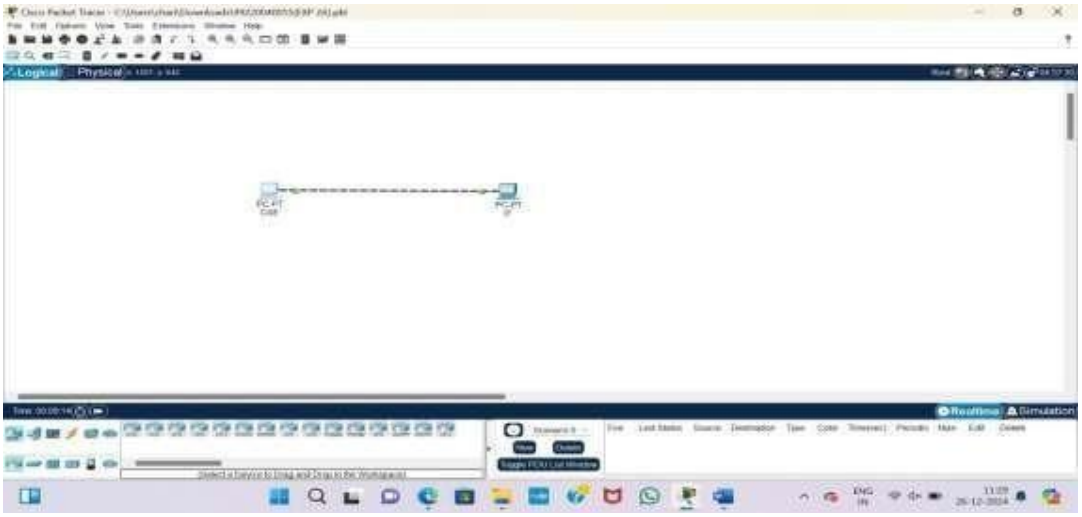


4. **Configuration details**:

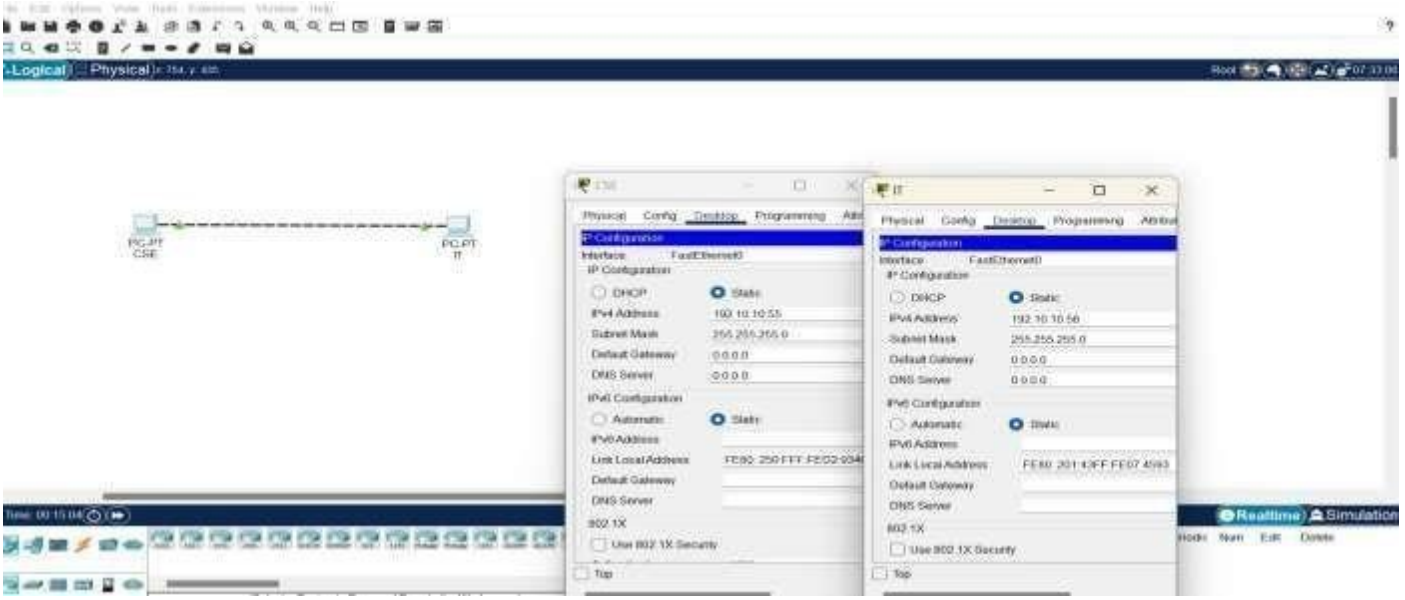
Device Name	Interface Name	IP Address	Subnet Mask
CSE	Fa0	192.10.10.55	255.255.255.0

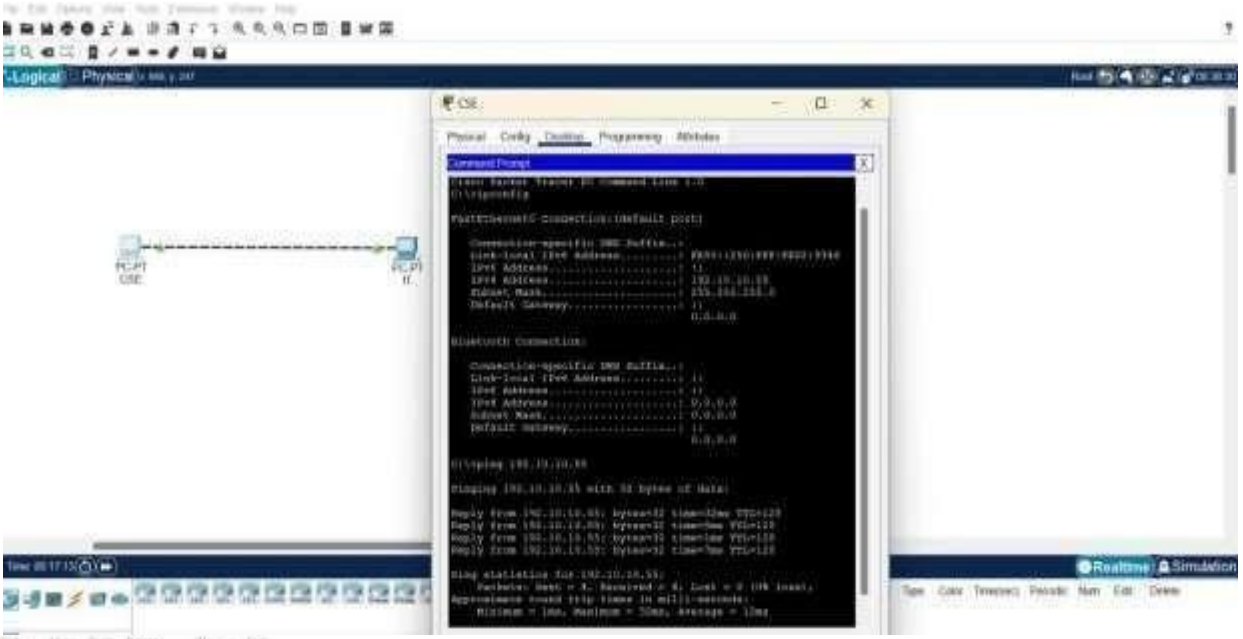
IT	Fa0	192.10.10.56	255.255.255.0

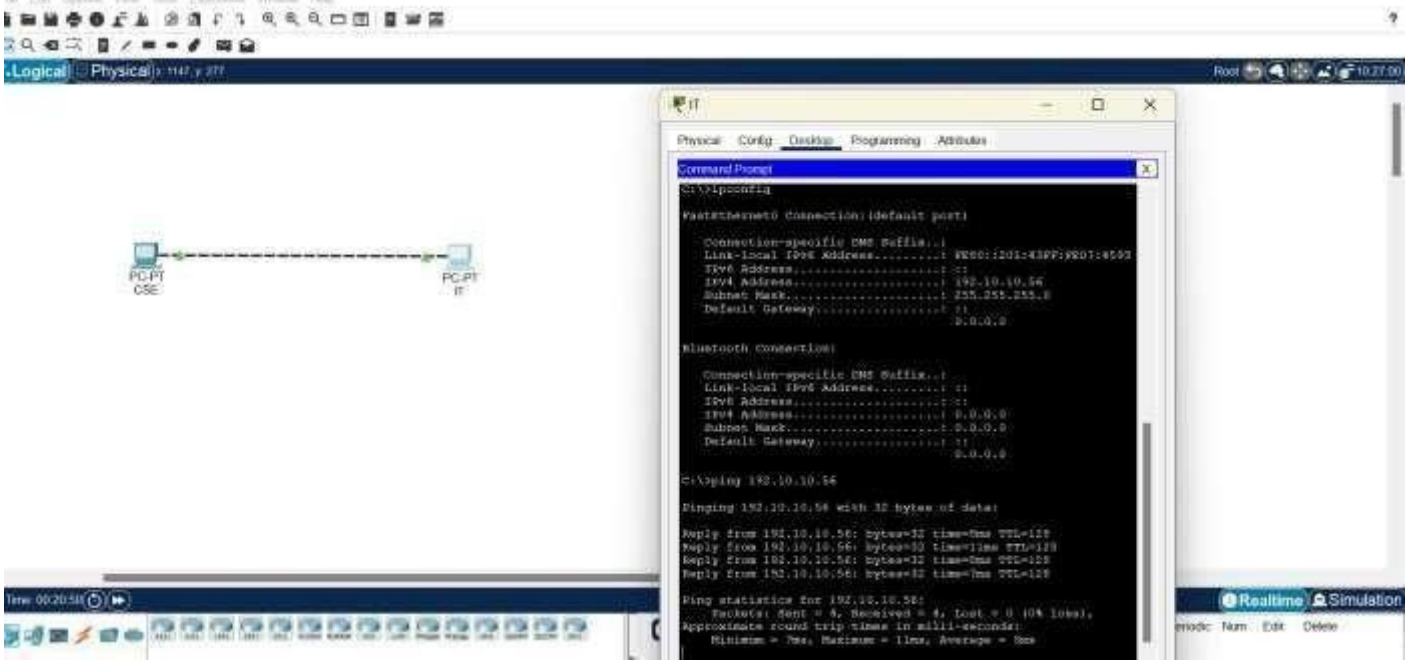
5. **Commands used** (List of commands to be used has to be listed (if any):
    1. Ipconfig
    2. Ping
6. **Output Diagram** (Minimum 3 screenshot):

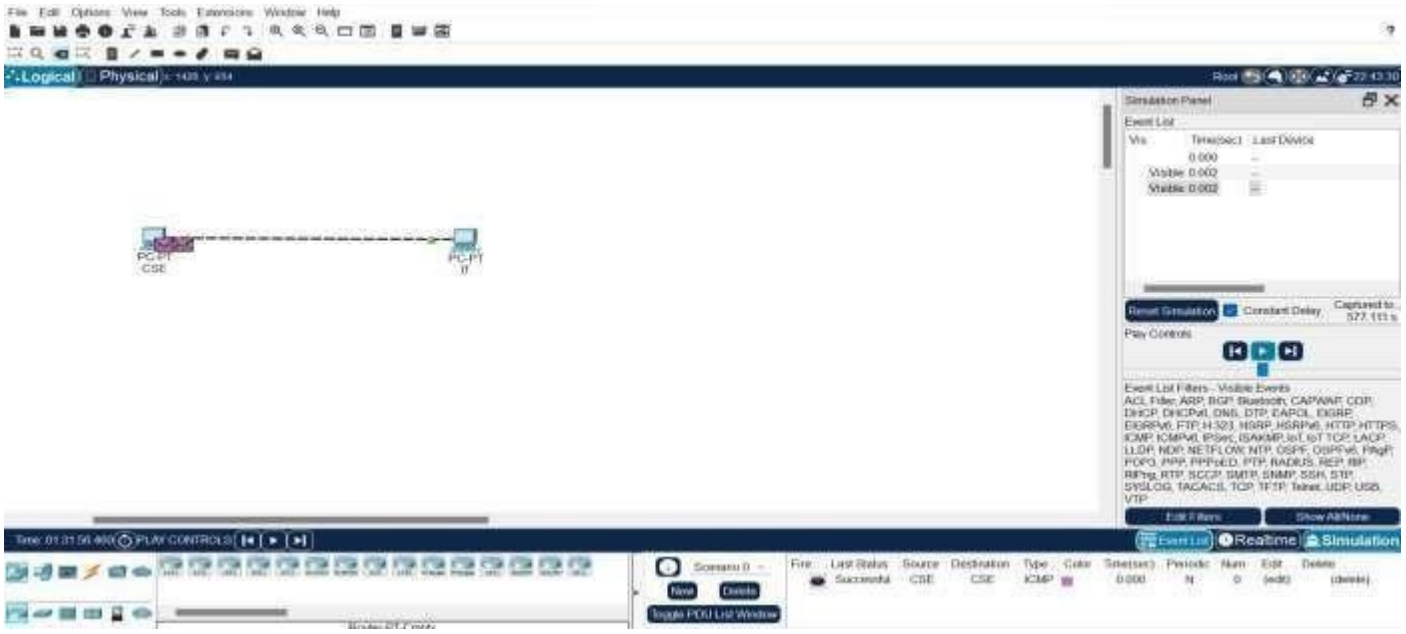


Network Diagram




CSE
Physical Config Desktop Programming Attributes
Command Prompt
C:\>ipconfig
FastEthernet0 Connection:(default port)
Connection-specific DNS Suffix..:
Link-local IPv6 Address.....: FE80::201:43FF:FE07:6553
IPv6 Address.....: ::
IPv4 Address.....: 192.10.10.55
Subnet Mask.....: 255.255.255.0
Default Gateway.....: ::
0.0.0.0
Bluetooth Connection:
Connection-specific DNS Suffix..:
Link-local IPv6 Address.....: ::
IPv6 Address.....: ::
IPv4 Address.....: 0.0.0.0
Subnet Mask.....: 0.0.0.0
Default Gateway.....: ::
0.0.0.0
C:\>ping 192.10.10.55
Pinging 192.10.10.55 with 32 bytes of data:
Reply from 192.10.10.55: bytes=32 time<1ms TTL=128
Reply from 192.10.10.55: bytes=32 time<1ms TTL=128
Reply from 192.10.10.55: bytes=32 time<1ms TTL=128
Reply from 192.10.10.55: bytes=32 time<1ms TTL=128
Ping statistics for 192.10.10.55:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 1ms, Maximum = 50ms, Average = 13ms
PC-PT
CSE
PC-PT
IT
Realtime
Simulation


IT
Physical Config Desktop Programming Attributes
Command Prompt
C:\>ipconfig
FastEthernet0 Connection:(default port)
Connection-specific DNS Suffix..:
Link-local IPv6 Address.....: FE80::201:43FF:FE07:6553
IPv6 Address.....: ::
IPv4 Address.....: 192.10.10.56
Subnet Mask.....: 255.255.255.0
Default Gateway.....: ::
0.0.0.0
Bluetooth Connection:
Connection-specific DNS Suffix..:
Link-local IPv6 Address.....: ::
IPv6 Address.....: ::
IPv4 Address.....: 0.0.0.0
Subnet Mask.....: 0.0.0.0
Default Gateway.....: ::
0.0.0.0
C:\>ping 192.10.10.56
Pinging 192.10.10.56 with 32 bytes of data:
Reply from 192.10.10.56: bytes=32 time=5ms TTL=128
Reply from 192.10.10.56: bytes=32 time=1ms TTL=128
Reply from 192.10.10.56: bytes=32 time=5ms TTL=128
Reply from 192.10.10.56: bytes=32 time=7ms TTL=128
Ping statistics for 192.10.10.56:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 7ms, Maximum = 11ms, Average = 9ms
PC-PT
CSE
PC-PT
IT
Realtime
Simulation


File Edit Options View Tools Extensions Window Help
Logical
Physical
Simulation Panel
Event List
Vis. Time(sec) Last Device
0.000
Visible: 0.002
Visible: 0.003
Reset Simulation
Constant Delay
Play Controls
Event List Filters - Visible Events
ACL Filter, ARP, BGP, Bluetooth, CAPWAP, CDP, DHCP, DHCPv6, DNS, DTP, EAPOL, EIGRP, EIGRPv6, FTP, H.323, HSRP, HSRPv6, HTTP, HTTPS, ICMP, ICMPv6, IPSec, ISAKMP, IoT, IoT TCP, LACP, LLDP, NDP, NETFLOW, NTP, OSPF, OSPFv6, PAgP, POP3, PPP, PPPoED, PTP, RADIUS, REP, RIP, RIPng, RTP, SCCP, SMTP, SNMP, SSH, STP, SYSLOG, TACACS, TCP, TFTP, Telnet, UDP, USB, VTP
Edit Filters
Show All/None
Realtime
Simulation
Scenario 0
New
Delete
Toggle PDU List Window
Fire
Last Status
Source
Destination
Type
Color
Time(sec)
Periodic
Num
Edit
Delete
Successful
CSE
CSE
ICMP
0.000
N
0
(edit)
(delete)

Rubrics for Experiment Assessment:

Rubrics	Good	Normal	Poor	Marks
Creation of Topology(4)	Created the topology, Identify the proper devices and making the connections **(4)**	Created the topology, Identify the proper devices, making the connections But missing some features **(3)**	Created wrong topology, Failed to Identify the proper devices and making connections **(1)**	
Verify the connectivity (4)	Verified the connectivity in all the levels **(4)**	Verified the connectivity at some levels (only some nodes) **(2)**	Verified the connectivity is not done. **(1)**	
Timely Completio n (2)	Completed the lab before the allotted time **(2)**	Completed the lab after the deadline **(1)**	Did not submitted before grading**(0)**	
Total				

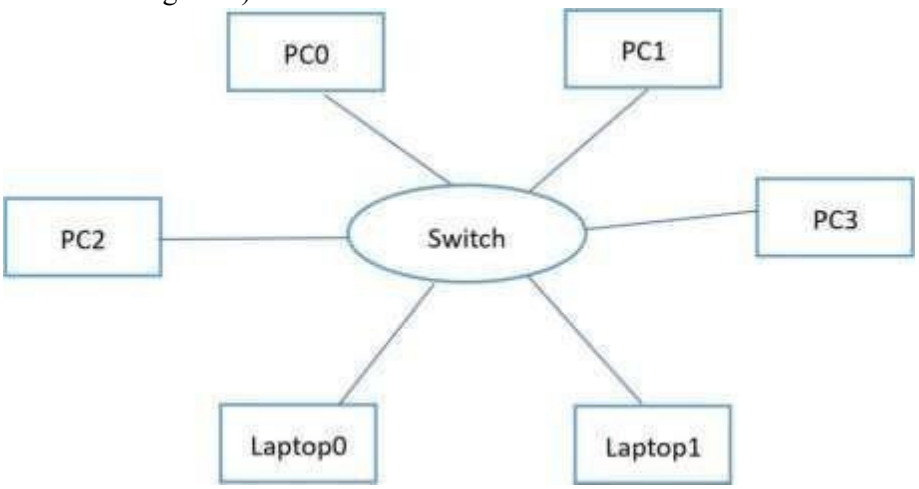
Result : Thus the Study of Network Devices has been done successfully.

Ex. No:	2b
Name of the Experiment	Study of Network Devices b) Design a simple LAN Network
Date	26-12-2024

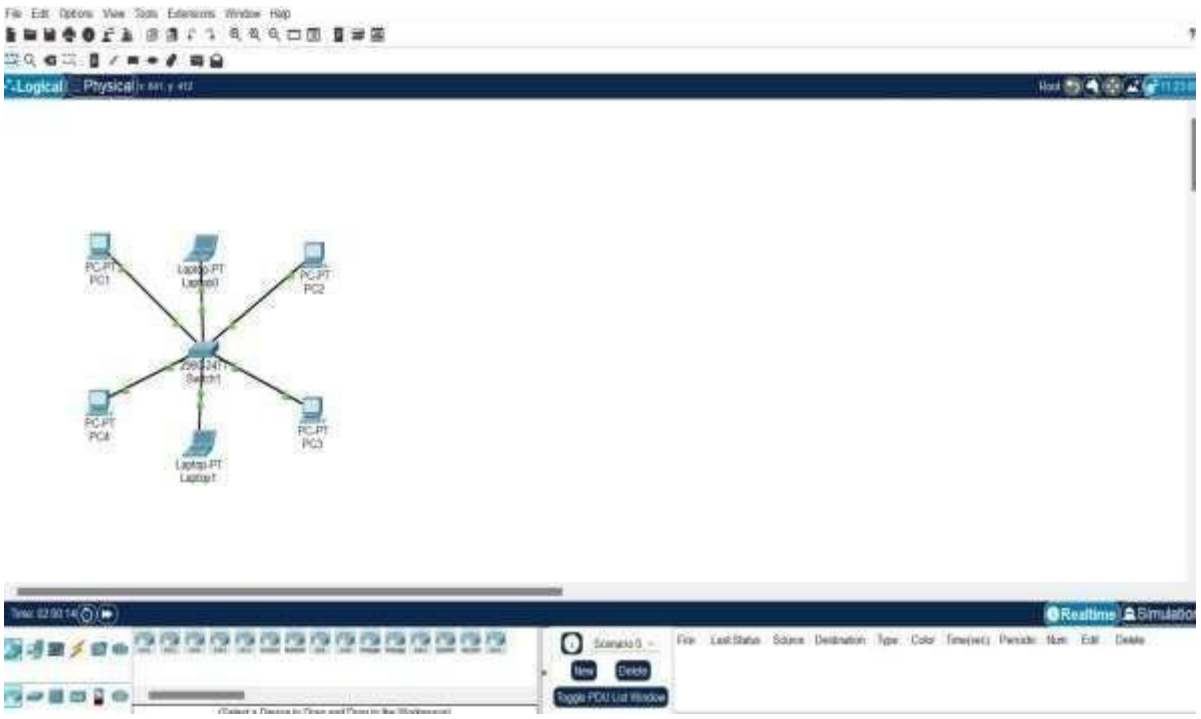
1. Device Requirements:

1. Switch
2. PC0
3. PC1
4. PC2
5. PC3
6. Laptop0
7. Laptopl
8. Wire

1. **Network Diagram for your experiment** (draw the diagram either hand drawing/ms paint or any other drawing tools)



2. **Network Diagram (packet tracer diagram before configuration):**



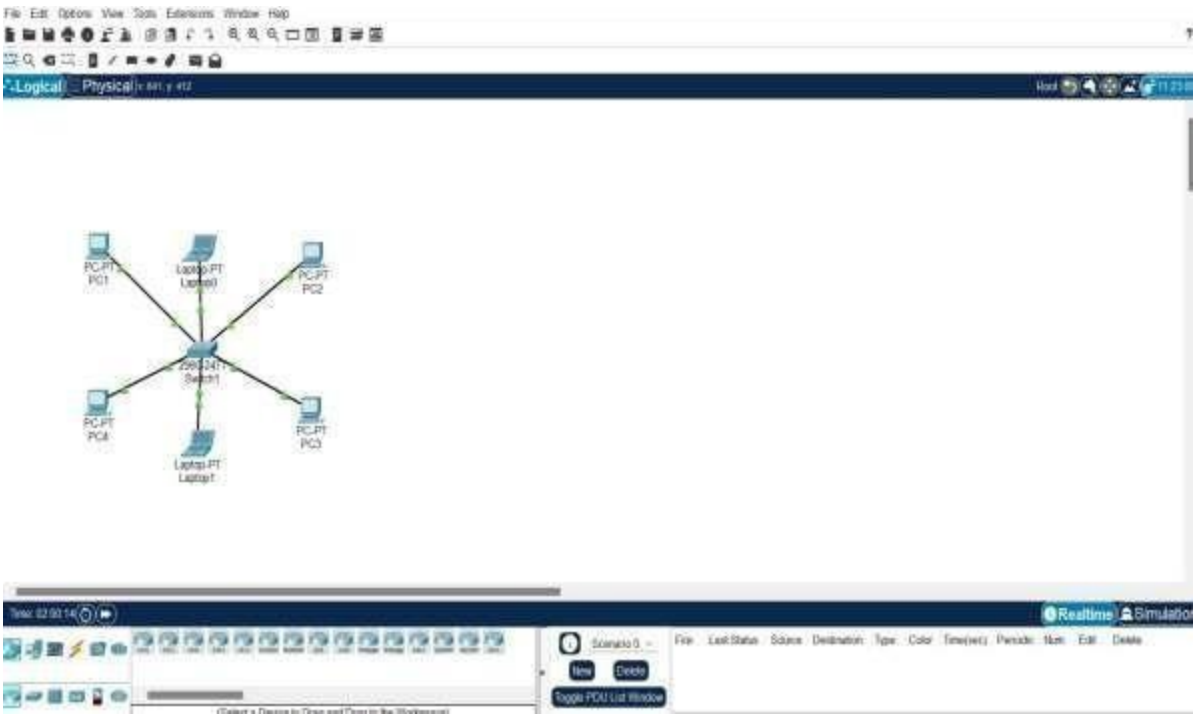
3. **Configuration details:**

Device Name	Interface Name	IP Address	Subnet mask
PC1	Fa0	172.16.103.55	255.255.0.0
PC2	Fa0/5	172.16.103.57	255.255.0.0
PC3	Fa0/6	172.16.103.58	255.255.0.0
PC4	Fa0/3	172.16.103.60	255.255.0.0
Laptop0	Fa0/2	172.16.103.56	255.255.0.0
Laptopl	Fa0/4	172.16.103.59	255.255.0.0
Switch			

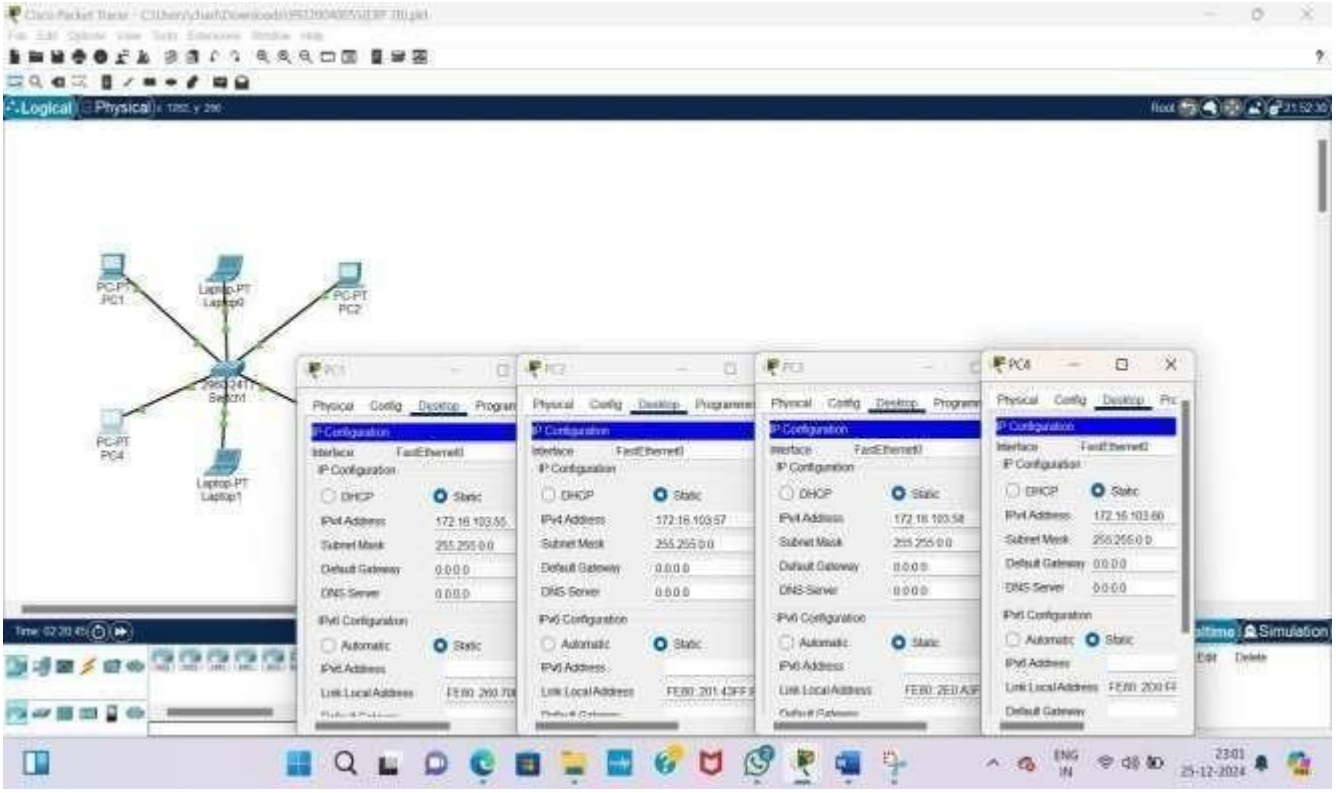
4. **Commands used** (List of commands to be used has to be listed (if any): a.Ipconfigb. Ping

5. **Output Diagram** (Minimum 3 screenshot):

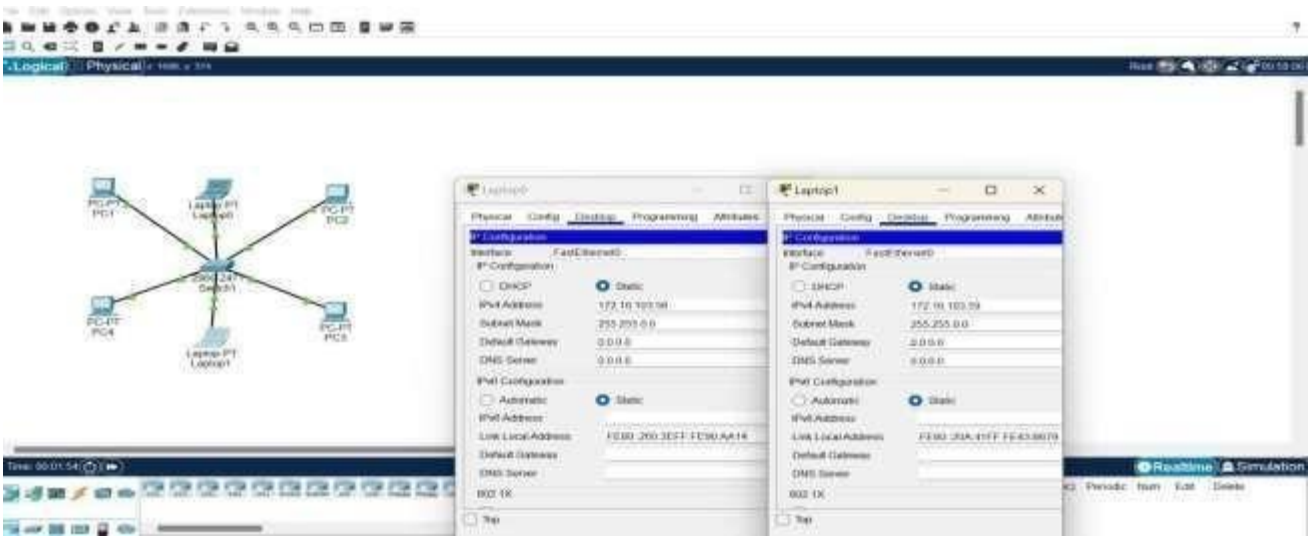
c s<o eackeime r O

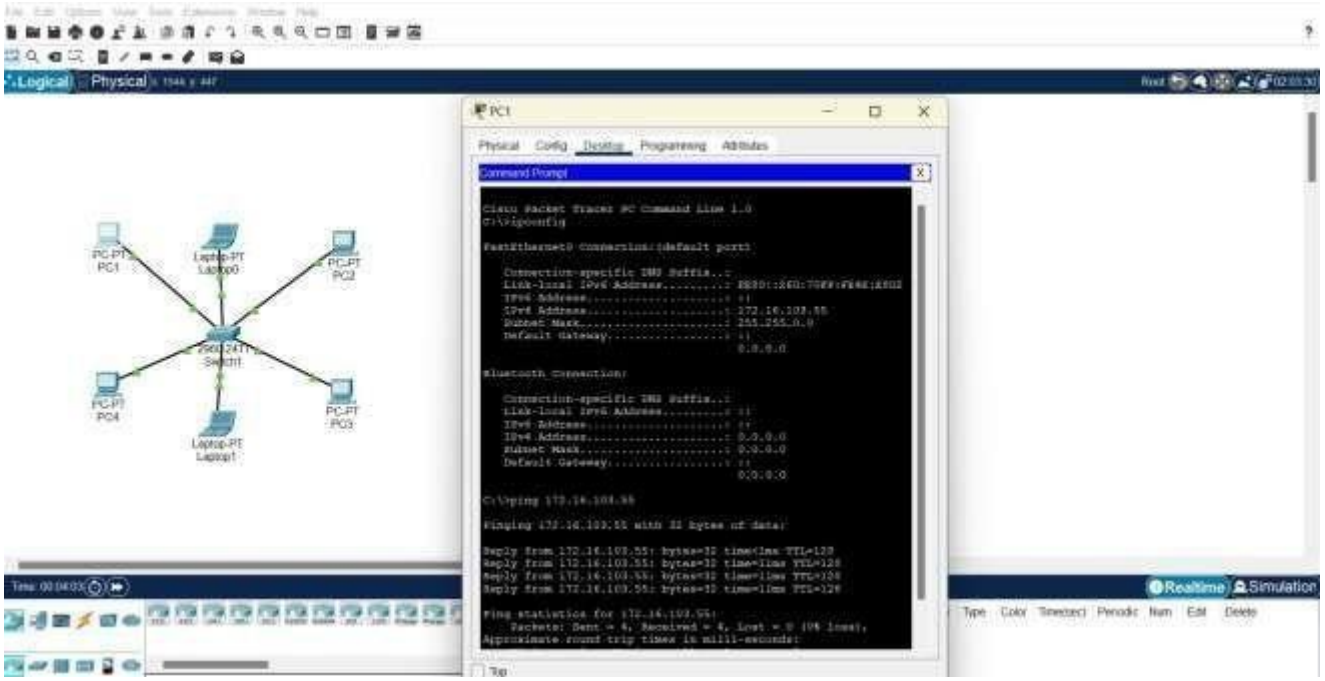


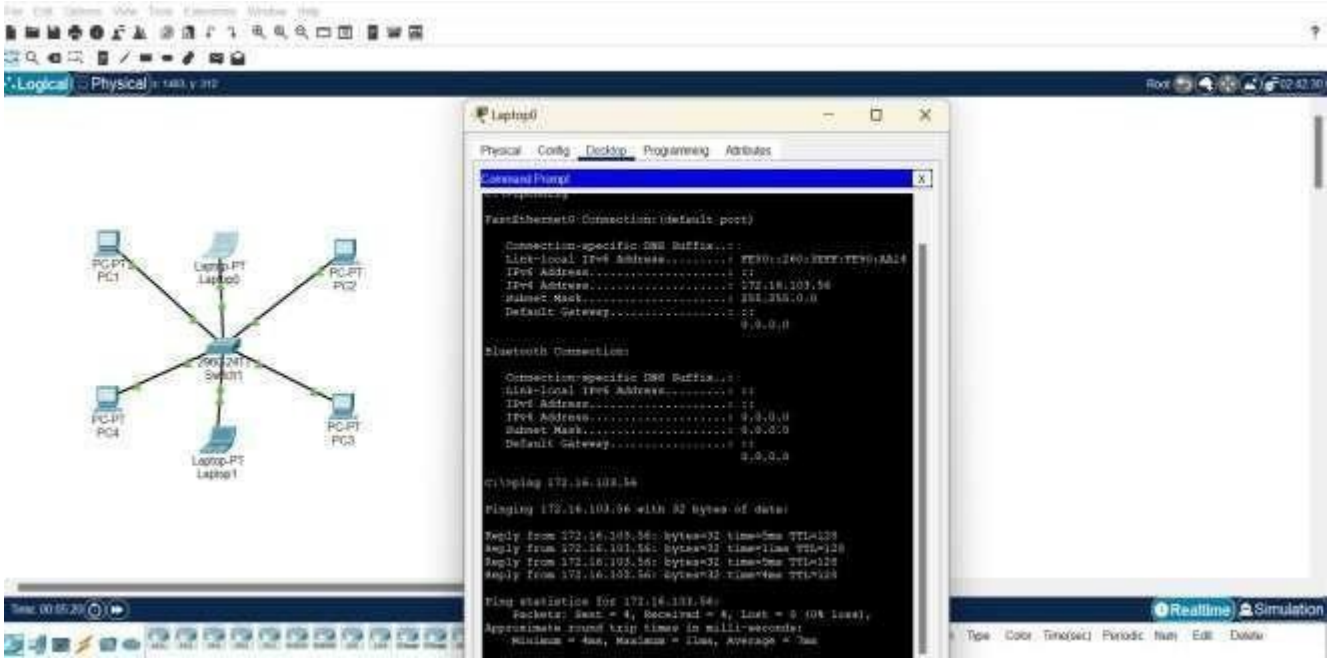
Network Diagram

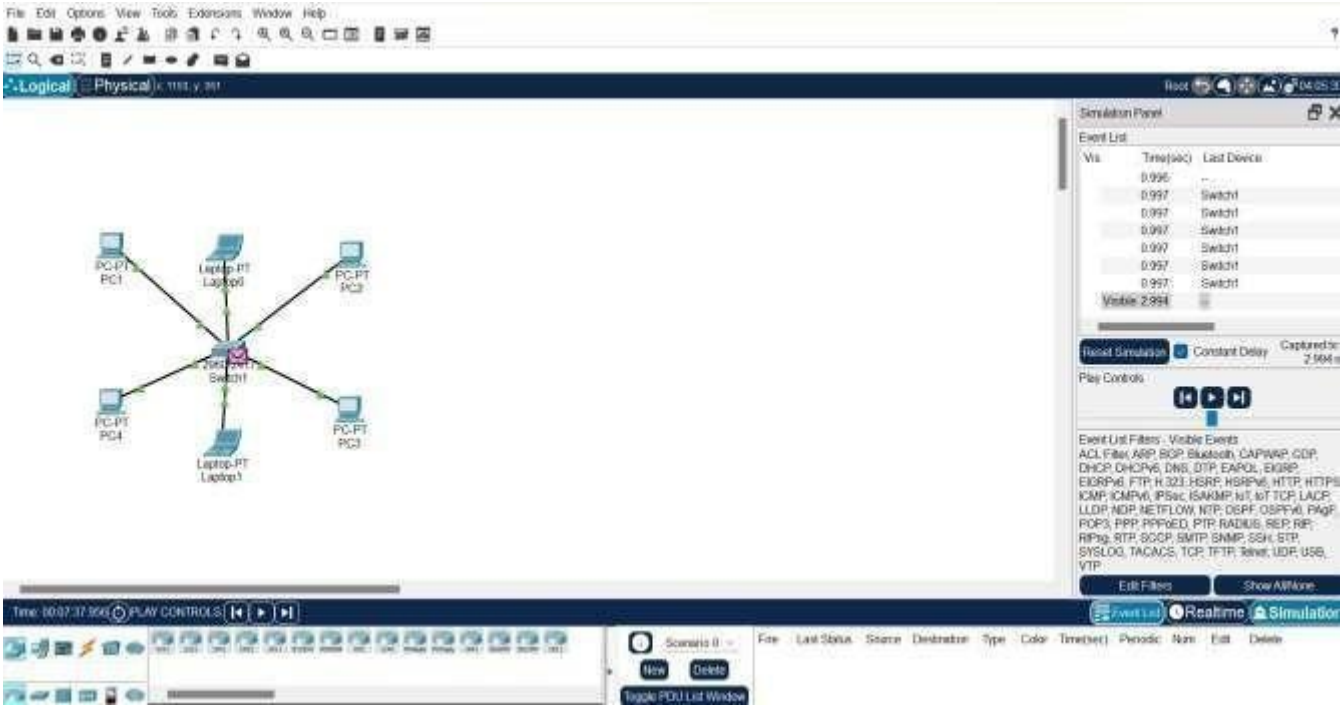


Assigning IP Address


Logical
Physical
PC-PT PC1
Laptop-PT Laptop0
PC-PT PC2
2960-24TT Switch1
PC-PT PC4
PC-PT PC3
Laptop-PT Laptop1
Laptop0
Laptop1
Physical
Config
Desktop
Programming
Attributes
IP Configuration
Interface
FastEthernet0
DHCP
Static
IPv4 Address
Subnet Mask
Default Gateway
DNS Server
IPv6 Configuration
Automatic
IPv6 Address
Link Local Address
802.1X
Top
Realtime
Simulation


Logical
Physical
PC1
Physical
Config
Desktop
Programming
Attributes
Command Prompt
Cisco Packet Tracer PC Command Line 1.0
C:\>ipconfig
FastEthernet0 Connection:(default port)
Connection-specific DNS Suffix..:
Link-local IPv6 Address.....: FE80::260:7099:9E4E:E502
IPv6 Address.....: ::
IPv4 Address.....: 172.16.103.55
Subnet Mask.....: 255.255.0.0
Default Gateway.....: ::
0.0.0.0
Bluetooth Connection:
Connection-specific DNS Suffix..:
Link-local IPv6 Address.....: ::
IPv6 Address.....: ::
IPv4 Address.....: 0.0.0.0
Subnet Mask.....: 0.0.0.0
Default Gateway.....: ::
0.0.0.0
C:\>ping 172.16.103.55
Pinging 172.16.103.55 with 32 bytes of data:
Reply from 172.16.103.55: bytes=32 time<1ms TTL=128
Reply from 172.16.103.55: bytes=32 time<1ms TTL=128
Reply from 172.16.103.55: bytes=32 time<1ms TTL=128
Reply from 172.16.103.55: bytes=32 time<1ms TTL=128
Ping statistics for 172.16.103.55:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Top
Realtime
Simulation


Logical
Physical
Laptop0
Physical
Config
Desktop
Programming
Attributes
Command Prompt
FastEthernet0 Connection:(default port)
Connection-specific DNS Suffix..:
Link-local IPv6 Address.....: FE80::260:3EFF:FE90:AA14
IPv6 Address.....: ::
IPv4 Address.....: 172.16.103.56
Subnet Mask.....: 255.255.0.0
Default Gateway.....: ::
0.0.0.0
Bluetooth Connection:
Connection-specific DNS Suffix..:
Link-local IPv6 Address.....: ::
IPv6 Address.....: ::
IPv4 Address.....: 0.0.0.0
Subnet Mask.....: 0.0.0.0
Default Gateway.....: ::
0.0.0.0
C:\>ping 172.16.103.56
Pinging 172.16.103.56 with 32 bytes of data:
Reply from 172.16.103.56: bytes=32 time=0ms TTL=128
Reply from 172.16.103.56: bytes=32 time=1ms TTL=128
Reply from 172.16.103.56: bytes=32 time=0ms TTL=128
Reply from 172.16.103.56: bytes=32 time=0ms TTL=128
Ping statistics for 172.16.103.56:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 1ms, Average = 0ms
Realtime
Simulation



Rubrics for Experiment Assessment:

Rubrics	**Good**	**Normal**	**Poor**	**Marks**
Creation of Topology(4)	Created the topology, Identify the proper devices and making the connections **(4)**	Created the topology, Identify the proper devices, making the connections But missing some features **(3)**	Created wrong topology, Failed to Identify the proper devices and making connections **(1)**	
Verify the connectivity (4)	Verified the connectivity in all the levels **(4)**	Verified the connectivity at some levels (only some nodes) **(2)**	Verified the connectivity is not done. **(1)**	
Timely Completio n (2)	Completed the lab before the allotted time **(2)**	Completed the lab after the deadline **(1)**	Did not submitted before grading**(0)**	
Total				

Result : Thus the Design a Simple LAN Network has been done successfully .

Ex.No:	**3**
Name of the Experiment	**Study of Different types of Network Cables**
Date	2-01-2025

Objective(s):
To Study of different types of Network cables and practically implement the Crossover wired and Straight through cable using Crimping Tool.

Components Required:

- CAT5, CAT6 Cable
- RJ45 Crimp-able Connector
- Crimping tools
- Splicer

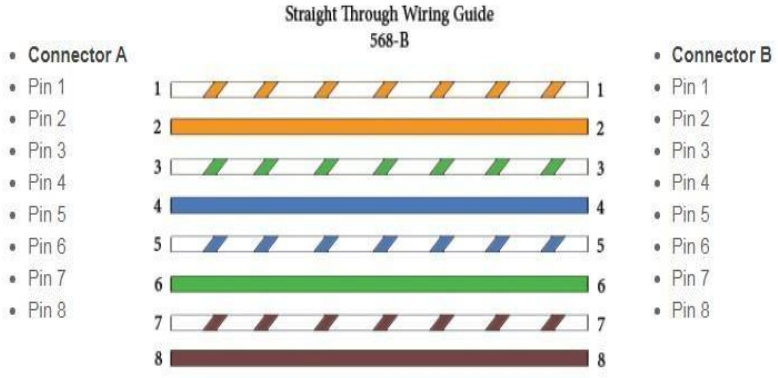
Description:
The Ethernet cables for connectivity in most office and home environments rely on twisted wire pairs within an overall cable -Cat5,Cat6 andCat7all used this format.

Straight-Through Wired Cables

Straight- Through refers to cables that have the pin assignments on each end of the cable. In other words, Pin1 connector A goes to P in 1 on connector B, Pin2 to Pin2,etc. Straight-Through wired cables are most commonly used to connect a host to a client. When we talk about cat5e patch cables, the Straight-Through wired cat5e patch cable is used to connect computers, printers, and other network client devices to the router switch or hub (the host device in this instance).

Use straight-through cables for the following connections:

- Switch to a router Ethernet port
- Computer to switch
- Computer to hub



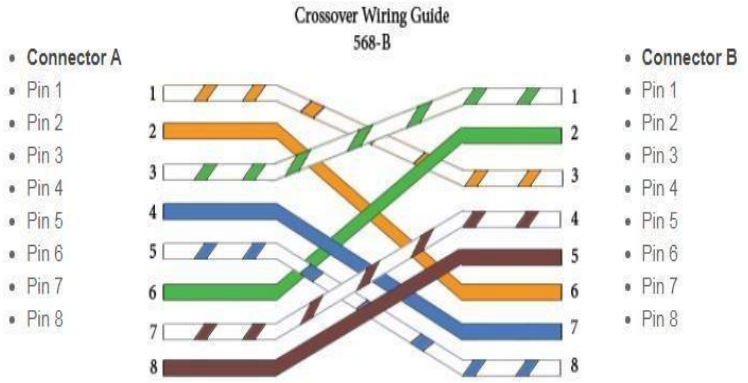
Crossover Wired Cables

Crossover wired cables (commonly called crossover cables) are very much like Straight-Through cables with the exception that TX and RX lines are crossed (they are at opposite positions on either end of the cable). Using the 568-B standard as an example below, you will see that Pin 1 on connector A goes to Pin 3 on connector B. Pin 2 on connector A goes to Pin 6 on connector B, etc.

Crossover cables are most commonly used to connect two hosts directly. Examples would be connecting a computer directly to another computer, connecting a switch directly to another switch, or connecting a router to a router.

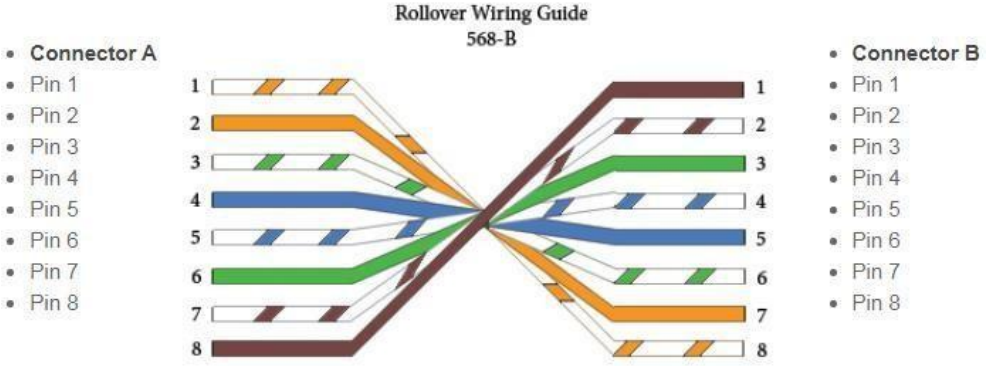
To summarize, crossover cables directly connect the following devices on a LAN:

- Switch to switch
- Switch to hub
- Hub to hub
- Computer to computer
- Computer to a router Ethernet port
- Router to router Ethernet port connection



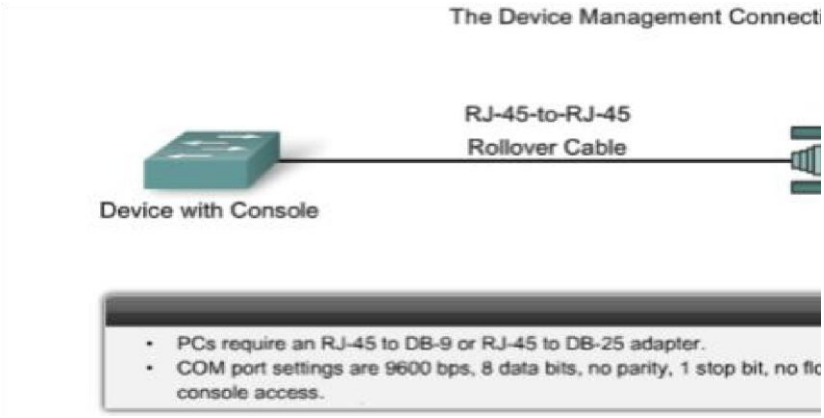
Rollover Wired Cables

Rollover wired cables, most commonly called rollover cables, have opposite Pin assignments on each end of the cable or, in other words, it is "rolled over." Pin 1 of connector A would be connected to Pin 8 of connector B. Pin 2 of connector A would be connected to Pin 7 of connector B, and so on. Rollover cables, sometimes referred to as Yost cables, are most commonly used to connect to a device's console port to make programming changes to the device. Unlike crossover and straight-wired cables, rollover cables are not intended to carry data but instead create an interface with the device.



Console Cables (RJ-45 to DB-9 Female).This cable is also known as Management Cable

The connection to the console is made by plugging the DB -9 connector into an available EIA/TIA 232 serial port on the computer. It is important to remember that if there is more than one serial port, note which port number is being used for the console connection. Once the serial connection to the computer is made, connect the RJ -45 end of the cable directly into the console interface on the router.



Video Reference:
Refer the following videos:

Categories of Cables: https://www.youtube.com/watch?v=_NX99ad2FUA

Crimpling :https://www.youtube.com/watch?v=8qTS2BiRZzU

Fabrication of Network Cables:https://youtu.be/sEc9oUYJZjQ?si=OSLuCj6nP7HQekXf **Answer the following VIVA Questions:**

1. Transmission media are directly controlled by _____**Physical**_____ Layer.

2. What are the three major classes of Guided Media?

Ans:

Twisted Pair Cable: Insulated copper wires twisted together, used in Ethernet and telecommunication. **Coaxial Cable**: Central conductor with insulating layers, used in cable TV and broadband. **Fiber Optic Cable**: Transmits data using light signals, ideal for high-speed and longdistance communication.

3. Why Cladding is used in Fiber Optics?

Ans:

- It ensures total internal reflection, keeping light signals confined within the core.
- Minimizes dispersion and loss by maintaining the light's pathway.
- Provides mechanical protection and preserves the integrity of the core.
- Improves transmission efficiency and reduces interference.

4. List the Categories of UTP cables.

Ans:

Category 1 (Cat 1): Used for voice communication (e.g., telephone lines).
Category 2 (Cat 2): Supports data up to 4 Mbps (obsolete).
Category 3 (Cat 3): Used in 10 Mbps Ethernet networks.
Category 4 (Cat 4): Supports data up to 16 Mbps (Token Ring networks).
Category 5 (Cat 5): Used in 100 Mbps Ethernet and 1 Gbps networks.
Category 5e (Cat 5e): Enhanced Cat 5 for reduced crosstalk; supports 1 Gbps.
Category 6 (Cat 6): Supports 10 Gbps over shorter distances, with improved performance.

Category 6a (Cat 6a): Augmented Cat 6, supports 10 Gbps over longer distances.
Category 7 (Cat 7): Shielded for higher performance; supports 10 Gbps. Category 8 (Cat 8): Designed for 25/40 Gbps data centers.

5. Mention the cause of attenuation and how will you measure it.

Ans: Attenuation is caused by the reduction in signal strength during transmission due to:

- Loss of energy as the signal interacts with the medium.
- Dispersion of signal energy due to imperfections in the medium.
- Signal leakage caused by bends in the transmission medium.
- External electromagnetic noise affecting the signal.

Measurement:

$$\text{Attenuation (dB)} = 10 \times \log_{10}(P_{output}/P_{input})$$

6. What are the advantages of Fiber Optics?

Ans:

- High Bandwidth
- Long-Distance Transmission
- Immunity to Electromagnetic Interference
- Security
- Lightweight and Durable

7. What is meant by LOS?

Ans: Line of Sight (LOS) refers to a direct, unobstructed path between the transmitting and receiving antennas in a communication system.

- Essential for high-frequency signals like microwaves and infrared.
- Obstructions like buildings, trees, or terrain can disrupt LOS communication.
- Commonly used in satellite, radio, and point-to-point wireless systems.

8. Mention the modes of propagation in unguided medium.

Ans:

- Ground Wave Propagation
- Sky Wave Propagation
- Space Wave Propagation

9. List out the connectors used in guided medium.

Ans:

- Twisted Pair Cable Connectors
- Coaxial Cable Connectors
- Fiber Optic Cable Connectors

10. Where you will use Straight through cable and Cross over cable?

Straight-Through Cable

- Connects a computer (or any device) to a network switch or router.
- Connects a router to a modem for internet access.
- For connecting a network switch to a hub.

Cross-Over Cable

- Directly connects two computers without a hub or switch.
- For connecting two switches together.
- Used when directly connecting two routers. For connecting two hubs directly.

Rubrics for Experiment Assessment:

Rubrics	**Good**	**Normal**	**Poor**	**Marks**
Creation of Topology (4)	Created the topology, Identify the proper devices and making the connections **(4)**	Created the topology, Identify the proper devices, making the connections But missing some features **(3)**	Created wrong topology, Failed to Identify the proper devices and making connections **(1)**	
Verify the connectivity (4)	Verified the connectivity in all the levels **(4)**	Verified the connectivity at some levels (only some nodes) **(2)**	Verified the connectivity is not done. **(1)**	
Timely Completion (2)	Completed the lab before the allotted time **(2)**	Completed the lab after the deadline **(1)**	Did not submitted before grading **(0)**	
Total				

RESULT:

Thus the different types of Network cables and the implementation the Crossover wired and Straight through cable using Crimping Tool was completed successfully.

Ex. No:	**4**
Name of the Experiment	**Study of Network Topologies**
Date	

Objective(s):

To design and implement network topologies using Cisco Packet Tracer

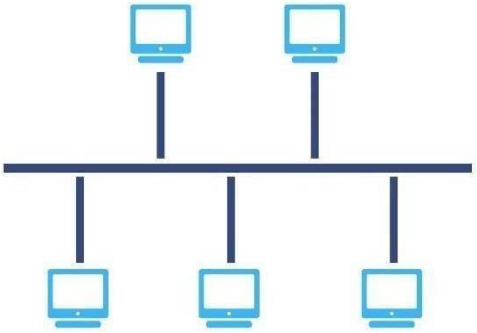
Introduction:

Network topology is the geometric representation of relationship of all the links connecting the devices or nodes. Network topology represent in two ways one is physical topology that define the way in which a network is physically laid out and other one is logical topology that defines how data actually flow through the network. In this lab, we will discuss how to design bus, star and mesh topology network and provide interfacing and simulation between end points using packet tracer software.

Theoretical Background:

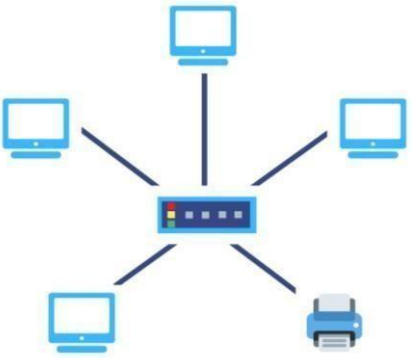
Bus Topology

In local area network, it is a single network cable runs in the building or campus and all nodes are connected along with this communication line with two endpoints called the bus or backbone. In other words, it is a multipoint data communication circuit that is easily control data flow between the computers because this configuration allows all stations to receive every transmission over the network. For bus topology we build network using three generic pc which are serially connected with three switches using copper straight through cable and switches are interconnected using copper cross over cable.

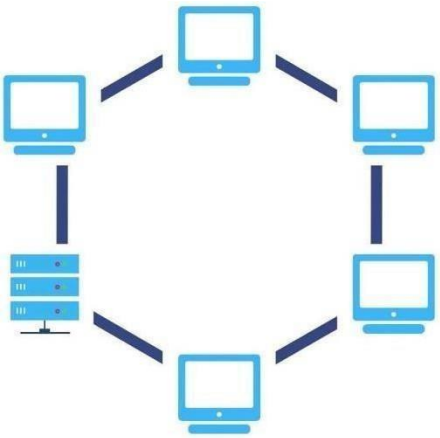


Star Topology

In star topology, all the cables run from the computers to a central location where they are all connected by a device called a hub. It is a concentrated network, where the end points are directly reachable from a central location when network is expanded. Ethernet 10 base T is a popular network based on the star topology. For star topology we build network using five generic pc which are centrally connected to single switch 2950-24 using copper straight through cable.



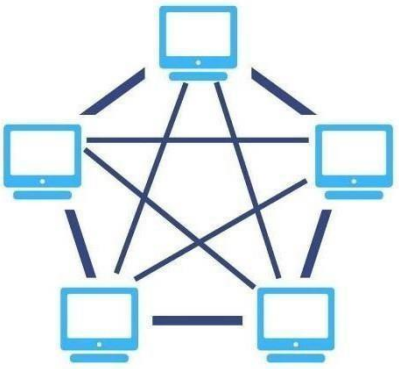
RING TOPOLOGY



As we mentioned earlier, the ring topology is similar to a daisy chain topology but with the loop closed so that the nodes are arranged in a ring or circle. Each node has exactly two peers and the data travels in one direction passing through each intermediate node on the ring until it reaches the destination node. Data can be made to pass in both directions by adding a second connection between the network nodes, creating a dual ring topology.

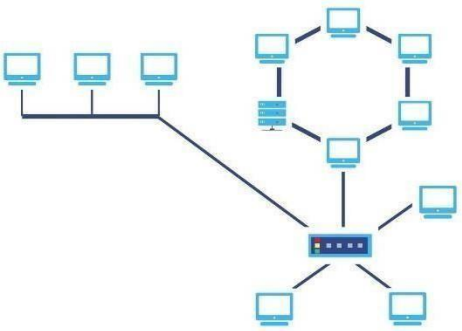
In a ring topology, an electrical "token" circulates around the network. Any node that wants to transmit data has to wait until it has possession of the token.

MESH TOPOLOGY



In mesh topology every device has a dedicated point to point link to every other device. The term dedicated stand for link carries traffic only between four devices it connects. It is a well-connected topology; in this, every node has a connection to every other node in the network. The cable requirements are high and it can include multiple topologies. Failure in one of the computers does not cause the network to break down, as they have alternative paths to other computers star topology, all the cables run from the computers to a central location.

Hybrid Topology



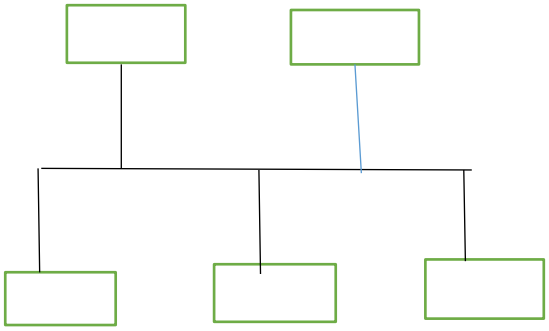
Hybrid topology combines two or more topologies. You can see in the above architecture in such a manner that the resulting network does not exhibit one of the standard topologies.

For example, as you can see in the above image that in an office in one department, Star and P2P topology is used. A hybrid topology is always produced when two different basic network topologies are connected.

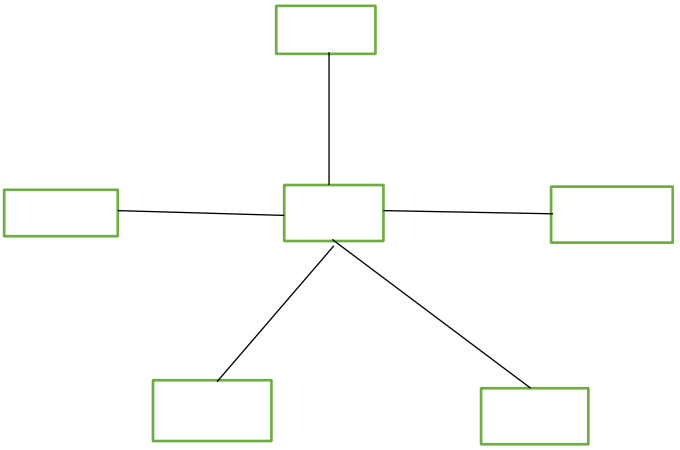
Design the above mentioned topologies and verify the connectivity.

1. **Device Requirements:**
   1.Switch
   2.Laptop
   3.PC's
   4.Copper cross over wire

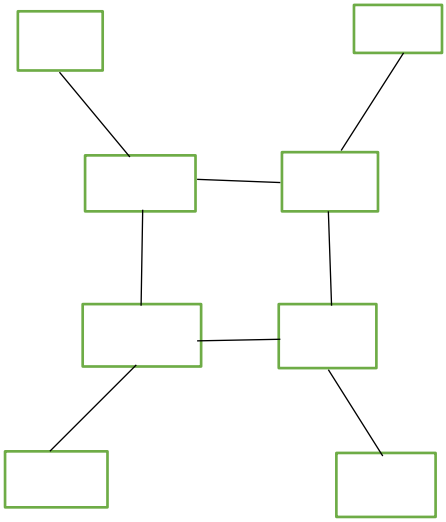
2. **Network Diagram for your experiment (draw the diagram either hand drawing/ms paint or any other drawing tools)**



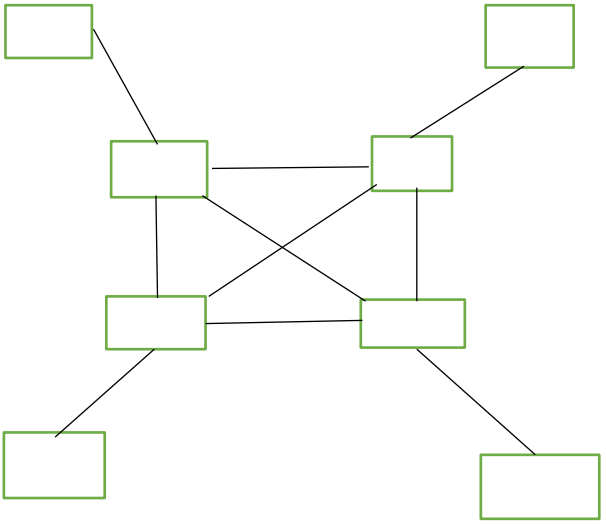
Bus topology



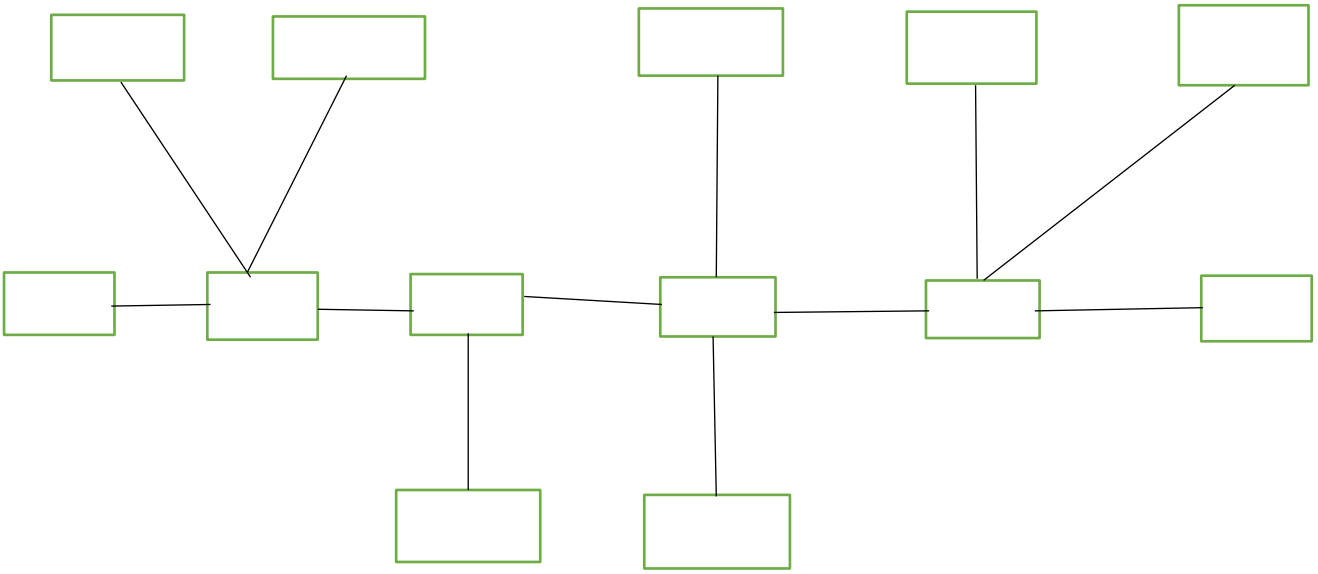
Star topology



Ring topology



Mesh topology



Hybrid topology

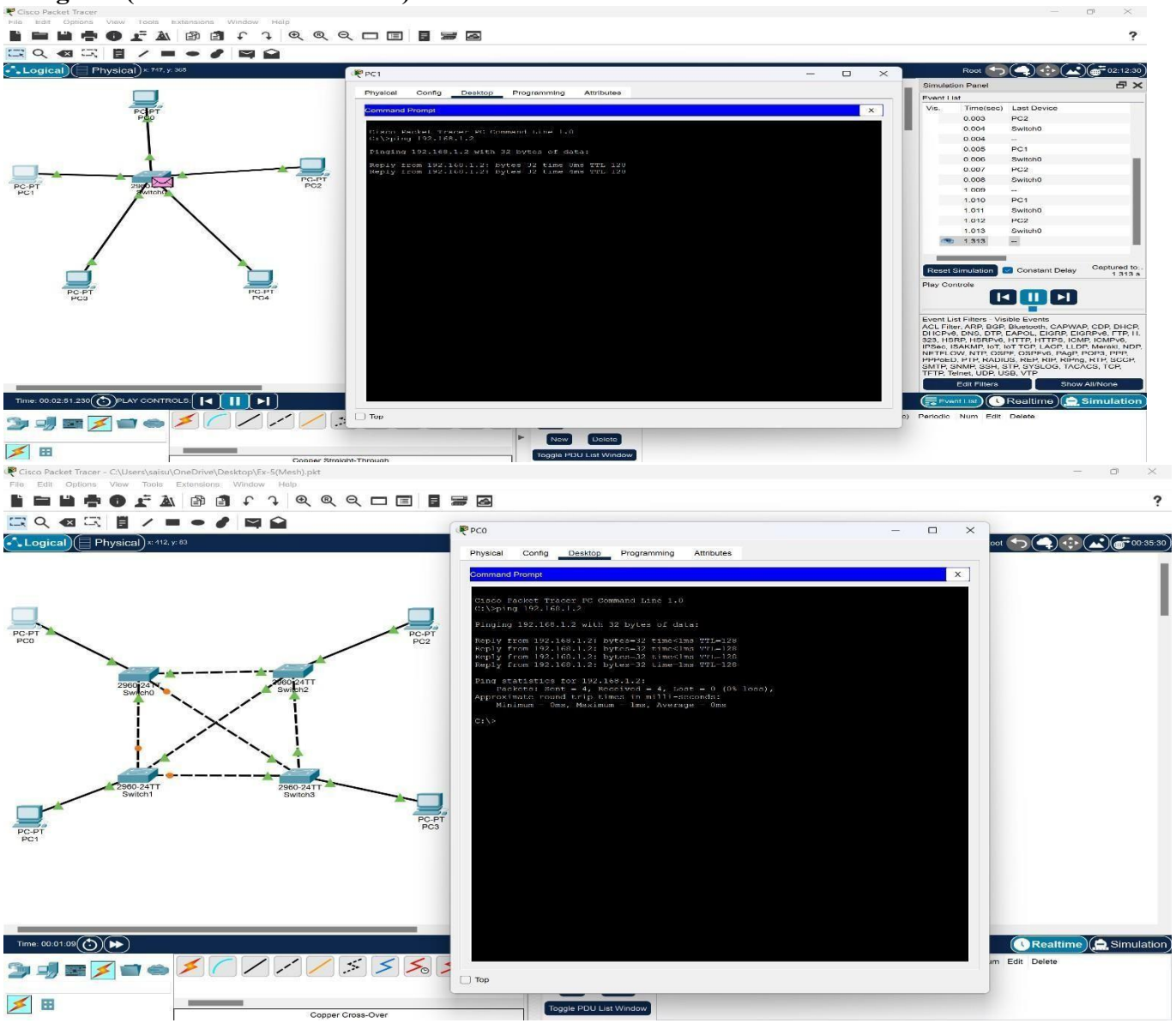
3. **Configuration details:**

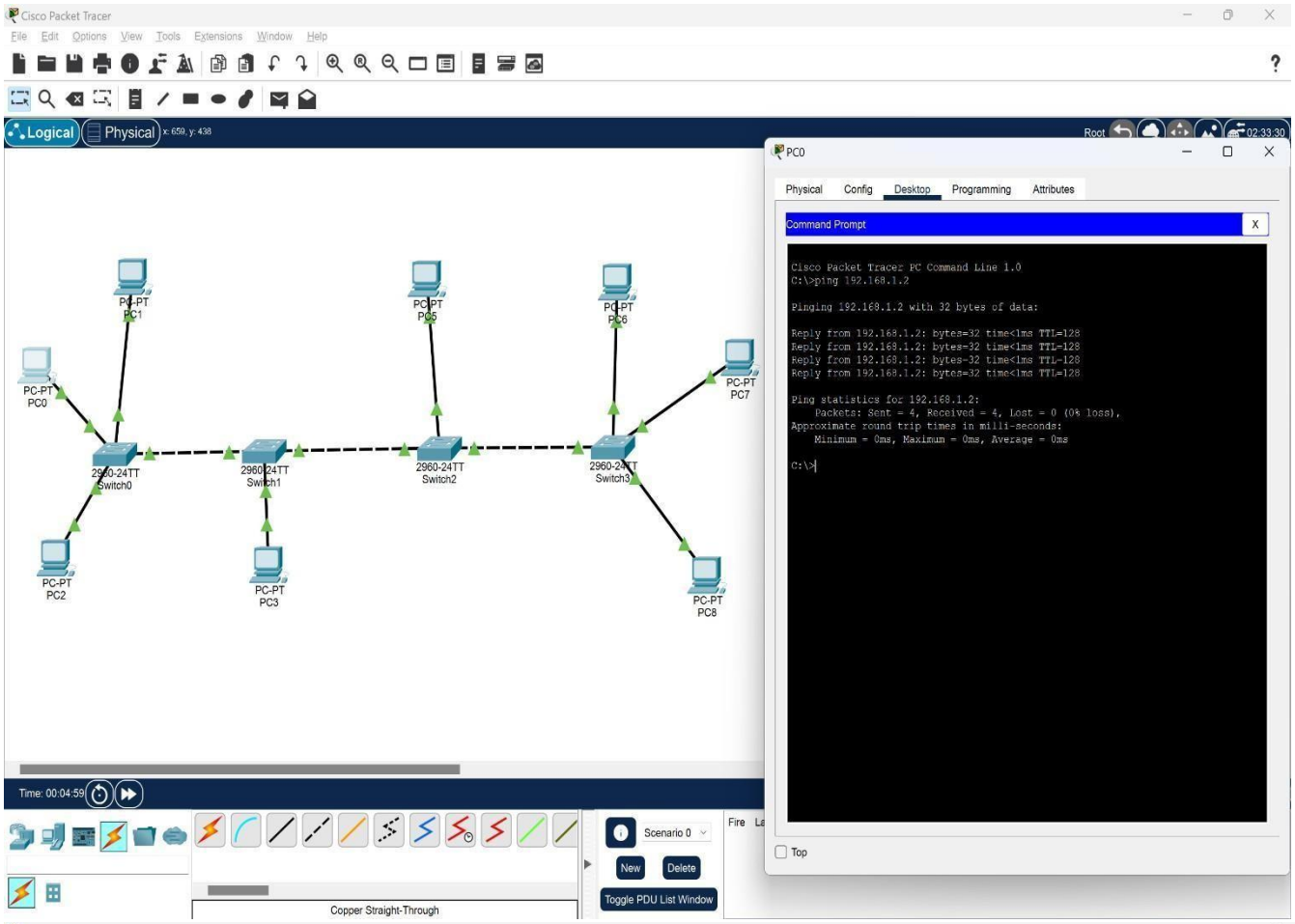
Device Name	Interface Name	IP Address	Subnet mask
PC0	Fa0/0	192.168.1.1	255.0.0.0
PC1	Fa0/1	192.168.1.2	255.0.0.0
PC2	Fa0/2	192.168.1.3	255.0.0.0
PC3	Fa0/3	192.168.1.4	255.0.0.0
PC4	Fa0/4	192.168.1.4	255.0.0.0
Laptop	Fa0/0	192.168.1.1	255.0.0.0

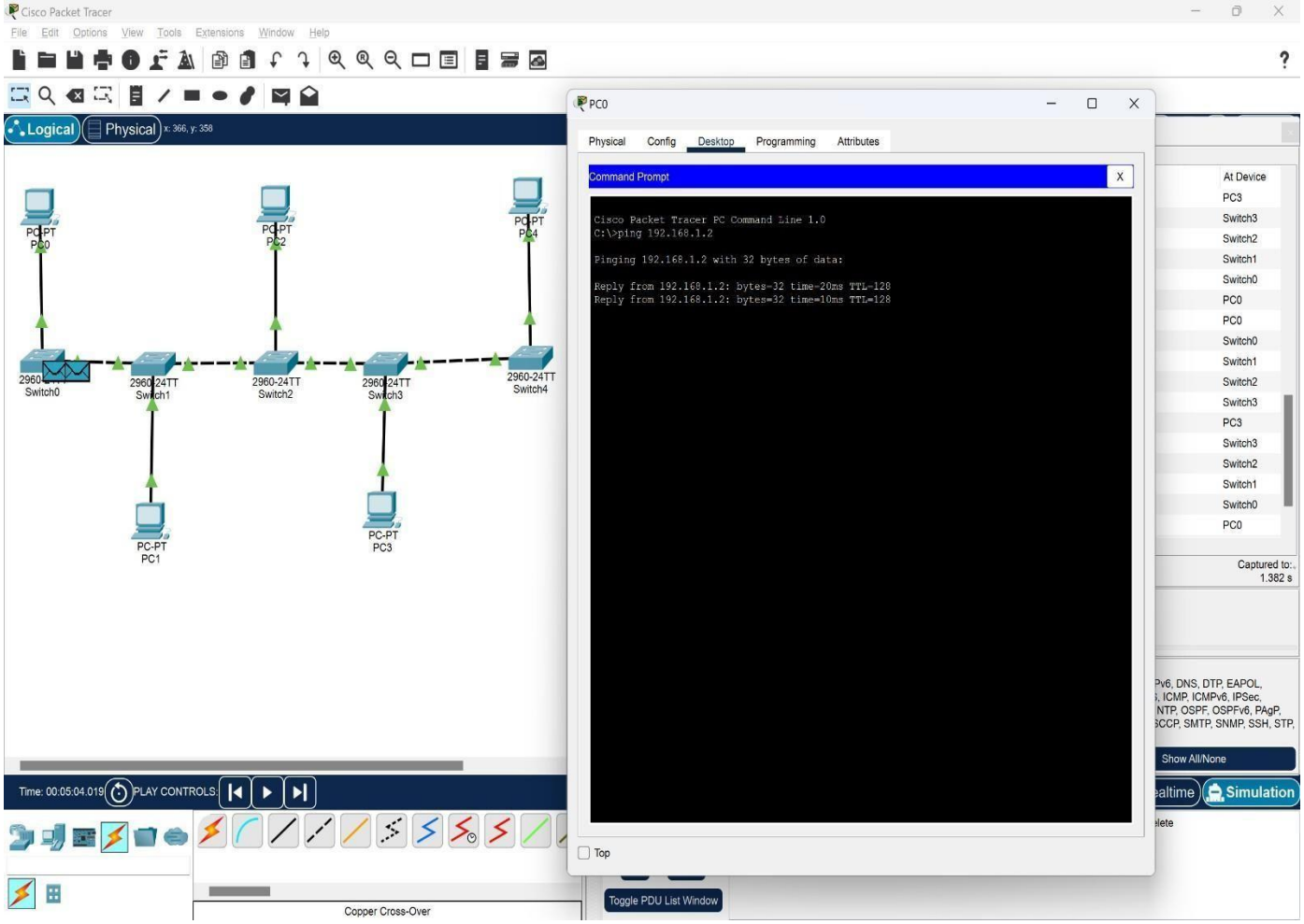
4. **Commands used in each of the diagram (if any):**

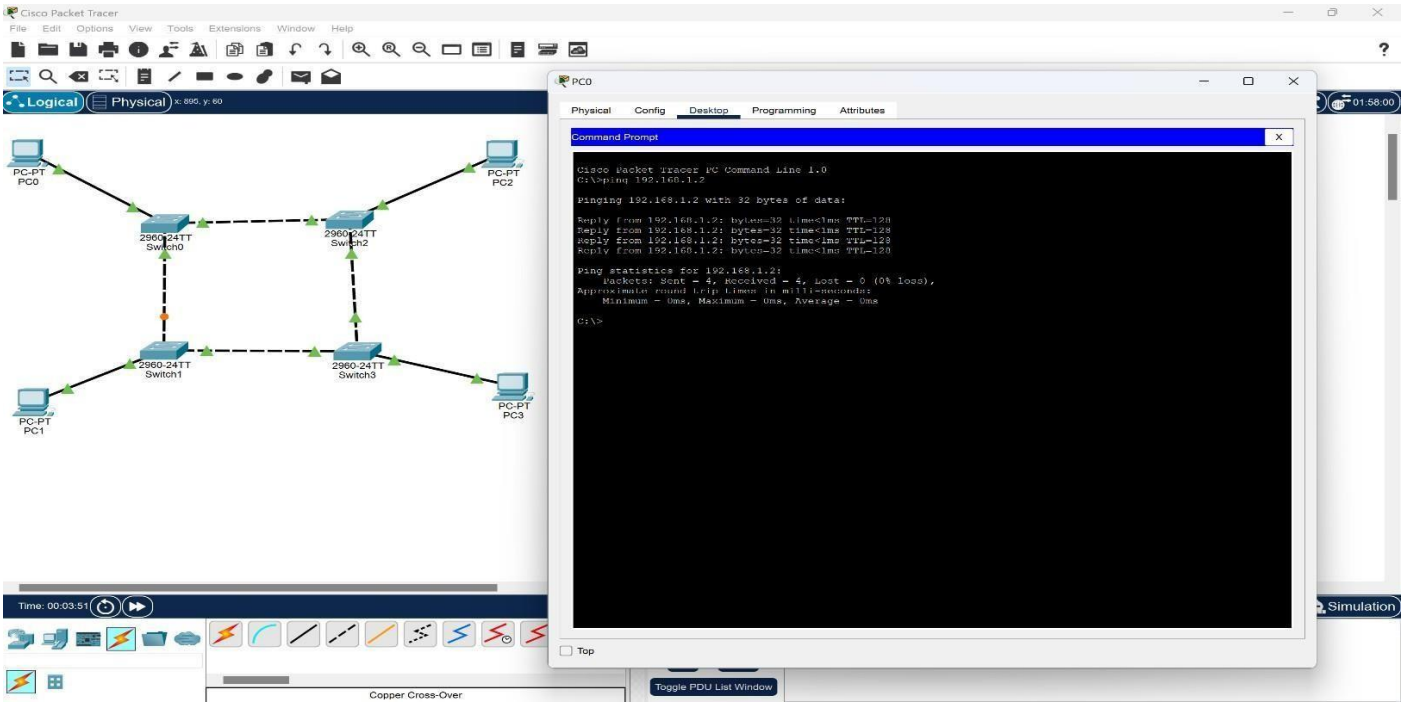
   - **IP config/all**
   - **Ping**

5. **Output Diagram (Minimum 3 screenshot):**




Cisco Packet Tracer
File Edit Options View Tools Extensions Window Help
Logical
Physical
PC0
Physical Config Desktop Programming Attributes
Command Prompt
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.1.2
Pinging 192.168.1.2 with 32 bytes of data:
Reply from 192.168.1.2: bytes=32 time<1ms TTL=128
Reply from 192.168.1.2: bytes=32 time<1ms TTL=128
Reply from 192.168.1.2: bytes=32 time<1ms TTL=128
Reply from 192.168.1.2: bytes=32 time<1ms TTL=128
Ping statistics for 192.168.1.2:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\>
PC-PT PC0
PC-PT PC1
PC-PT PC2
PC-PT PC3
PC-PT PC5
PC-PT PC6
PC-PT PC7
PC-PT PC8
2960-24TT Switch0
2960-24TT Switch1
2960-24TT Switch2
2960-24TT Switch3
Time: 00:04:59
Scenario 0
New
Delete
Toggle PDU List Window
Copper Straight-Through
Top


Cisco Packet Tracer
File Edit Options View Tools Extensions Window Help
Logical
Physical
PC0
Physical Config Desktop Programming Attributes
Command Prompt
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.1.2
Pinging 192.168.1.2 with 32 bytes of data:
Reply from 192.168.1.2: bytes=32 time=20ms TTL=128
Reply from 192.168.1.2: bytes=32 time=10ms TTL=128
PC-PT PC0
PC-PT PC1
PC-PT PC2
PC-PT PC3
PC-PT PC4
2960-24TT Switch0
2960-24TT Switch1
2960-24TT Switch2
2960-24TT Switch3
2960-24TT Switch4
Time: 00:05:04.019
PLAY CONTROLS:
Toggle PDU List Window
Copper Cross-Over
Top
At Device
Captured to:
1.382 s
Show All/None
Simulation



CONCLUSION (provide conclusion about this experiment):

Successfully created and executed network topologies (star,ring,bus,mesh and hybrid) using cisco packet tracer

Rubrics for Experiment Assessment:

Rubrics	Good	Normal	Poor	Marks
Creation of Topology (4)	Created the topology, Identify the proper devices and making the connections **(4)**	Created the topology, Identify the proper devices, making the connections But missing some features **(3)**	Created wrong topology, Failed to Identify the proper devices and making connections **(1)**	
Verify the connectivity (4)	Verified the connectivity in all the levels **(4)**	Verified the connectivity at some levels (only some nodes) **(2)**	Verified the connectivity is not done. **(1)**	
Timely Completion (2)	Completed the lab before the allotted time **(2)**	Completed the lab after the deadline **(1)**	Did not submitted before grading **(0)**	
			Total	

Ex. No:	**5**
Nameofthe Experiment	**Configuration of intra VLAN network**
Date	9-01-2025

Objective(s):

To design and implement Intra VLAN using switch configuration

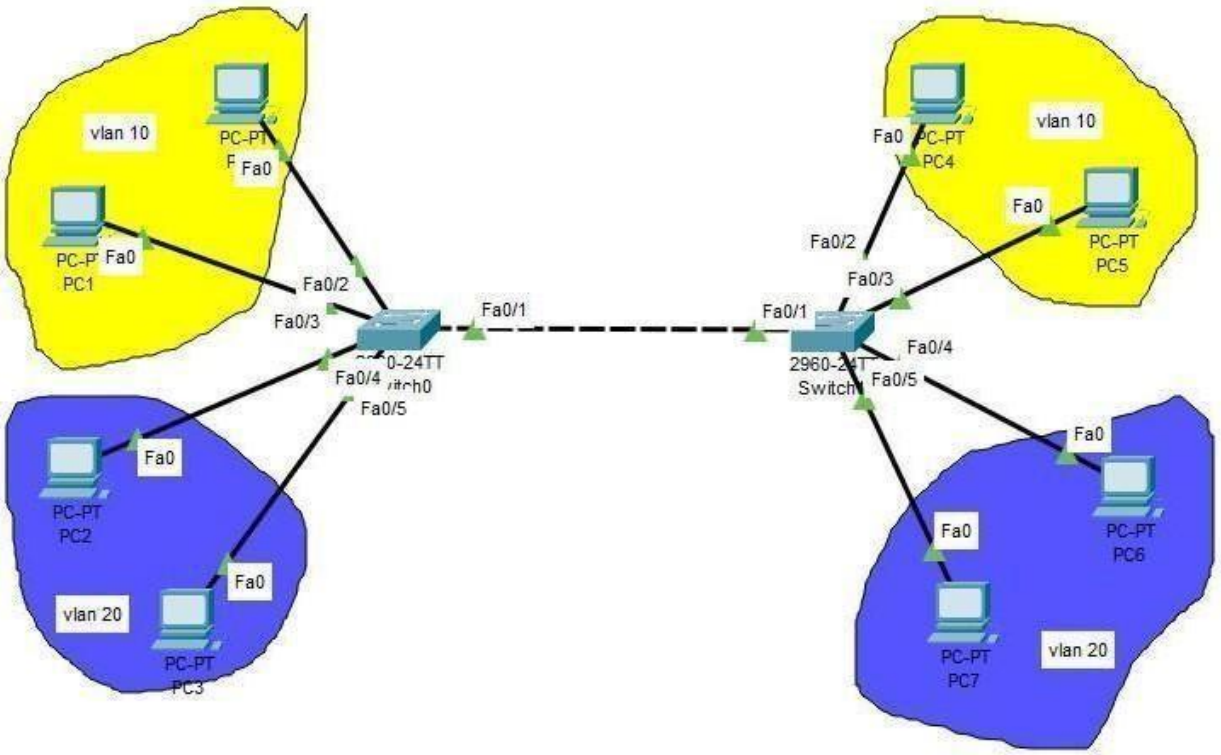
Introduction:

A VLAN is a group of devices on one or more LANs that are configured to communicate as if they were attached to the same wire, when in fact they are located on a number of different LAN segments. Because VLANs are based on logical instead of physical connections, they are extremely flexible.

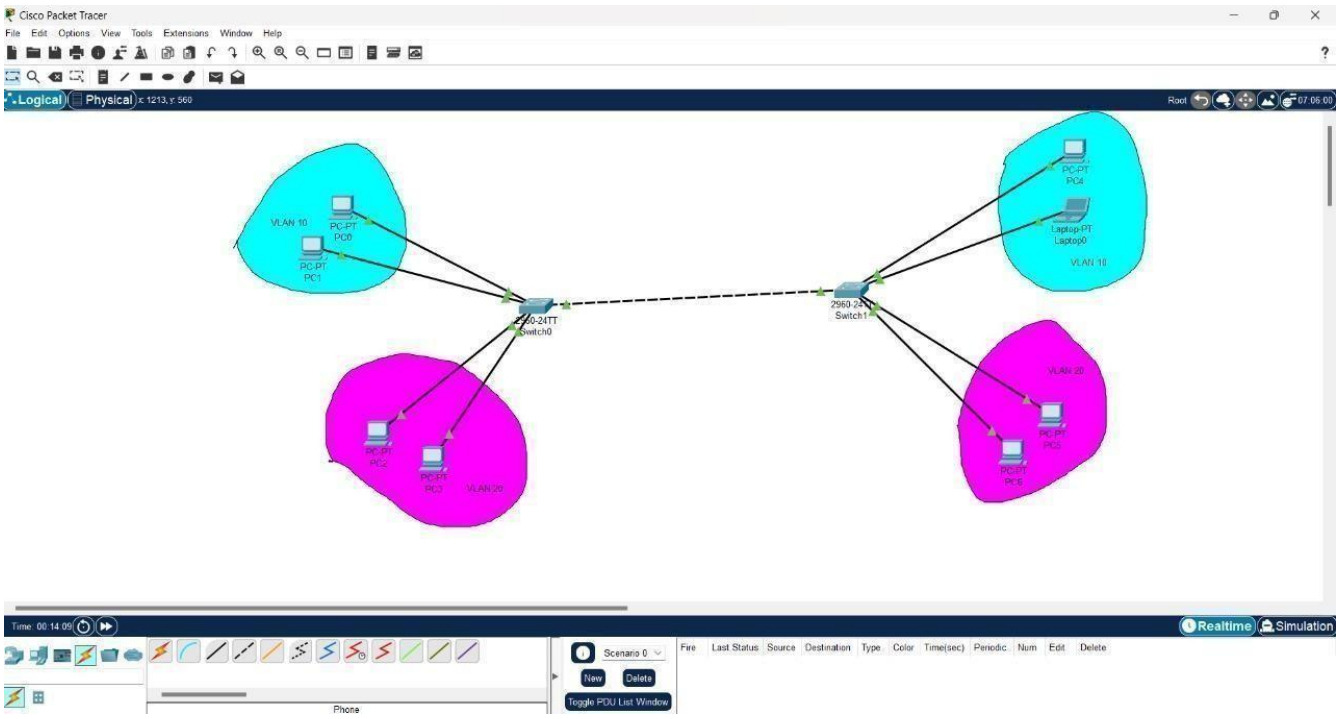
VLANs define broadcast domains in a Layer 2 network. A broadcast domain is the set of all devices that will receive broadcast frames originating from any device within the set. Broadcast domains are typically bounded by routers because routers do not forward broadcast frames. Layer 2 switches create broadcast domains based on the configuration of the switch. Switches are multiport bridges that allow you to create multiple broadcast domains. Each broadcast domain is like a distinct virtual bridge within a switch.

Design the above-mentioned topologies and verify the connectivity.

1. **DeviceRequirements:**
   1. PC0,PC01,PC02,PC03,PC04,PC05,PC06.
   2. Switch0,Switch1.
   3. Laptop0.
   4. Wire(CopperStraight-Through)
   5. Wire(CopperCrossOver)
2. **NetworkDiagramforyourexperiment(drawthediagrameitherhanddrawing/mspaintoranyother drawing tools)**



3. Network Diagram(Packet tracer diagram before configuration):



4. Configuration details:

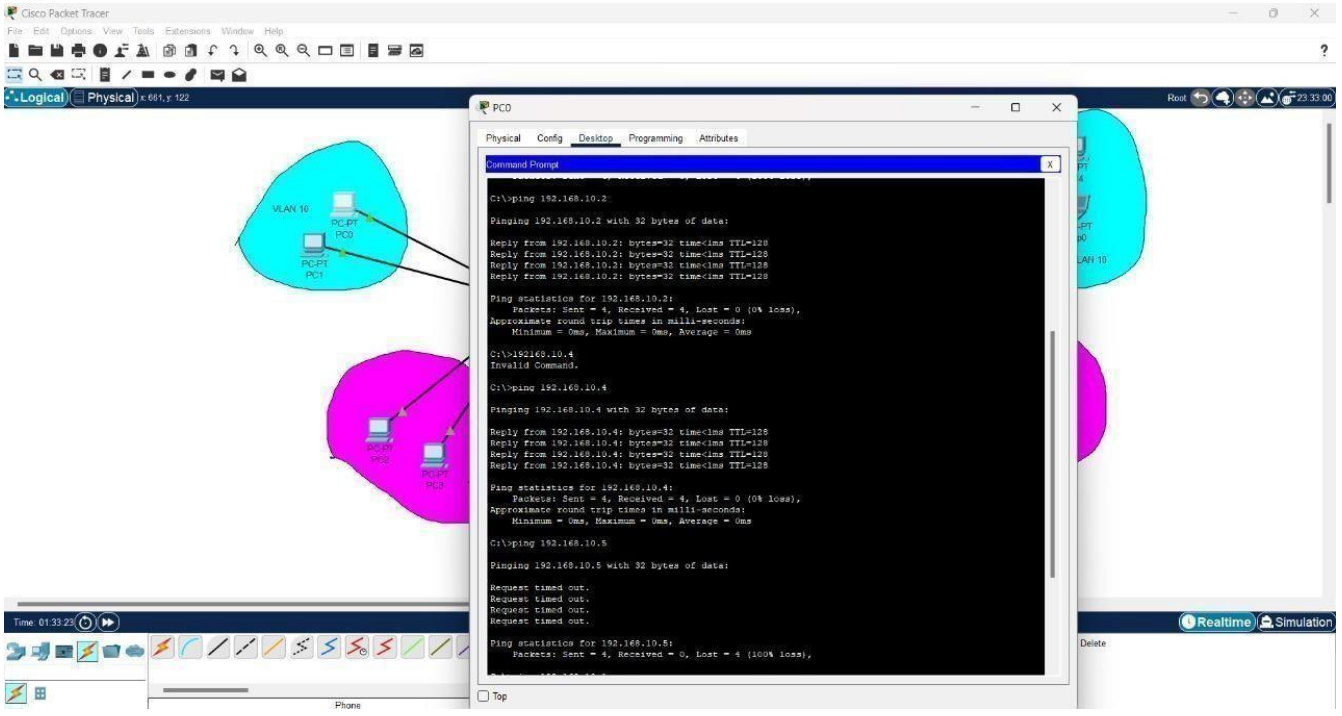
Device Name	Interface Name	IP Address	Subnetmask
PC0	Fa0	192.168.10.1	255.255.255.0
PC1	Fa0	192.168.10.2	255.255.255.0

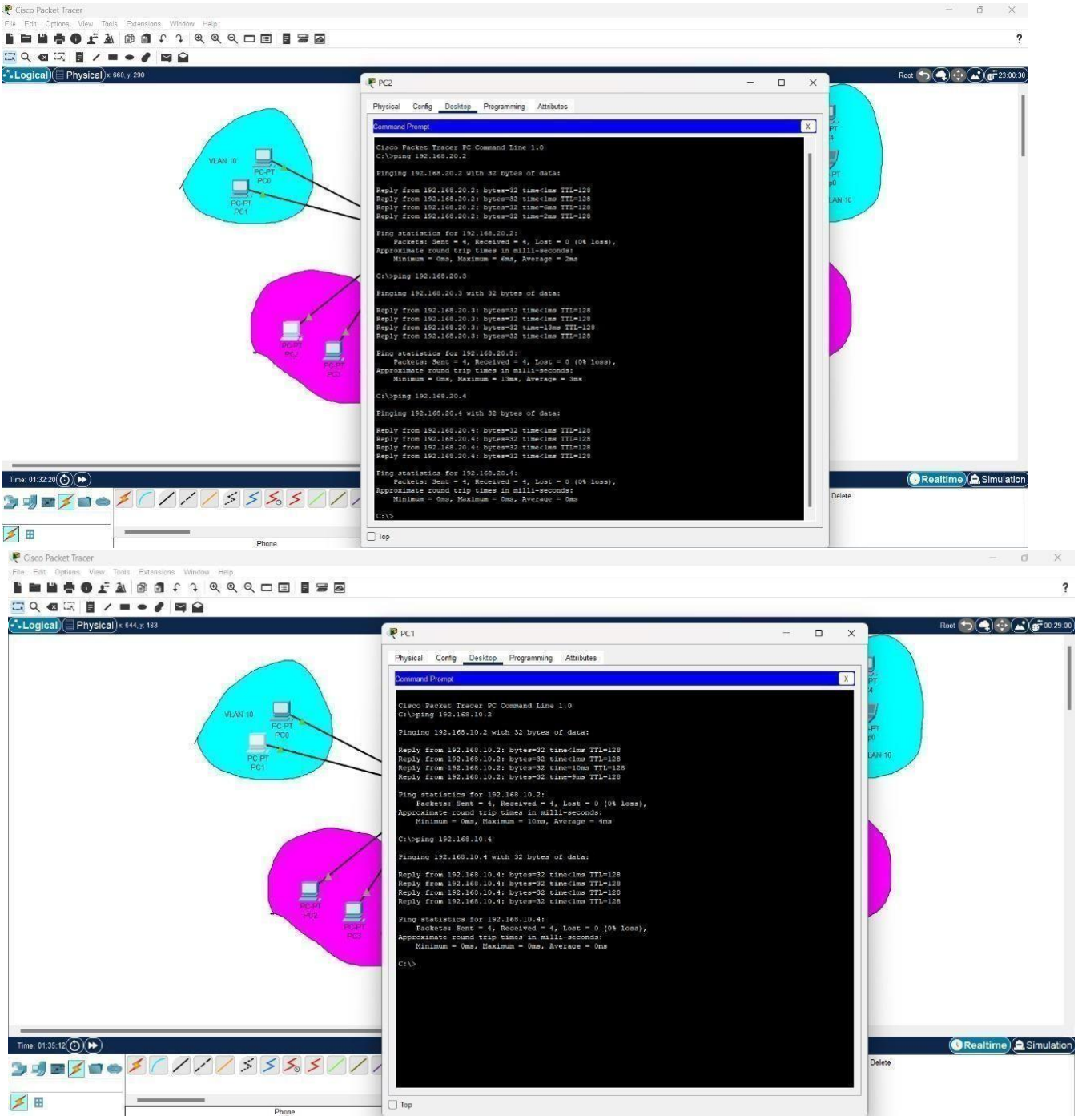
PC2	Fa0	192.168.20.2	255.255.255.0
PC3	Fa0	192.168.20.3	255.255.255.0
PC4	Fa0	192.168.10.3	255.255.255.0
PC5	Fa0	192.168.20.5	255.255.255.0
PC6	Fa0	192.168.20.4	255.255.255.0
Laptop0	Fa0	192.168.10.4	255.255.255.0
Switch0	Fa03		
Switch1	Fa01		

Describe step by step configuration steps properly

1. **Create VLANs**
2. **Configure interfaces**
3. **Configure trunking**

5. Output Diagram





CONCLUSION:

The study concludes that Cisco Packet Tracer is a robust and user-friendly tool for configuration of Intra VLAN networks. It plays a critical role in preparing individuals for industry certifications and real-world network management challenges. Its capabilities, combined with its accessibility, make it a cornerstone in the toolkit of network engineers and educators.

Rubrics for Experiment Assessment:

Rubrics	Good	Normal	Poor	Marks
Creation of Topology(4)	Created the topology, Identify the proper devices and making the connections **(4)**	Created the topology, Identify the proper devices, making the connections But missing some features **(3)**	Created wrong topology, Failed to Identify the proper devices and making connections **(1)**	
Verify the connectivity (4)	Verified the connectivity in all the levels **(4)**	Verified the connectivity at some levels (only some nodes) **(2)**	Verified the connectivity is notdone.**(1)**	
Timely Completion (2)	Completed the lab before the allotted time **(2)**	Completedthelabafterthe deadline **(1)**	Did not submitted before grading **(0)**	
			Total	

Result : Thus, the design and configuration of the Intra VLAN network has been done successfully.

Ex.No:	**6a**
Name of the Experiment	**Configuration of Inter VLAN network using L3 switch**
Date:	

Objective(s):

To design and implement Inter VLAN using switch configuration

Introduction:

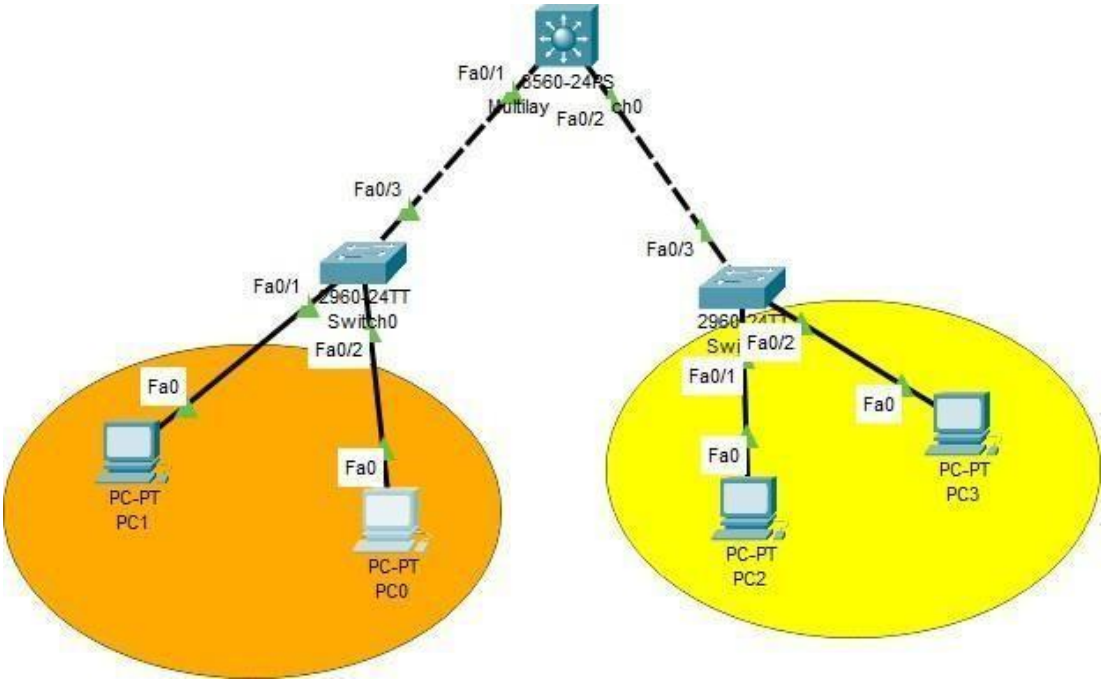
Normally, Routers are used to divide the broadcast domain and switches (at layer 2) Operate in a single broadcast domain but Switches can also divide the broadcast domain by using the concept of **VLAN (Virtual LAN)**.

VLAN is the logical grouping of devices in the same or different broadcast domains. By default, all the switch ports are in VLAN 1. As the single broadcast domain is divided into multiple broadcast domains, Routers or layer 3 switches are used for intercommunication between the different VLANs.The process of intercommunication of the different Vlans is known as Inter Vlan Routing (IVR).
Suppose we have made 2 logical groups of devices (VLAN) named sales and finance. If a device in the sales department wants to communicate with a device in the finance department, inter-VLAN routing has to be performed. These can be performed by either router or layer 3 switches.

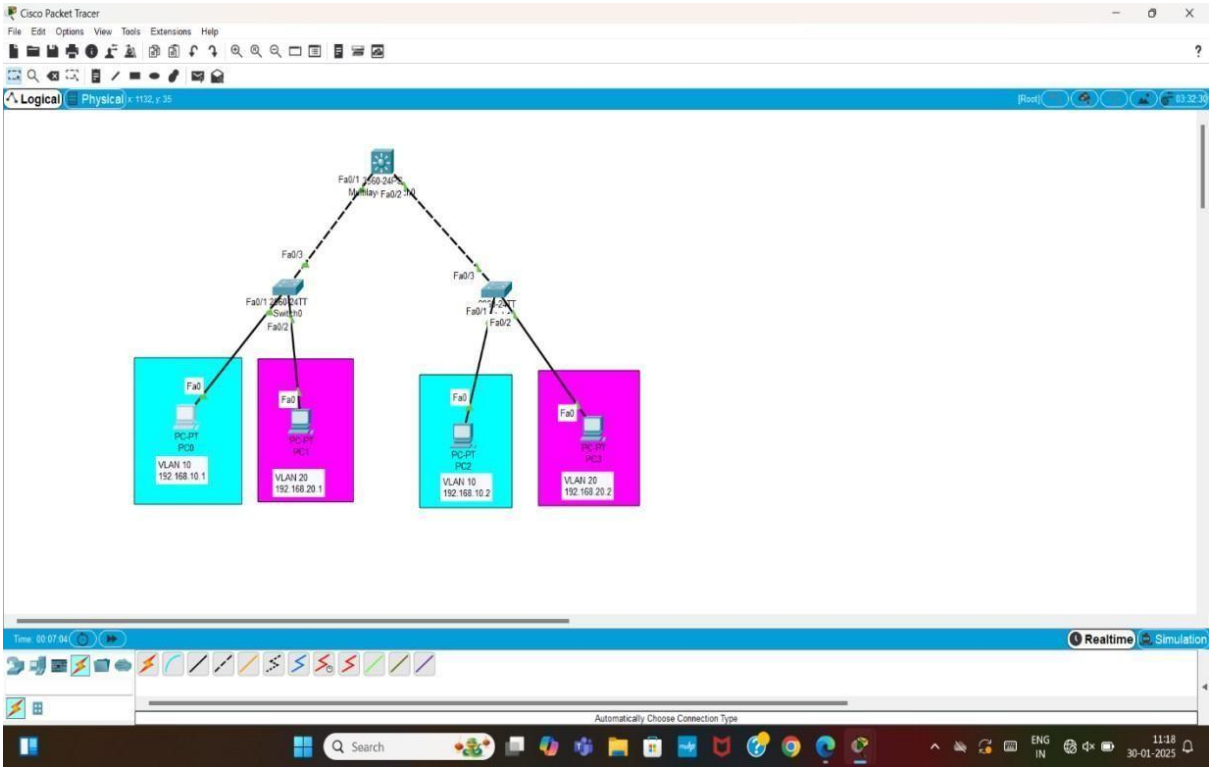
Switch Virtual Interface (SVI): SVI is a logical interface on a multilayer switch that provides layer 3 processing for packets to all switch ports associated with that VLAN. A single SVI can be created for a VLAN. SVI on the layer 3 switch provides both management and routing services while SVI on layer 2 switch provides only management services like creating VLANs or telnet/SSH services.

Process of Inter Vlan Routing by Layer 3 Switch: The SVI created for the respective VLAN acts as a default gateway for that VLAN just like the sub-interface of the router (in the process of Router On a stick). If the packet is to be delivered to different VLANs i.e inter VLAN Routing is to be performed on the layer 3 switch then first the packet is delivered to the layer 3 switch and then to the destination just like in the process of the router on a stick.

1. **Device Requirements:**
   1. PC0
   2. PC1
   3. PC2
   4. PC3
   5. Switch0
   6. Switch1
   7. Router
2. **Network Diagram for your experiment (draw the diagram either hand drawing/ms paint or any other drawing tools)**



3. **Network Diagram (Packet Tracer diagram before configuration):**



4. **Configuration details:**

Device Name	Interface Name	IP Address	Subnet mask
PC0	fa 0/1	192.168.10.1	255.255.255.0

PC1	fa 0/2	192.168.20.1	255.255.255.0
PC2	fa 0/1	192.168.10.2	255.255.255.0
PC3	fa 0/2	192.168.20.2	255.255.255.0

5. **Describe step by step configuration steps properly (you may copy the commands used in the configuration tab and paste it.)**

SWITCH-0

1.Create VLANS

Switch>enable

Switch#config ter

Enter configuration commands, one per line. End with CNTL/2.

Switch (config)#vlan 10

Switch (config-vlan)#exit Switch

(config)#vlan 20

Switch (config-vlan)#exit

2.Configure Interfaces

Switch (config)#interface fa0/1

Switch (config-if)#switchport mode access

Switch (config-if)#switchport access vlan 10

Switch (config-if)#exit

Switch (config)#interface fa0/2

Switch (config-if)#switchport mode access

Switch (config-if)#switchport access vlan 20 Switch

(config-if)#exit

3.Configure trunking

Switch (config)#interface fa0/3

Switch (config-if)#switchport mode trunk

%LINEPROTO-S-UPDOWN: Line protocol on Interface FastEthernet0/3, changed state to down %LINE PROTO-5-UPDOWN: Line Protocol on Interface FastEthernet0/3. changed state to up.

SWITCH-1

1.Create VLANS

Switch>enable

Switch#config ter

Enter configuration commands, one per line. End with CNTL/2.
Switch (config)#vlan 10
Switch (config-vlan)#exit Switch
(config)#vlan 20
Switch (config-vlan)#exit

2.Configure Interfaces

Switch (config)#interface fa0/1
Switch (config-if)#switchport mode access
Switch (config-if)#switchport access vlan 10
Switch (config-if)#exit
Switch (config)#interface fa0/2
Switch (config-if)#switchport mode access
Switch (config-if)#switchport access vlan 20 Switch
(config-if)#exit

3.Configure trunking

Switch (config)#interface fa0/3
Switch (config-if)#switchport mode trunk
%LINEPROTO-S-UPDOWN: Line protocol on Interface FastEthernet0/3, changed state to down
%LINE PROTO-5-UPDOWN: Line Protocol on Interface FastEthernet0/3. changed state to up.

MULTI LAYER SWITCH 0

1.Create VLANS

Switch>enable
Switch#config ter
Enter configuration commands, one per line. End with CNTL/2.
Switch (config)#vlan 10
Switch (config-vlan)#exit
Switch (config)#vlan 20
Switch (config-vlan)#exit

2.Configure Interfaces

Switch (config)#interface fa0/1

3.Configure trunking

Switch (config-if)#switchport mode trunk

Command rejected: An interface whose trunk, encapsulation is "Auto" Can not be Configured to "trunk" mode

Switch (config-if)#switchport trunk encapsulation dot1q

Switch (config-if)#switchport mode trunk

Switch (config-if)#exit

Switch (config)#interface mode trunk

Switch (config)#exit

Switch (config)#exit

%SYS-5-CONFIG-I: Configured from Console by Console.

Switch (Config)# ip routing.

Switch (Config)#interface vlan 10

%LINK-5-CHANGED: Interface VLANIO, changed state to up

%LINEPROTO-5-UPDOWN: LINE Protocol on Interface vlan10, Changed state to up.

Switch (Config-if)#ip address 192.168.10.100 255.255.255.0.

Switch (config-if)#interface vlan 20

%LINK-5-CHANGED: Interface to up. Wan 20, changed State

%LINE PROTO-5-UPDOWN: LINE Protocol on Interface VLan20, Changed stats to up.

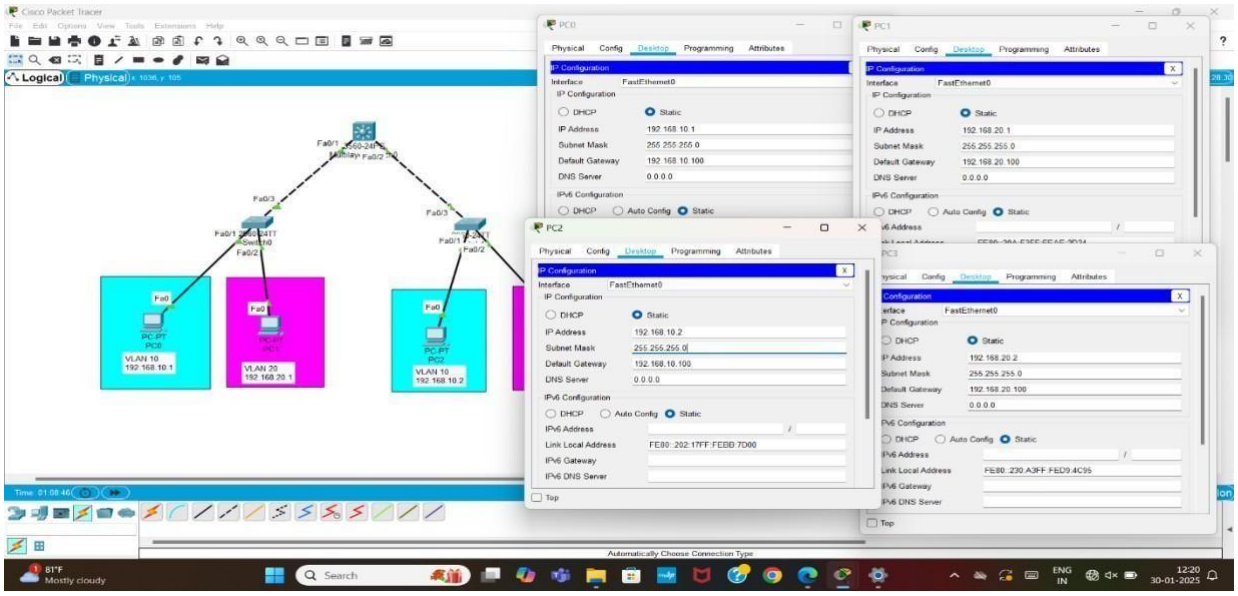
Switch (Config-if)#ip address 192.168.20.100 255.255.255.0

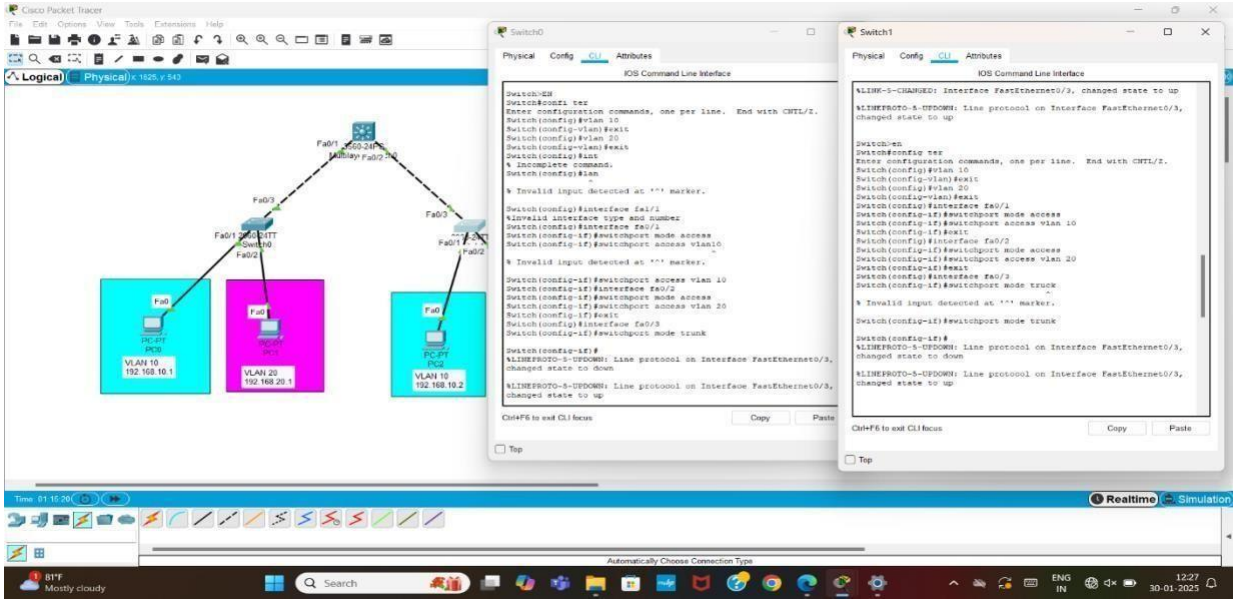
Switch (Config-if)#exit

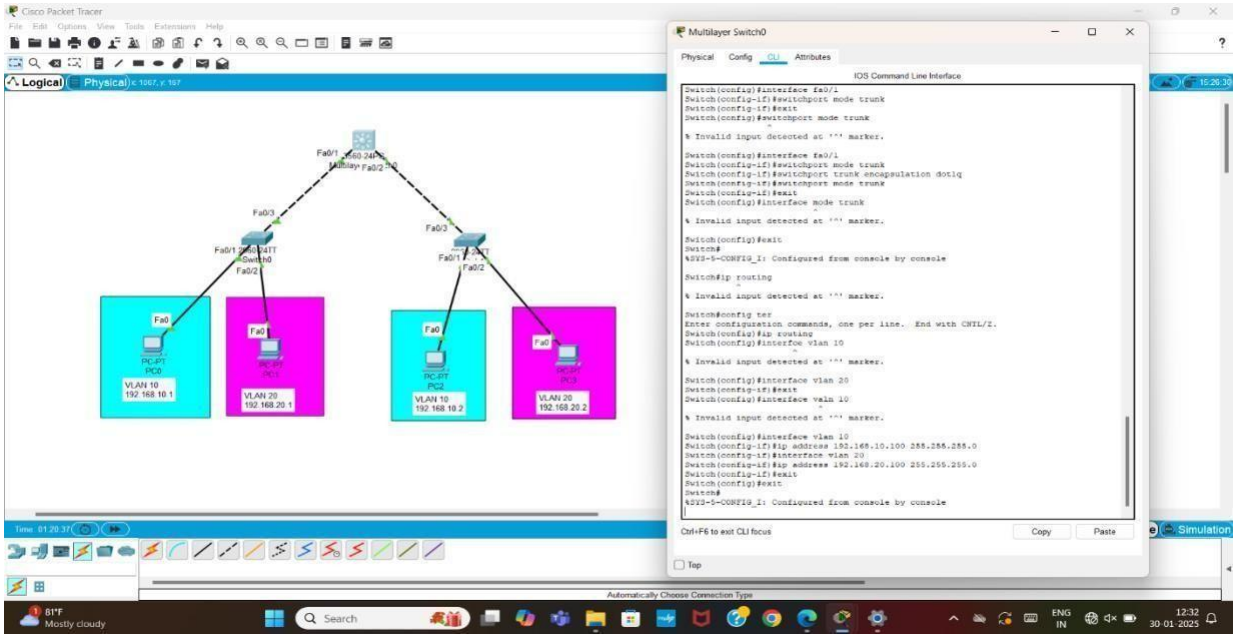
Switch (Config-if)#exit

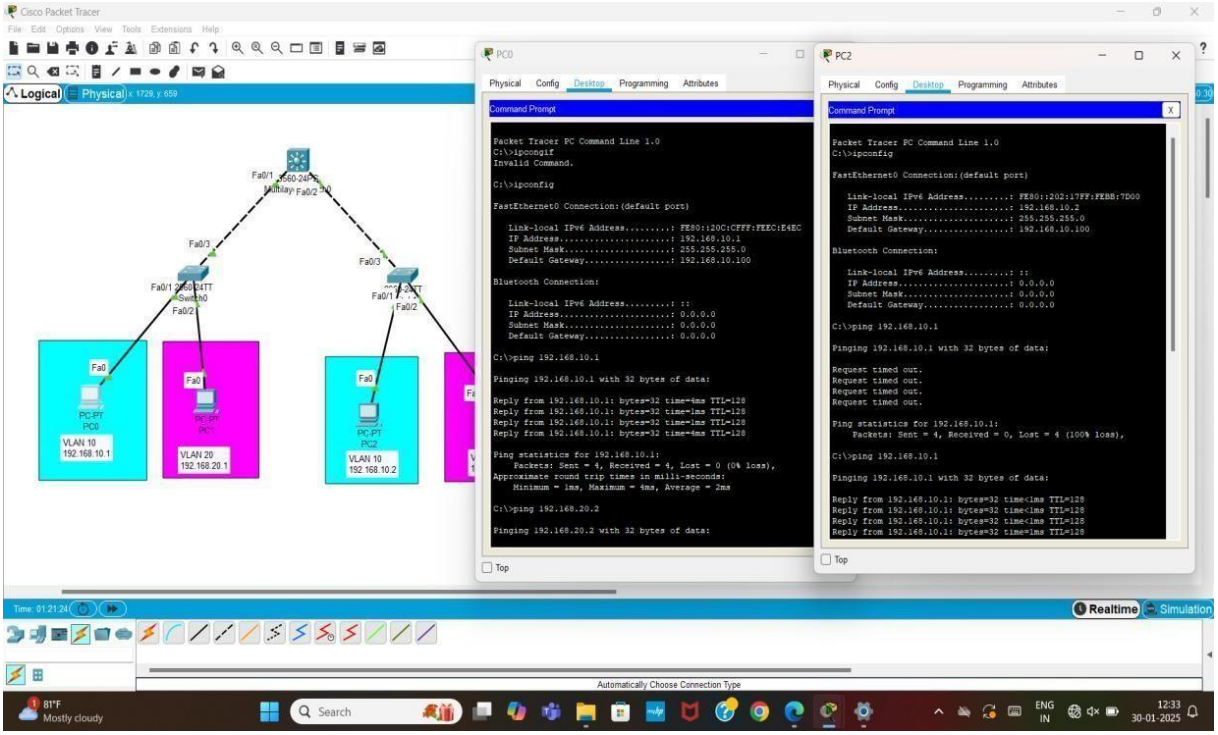
%SYS-5-CONFIG_I: Configured from Console by Console

6. Output Diagram (Minimum 3 screenshot)


Cisco Packet Tracer
PC0
PC1
PC2
PC3
Physical
Config
Desktop
Programming
Attributes
IP Configuration
DHCP
Static
IP Address
Subnet Mask
Default Gateway
DNS Server
IPv6 Configuration
Auto Config
IPv6 Address
Link Local Address
IPv6 Gateway
IPv6 DNS Server
FastEthernet0
192.168.10.1
192.168.10.2
192.168.20.1
192.168.20.2
255.255.255.0
192.168.10.100
192.168.20.100
0.0.0.0
FE80::202:17FF:FEDB:7D00
FE80::230:A3FF:FED9:4C95
Top
Logical
Physical
Automatically Choose Connection Type
81°F
Mostly cloudy
Search
ENG
IN
12:20
30-01-2025


Cisco Packet Tracer
Switch0
Switch1
Physical
Config
CLI
Attributes
IOS Command Line Interface
Switch>en
Switch#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#vlan 10
Switch(config-vlan)#exit
Switch(config)#vlan 20
Switch(config-vlan)#exit
Switch(config)#int
% Incomplete command.
Switch(config)#ian
% Invalid input detected at '^' marker.
Switch(config)#interface fa1/1
%Invalid interface type and number
Switch(config)#interface fa0/1
Switch(config-if)#switchport mode access
Switch(config-if)#switchport access vlan10
% Invalid input detected at '^' marker.
Switch(config-if)#switchport access vlan 10
Switch(config-if)#interface fa0/2
Switch(config-if)#switchport mode access
Switch(config-if)#switchport access vlan 20
Switch(config-if)#exit
Switch(config)#interface fa0/3
Switch(config-if)#switchport mode trunk
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/3, changed state to down
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/3, changed state to up
%LINK-5-CHANGED: Interface FastEthernet0/3, changed state to up
Ctrl+F6 to exit CLI focus
Copy
Paste
Top
Realtime
Simulation
Automatically Choose Connection Type
81°F
Mostly cloudy
Search
ENG
IN
12:27
30-01-2025


Cisco Packet Tracer
Multilayer Switch0
Physical
Config
CLI
Attributes
IOS Command Line Interface
Switch(config)#interface fa0/1
Switch(config-if)#switchport mode trunk
Switch(config-if)#exit
Switch(config)#switchport mode trunk
% Invalid input detected at '^' marker.
Switch(config)#interface fa0/1
Switch(config-if)#switchport mode trunk
Switch(config-if)#switchport trunk encapsulation dot1q
Switch(config-if)#switchport mode trunk
Switch(config-if)#exit
Switch(config)#interfece mode trunk
% Invalid input detected at '^' marker.
Switch(config)#exit
Switch#
%SYS-5-CONFIG_I: Configured from console by console
Switch#ip routing
% Invalid input detected at '^' marker.
Switch#config ter
Enter configuration commands, one per line. End with CNTL/Z.
Switch(config)#ip routing
Switch(config)#interfce vlan 10
% Invalid input detected at '^' marker.
Switch(config)#interface vlan 20
Switch(config-if)#exit
Switch(config)#interfece vain 10
% Invalid input detected at '^' marker.
Switch(config)#interface vlan 10
Switch(config-if)#ip address 192.168.10.100 255.255.255.0
Switch(config-if)#interface vlan 20
Switch(config-if)#ip address 192.168.20.100 255.255.255.0
Switch(config-if)#exit
Switch(config)#exit
Switch#
%SYS-5-CONFIG_I: Configured from console by console
Ctrl+F6 to exit CLI focus
Copy
Paste
Top
VLAN 10
192.168.10.1
VLAN 20
192.168.20.1
192.168.10.2
192.168.20.2
Automatically Choose Connection Type
81°F
Mostly cloudy
Search
ENG
IN
12:32
30-01-2025



CONCLUSION (provide conclusion about this experiment):

- A 3 - layer switch for Inter Wheeler routing provides a cost effective and efficient way to enable communication between different VLANS within a network.
- VLANS allows us to define logical segments of our network infrastructure

Rubrics for Experiment Assessment:

Rubrics	Good	Normal	Poor	Marks
Creation of Topology (4)	Created the topology, Identify the proper devices and making the connections **(4)**	Created the topology, Identify the proper devices, making the connections But missing some features **(3)**	Created wrong topology, Failed to Identify the proper devices and making connections **(1)**	
Verify the connectivity (4)	Verified the connectivity in all the levels **(4)**	Verified the connectivity at some levels (only some nodes) **(2)**	Verified the connectivity is not done. **(1)**	
Timely Completion (2)	Completed the lab before the allotted time **(2)**	Completed the lab after the deadline **(1)**	Did not submitted before grading **(0)**	
			Total	

Result: Thus the implement of configuration of intervalence network using multi layer-3 speech has been executed successfully in Cisco Packet

Ex.No:	6b
Name of the Experiment	Configuration of Inter VLAN using Router on a stick method
Date:	

Objective(s):

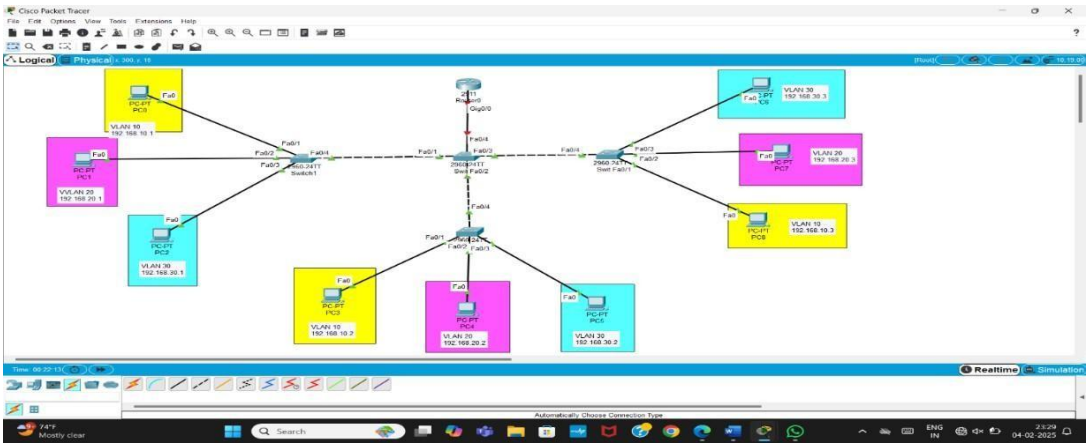
To design and implement Inter VLAN using switch configuration

Introduction:

'Router on a Stick' allows routing between VLANs with only one interface. Each VLAN represents a different Subnet. In general, routers can take traffic from only one subnet and transfer it to another subnet. And we can assign only one IP Address to a router interface. 'Router on a stick' allow us to create sub-interfaces, and assign IP Addresses to those sub-interfaces. To make it work, we have to create a truck connection between the switch and a router so that traffic from multiple VLANs can be sent to the router.

If we create a route between VLANs without the 'Router on a Stick' method, then we have to waste interfaces on the switches and routers. And if we enable routing between multiple VLANs then it will become practically inefficient as the switches and the routers will use those multiple interfaces.

The image below is an alternative method for allowing routing between VLANs. As you can see, we are using two interfaces on both the router and a switch to allow routing between VLANs. We have not created sub-interface in the below figure.



The network consists of multiple VLANs (Virtual Local Area Networks) interconnected through switches and a central router. Each VLAN is represented by a distinct color, with devices assigned specific IP addresses in their respective VLAN subnets.

1. VLAN Segmentation: VLANs are used to logically separate devices within the same physical network, improving security and reducing broadcast traffic.

Each VLAN (e.g., VLAN 10, 20, 30) is assigned its subnet range.

2. Switches: The switches connect devices within the same VLAN.

Trunk links (dashed lines) between switches allow traffic from multiple VLANs to pass through.

3. Router-on-a-Stick Configuration: The central router facilitates inter-VLAN communication by using sub-interfaces for each VLAN, configured with appropriate IP addresses and VLAN tagging.

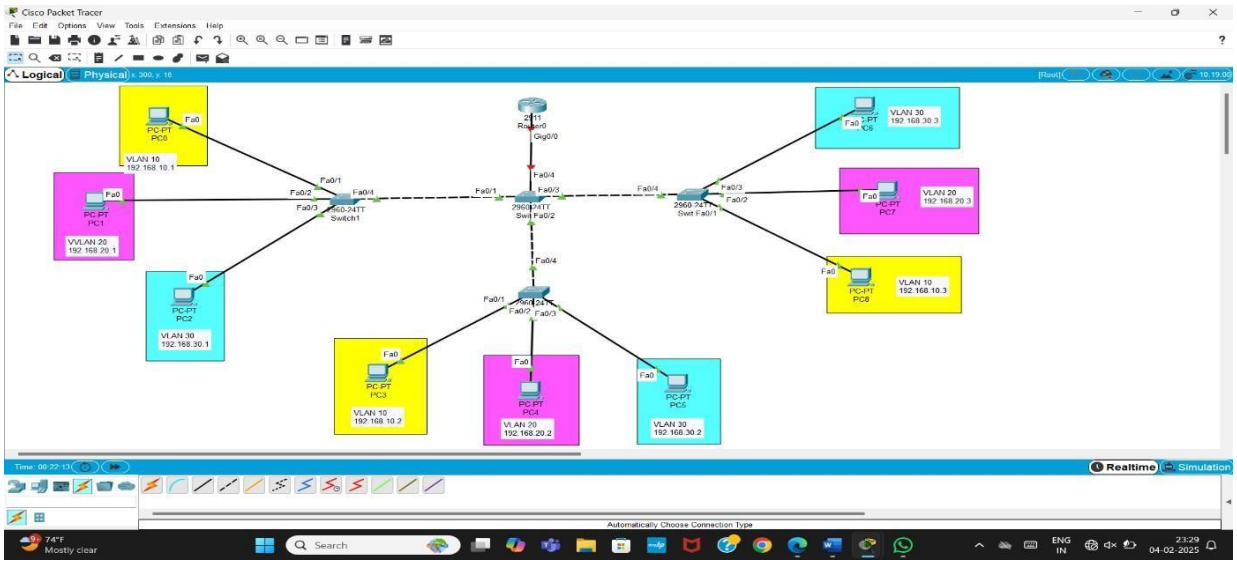
4. Devices: PCs are connected to specific switches within VLANs and assigned IP addresses corresponding to their VLAN subnet (e.g., 192.168.x.x).

5. Connectivity: The dashed lines represent connections between switches and the router, indicating trunk links that handle traffic for multiple VLANs.

Access ports are used for connecting end devices (e.g., PCs) to specific VLANs.

In the below lab, we will configure 'Router on a Stick' that would allow routing between the VLANs. Some of the important concepts in this lab are – to create sub-interfaces, use encapsulation dot1Q command to encapsulate the traffic, and mentioning the VLAN number to ascertain that for which VLAN the sub-interface should respond.

1. **Device Requirements:**
    1. PC0
    2. PC1
    3. PC2
    4. PC3
    5. PC4
    6. PC5
    7. PC6
    8. PC7
    9. PC 8
    10. Switch 0
    11. Switch 1
    12. Switch 2
    13. Switch 3
    14. Routes 0
2. **Network Diagram (Packet tracer diagram before configuration):**



3. Configuration details:

Device Name	Interface Name	IP Address	Subnet mask
PC0	fa 0/1	192.168.10.1	255.255.255.0
PC1	fa 0/2	192.168.20.1	255.255.255.0
PC2	fa 0/3	192.168.30.1	255.255.255.0
PC3	fa 0/1	192.168.10.2	255.255.255.0
PC4	fa 0/2	192.168.20.2	255.255.255.0
PC5	fa 0/3	192.168.30.2	255.255.255.0

4. Describe step by step configuration steps properly (you may copy the commands used in the configuration tab and paste it.)

SWITCH-1

Switch > enable

Switch #Configure terminal

Enter Configuration Commands, one per Line. End with CNTL/Z.

Switch (Config)#vlan 10

Switch (Config-vlan)# vlan 20

Switch (Config-vlan) #vlaan 30

Switch (Config-vlan)#exit

Switch (Config)#exit

%SYS-5-CONFIG_I: Configured from Console by Console.

Switch #Show vlan

Switch# Configure terminal

Enter Configuration Commands, one per line. End with CNTL/Z.

Switch (Config)#interface fa0/1,

Switch (Config-if)#switchport mode access

Switch (Config-if)#switchport access vlan 10

Switch (Config-if)#exit

Switch (Config-if)#interface fa0/2

Switch (Config-if)#switchport mode access

Switch (Config-if)#switchport access vlan 20

Switch (Config-if)#exit

Switch (config-if)# interface fa0/3

Switch (Config-if)#switchport mode access.

Switch (Config-if)#switchport access vlan 30

Switch (config-if)#exit

Switch > Show interfaces trunk

Switch>% SPANTREE_2_RECV_PVID_ERR: Received 802.1Q

BPDU on non trunk FastEthernet0/4 VLAN1

%SPANTREE_2_BLOCK_PVID_LOCAL: Blocking FastEthernet0/4 on VLAN0001, INConsistent port type

CENTER SWITCH-0

Switch >enable

Switch #Configure terminal

Enter Configuration Commands, one per love. Endi with CNTL/Z.

Switch (Config)#vlan 10

Switch (Config-vlan)#vlan 20

Switch (Config-vlan)#vlan 30

Switch (Config-vlan)#exit

Switch (config)#interface range fa0/1-3

Switch (Config-if-range)#switchport mode access

Switch (Config-if-range)#switchport mode trunk

Switch (Config if-range)#

%LINEPROTO-5-UPDOWN: LINE Protocol on interface FastEthernet0/1, changed state to down

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up

%LINE PROTO-5-UPDOWN: Line Protocol on Interface FastEthernet0/2, changed state to down.
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/2, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/3, changed state to down

%LINEPROTO-5-UPDOWN: Line Protocol on Interface FastEthernet0/3, changed state to up.

Switch (Config-if-range)#exit

Switch (Config)#exit

%SYS-5-CONFIG-I: Configured from Console by console

Switch #show interfaces trunk

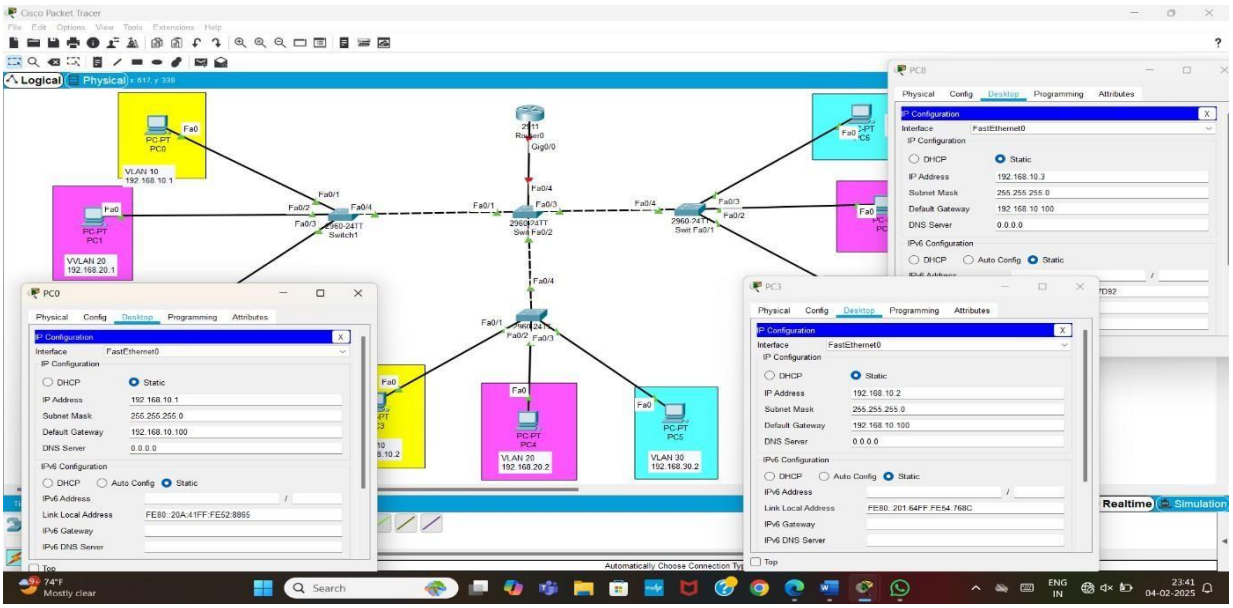
Switch > enable

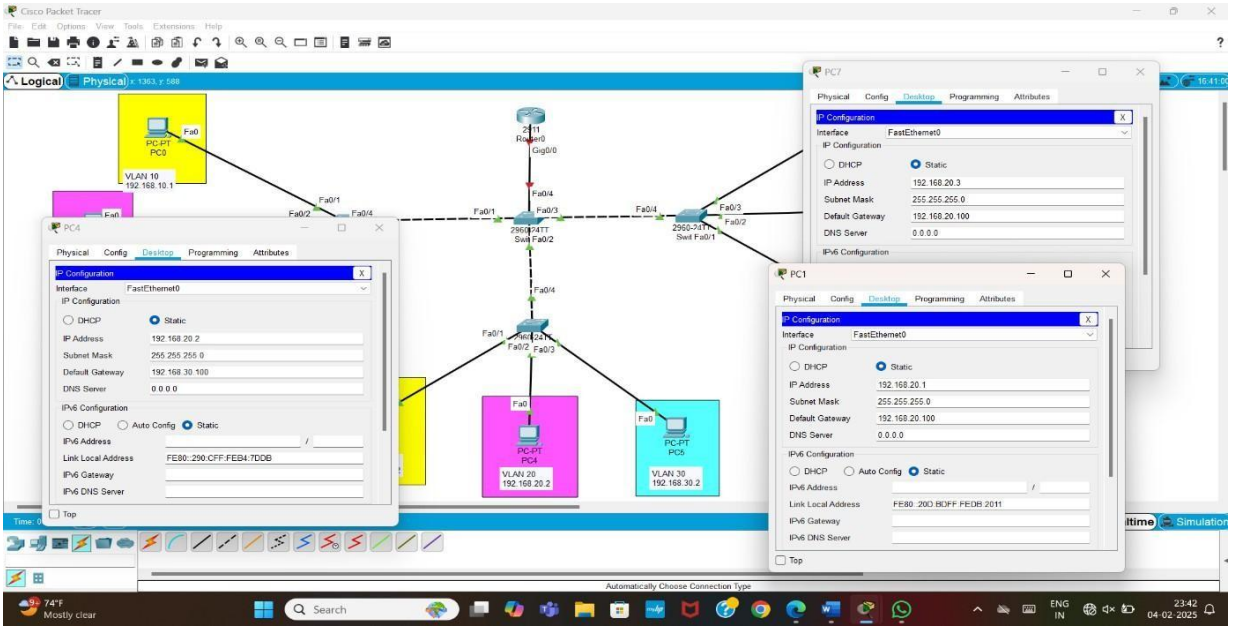
Switch# Configure terminal
Enter Configuration Commands, one per Line. End with CNTL/Z
Switch (config)#interface Fa0/4
Switch (config-if)#switchport mode trunk
%LINEPROTO-5-UPDOWN: LINE protocol on Interface FastEthernet0/4, changed state to down
%LINEAROTO -5-UPDOWN: Line protocol on Interface FastEthernet0/4, changed state to up.
Switch (Config-if)#exit
Switch (config)#exit
%SYS-5-CONFIG-I: Configured from Console by console Switch
#show interfaces trunk

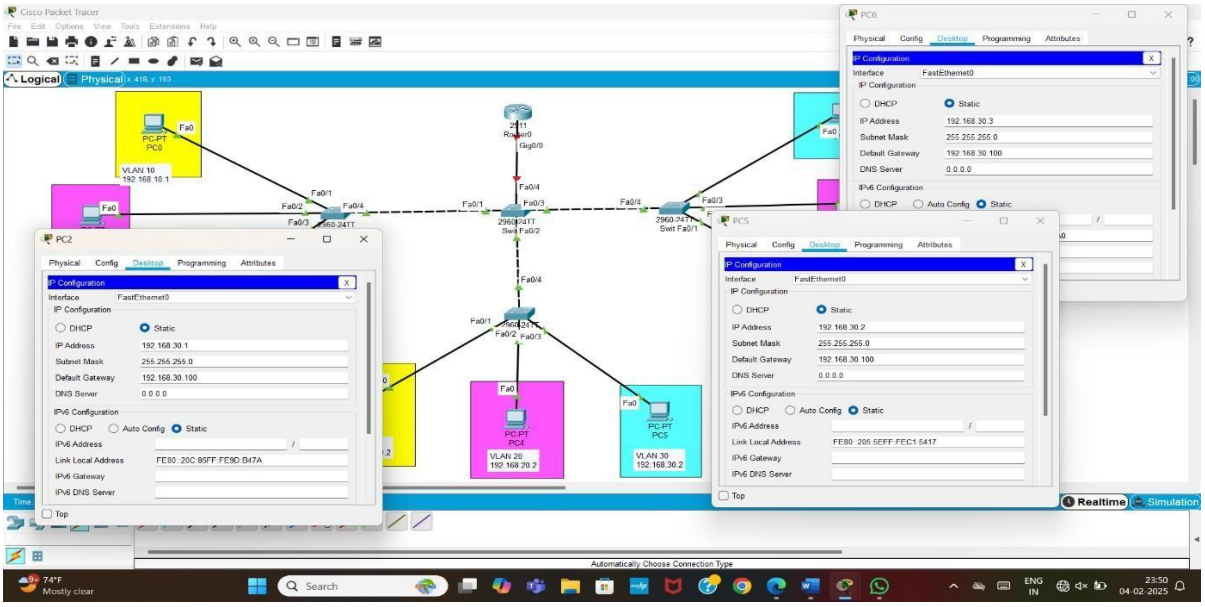
ROUTER-O

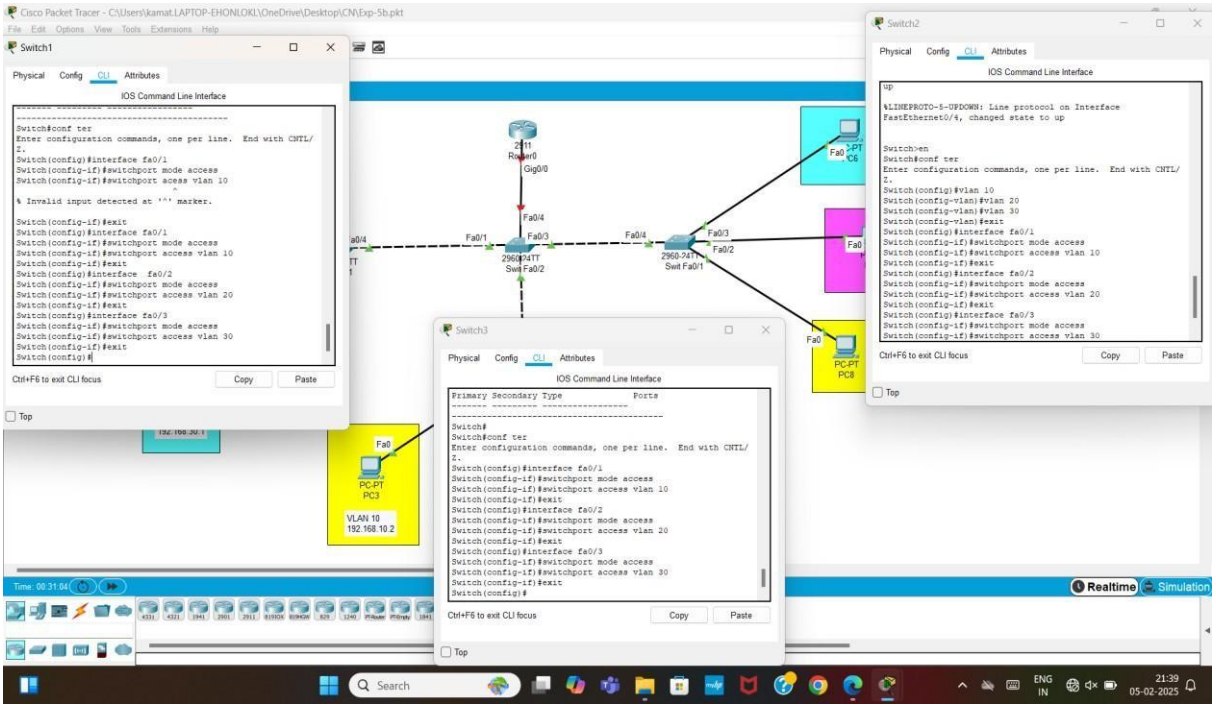
Router> enable
Router# show ip route
Codes: L-Local, C-Connected, S-Static, R-RIP, M-mobile, B-BGP, D-EIGRP, EX-EIGRP external, 0-08FF, IA-O8PF inter area. NI-OSPE NSSA external type 1, N2-OSPF NSSA external type 2, EI-OSPF external type 1, E2-OSPF external type 2, E-EGP i-IS-IS, LI-IS-IS Level-1, L2-IS-IS Level-2, ia-IS-IS inter area * - Candidate default, U-per-user Static route, O-ODR P-Periodic download ed Static route.
Router #show if interface brief
Router #Configure terminal
Enter Configuration Commands, one per Line. End with CNTLE.
Router (config)#interface gigabit Ethernet 0/0.10
Router (Config-Subif)#encapsuiation dot 1Q 10
Router (Config-Subif)#ip address 192.168.10.100 255.255.255.0
Router (Config-Subif)#exit
Router(config)#interface gigabit Ethernet 0/0.20
Router (Config-Sublf) #encapsulation dot 1Q 20
Router (Config-Subif)#ip address 192.168.20.100 255.255.255.0
Router (Config-Subif)#exit.
Router (Config)# interface gigabit Ethernet 0/0-30
Router (Config-Subif) # encapsulation dot 1Q 80 address 192.168-30-100
Router (Config-subilf)#ip address 192.168-30-100 255-2.55.255.0
Router (Config)# exit
Router (Config)#exit
%SYS-5-CONFIG-I: Configured from Console by Console
Router #show ip interface brief

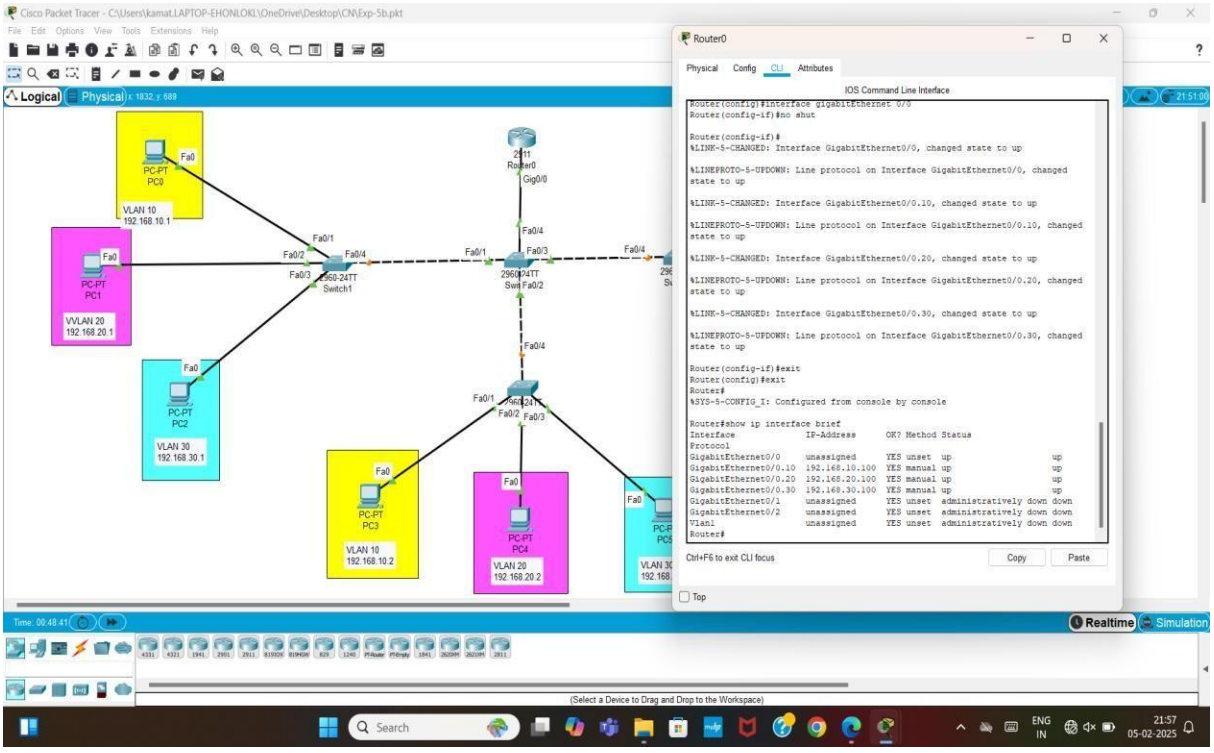
5. Output Diagram (Minimum 3 screenshot):











CONCLUSION (provide conclusion about this experiment):

- A router is used in interval grouping to enable communication between different VLANS by acting as a layer 3 device that directs traffic based on an ip address

Rubrics for Experiment Assessment:

Rubrics	Good	Normal	Poor	Marks
Creation of Topology (4)	Created the topology, Identify the proper devices and making the connections **(4)**	Created the topology, Identify the proper devices, making the connections But missing some features **(3)**	Created wrong topology, Failed to Identify the proper devices and making connections **(1)**	
Verify the connectivity (4)	Verified the connectivity in all the levels **(4)**	Verified the connectivity at some levels (only some nodes) **(2)**	Verified the connectivity is not done. **(1)**	
Timely Completion (2)	Completed the lab before the allotted time **(2)**	Completed the lab after the deadline **(1)**	Did not submitted before grading **(0)**	
			Total	

Results: Thus the implementation of configuration of Inter VLAN network using router monistic method has been executed successfully in Cisco Packet.

Ex. No:	7(a)
Name of the Experiment:	Link State Routing
Date:	

Objective(s):

To design and implement Link state routing using packet tracer.

Introduction:

Link State Routing Protocols used to select the path for data packet in an internetwork. Link state routing protocols uses link state routers to share information of connected network devices. This is a learning process. By learning process each router maintain the routing table to select the shortest path for data packet transmission. Each router update the network topology to nearby router only. Link state routing protocols are also known as **shortest path first protocol.**

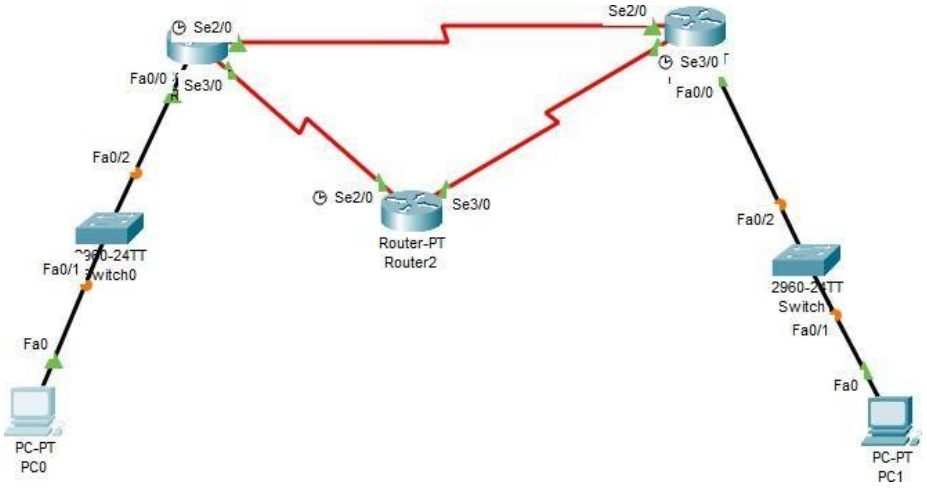
Link state protocols allow routers to share the information about network connected to it. This information passed to neighbour router only. An accurate information of network topology around the router updated in routing table. By help of the routing table better routing path selected by the router.

The information passes by router is known as link state advertisements(LSAs). In distance vector the information message passes in a fix time interval. Link state advertisements shared only when any changes done in the network topology. The bandwidth less consumed by link state routing protocol. The time of convergence is less than in distance vector protocol.

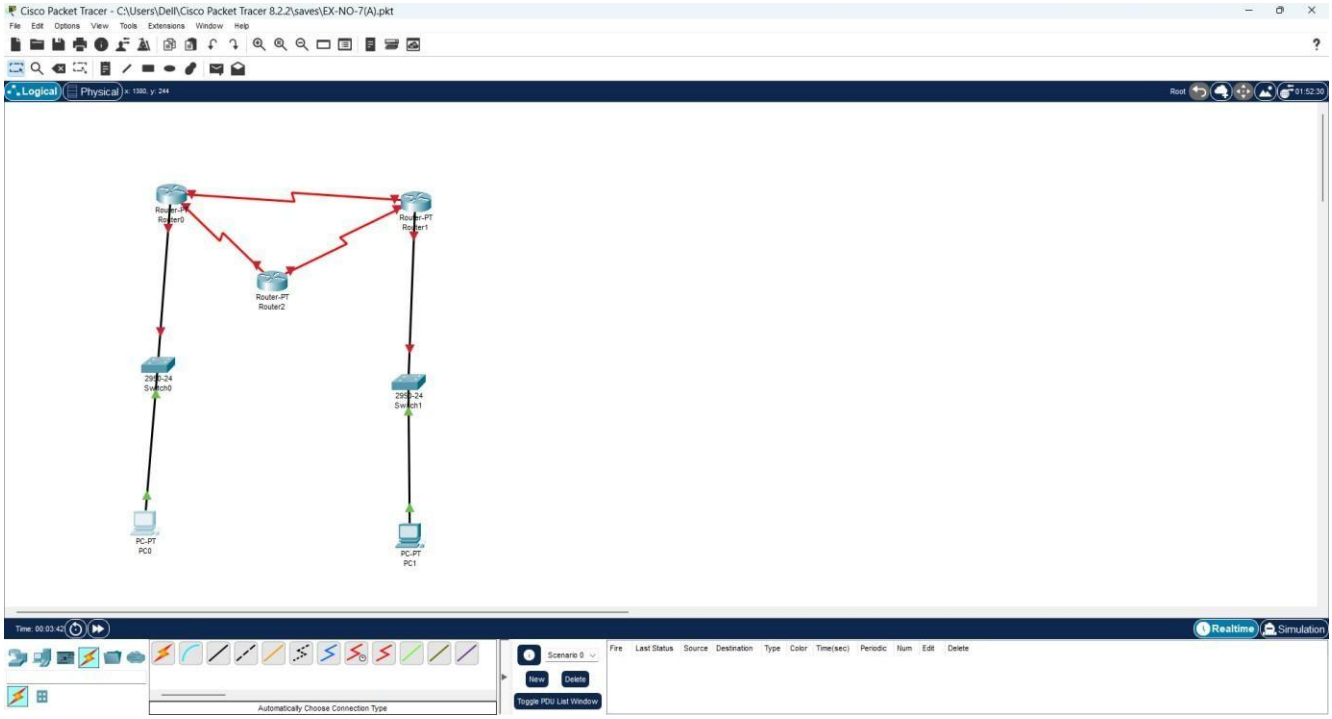
Function of link state routing protocol

Important terms of link state routing protocol are link state packet, database, algorithm, routing table etc. Link state packets contains the routing information and sent to neighbour only when any changes occurs in connected network. Link state packets update the routing table in nearby routers. The information collected by link state packets stored in link state database.

1. **Device Requirements:**
   1. Router
   2. Switch
   3. PC
   4. Wires
2. **Network Diagram for your experiment (draw the diagram either hand drawing/ms paint or any other drawing tools)**



3. Network Diagram (Packet tracer diagram before configuration):



4. Configuration details:

Device Name	Interface Name	IP Address	Subnet mask
PC0	Fa0/1	10.0.0.2	255.0.0.0
PC1	Fa0/1	20.20.20.2	255.0.0.0
Switch 0	Fa0/2		
Switch 1	Fa0/2		
Router 0	Fa0/0		
Router 1	Se 2-3/0		
Router 2	Se 2-3/0		

5. **Describe step by step configuration steps properly (you may copy the commands used in the configuration tab and paste it.)**

ROUTER0

Router>enable

Router#

Router#configure terminal

Enter configuration commands, one per line. End with CNTL/Z.

Router(config)#interface FastEthernet0/0

Router(config-if)#no shutdown

Router(config-if)#

%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up no ip address

Router(config-if)#ip address 10.10.10.1 255.0.0.0

Router(config-if)#exit

Router(config)#interface Serial2/0

Router(config-if)#no shutdown

Router(config-if)#clock rate 64000

Router(config-if)#ip address 30.30.30.1 255.0.0.0

Router(config-if)#ip address 30.30.30.1 255.0.0.0

Router(config-if)#

%LINK-5-CHANGED: Interface Serial2/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up
Router(config-if)#exit

Router(config)#interface Serial3/0

Router(config-if)#no shutdown

Router(config-if)#clock rate 64000

Router(config-if)#ip address 40.40.40.1 255.0.0.0

Router(config-if)#ip address 40.40.40.1 255.0.0.0

Router(config-if)#

%LINK-5-CHANGED: Interface Serial3/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial3/0, changed state to up

Router(config-if)#exit

Router(config)#router ospf 1

Router(config-router)#network 10.0.0.0 0.255.255.255 area 0
Router(config-router)#network 30.0.0.0 0.255.255.255 area 0
Router(config-router)#network 40.0.0.0 0.255.255.255 area 0
Router(config-router)#exit
Router(config)#exit
Router#

ROUTER1

Router>enable
Router#
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface FastEthernet0/0
Router(config-if)#no shutdown Router(config-
if)#

%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up ip address 20.20.20.1 255.0.0.0
Router(config-if)#ip address 20.20.20.1 255.0.0.0
Router(config-if)#
Router(config-if)#exit
Router(config)#interface Serial2/0
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface Serial2/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up ip address 30.30.30.3 255.0.0.0
Router(config-if)#ip address 30.30.30.3 255.0.0.0
Router(config-if)#exit
Router(config)#interface Serial3/0
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface Serial3/0, changed state to up clock rate 64000
This command applies only to DCE interfaces

Router(config-if)#ip address 50.50.50.2 255.0.0.0
Router(config-if)#ip address 50.50.50.2 255.0.0.0
Router(config-if)#exit
Router(config)#router ospf 1
Router(config-router)#network 30.0.0.0 0.255.255.255 area 0
Router(config-router)#
00:18:05: %OSPF-5-ADJCHG: Process 1, Nbr 40.40.40.1 on Serial2/0 from LOADING to FULL, Loading Done

Router(config-router)#network 0.0.0.0 0.255.255.255 area 0
Router(config-router)#network 20.0.0.0 0.255.255.255 area 0
Router(config-router)#network 50.0.0.0 0.255.255.255 area 0
Router(config-router)#exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console

ROUTER2

Router>enable
Router#
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface FastEthernet0/0
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up ip address 20.20.20.1 255.0.0.0
Router(config-if)#ip address 20.20.20.1 255.0.0.0
Router(config-if)#exit
Router(config)#interface Serial2/0
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface Serial2/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up ip address 30.30.30.3 255.0.0.0

Router(config-if)#ip address 30.30.30.3 255.0.0.0

Router(config-if)#exit

Router(config)#interface Serial3/0

Router(config-if)#no shutdown

Router(config-if)#

%LINK-5-CHANGED: Interface Serial3/0, changed state to up clock

rate 64000

This command applies only to DCE interfaces

Router(config-if)#ip address 50.50.50.2 255.0.0.0

Router(config-if)#ip address 50.50.50.2 255.0.0.0

Router(config-if)#exit

Router(config)#router ospf 1

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

Router(config-router)#

00:18:05: %OSPF-5-ADJCHG: Process 1, Nbr 40.40.40.1 on Serial2/0 from LOADING to FULL, Loading Done

Router(config-router)#network 20.0.0.0 0.255.255.255 area 0

Router(config-router)#network 50.0.0.0 0.255.255.255 area 0

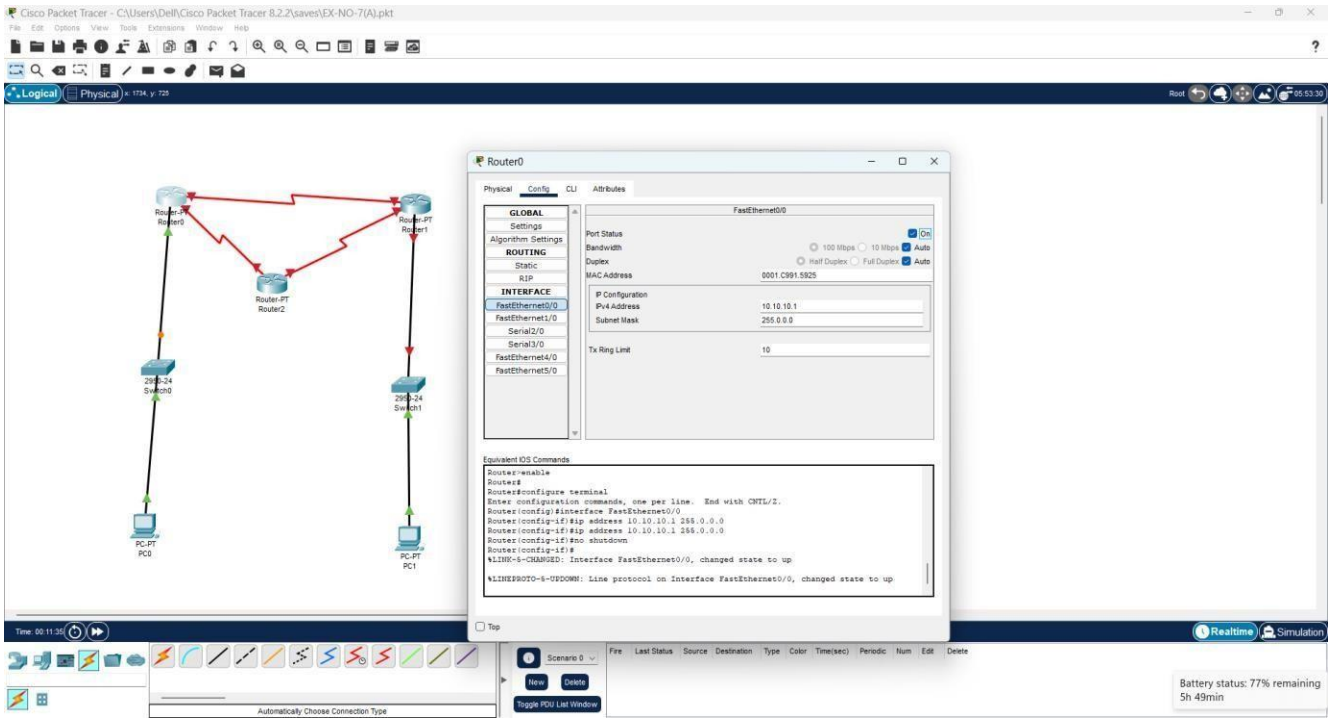
Router(config-router)#exit

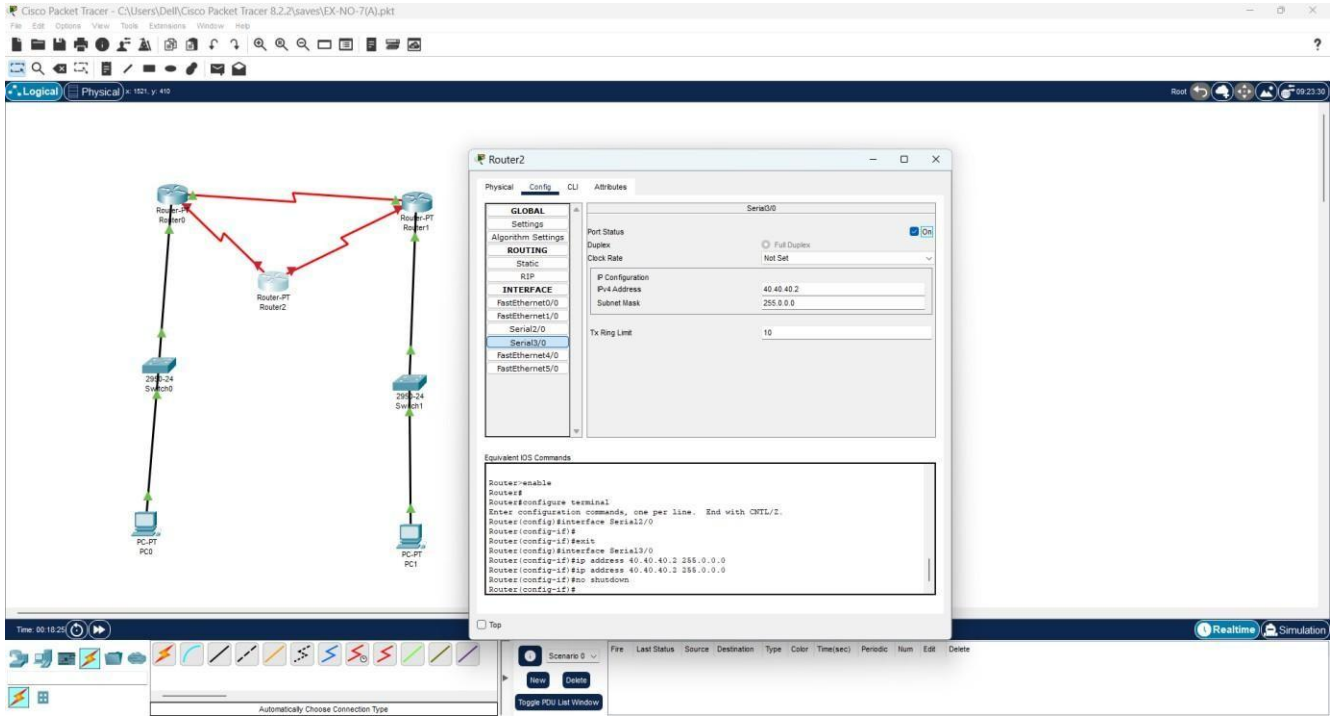
Router(config)#exit

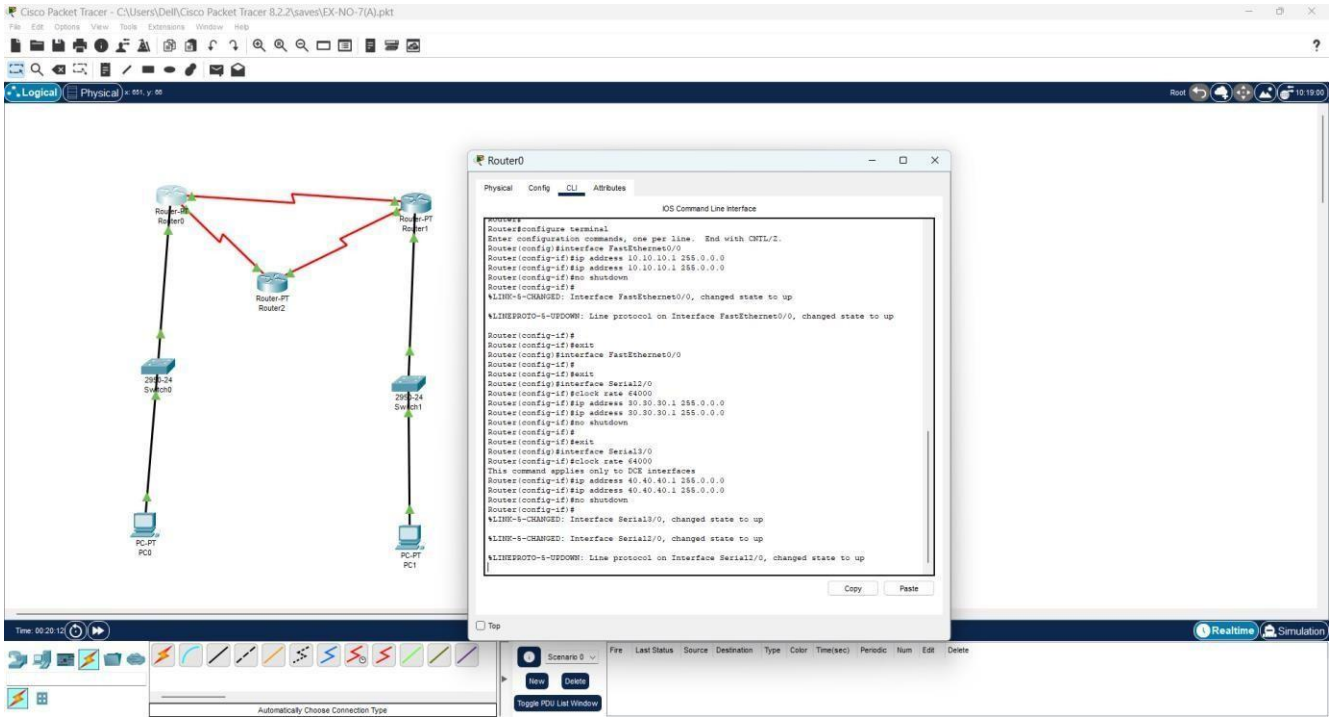
Router#

%SYS-5-CONFIG_I: Configured from console by console

6. Output Diagram (Minimum 3 screenshot):







Rubrics for Experiment Assessment:

Rubrics	Good	Normal	Poor	Marks
Creation of Topology (4)	Created th e topology, Identify the proper devices and making the connections **(4)**	Created the topology, Identify the proper devices, making the connections But missing some features **(3)**	Created wron g topology, Failed to Identify the proper devices and making connections **(1)**	

Verify the connectivity (4)	Verified the connectivity in all the levels **(4)**	Verified the connectivity at some levels (only some nodes) **(2)**	Verified the connectivity is not done. **(1)**	
Timely Completion (2)	Completed the lab before the allotted time **(2)**	Completed the lab after the deadline **(1)**	Did not submitted before grading **(0)**	
			Total	

CONCLUSION (provide conclusion about this experiment):

In this experiment, we configured Link-State Routing, which uses a dynamic routing protocol like OSPF (Open Shortest Path First) to determine the best paths in a network. This method ensures faster convergence, scalability, and efficient use of network resources.

Result:

Thus, the implementation of Link State Router using packet tracer was implemented and executed successfully.

Ex. No:	7(b)
Name of the Experiment:	Distance Vector Routing
Date:	

Objective(s):

To design and implement Distance Vector routing using packet tracer **Introduction:**

Distance-Vector routing protocols select the best path for data packets. Here distance is reference of hop in network. Distance-Vector protocols calculate the distance between source and destination on the basis of hop count. Suppose there are two path available for data packet from source and destination. Distance-Vector protocol select the path in which the number of hopes are less. RIP and IGRP are example of Distance-Vector routing protocol.

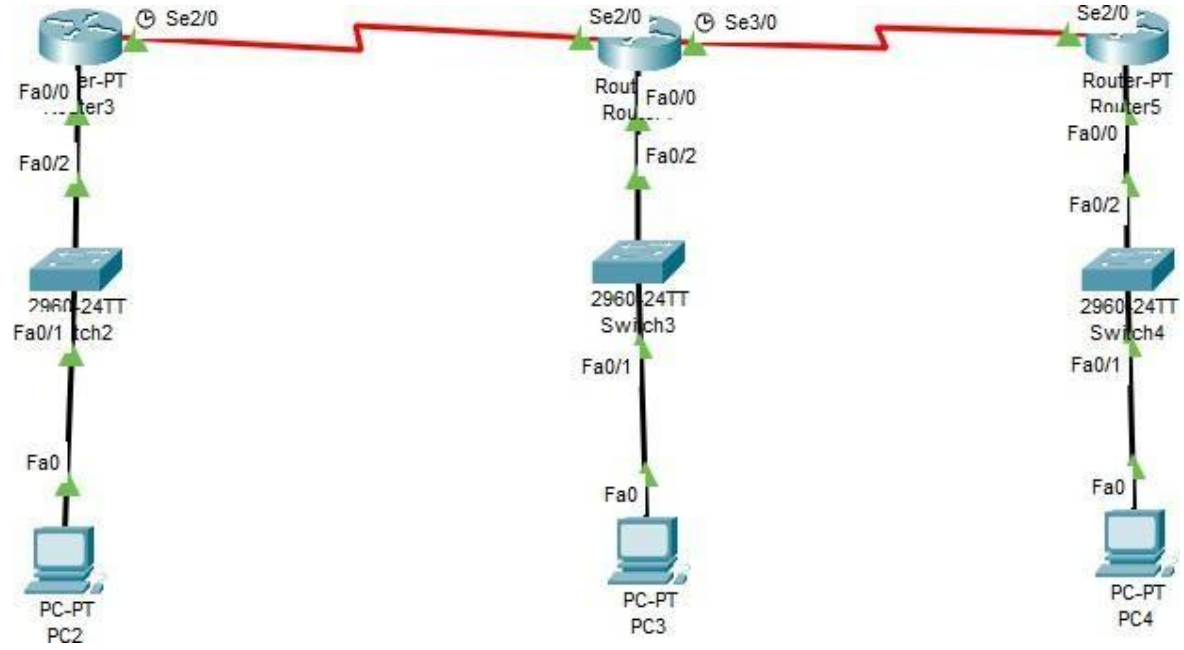
Distance vector routing protocols manage the selection of best path for data packets by routers. Routing table of all routers update by sharing the information on the network. The destination network path defines by hop count up to destination network. Distance vector routing protocols generally known as DVRP. Distance vector routing protocols is mostly used protocol in present scenario. DVRP sent the data packets over the internet protocol.

There are two terms in DVRP. The first term is distance and second is vector. Distance is number of hop or step to send the data packets up to destination network. Path selection for a data packet is depends on the hop count. **Minimum hope count path selected by the Distance vector routing protocols**. The term vector refers to the propagating of the packet on a given set of network nodes. Routers broadcast the information of remote network to next router. **Every router does the same thing so the routing table of all routers updated automatically**. All router informs about the connected networks to next router then router update its own routing table.

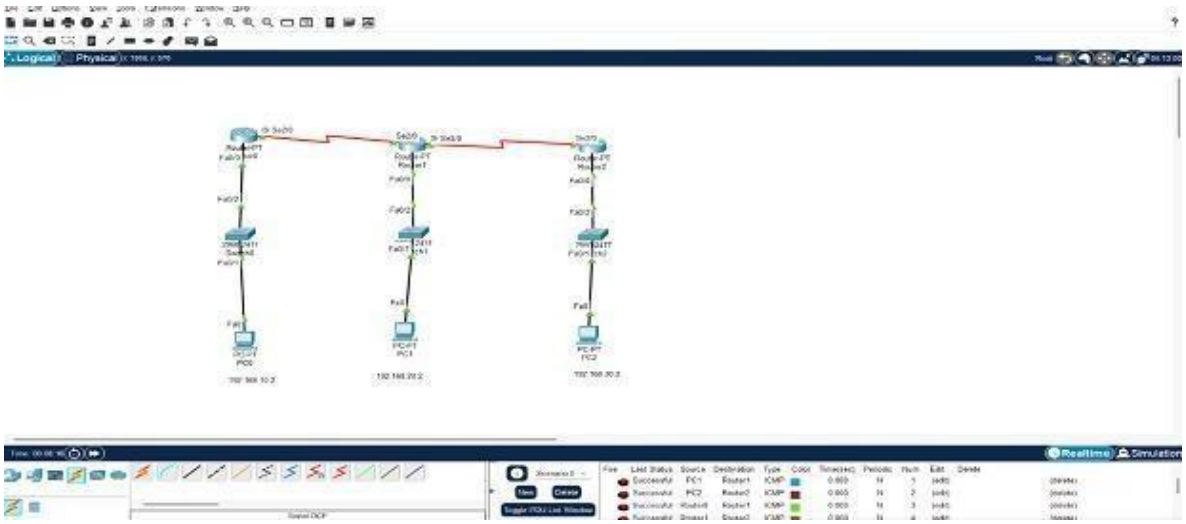
Network topology changes time to time. Adding or removing a router in a network is very common phenomena. **Any change in network should be updated in all router's routing table**. Doing this manually is very critical work. Distance vector routing protocols do this job automatically. The process of broadcasting any update in routing table and updation in all routing tables is known as convergence. The algorithm distance vector routing protocol find the routes on a internetwork. The other algorithm used to select the best path for data packets is Link State routing protocols. DVRP algorithm allow routers to exchange the routing tables with each other. Each router received the routing table from neighbour router, update own routing table and share the updated table to next neighbour router. This process repeat after a fix predefined time interval. By repeating this process all devices connected in the network maintain the routing table which allow the flow of data packets efficiently.

1. **Device Requirements:**
   1. Routers
   2. Switches
   3. PC
   4. Wires

2. **Network Diagram for your experiment (draw the diagram either hand drawing/ms paint or any other drawing tools)**



3. Network Diagram (Packet tracer diagram before configuration):



4. Configuration details:

Device Name	Interface Name	IP Address	Subnet mask
PC0	Fa0/1	192.168.10.2	255.255.255.0
PC1	Fa0/1	192.168.20.2	255.255.255.0
PC2	Fa0/1	192.168.30.2	255.255.255.0
Switch 0	Fa0/2		
Switch 1	Fa0/2		
Switch 2	Fa0/2		
Router 0	Se2/0		

Router 1	Se2/0, Se3/0		
Router 2	Se2/0		

5. **Describe step by step configuration steps properly (you may copy the commands used in the configuration tab and paste it.)**

ROUTER0

Router>enable
Router#
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface FastEthernet0/0
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up ip address 192.168.10.1 255.255.255.0
Router(config-if)#ip address 192.168.10.1 255.255.255.0
Router(config-if)#
Router(config-if)#exit
Router(config)#interface Serial2/0
Router(config-if)#no shutdown
Router(config-if)#clock rate 64000
Router(config-if)#ip address 10.0.0.2 255.0.0.0
Router(config-if)#ip address 10.0.0.2 255.0.0.0
Router(config-if)#
%LINK-5-CHANGED: Interface Serial2/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up

Router(config-if)#exit
Router(config)#router rip
Router(config-router)#network 10.0.0.0
Router(config-router)#network 192.168.10.0

ROUTER1

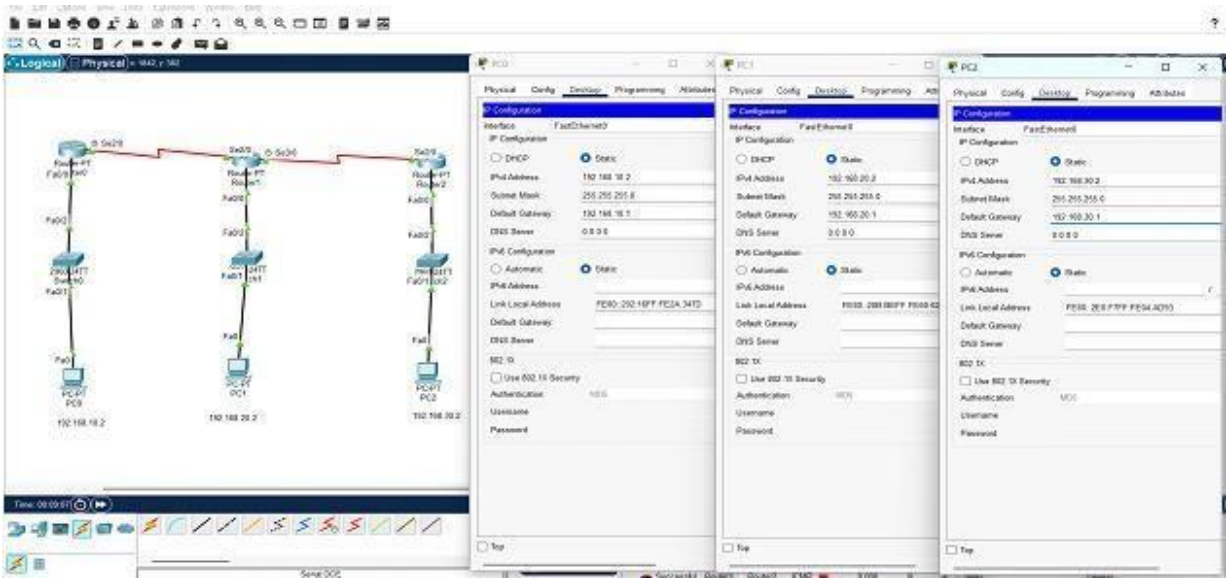
Router>enable

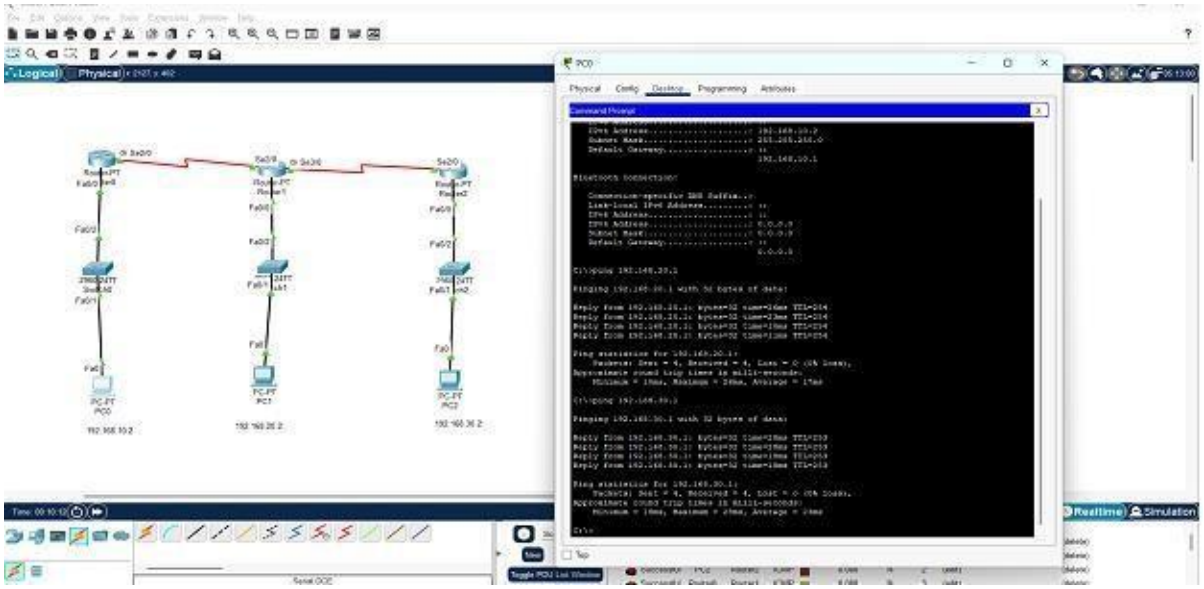
```
Router#
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface FastEthernet0/0
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up ip
address 192.168.20.1 255.255.255.0
Router(config-if)#ip address 192.168.20.1 255.255.255.0
Router(config-if)#
Router(config-if)#exit
Router(config)#interface Serial2/0
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface Serial2/0, changed state to up ip
address 10.0.0.3 255.0.0.0
Router(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up ip
address 10.0.0.3 255.0.0.0
Router(config-if)#
Router(config-if)#exit
Router(config)#interface Serial3/0
Router(config-if)#no shutdown
Router(config-if)#clock rate 64000
Router(config-if)#ip address 20.0.0.2 255.0.0.0
Router(config-if)#ip address 20.0.0.2 255.0.0.0
Router(config-if)#
%LINK-5-CHANGED: Interface Serial3/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial3/0, changed state to up

Router(config-if)#exit
Router(config)#router rip
Router(config-router)#network 10.0.0.0
Router(config-router)#network 20.0.0.0
Router(config-router)#network 192.168.20.0
```

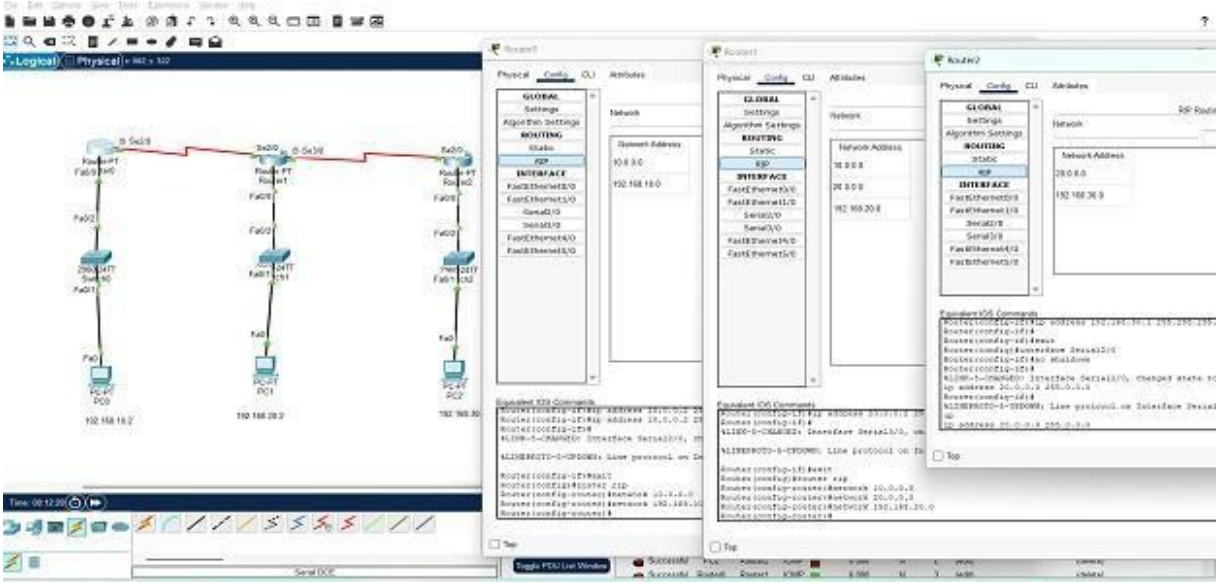
ROUTER2

```
Router>enable
Router#
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface FastEthernet0/0
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up ip
address 192.168.30.1 255.255.255.0
Router(config-if)#ip address 192.168.30.1 255.255.255.0
Router(config-if)#
Router(config-if)#exit
Router(config)#interface Serial2/0
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface Serial2/0, changed state to up ip
address 20.0.0.3 255.0.0.0
Router(config-if)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed state to up ip
address 20.0.0.3 255.0.0.0
Router(config-if)#
Router(config-if)#exit
Router(config)#router rip
Router(config-router)#network 20.0.0.0
Router(config-router)#network 192.168.30
```

6. Output Diagram (Minimum 3 screenshot):







Rubrics for Experiment Assessment:

Rubrics	Good	Normal	Poor	Marks
Creation of Topology (4)	Created the topology, Identify the proper devices and making the connections **(4)**	Created the topology, Identify the proper devices, making the connections But missing some features **(3)**	Created wrong topology, Failed to Identify the proper devices and making connections **(1)**	
Verify the connectivity (4)	Verified the connectivity in all the levels **(4)**	Verified the connectivity at some levels (only some nodes) **(2)**	Verified the connectivity is not done. **(1)**	
Timely Completion (2)	Completed the lab before the allotted time **(2)**	Completed the lab after the deadline **(1)**	Did not submitted before grading **(0)**	
			Total	

CONCLUSION (provide conclusion about this experiment):

In this experiment, we configured Distance Vector Routing, which determines the best path based on hop count using protocols like RIP (Routing Information Protocol). Routers exchange periodic updates with neighbours', making this method simple to implement but slower to converge. While effective for small networks, distance vector routing can suffer from routing loops and inefficiencies in larger networks.

Result:

Thus, the implementation of Distance Vector router using packet tracer was implemented and executed successfully.

Ex. No:	8
Name of the Experiment:	Subnetting
Date:	

Objective(s):

To design and implement Subnetting configuration using packet tracer.

Introduction:

Subnetting is the process of dividing a large network into smaller networks called “subnets.” Subnets provide each group of devices with their own space to communicate, which ultimately helps the network to work easily. This also boosts security and makes it easier to manage the network, as each subnet can be monitored and controlled separately.

An IP address is made up of different parts, each serving a specific purpose in identifying a device on a network. An IPv4 address consists of four parts called “octets,” separated by dots (e.g., 192.168.1.1). It has two main sections:

- **Network Portion**: Identifies the network the device belongs to.
- **Host Portion**: Uniquely identifies a device within the network.

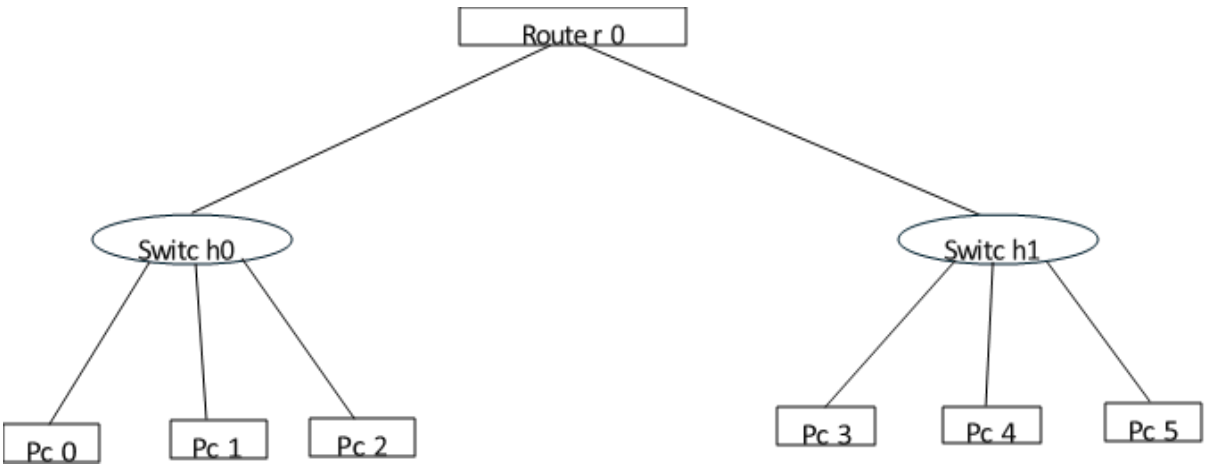
IPv4 addresses are divided into classes based on the length of the network and host portions:

- **Class A**: 8-bit network ID, 24-bit host ID.
- **Class B**: 16-bit network ID, 16-bit host ID.
- **Class C**: 24-bit network ID, 8-bit host ID.

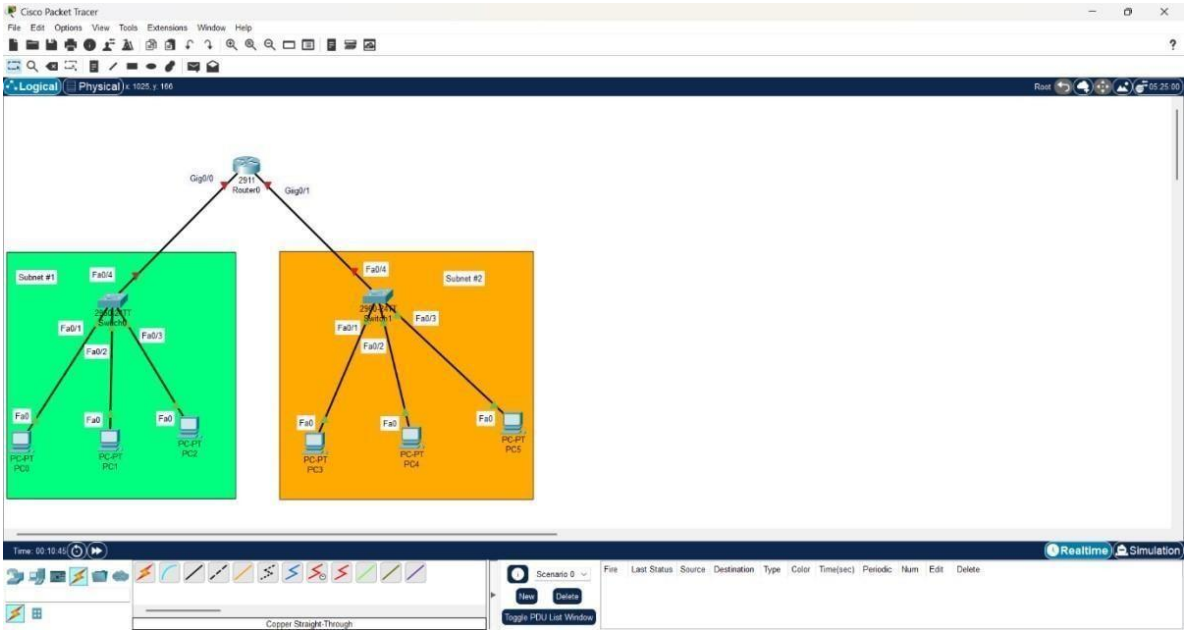
1. Device Requirements:

1. PC
2. Router
3. Switch
4. Wires

2. Network Diagram for your experiment (draw the diagram either hand drawing/ms paint or any other drawing tools)



3. Network Diagram (Packet tracer diagram before configuration):



4. Configuration details:

Device Name	Interface Name	IP Address	Subnet mask	Default Gateway
PC0	Fa0	192.168.10.1	255.255.255.128	192.168.10.4
PC1	Fa0	192.168.10.2	255.255.255.128	192.168.10.4
PC2	Fa0	192.168.10.3	255.255.255.128	192.168.10.4
PC3	Fa0	192.168.10.129	255.255.255.128	192.168.10.132
PC4	Fa0	192.168.10.130	255.255.255.128	192.168.10.132
PC5	Fa0	192.168.10.131	255.255.255.128	192.168.10.132
Router 0	Gig0/0, Gig0/1	192.168.10.4, 192.168.10.132	255.255.255.128, 255.255.255.128	
Switch 1	Fa0/4			
Switch 2	Fa0/4			

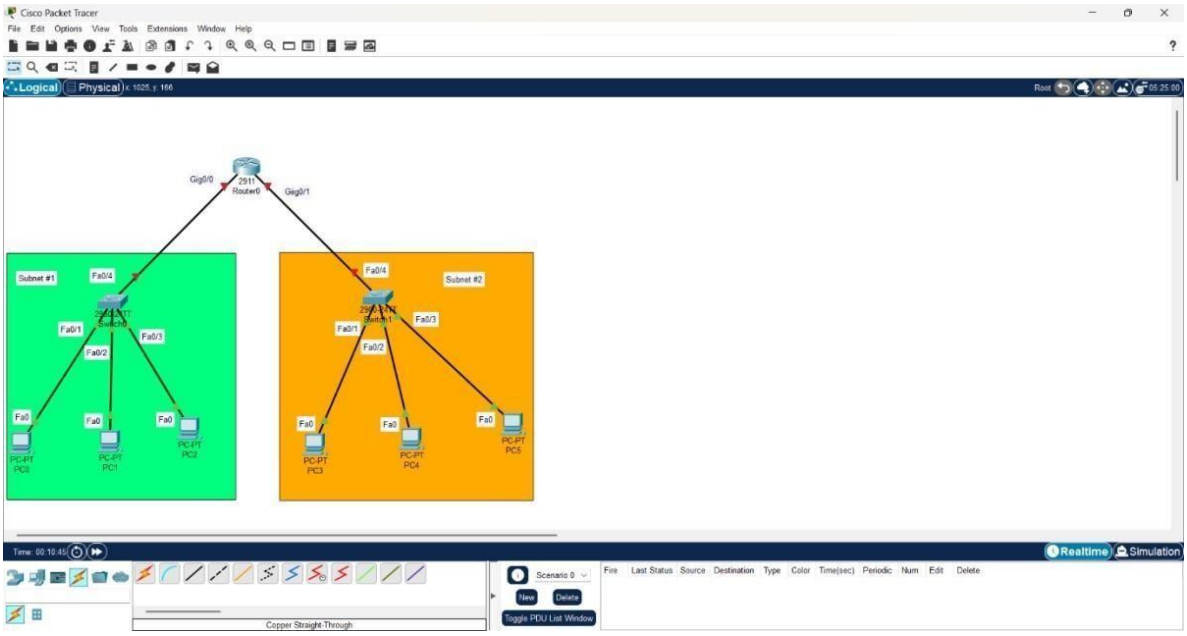
5. Describe step by step configuration steps properly (you may copy the commands used in the configuration tab and paste it.) <u>Router 0 :</u>

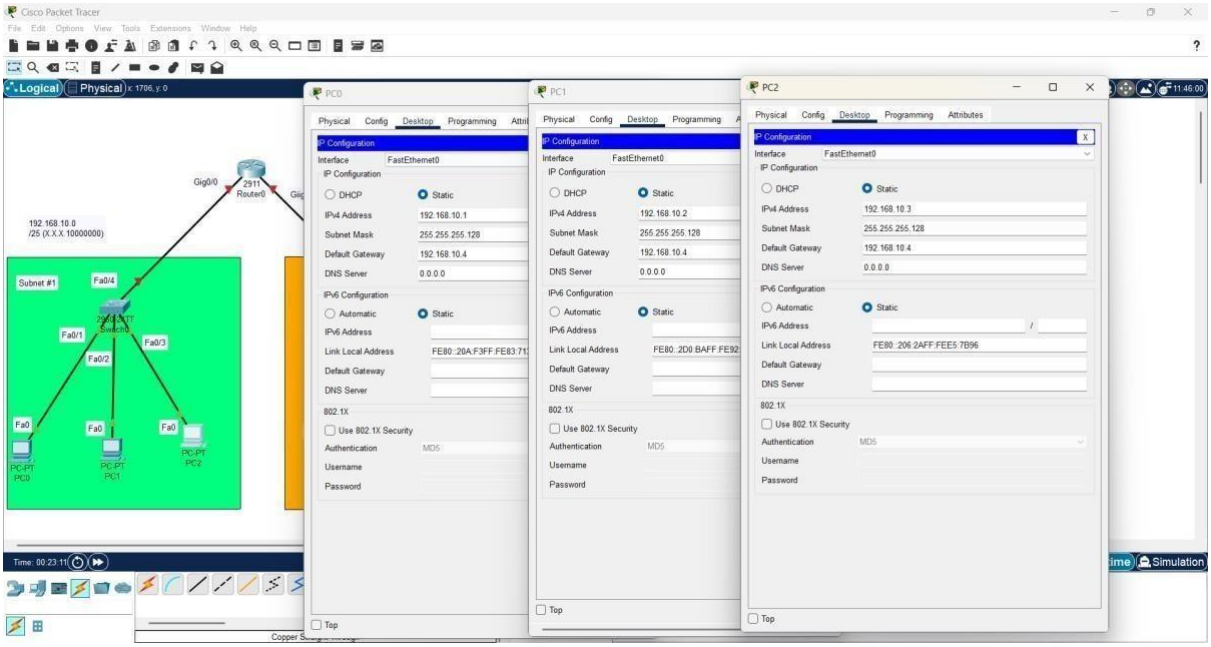
Router>enable
Router#
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface GigabitEthernet0/0
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up

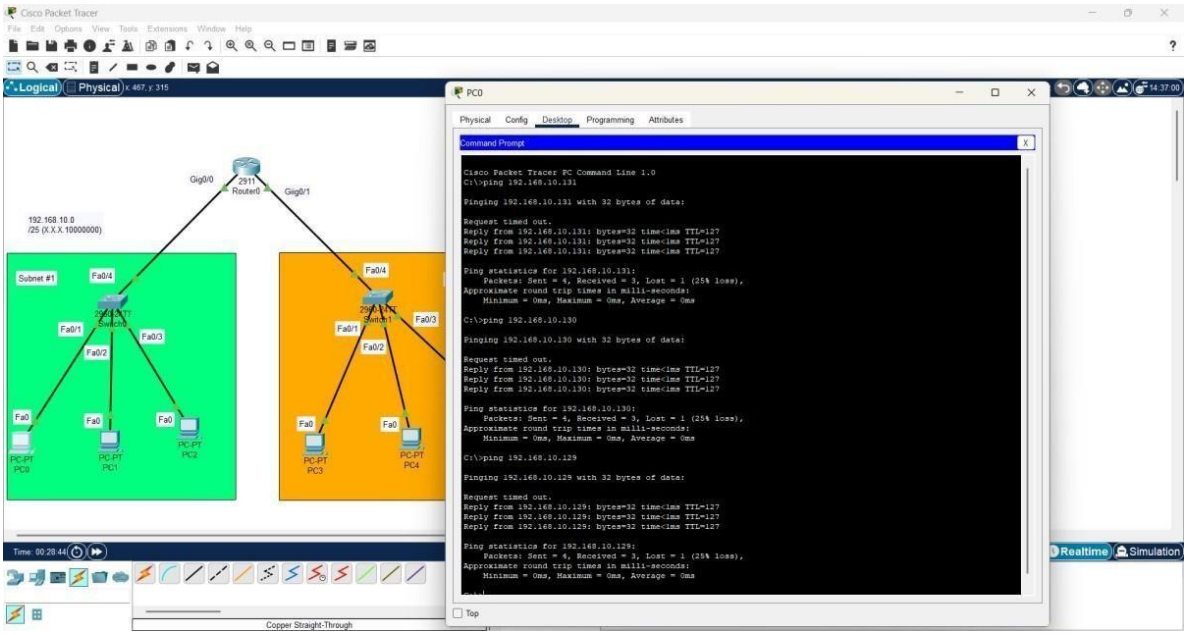
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up
ip address 192.168.10.4 255.255.255.0
Router(config-if)#ip address 192.168.10.4 255.255.255.0
Router(config-if)#ip address 192.168.10.4 255.255.255.128 Router(config-
if)#ip address 192.168.10.4 255.255.255.128
Router(config-if)#
Router(config-if)#exit
Router(config)#interface GigabitEthernet0/1
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/1, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed state to up
ip address 192.168.10.132 255.255.255.128
Router(config-if)#ip address 192.168.10.132 255.255.255.128
Router(config-if)#
Router(config-if)#
Router(config-if)#exit
Router(config)#interface GigabitEthernet0/1
Router(config-if)#
Router(config-if)#exit
Router(config)#interface GigabitEthernet0/0
Router(config-if)#exit

6. Output Diagram (Minimum 3 screenshot):




Cisco Packet Tracer
Logical
Physical
PC0
PC1
PC2
Physical
Config
Desktop
Programming
Attributes
IP Configuration
Interface
FastEthernet0
DHCP
Static
IPv4 Address
192.168.10.1
192.168.10.2
192.168.10.3
Subnet Mask
255.255.255.128
Default Gateway
192.168.10.4
DNS Server
0.0.0.0
IPv6 Configuration
Automatic
Static
IPv6 Address
Link Local Address
FE80::206:2AFF:FEE5:7B96
802.1X
Use 802.1X Security
Authentication
MD5
Username
Password
Top
192.168.10.0
/25 (X.X.X.10000000)
Subnet #1
Gig0/0
2911
Router0
2960-24TT
Switch0
Fa0/1
Fa0/2
Fa0/3
Fa0/4
Fa0
PC-PT
PC0
PC1
PC2
Time: 00:23:11
Simulation


Cisco Packet Tracer
Logical
Physical
PC0
Physical
Config
Desktop
Programming
Attributes
Command Prompt
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.10.131
Pinging 192.168.10.131 with 32 bytes of data:
Request timed out.
Reply from 192.168.10.131: bytes=32 time<1ms TTL=127
Reply from 192.168.10.131: bytes=32 time<1ms TTL=127
Reply from 192.168.10.131: bytes=32 time<1ms TTL=127
Ping statistics for 192.168.10.131:
Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\>ping 192.168.10.130
Pinging 192.168.10.130 with 32 bytes of data:
Request timed out.
Reply from 192.168.10.130: bytes=32 time<1ms TTL=127
Reply from 192.168.10.130: bytes=32 time<1ms TTL=127
Reply from 192.168.10.130: bytes=32 time<1ms TTL=127
Ping statistics for 192.168.10.130:
Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\>ping 192.168.10.129
Pinging 192.168.10.129 with 32 bytes of data:
Request timed out.
Reply from 192.168.10.129: bytes=32 time<1ms TTL=127
Reply from 192.168.10.129: bytes=32 time<1ms TTL=127
Reply from 192.168.10.129: bytes=32 time<1ms TTL=127
Ping statistics for 192.168.10.129:
Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
Top
192.168.10.0
/25 (X.X.X.10000000)
Gig0/0
Gig0/1
2911
Router0
Subnet #1
2960-24TT
Switch0
Switch1
Fa0/1
Fa0/2
Fa0/3
Fa0/4
Fa0
PC-PT
PC0
PC1
PC2
PC3
PC4
Time: 00:28:44
Realtime
Simulation
Copper Straight-Through

CONCLUSION (provide conclusion about this experiment):

In Conclusion, Subnetting is an essential skill for network administrators and IT professionals. By understanding how to divide IP address spaces into smaller subnets, you can significantly improve network efficiency, organization, and security

Rubrics for Experiment Assessment:

Rubrics	Good	Normal	Poor	Marks
Creation of Topology (4)	Created the topology, Identify the proper devices and making the connections **(4)**	Created the topology, Identify the proper devices, making the connections but missing some features **(3)**	Created wrong topology, Failed to Identify the proper devices and making connections **(1)**	
Verify the connectivity (4)	Verified the connectivity in all the levels **(4)**	Verified the connectivity at some levels (only some nodes) **(2)**	Verified the connectivity is not done. **(1)**	
Timely Completion (2)	Completed the lab before the allotted time **(2)**	Completed the lab after the deadline **(1)**	Did not submitted before grading **(0)**	
			Total	

Result:

Thus, the designing the Subnetting configuration using packet tracer was implemented and executed successfully.

Ex. No:	9
Name of the Experiment:	DHCP Configuration
Date:	

Objective(s):

To design and implement DHCP configuration using packet tracer.

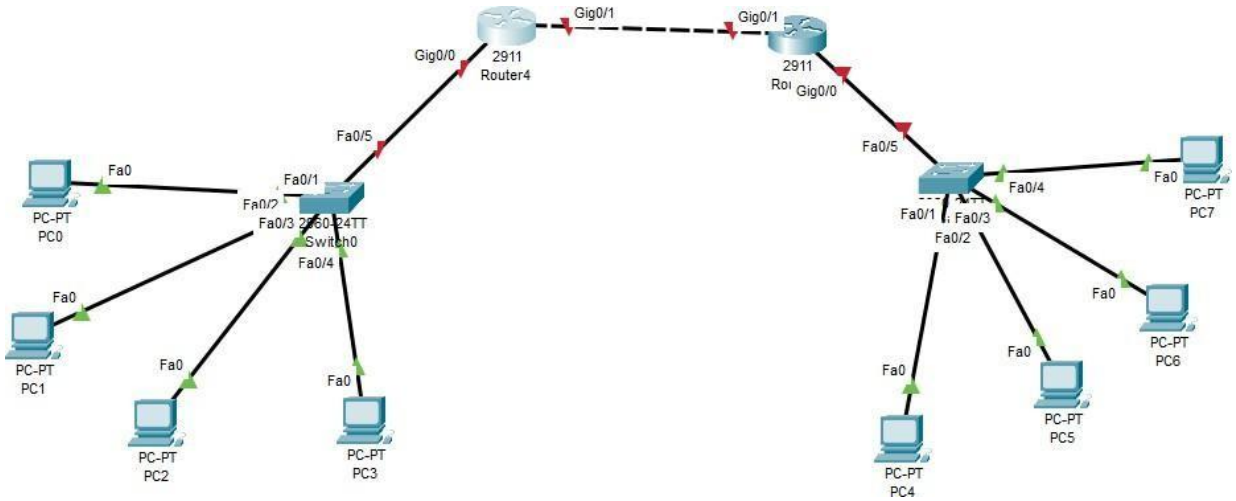
Introduction:

In this activity, you will continue to configure the Cisco 1841 ISR router for the customer network by configuring the DHCP service. The customer has several workstations that need to be automatically configured with IP addresses on the local subnet and appropriate DHCP options to allow access to the Internet.

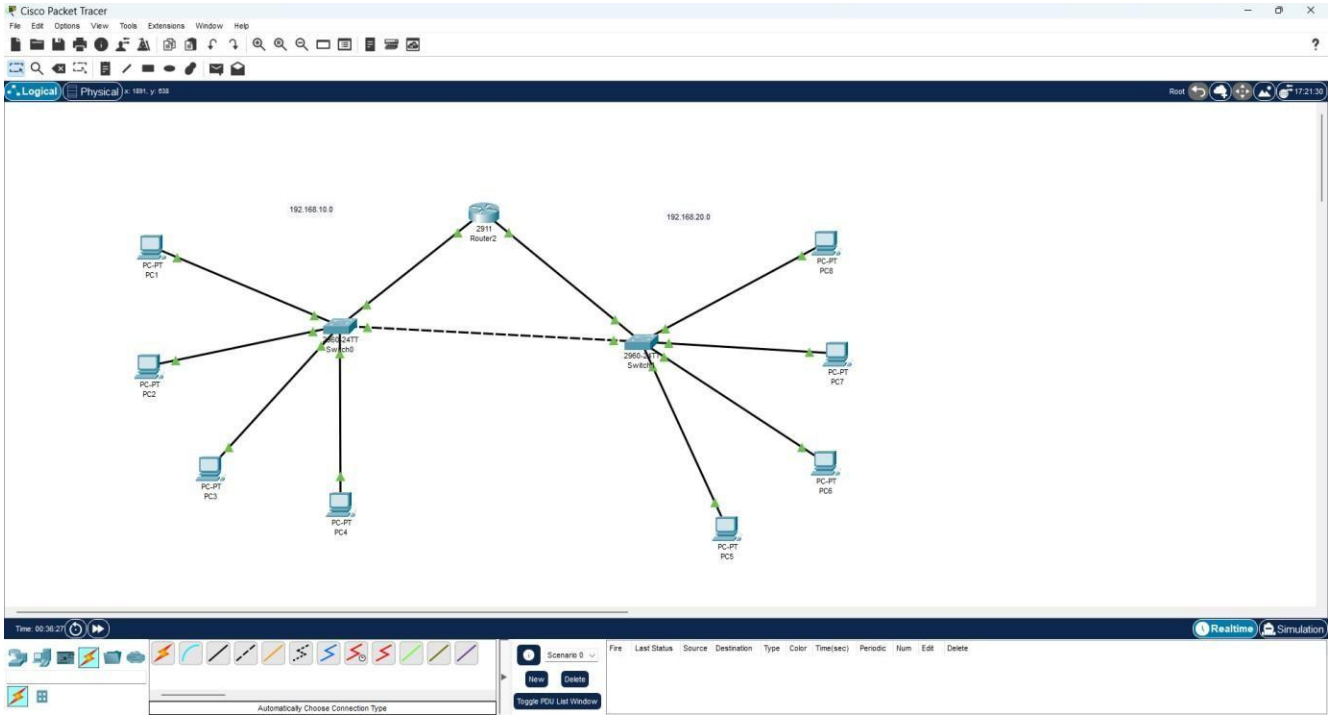
The DHCP pool will use the 192.168.1.0/24 network but the first 49 addresses are excluded. The default gateway and DNS server also need to be configured as 192.168.1.1 and 192.168.1.10. For this activity, both the user and privileged EXEC passwords are cisco.

Note: Packet Tracer does not currently support the domain name and lease period options. These options are not used in this activity.

1. **Device Requirements:**
   1. Router
   2. Switch
   3. PC
   4. Wires
2. **Network Diagram for your experiment (draw the diagram either hand drawing/ms paint or any other drawing tools)**



3. **Network Diagram (Packet tracer diagram before configuration):**



4. Configuration details:

Device Name	Interface Name	IP Address	Subnet mask	Default Gateway
Router 2	Gig0/0, Gig0/1,	192.168.10.1, 192.168.20.1	255.255.255.0, 255.255.255.0	
Switch 1	Fa0/4			
Switch 0	Fa0/4			
PC1	Fa0	192.168.10.2	255.255.255.0	192.168.10.1
PC2	Fa0	192.168.10.7	255.255.255.0	192.168.10.1
PC3	Fa0	192.168.10.8	255.255.255.0	192.168.10.1
PC4	Fa0	192.168.10.9	255.255.255.0	192.168.10.1
PC5	Fa0	192.168.20.2	255.255.255.0	192.168.20.1
PC6	Fa0	192.168.20.3	255.255.255.0	192.168.20.1
PC7	Fa0	192.168.20.4	255.255.255.0	192.168.20.1
PC8	Fa0	192.168.20.5	255.255.255.0	192.168.20.1

5. Describe step by step configuration steps properly (you may copy the commands used in the configuration tab and paste it.)

Router>

Router>

Router>en

Router#config ter

Enter confiquration commands, one per line. End vith CNTL/2.

Router (config)#inter

Router (config)#interface g
Router (config)#interface gigabitEthernet 0/0
Router (config-if)#ip add
Router (config-if)#ip address 192.168.10.1 255.255.255.0
Router (config-if)#no shut Router
(config-if):
LINK-5-CHANGED: Interface GigabitEtherner0/0, changed state to up
SLINEPROTO-5-UPDON: Line protocol on Interface GigabitEtherner0/0, changed state to
up Router(config-if)#exit Router (config):
Router (config)#int
Router (config)#interface g
Router (config)#interface gigabitEthernet 0/1
Router (config-if)#ip ad
Router (config-if)#ip address 152.168.20.1 255.255.255.0
Router (config-if)#no shut
Router (config-if)
SLINK-5-CHANGED: Interface GigabitEtherneto/1, changed state to up
SLINEPROTO-5-UPDOWN:Line protocol on InterfaceGigabitEthernet0/1, changed state to up
Router (config-if)#exit Router (config):
Router (config):
Router (config) #ip dh
Router (config)#ip dhcp p
Router (config)#ip dhcp poo1 10
Router (dhcp-config) #net
Router (dhcp-config)#network 192.168.10.0 255.255.255.0
Router (dhcp-config) #def
Router (dhcp-config) #default-router 192.168.10.1
Router (dhcp-config)#exit
Router (config)sipdh
Router (config)#ip dh
Router (config)#ip dhcp ex
Router (config)#ip dhcp excluded-address 152.168.10.3 152.168.10.6
Router (config):
Router (config)#dh
Router (config) #dhcp p
Router (config)#exit
Router#
SYS-5-CONFIG_I: Configured from console by console
Router#config ter
Enter configuration commands, one per line. End with CNTL/2.
Router (config)#ip dh
Router (config)#ip dhcp p
Router (config)t#p dhdp pool 20

Router (dhcp-config)#net

Router (dhcp-config) #network 192.168.20.0 255.255.255.0

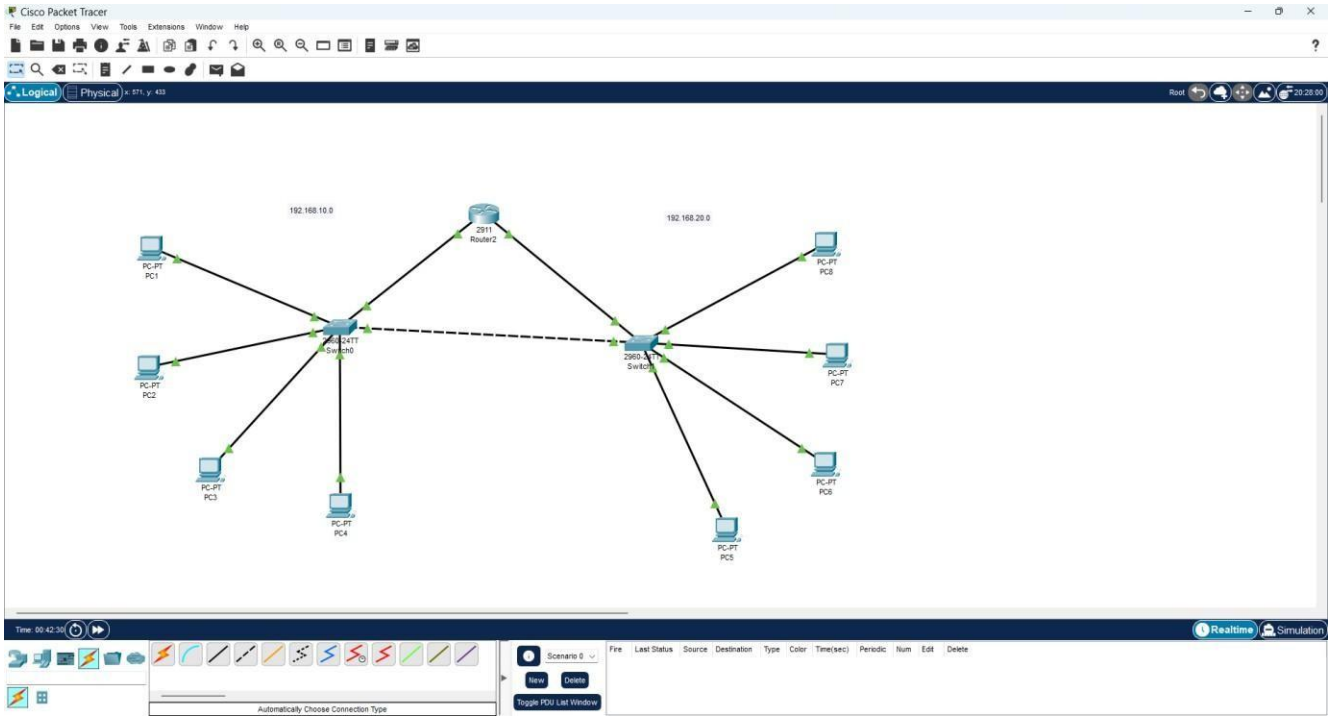
Router (dhdp-config)#def

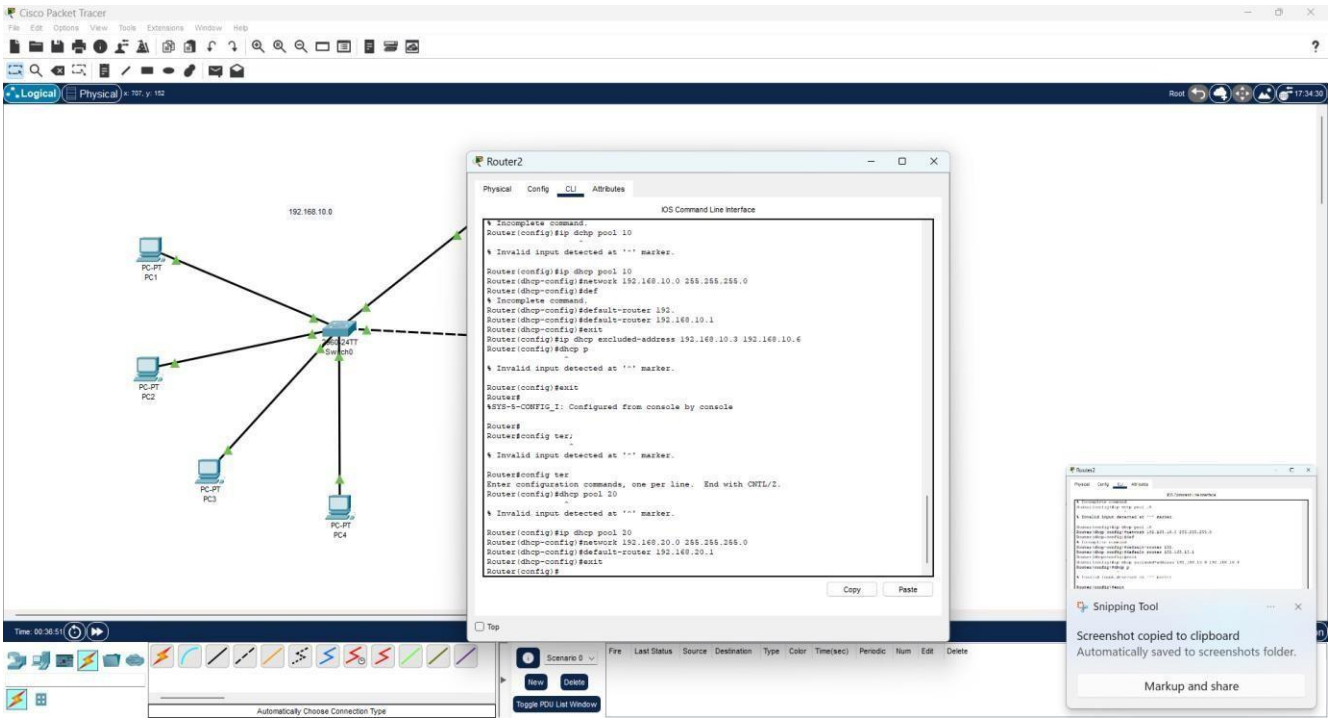
Router (dhdp-config)#default-router 192.168.20.1

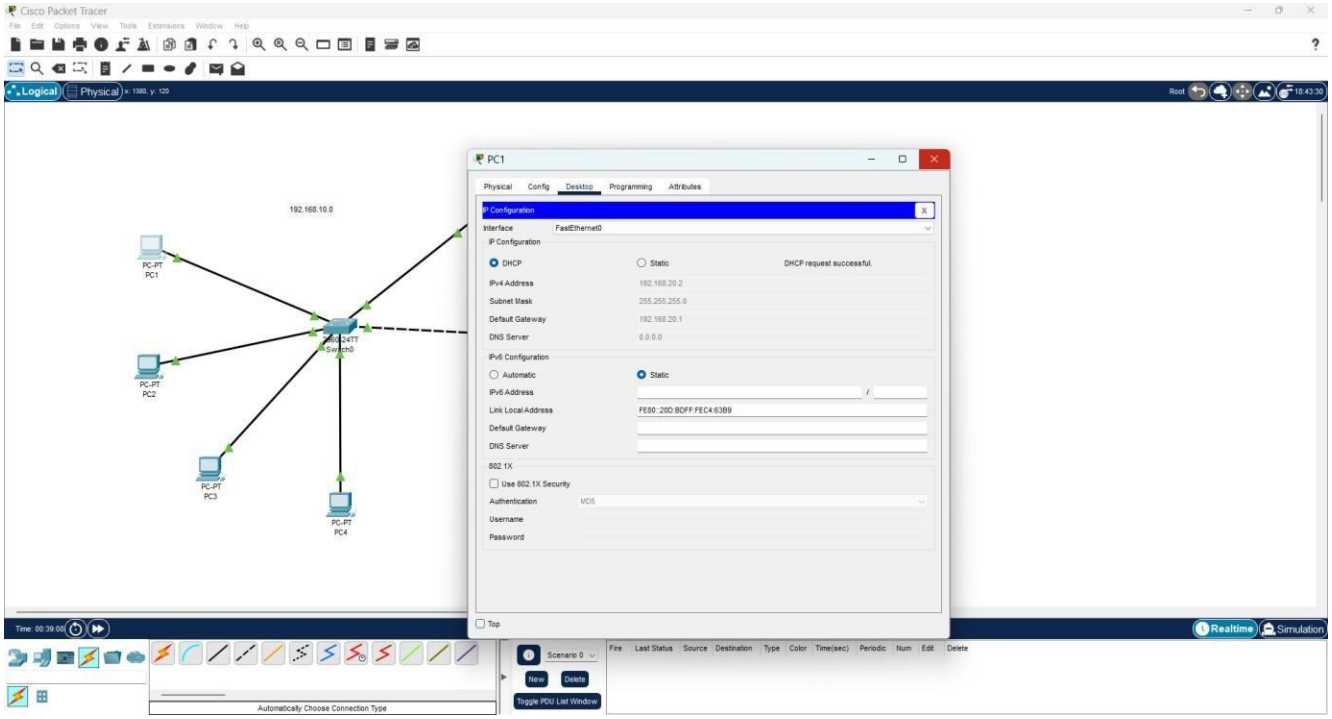
Router (dhdp-config)#exit

Router (config)#

6. Output Diagram (Minimum 3 screenshot):







CONCLUSION (provide conclusion about this experiment):

By configuring DHCP on a Cisco router within Packet Tracer, network administrators can automate the process of assigning IP addresses to devices on a network, streamlining network management by eliminating the need for manual configuration on each device, thus improving efficiency and reducing potential errors associated with manual IP address allocation.

Rubrics for Experiment Assessment:

Rubrics	Good	Normal	Poor	Marks
Creation of Topology (4)	Created the topology, Identify the proper devices and making the connections **(4)**	Created the topology, Identify the proper devices, making the connections But missing some features **(3)**	Created wrong topology, Failed to Identify the proper devices and making connections **(1)**	
Verify the connectivity (4)	Verified the connectivity in all the levels **(4)**	Verified the connectivity at some levels (only some nodes) **(2)**	Verified the connectivity is not done. **(1)**	
Timely Completion (2)	Completed the lab before the allotted time **(2)**	Completed the lab after the deadline **(1)**	Did not submitted before grading **(0)**	
			Total	

Result:

Thus, the designing the DHCP Configuration using packet tracer was implemented and executed successfully.

Ex No :	10
Name of the Experiment	**Capture and Analyse TCP and IP packets**
Date :	

Objective(s):

To capture and analyse TCP and IP packet using Wireshark.

Introduction

Packet Analysis is a technique used to intercept data in information security, where many of the tools that are used to secure the network can also be used by attackers to exploit and compromise the same network. The core objective of sniffing is to steal data, such as sensitive information, email text, etc., or sniff the traffic that is being transmitted between two parties.

Packet Analysis involves intercepting network traffic between two target network nodes and capturing network packets exchanged between nodes. A packet sniffer is referred to as a network monitor that is used legitimately by a network administrator to monitor the network for vulnerabilities by capturing the network traffic and should there be any issues, proceeds to troubleshoot the same. Similarly, sniffing tools can be used by attackers in promiscuous mode to capture and analyze all the network traffic. Once attackers have captured the network traffic they can analyze the packets and view the user name and password information in a given network as this information is transmitted in a cleartext format. An attacker can easily intrude into a network using this login information and compromise other systems on the network.

Hence, it is very crucial for an Information Security Auditor or a Penetration Tester to be familiar with network traffic analyzers and he or she should be able to maintain and monitor a network to detect rogue packet sniffers, MAC attacks, DHCP attacks, ARP poisoning, spoofing, or DNS poisoning, and know the types of information that can be detected from the captured data and use the information to keep the network running smoothly.

Exercise:

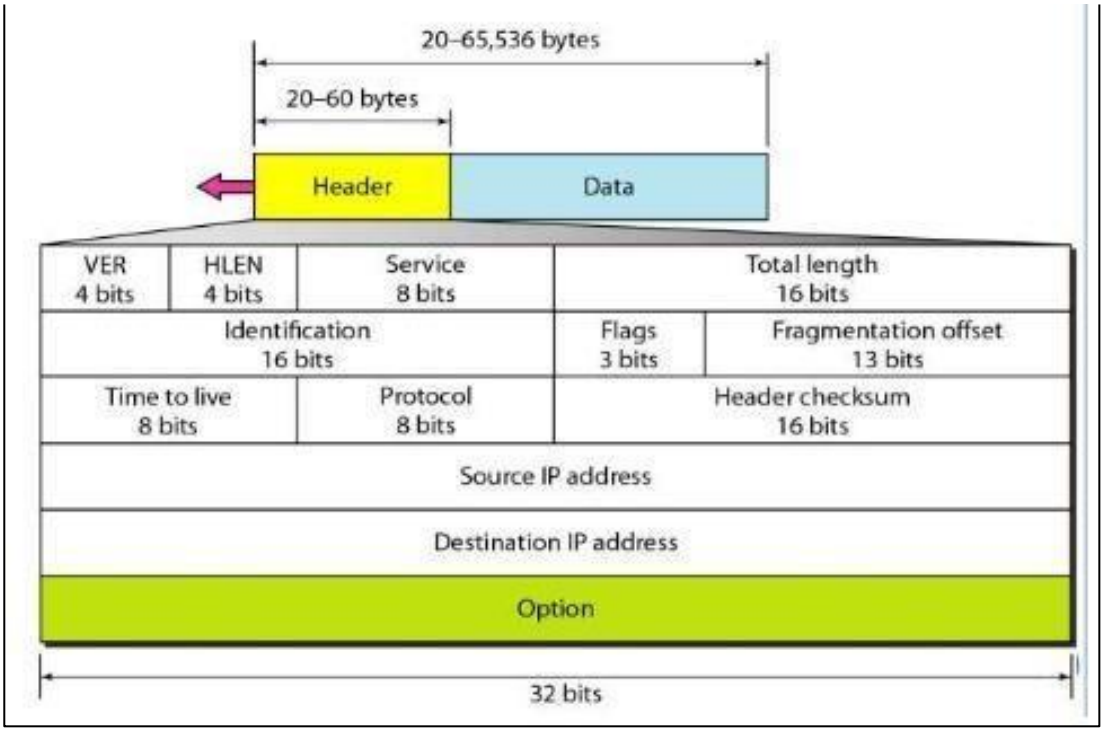
1. Visit any one website by opening a browser and fill your machine details (attach relevant screenshots).

Parameter	Value
Your Machine IP Address.	10.1.10.139
Your Machine MAC Address	50-5A-65-8F-24-97
Default Gateway address	10.1.0.1
Website URL	https://www.kalasalingam.ac.in
Website IP Address	18.67.161.45

2. Fill the following IP packet details:

Field Name	Field Length (no of bits)	Field value
Destination MAC address	48 bits	c8-4f-86-fc-00-0f
Source MAC address	48 bits	50-5A-65-8F-24-97
Destination IP address	32 bits	18.67.161.45
Source IP Address	32 bits	10.1.10.139
Destination TCP port	16 bits	59960
Source TCP port	16 bits	443

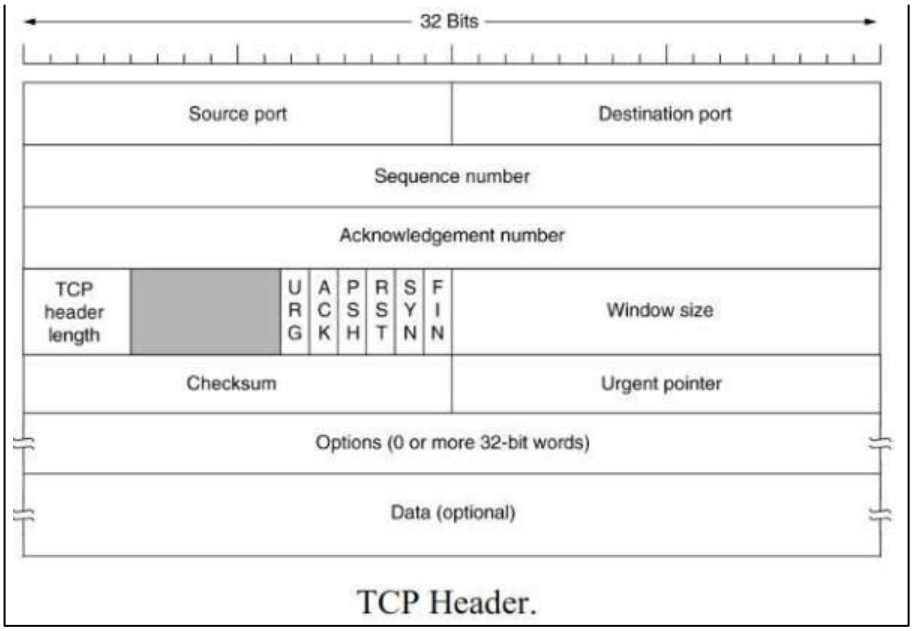
3. Fill the details as per the IP frame format .(highlight the details for each of the output and paste screenshot)



Field Name	Field Value (# of bits)	Field Value (Either Binary or Hex Value)
Version	4	0100

Header Length	4	0101
Type of service	8	0x00
Datagram Length	16	1181
16 bit Identifier	16	0xc3c2
Flags	3	010
13-bit Fragment offset	13	0 0000 0000 0000
Time-to-live	8	fa(250)
Upper layer protocol	8	6
Header Checksum	16	0xaec7
32 bit Source Address	32	18.67.161.45
32 bit destination address	32	10.1.10.139
Options (if any)	-	-
Date	-	08-03-2025

TCP Header Format:

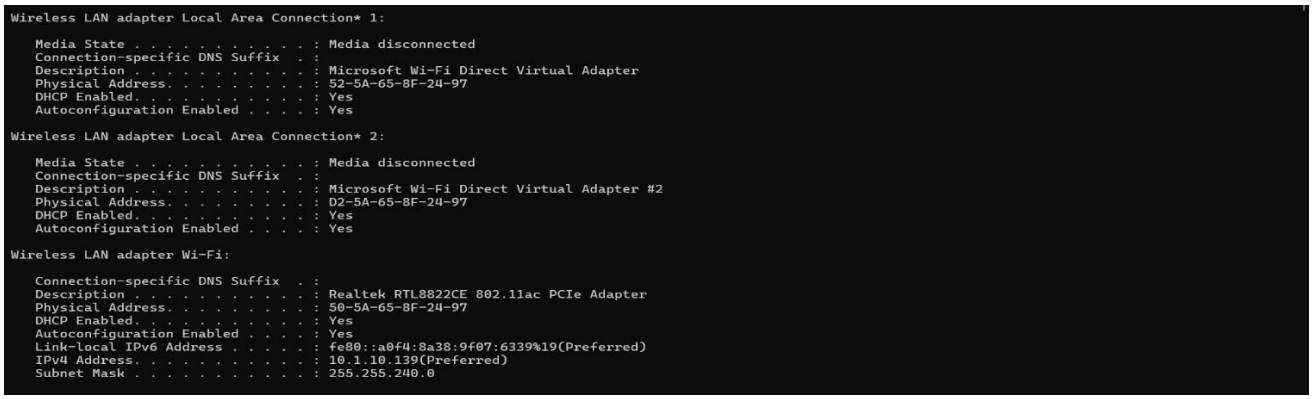


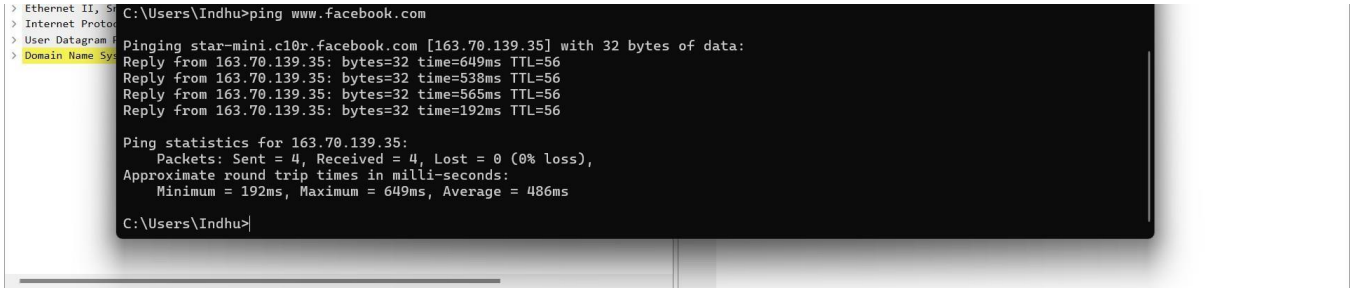
TCP Header.

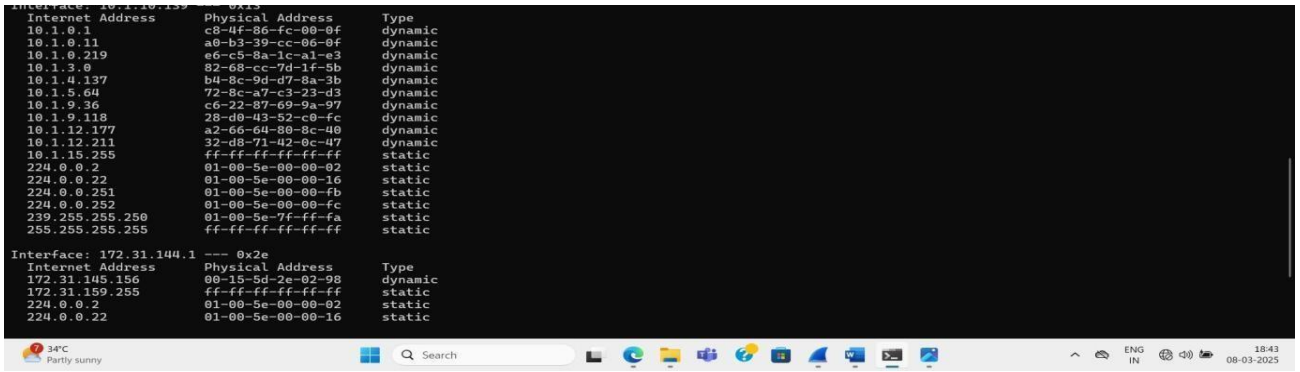
Field Name	Field Value (# of bits)	Field Value (Either Binary or Hex Value)
Source Port	16 bits	443
Destination Port	16 bits	59960
Sequence No.	32 bits	51595
Acknowledgement No	32 bits	2845
Header Length	4 bits	5

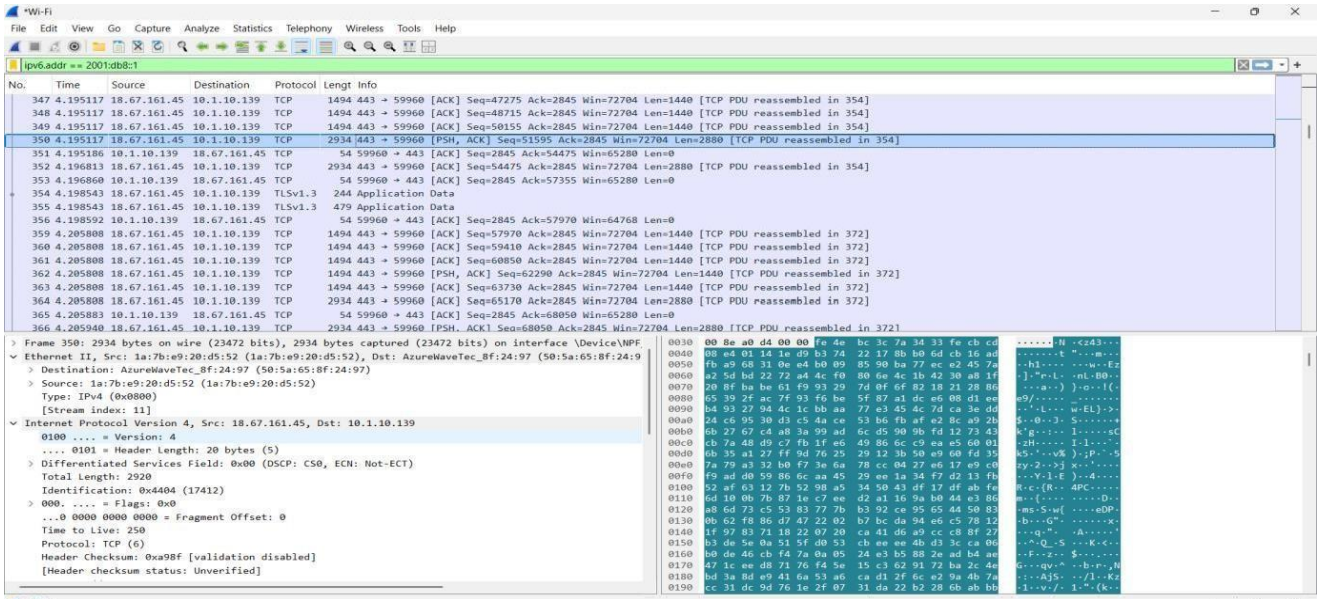
FLSGS (URG,PSH,ACK,RST,SYN,FIN)	6 bits	011000
Receive Window Size	16 bits	72704
Checksum	16 bits	0xa0d4
Urgent Pointer	6 bits	0
Options	-	-
Data	--	-

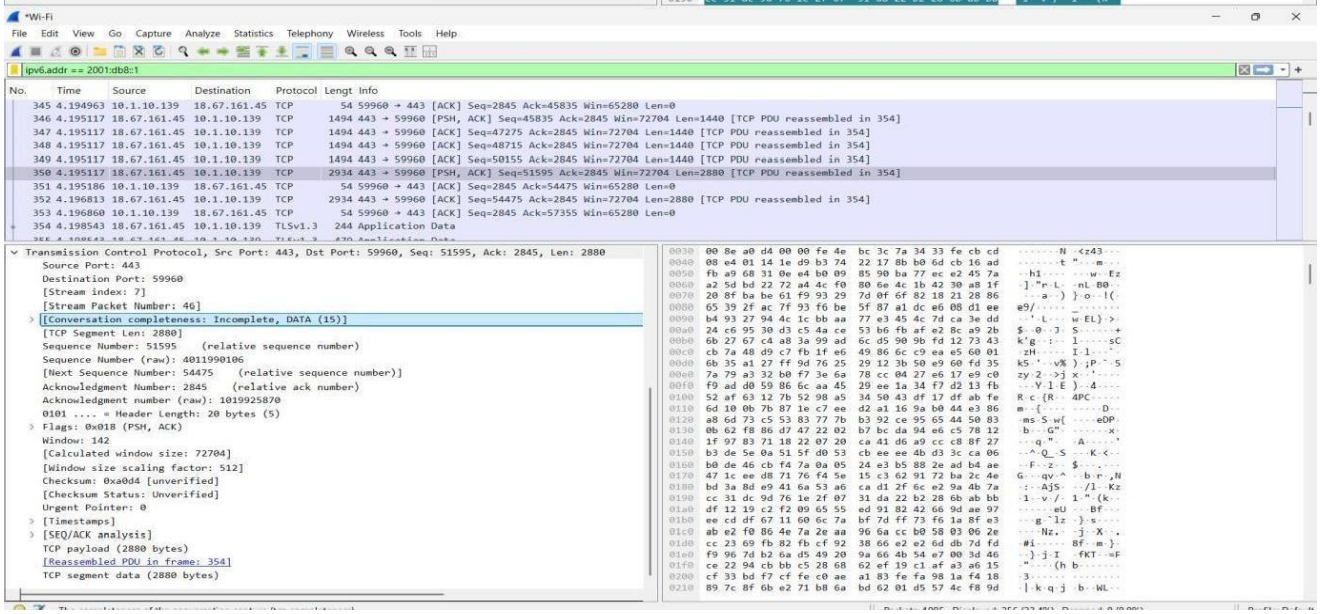
Paste the screenshot and highlight the above details:

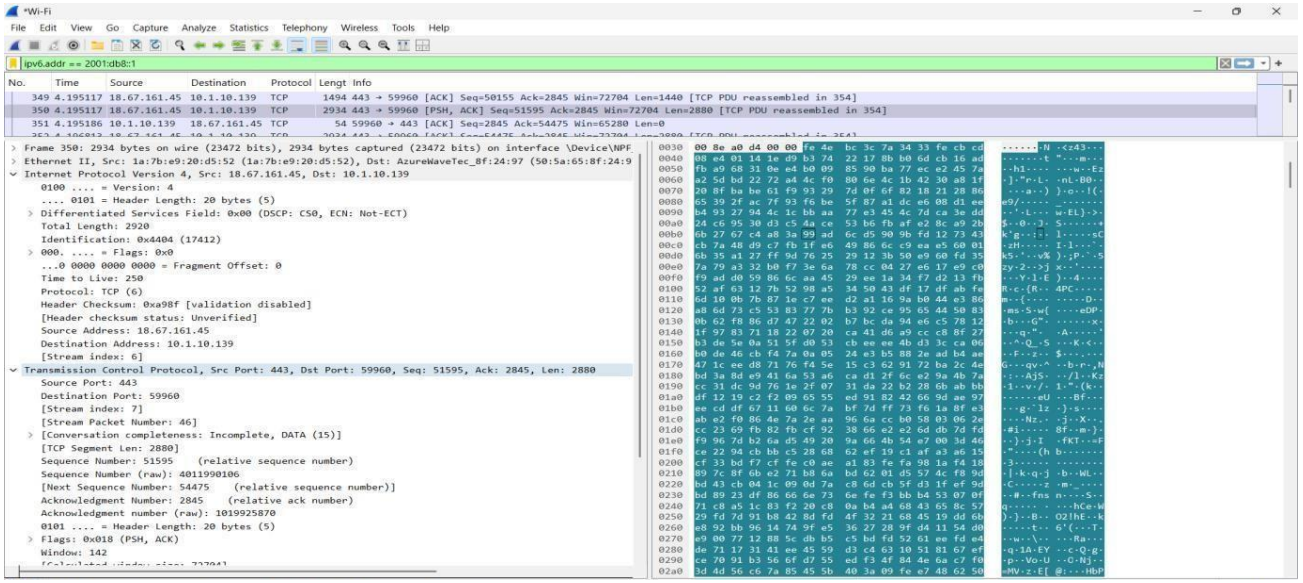







*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
ipv6.addr == 2001:db8::1
No. Time Source Destination Protocol Lengt Info
347 4.195117 18.67.161.45 10.1.10.139 TCP 1494 443 → 59960 [ACK] Seq=47275 Ack=2845 Win=72704 Len=1440 [TCP PDU reassembled in 354]
348 4.195117 18.67.161.45 10.1.10.139 TCP 1494 443 → 59960 [ACK] Seq=48715 Ack=2845 Win=72704 Len=1440 [TCP PDU reassembled in 354]
349 4.195117 18.67.161.45 10.1.10.139 TCP 1494 443 → 59960 [ACK] Seq=50155 Ack=2845 Win=72704 Len=1440 [TCP PDU reassembled in 354]
350 4.195117 18.67.161.45 10.1.10.139 TCP 2934 443 → 59960 [PSH, ACK] Seq=51595 Ack=2845 Win=72704 Len=2880 [TCP PDU reassembled in 354]
351 4.195186 10.1.10.139 18.67.161.45 TCP 54 59960 → 443 [ACK] Seq=2845 Ack=54475 Win=65280 Len=0
352 4.196813 18.67.161.45 10.1.10.139 TCP 2934 443 → 59960 [ACK] Seq=54475 Ack=2845 Win=72704 Len=2880 [TCP PDU reassembled in 354]
353 4.196860 10.1.10.139 18.67.161.45 TCP 54 59960 → 443 [ACK] Seq=2845 Ack=57355 Win=65280 Len=0
354 4.198543 18.67.161.45 10.1.10.139 TLSv1.3 244 Application Data
355 4.198543 18.67.161.45 10.1.10.139 TLSv1.3 479 Application Data
356 4.198592 10.1.10.139 18.67.161.45 TCP 54 59960 → 443 [ACK] Seq=2845 Ack=57970 Win=64768 Len=0
359 4.205808 18.67.161.45 10.1.10.139 TCP 1494 443 → 59960 [ACK] Seq=57970 Ack=2845 Win=72704 Len=1440 [TCP PDU reassembled in 372]
360 4.205808 18.67.161.45 10.1.10.139 TCP 1494 443 → 59960 [ACK] Seq=59410 Ack=2845 Win=72704 Len=1440 [TCP PDU reassembled in 372]
361 4.205808 18.67.161.45 10.1.10.139 TCP 1494 443 → 59960 [ACK] Seq=60850 Ack=2845 Win=72704 Len=1440 [TCP PDU reassembled in 372]
362 4.205808 18.67.161.45 10.1.10.139 TCP 1494 443 → 59960 [PSH, ACK] Seq=62290 Ack=2845 Win=72704 Len=1440 [TCP PDU reassembled in 372]
363 4.205808 18.67.161.45 10.1.10.139 TCP 1494 443 → 59960 [ACK] Seq=63730 Ack=2845 Win=72704 Len=1440 [TCP PDU reassembled in 372]
364 4.205808 18.67.161.45 10.1.10.139 TCP 2934 443 → 59960 [ACK] Seq=65170 Ack=2845 Win=72704 Len=2880 [TCP PDU reassembled in 372]
365 4.205883 10.1.10.139 18.67.161.45 TCP 54 59960 → 443 [ACK] Seq=2845 Ack=68050 Win=65280 Len=0
366 4.205940 18.67.161.45 10.1.10.139 TCP 2934 443 → 59960 [PSH, ACK] Seq=68050 Ack=2845 Win=72704 Len=2880 [TCP PDU reassembled in 372]
Frame 350: 2934 bytes on wire (23472 bits), 2934 bytes captured (23472 bits) on interface \Device\NPF
Ethernet II, Src: 1a:7b:e9:20:d5:52 (1a:7b:e9:20:d5:52), Dst: AzureWaveTec_8f:24:97 (50:5a:65:8f:24:9
Destination: AzureWaveTec_8f:24:97 (50:5a:65:8f:24:97)
Source: 1a:7b:e9:20:d5:52 (1a:7b:e9:20:d5:52)
Type: IPv4 (0x0800)
[Stream index: 11]
Internet Protocol Version 4, Src: 18.67.161.45, Dst: 10.1.10.139
0100 .... = Version: 4
.... 0101 = Header Length: 20 bytes (5)
Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
Total Length: 2920
Identification: 0x4404 (17412)
000. .... = Flags: 0x0
...0 0000 0000 0000 = Fragment Offset: 0
Time to Live: 250
Protocol: TCP (6)
Header Checksum: 0xa98f [validation disabled]
[Header checksum status: Unverified]


*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
ipv6.addr == 2001:db8::1
No. Time Source Destination Protocol Lengt Info
345 4.194963 10.1.10.139 18.67.161.45 TCP 54 59960 → 443 [ACK] Seq=2845 Ack=45835 Win=65280 Len=0
346 4.195117 18.67.161.45 10.1.10.139 TCP 1494 443 → 59960 [PSH, ACK] Seq=45835 Ack=2845 Win=72704 Len=1440 [TCP PDU reassembled in 354]
347 4.195117 18.67.161.45 10.1.10.139 TCP 1494 443 → 59960 [ACK] Seq=47275 Ack=2845 Win=72704 Len=1440 [TCP PDU reassembled in 354]
348 4.195117 18.67.161.45 10.1.10.139 TCP 1494 443 → 59960 [ACK] Seq=48715 Ack=2845 Win=72704 Len=1440 [TCP PDU reassembled in 354]
349 4.195117 18.67.161.45 10.1.10.139 TCP 1494 443 → 59960 [ACK] Seq=50155 Ack=2845 Win=72704 Len=1440 [TCP PDU reassembled in 354]
350 4.195117 18.67.161.45 10.1.10.139 TCP 2934 443 → 59960 [PSH, ACK] Seq=51595 Ack=2845 Win=72704 Len=2880 [TCP PDU reassembled in 354]
351 4.195186 10.1.10.139 18.67.161.45 TCP 54 59960 → 443 [ACK] Seq=2845 Ack=54475 Win=65280 Len=0
352 4.196813 18.67.161.45 10.1.10.139 TCP 2934 443 → 59960 [ACK] Seq=54475 Ack=2845 Win=72704 Len=2880 [TCP PDU reassembled in 354]
353 4.196860 10.1.10.139 18.67.161.45 TCP 54 59960 → 443 [ACK] Seq=2845 Ack=57355 Win=65280 Len=0
354 4.198543 18.67.161.45 10.1.10.139 TLSv1.3 244 Application Data
Transmission Control Protocol, Src Port: 443, Dst Port: 59960, Seq: 51595, Ack: 2845, Len: 2880
Source Port: 443
Destination Port: 59960
[Stream index: 7]
[Stream Packet Number: 46]
[Conversation completeness: Incomplete, DATA (15)]
[TCP Segment Len: 2880]
Sequence Number: 51595 (relative sequence number)
Sequence Number (raw): 4011990106
[Next Sequence Number: 54475 (relative sequence number)]
Acknowledgment Number: 2845 (relative ack number)
Acknowledgment number (raw): 1019925870
0101 .... = Header Length: 20 bytes (5)
Flags: 0x018 (PSH, ACK)
Window: 142
[Calculated window size: 72704]
[Window size scaling factor: 512]
Checksum: 0xa0d4 [unverified]
[Checksum Status: Unverified]
Urgent Pointer: 0
[Timestamps]
[SEQ/ACK analysis]
TCP payload (2880 bytes)
[Reassembled PDU in frame: 354]
TCP segment data (2880 bytes)


*Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
ipv6.addr == 2001:db8::1
No. Time Source Destination Protocol Lengt Info
349 4.195117 18.67.161.45 10.1.10.139 TCP 1494 443 → 59960 [ACK] Seq=50155 Ack=2845 Win=72704 Len=1440 [TCP PDU reassembled in 354]
350 4.195117 18.67.161.45 10.1.10.139 TCP 2934 443 → 59960 [PSH, ACK] Seq=51595 Ack=2845 Win=72704 Len=2880 [TCP PDU reassembled in 354]
351 4.195186 10.1.10.139 18.67.161.45 TCP 54 59960 → 443 [ACK] Seq=2845 Ack=54475 Win=65280 Len=0
Frame 350: 2934 bytes on wire (23472 bits), 2934 bytes captured (23472 bits) on interface \Device\NPF
Ethernet II, Src: 1a:7b:e9:20:d5:52 (1a:7b:e9:20:d5:52), Dst: AzureWaveTec_8f:24:97 (50:5a:65:8f:24:9
Internet Protocol Version 4, Src: 18.67.161.45, Dst: 10.1.10.139
0100 .... = Version: 4
.... 0101 = Header Length: 20 bytes (5)
Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
Total Length: 2920
Identification: 0x4404 (17412)
000. .... = Flags: 0x0
...0 0000 0000 0000 = Fragment Offset: 0
Time to Live: 250
Protocol: TCP (6)
Header Checksum: 0xa98f [validation disabled]
[Header checksum status: Unverified]
Source Address: 18.67.161.45
Destination Address: 10.1.10.139
[Stream index: 6]
Transmission Control Protocol, Src Port: 443, Dst Port: 59960, Seq: 51595, Ack: 2845, Len: 2880
Source Port: 443
Destination Port: 59960
[Stream index: 7]
[Stream Packet Number: 46]
[Conversation completeness: Incomplete, DATA (15)]
[TCP Segment Len: 2880]
Sequence Number: 51595 (relative sequence number)
Sequence Number (raw): 4011990106
[Next Sequence Number: 54475 (relative sequence number)]
Acknowledgment Number: 2845 (relative ack number)
Acknowledgment number (raw): 1019925870
0101 .... = Header Length: 20 bytes (5)
Flags: 0x018 (PSH, ACK)
Window: 142

Rubrics for Wireshark labs:

Rubrics	Excellent	Fair	Poor	Marks
Understanding (2)	Understand the Concept very well. (2)	Understand the Concept (1)	Poor Understand the Concept (0)	
Usage of filters (3)	Identified and applied the filter correctly (3)	Identified the filter, but not applied correctly (2-1)	Couldn't identify and apply the filter. Just captured the packets (1)	
Attach relevant Screenshots (3)	clearly Highlighted the answers and attached the screenshots (3)	attached the screenshots, but not highlighted. (2-1)	Did not attach the screenshots (0)	
On time Submission (2)	Early or on time submission (2)	Submitted after deadline (1)	Did not Submit (0)	
			Total	

CONCLUSION:

Thus, the Capture and Analyze TCP and IP packets has implemented successfully by using WIRESHARK.

Ex.No:11	
Date :	**Capture and Analyzing TCP 3 way handshake**

Objective(s):

To capture and analyse TCP 3-way handshake packet using Wireshark.

Introduction:

TCP or Transmission Control Protocol is one of the most important protocols or standards for enabling communication possible amongst devices present over a particular network. It has algorithms that solve complex errors arising in packet communications, i.e. corrupted packets, invalid packets, duplicates, etc. Since it is used with IP(Internet Protocol), many times it is also referred to as TCP/IP. In order to start a communication, the TCP first establishes a connection using the three-way-handshake. TCP's efficiency over other protocols lies in its error detecting and correction attribute. Not only this, it organizes packets and segments larger data into a number of packets without disrupting the integrity of the data.

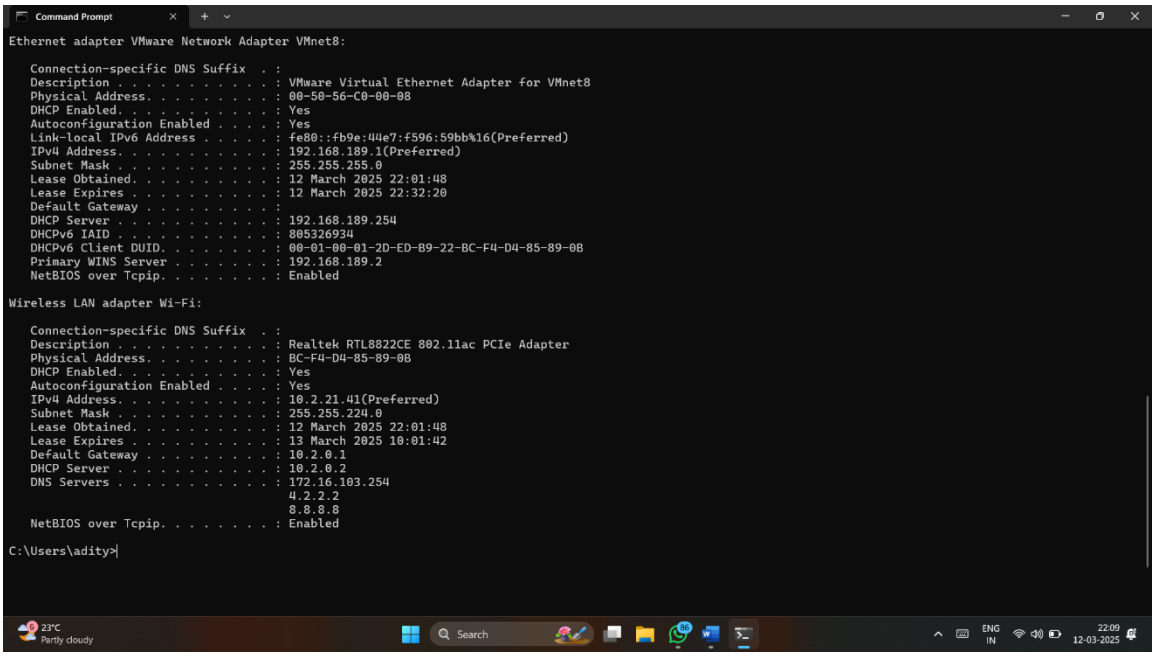
So now we are a bit familiar with TCP, let's look at how we can analyze TCP using Wireshark, which is the most widely used protocol analyzer in the world.

Here you will have the list of TCP packets. The first three packets of this list are part of the three-way handshake mechanism of TCP to establish a connection. Let's get a basic knowledge of this mechanism which happens in the following 3 steps:

- A synchronization packet (SYN) is sent by your local host IP to the server it desires to connect to.
- The server reciprocates by sending an acknowledgment packet (ACK) to the local host signaling that it has received the SYN request of the host IP to connect and also sends a synchronization packet (SYN) to the local host to confirm the connection. So this one is basically an SYN+ACK packet.
- The host answers this request by sending the ACK on receiving the SYN of the server.

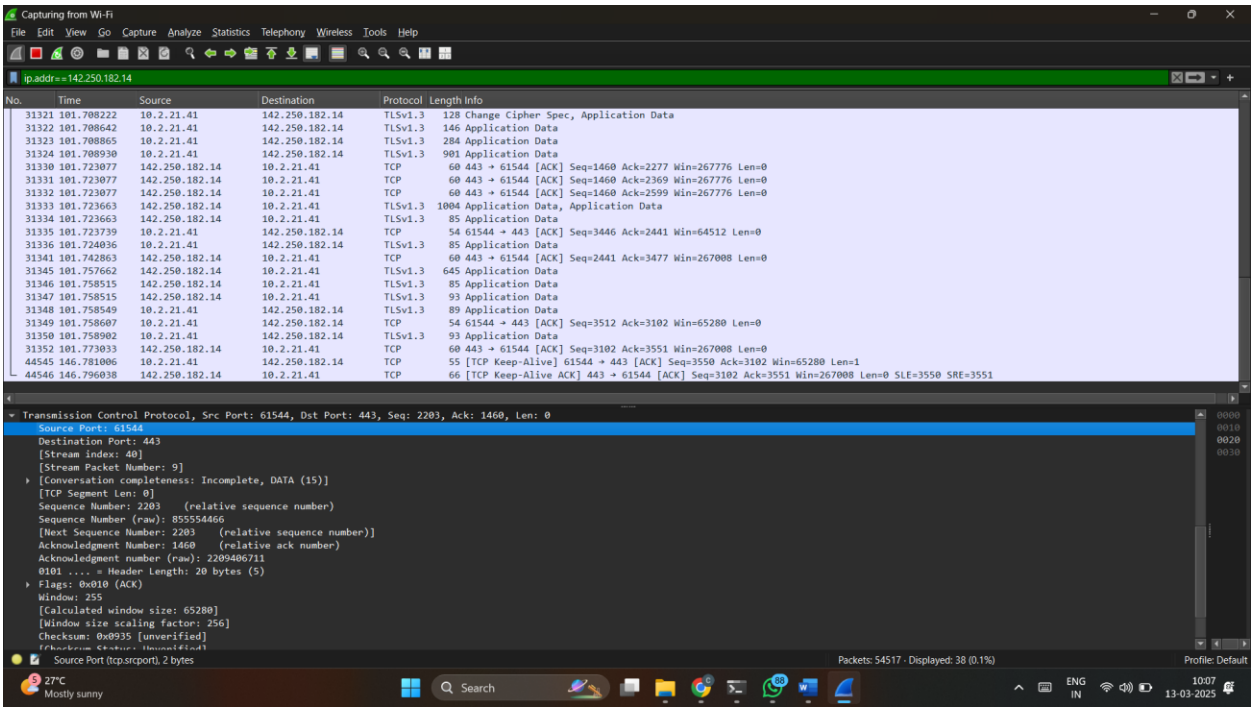
1. Visit any one website by opening a browser fill your machine details (attach relevant screenshots).

Parameter	Value
Your Machine IP Address.	10.2.21.41
Your Machine MAC Address	BC-F4-D4-85-89-0B
Default Gateway address	10.2.0.1
Website URL	https://google.com
Website IP Address	142.250.182.14

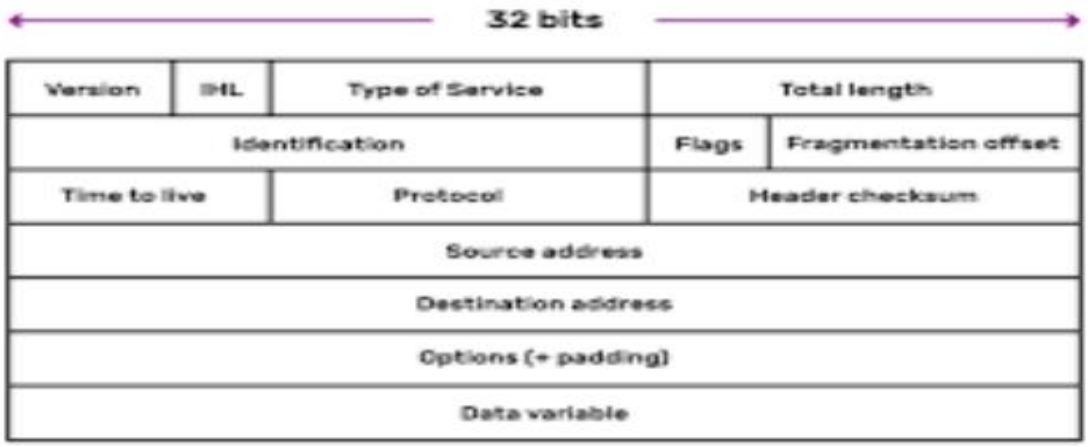


2. Fill the following details:

Field Name	Field Length (no of bits)	Field value
Destination MAC address	48	C8:4f:86:fc:00:10
Source MAC address	48	BC-F4-D4-85-89-0B
Destination IP address	32	142.250.182.14
Source IP Address	32	10.2.21.41
Destination TCP port	16	443
Source TCP port	16	61544

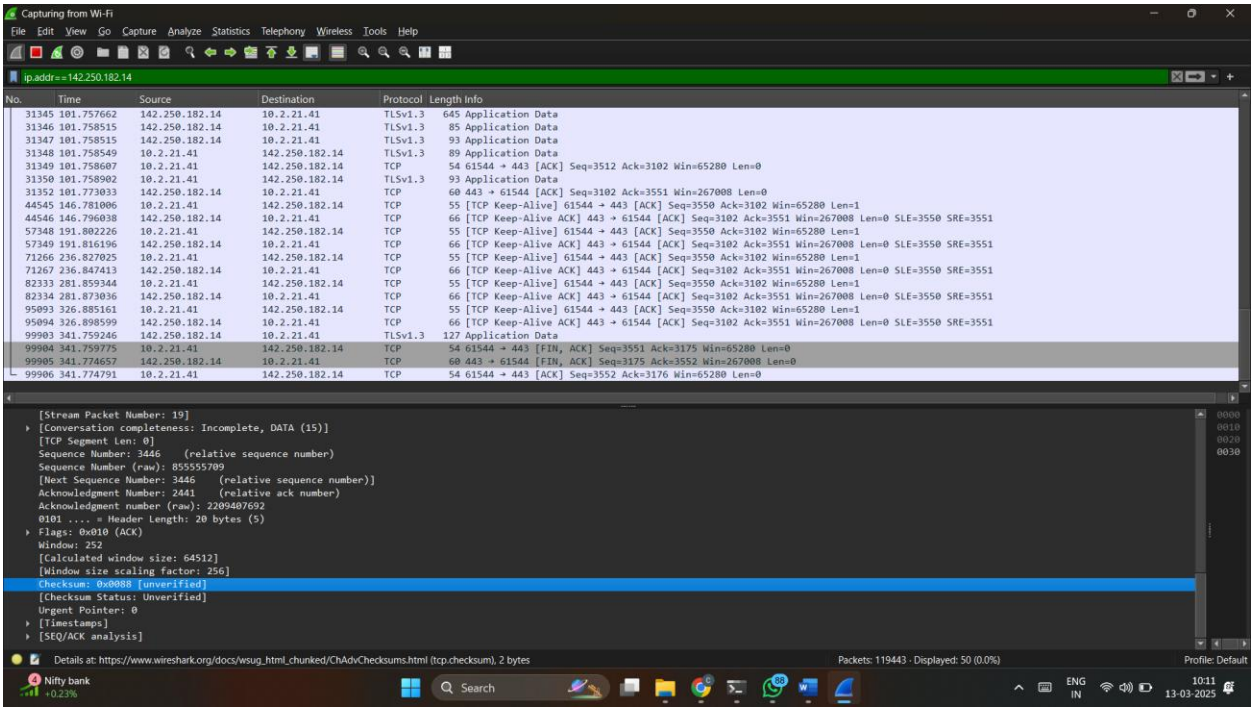


3. Fill the details as per the IP frame format .(highlight the details for each of the output and paste screenshot)

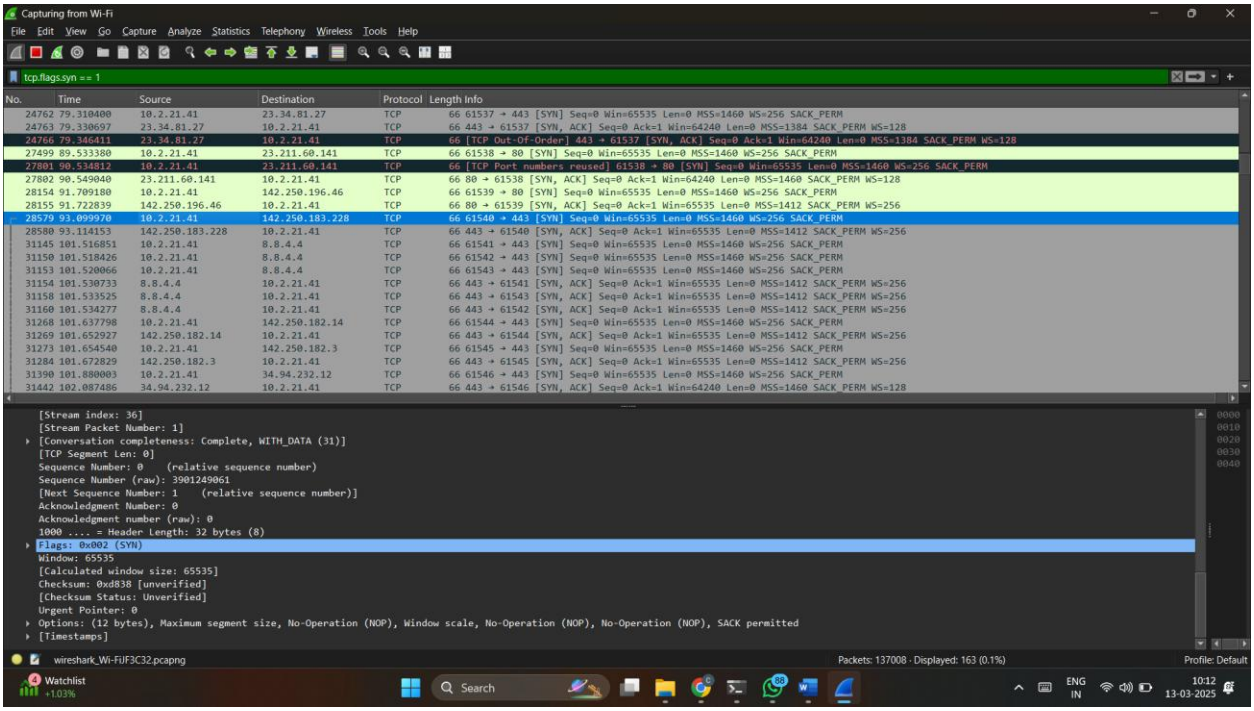


Version : 4	IHL : 20	TOS :0	Total Length : 40

Identification : 0x3e32(15922)		Flags: 0010	Fragmentation offset: 0
TTL : 128	Protocol : TCP(6)	Header Checksum : 0x0088	
Source Address : 10.2.21.41			
Destination address : 142.250.182.14			
Options (+ Padding)			
Data Variable			

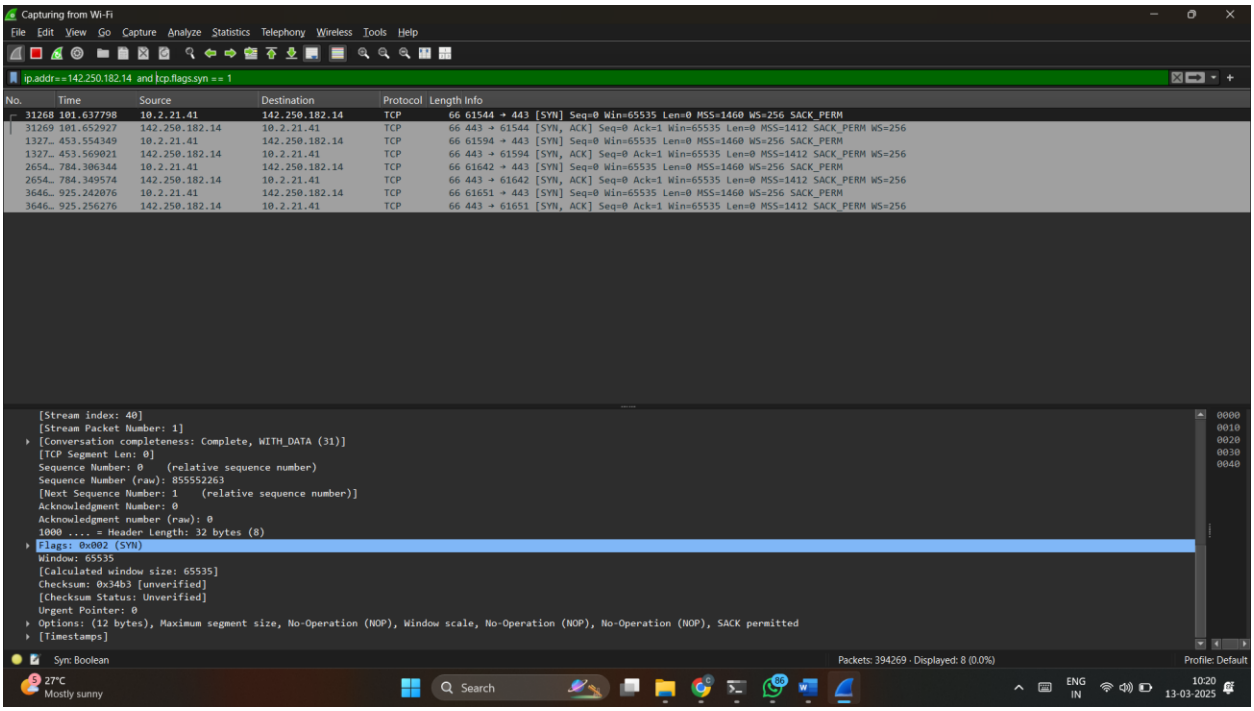


4. Using the Wireshark capture of the first TCP session startup (SYN bit set to 1), fill in information about the TCP header. (paste screenshot for each of the output). Capture the packet and analyze it.



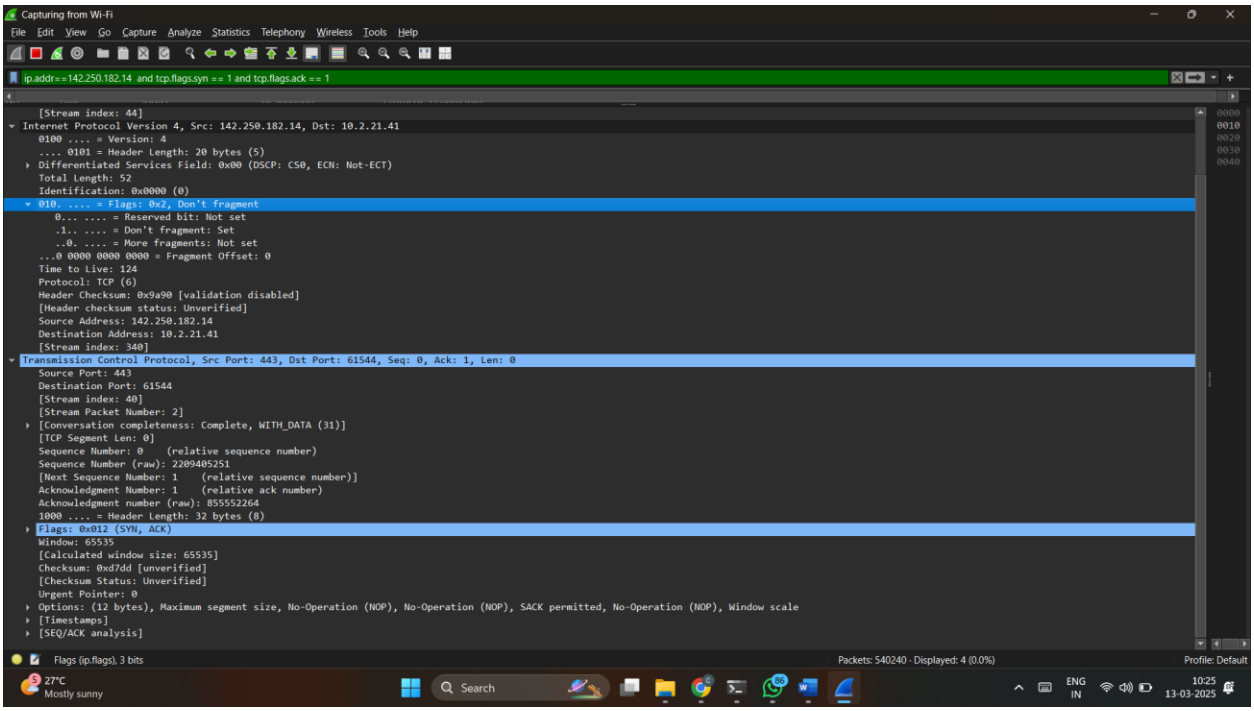
5. Fill in the following information regarding the SYN message.(highlight the details for each of the output and paste screenshot)

Source IP address	10.2.21.41
Destination IP address	142.250.182.14
Source port number	443
Destination port number	61544
Sequence number	0
Acknowledgement number	1
Flags	0000010
Header length	32 bytes(8)
Window size	65535
Checksum	0x34b3



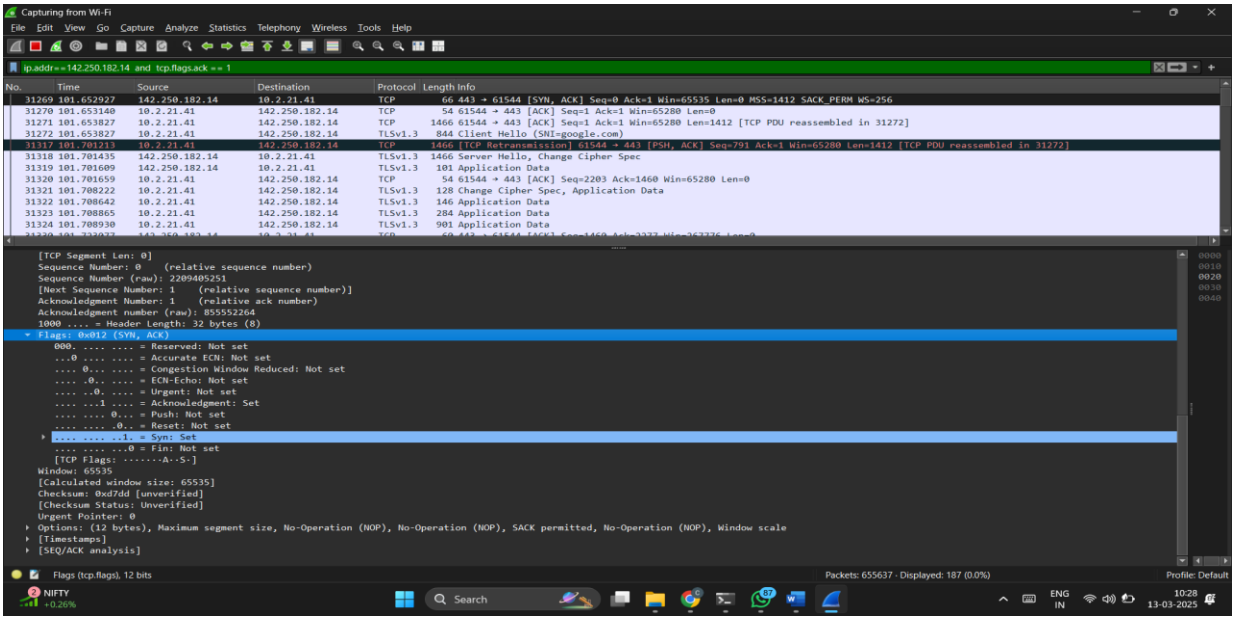
6. Fill in the following information regarding the SYN-ACK message .(highlight the details for each of the output and paste screenshot)

Source IP address	10.2.21.41
Destination IP address	142.250.182.14
Source port number	443
Destination port number	61544
Sequence number	0
Acknowledgement number	1
Header length	32 bytes(8)
Window size	65535
Flags	0010
Checksum	0xd7dd



7. Fill in the following information regarding the ACK message. .(highlight the details for each of the output and paste screenshot)

Source IP address	10.2.21.41
Destination IP address	142.250.182.14
Source port number	443
Destination port number	61544
Sequence number	0
Acknowledgement number	1
Header length	32 bytes(8)
Window size	65535
Flags	00000100
Checksum	0xd7dd



Rubrics for Wireshark labs: (To be Filled by the Class Teacher)

Rubrics	**Excellent**	**Fair**	**Poor**	**Marks**
Understanding (2)	Understand the Concept very well. (2)	Understand the Concept (1)	Poor Understand the Concept (0)	
Usage of filters (3)	Identified and applied the filter correctly (3)	Identified the filter, but not applied correctly (2-1)	Couldn't identify and apply the filter. Just captured the packets (1)	
Attach relevant Screenshots (3)	clearly Highlighted the answers and attached the screenshots (3)	attached the screenshots, but not highlighted. (2-1)	Did not attach the screenshots (0)	
On time Submission (2)	Early or on time submission (2)	Submitted after deadline (1)	Did not Submit (0)	
Total				

Result: Thus the implementation of **Capture and Analyzing TCP 3 way handshake** the has been successfully executed using cisco packet racer.

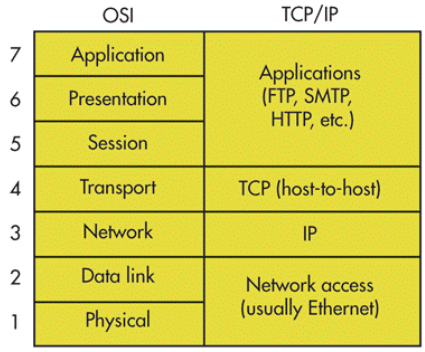
Ex.No:12 Date :	Capture and Analyse HTTP packets

Objective(s):

To capture and analyse TCP and IP packet using Wireshark.

Introduction:

Full form of HTTP is HyperText Transfer Protocol. HTTP is an application layer protocol in ISO or TCP/IP model. See below picture to find out HTTP which resides under application layer.



HTTP is used by the World Wide Web (w.w.w) and it defines how messages are formatted and transmitted by browser. So HTTP define rules what action should be taken when a browser receives HTTP command. And also HTTP defines rules for transmitting HTTP command to get data from server.

For example, when you enter a url in browser (Internet explorer, Chrome, Firefox, Safari etc) it actually sends an HTTP command to server. And server replies with appropriate command.

HTTP Methods:

There are some set of methods for HTTP/1.1 (This is HTTP version)

GET, HEAD, POST, PUT, DELETE, CONNECT, OPTION and TRACE.

We will not go in details of each method instead we will get to know about the methods which are seen quite often. Such as

GET: GET request asks data from web server. This is a main method used document retrival. We will see one practical example of this method.

POST: POST method is used when it's required to send some data to server.

HTTP is Wiresahark:

Let's try something practical to understand how HTTP works ?

So in this example we will download **"alice.txt" (Data file present in server)** from **"gaia.cs.umass.edu"** server.

Setps:

1. Open any URL http://gaia.cs.umass.edu/wireshark-labs/alice.txt
2. Now we see the downloaded file in browser. Here is the screenshot

3. In parallel we have capture the packets in Wireshark.

HTTP packets exchanges in Wireshark:

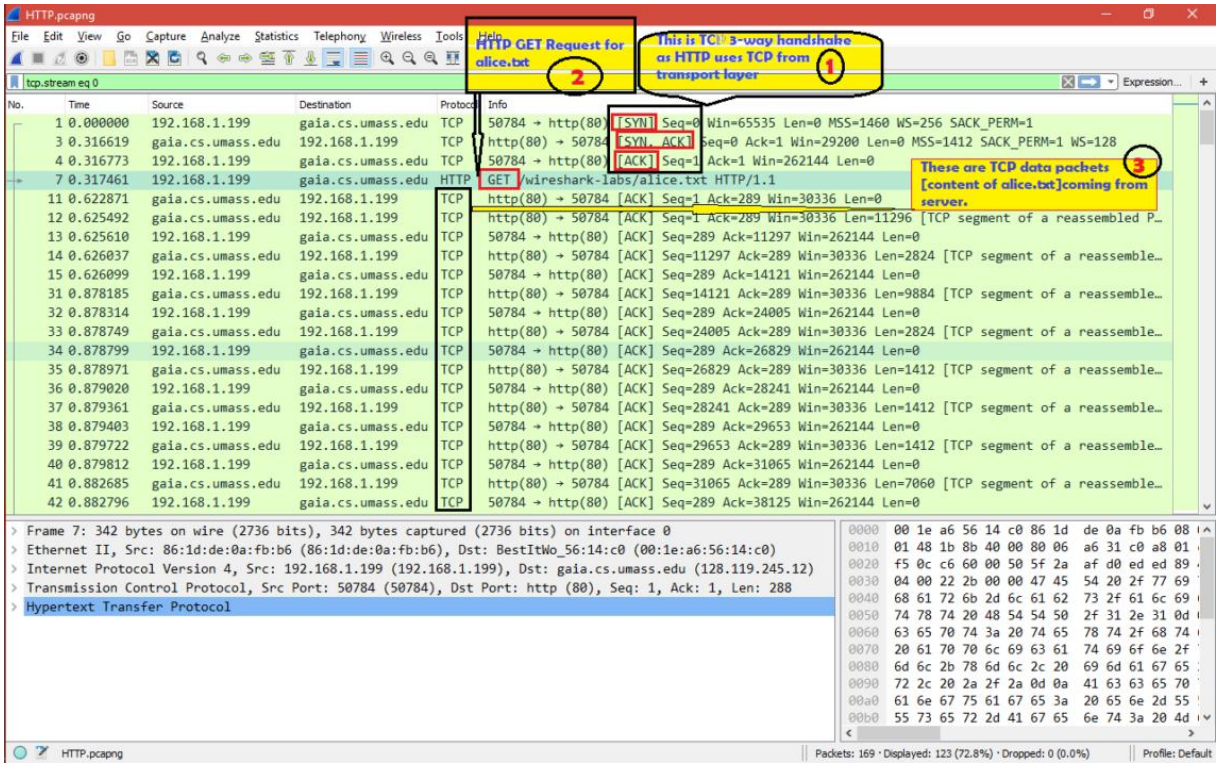
Before we go into HTTP we should know that HTTP uses port 80 and TCP as transport layer protocol [We will explain TCP in another topic discussion].

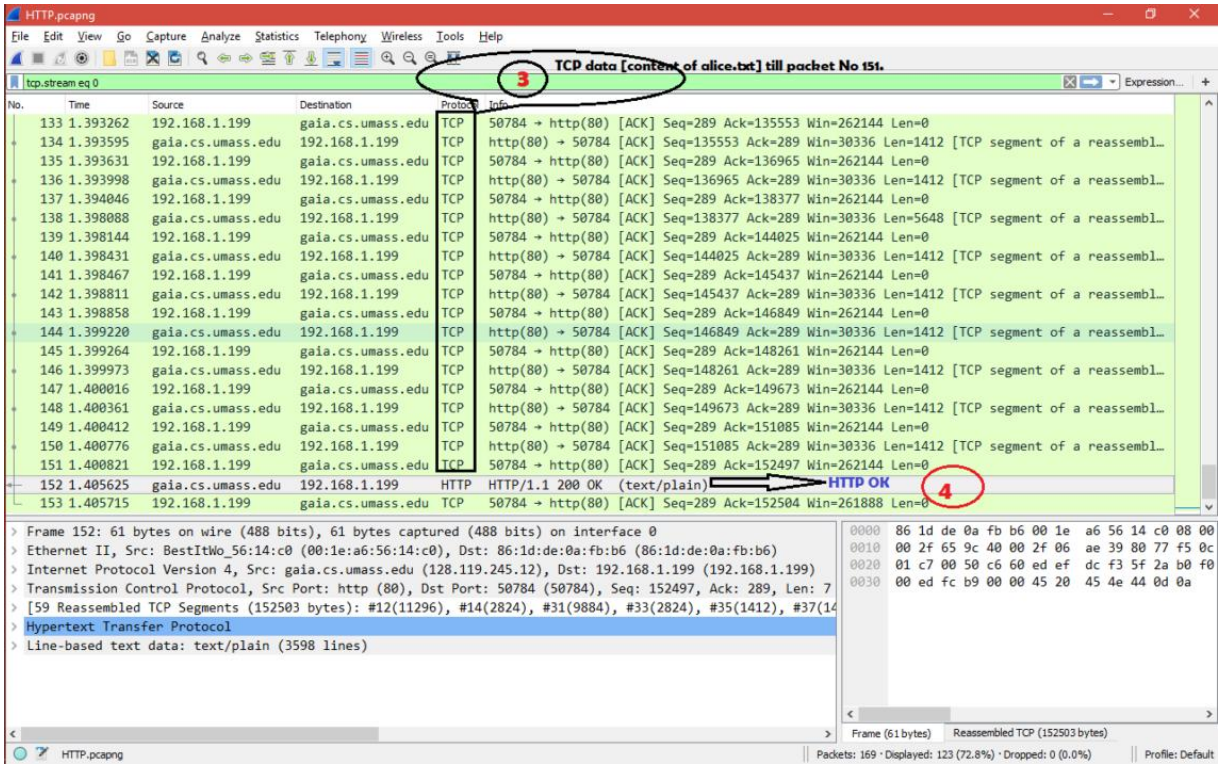
Now let's see what happens in network when we put that URL and press enter in browser.

Here is the screenshot for

TCP 3-way handshake ——-> HTTP OK ——-> TCP Data [content of alice.txt] ——->

HTTP-OK





Now let's see what's there inside HTTP GET and HTTP OK packets.

Note: We will explain TCP exchanges in another topic discussion.

HTTP GET:

After TCP 3-way handshake [SYN, SYN+ACK and ACK packets] is done HTTP GET request is sent to the server and here are the important fields in the packet.

1.Request Method: GET ==> The packet is a HTTP GET .

2.Request URI: /wireshark-labs/alice.txt ==> The client is asking for file alice.txt present under /Wireshark-labs

3.Request version: HTTP/1.1 ==> It's HTTP version 1.1

4.Accept: text/html, application/xhtml+xml, image/jxr, */* ==> Tells server about the type of file it [client side browser] can accept. Here the client is expecting alice.txt which is text type.

5.Accept-Language: en-US ==> Accepted language standard.

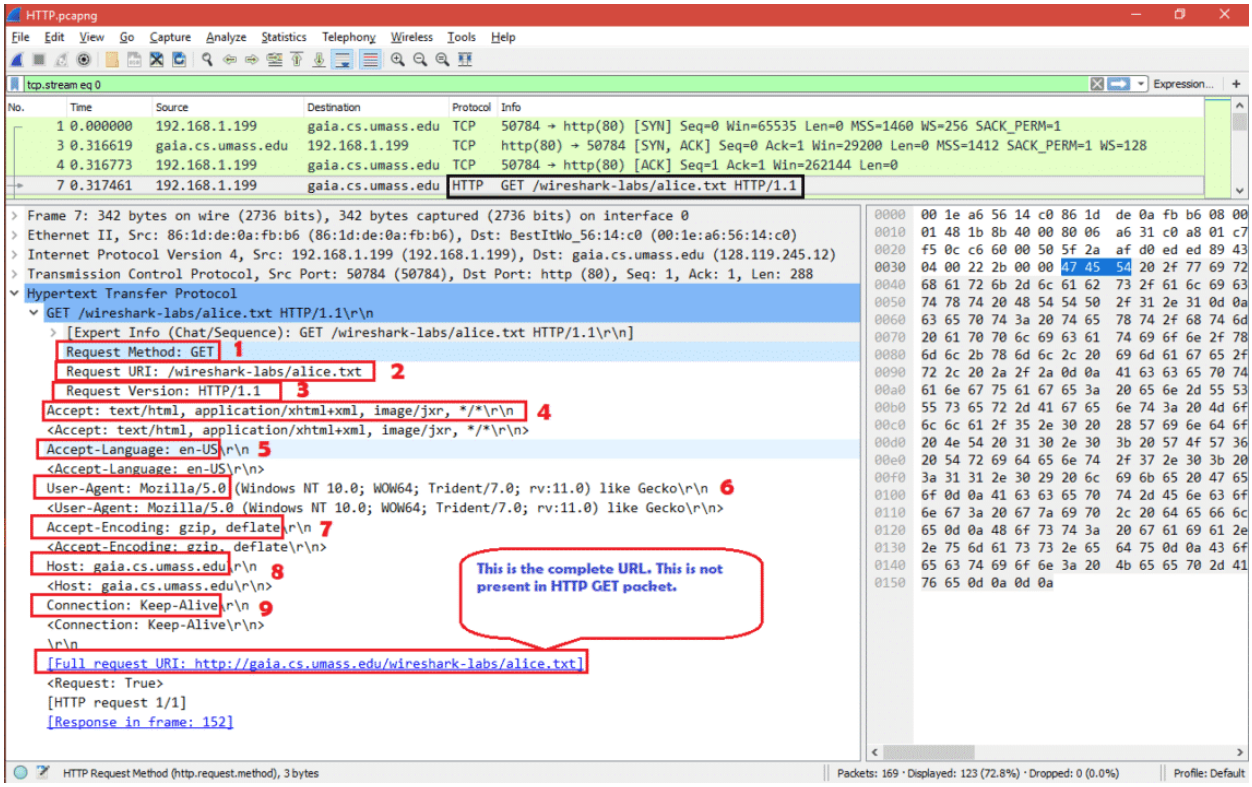
6.User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; Trident/7.0; rv:11.0) like Gecko ==> Client side browser type. Even if we used internet explorer but we see it always/maximum time says Mozilla

7.Accept-Encoding: gzip, deflate ==> Accepted encoding in client side.

8.Host: gaia.cs.umass.edu ==> This is the web server name where client is sending HTTP GET request.

9.Connection: Keep-Alive ==> Connection controls whether the network connection stays open after the current transaction finishes. Connection type is keep alive.

Here is the screenshot for HTTP-GET packet fields



HTTP OK:

After TCP data [content of alice.txt] is sent successfully HTTP OK is sent to the client and here are the important fields in the packet.

1. Response Version: HTTP/1.1 ==> Here server also in HTTP version 1.1

2.Status Code: 200 ==> Status code sent by server.

3.Response Phrase: OK ==> Response phrase sent by server.

So the from 2 and 3 we get 200 OK which means the request [HTTP GET] has succeeded.

4.Date: Sun, 10 Feb 2019 06:24:19 GMT ==> Current date , time in GMT when HTTP GET was received by server.

5.Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips PHP/5.4.16 mod_perl/2.0.10 Perl/v5.16.3 ==> Server details and configurations versions.

6.Last-Modified: **Sat, 21 Aug 2004 14:21:11 GMT** ==> Last modified date and time for the file “alice.txt”.

7.ETag: “2524a-3e22aba3a03c0” ==> The ETag indicates the content is not changed to assist caching and improve performance. Or if the content has changed, etags are useful to help prevent simultaneous updates of a resource from overwriting each other.

8. Accept-Ranges: bytes ==> Byte is the unit used in server for content.

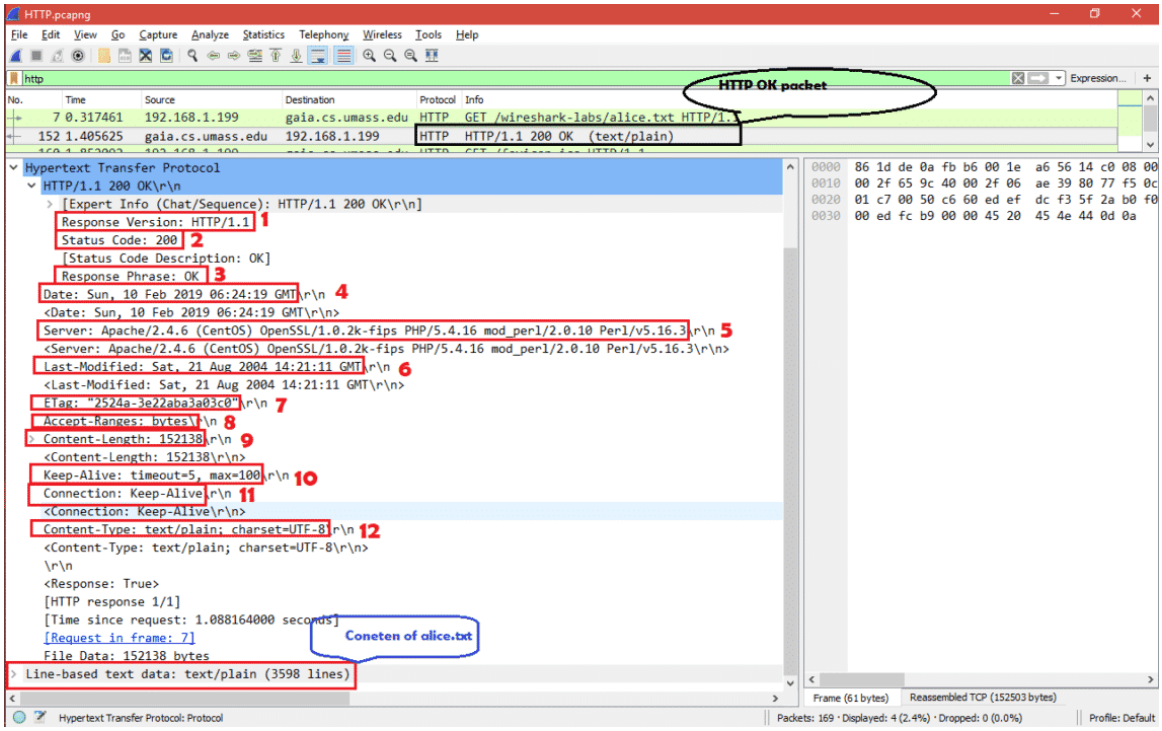
9.Content-Length: 152138 ==> This is the total length of the alice.txt in bytes.

10. Keep-Alive: timeout=5, max=100 ==> Keep alive parameters.

11.Connection: Keep-Alive ==> Connection controls whether the network connection stays open after the current transaction finishes. Connection type is keep alive.

12.Content-Type: text/plain; charset=UTF-8 ==> The content [alice.txt] type is text and charset standard is UTF-8.

Here is the screenshot for different fields of HTTP OK packet.



So now we know what happens when we request for any file that is present in web server.

Conclusion:

HTTP is simple application protocol that we use every day in our life. But it's not secure so HTTPS has been implemented. That "S" stands for secure. That's why you so maximum web server name start with http***s://[websitename]***. This means all communication between you and server are encrypted. We will have separate discussion on this HTTPS in future.

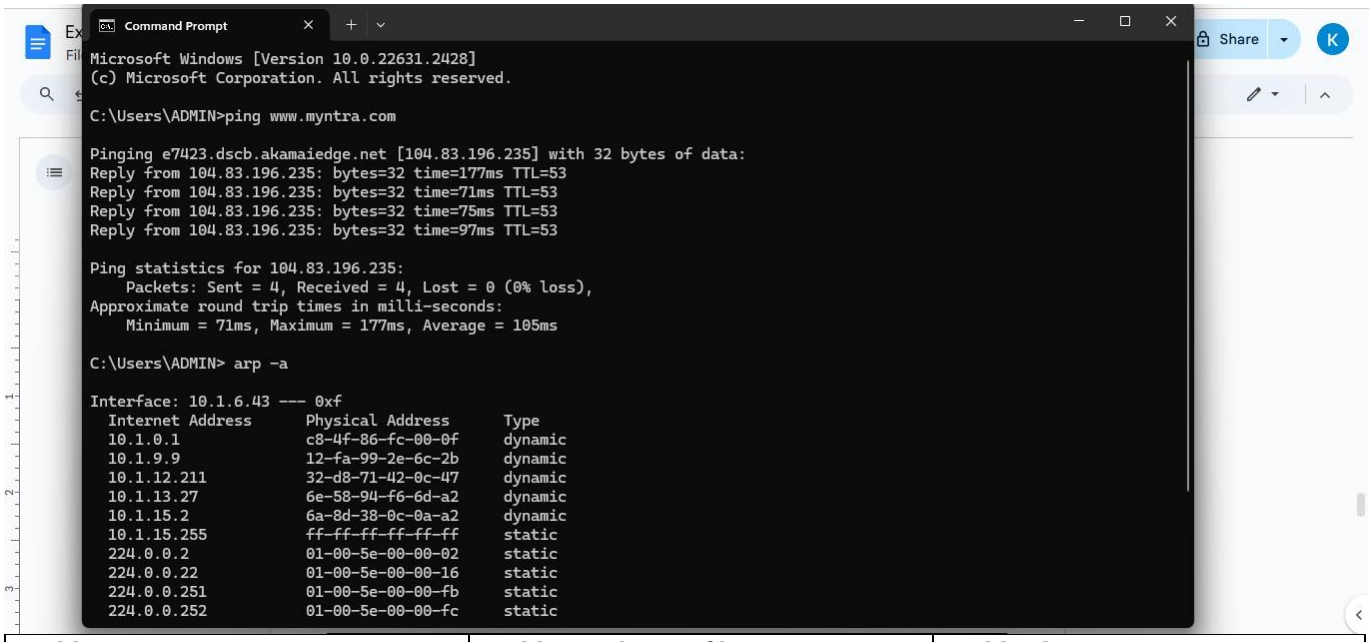
Exercise:

Analyze the HTTP protocol using Wireshark by visiting any URL.

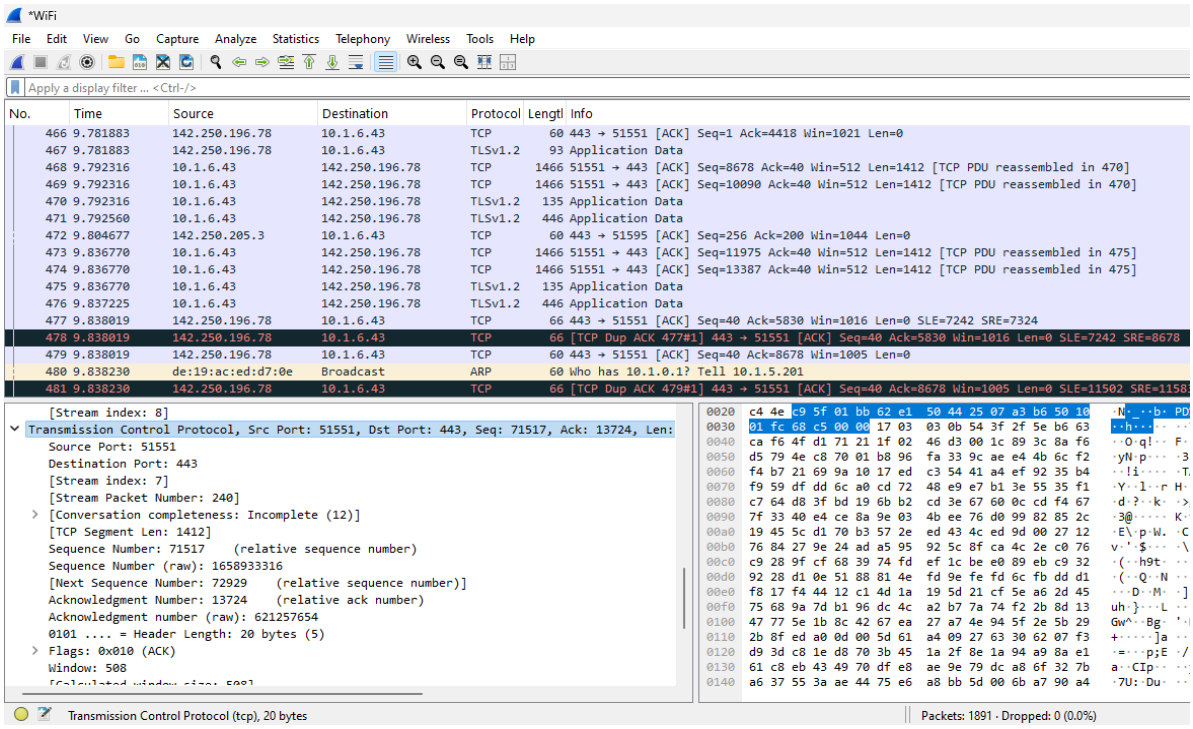
1. Visit any one website by opening a browser fill your machine details (attach relevant screenshots).

Parameter	Value
Your Machine IP Address.	192.168.213.64
Your Machine MAC Address	CC-47-40-59-DC-B8
Default Gateway address	192.168.213.158
Website URL	www.myntra.com
Website IP Address	104.83.196.235

2. Fill the TCP connection segment details:



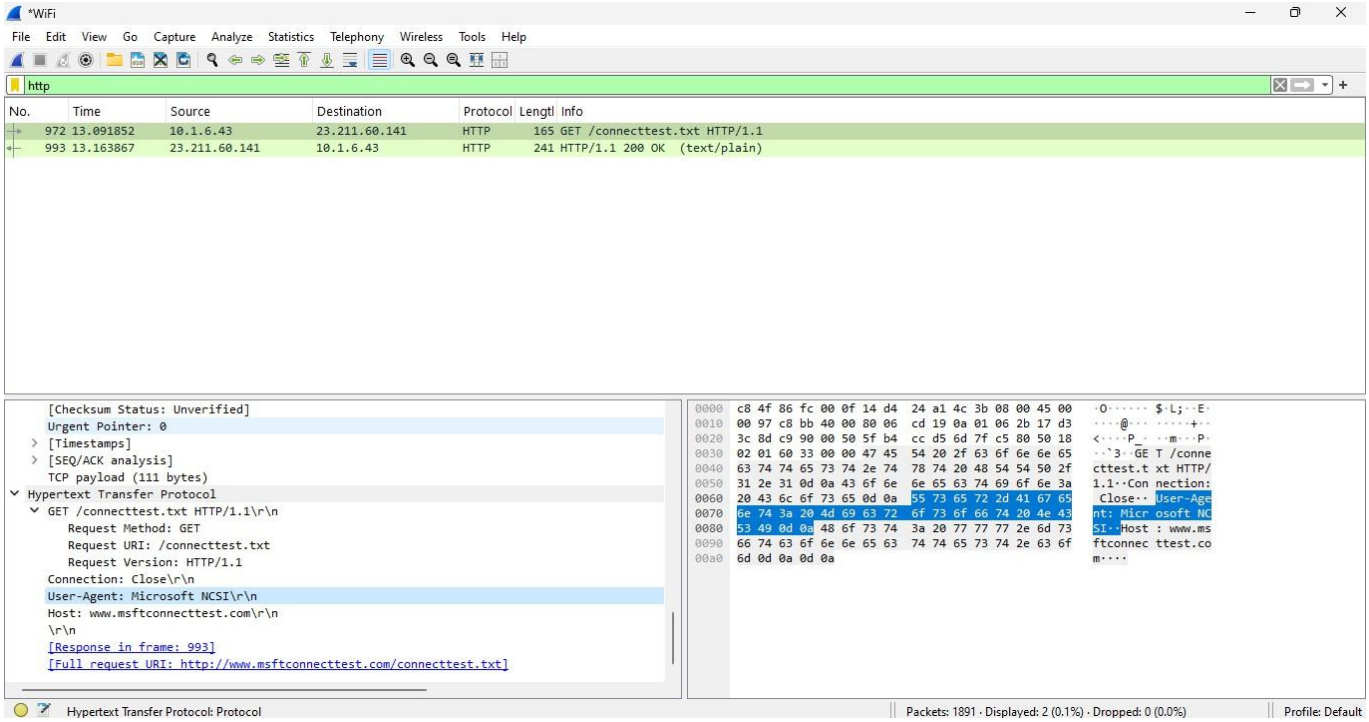
Field Name	Field Length (no of bits)	Field value
Destination MAC address	48	C8:4f:86:fc:00:10
Source MAC address	48	CC-47-40-59-DC-B8
Destination IP address	32	104.83.196.235
Source IP Address	32	192.168.213.64
Destination TCP port	16	51551
Source TCP port	16	443



3. HTTP Request Message Details.

Field Name	Field Length (# of Bits)	Field Value (Binary or Hexa value)
Method	24	GET
Host	104	www.msftconnecttest.com
Accept	304	
User-Agent	144	Microsoft NCSI
Accept-Language	40	
Accept-Encoding	24	
Connection	80	Close

Paste the HTTP Response Screenshot:



Paste the Wireshark File with view permission:

https://drive.google.com/drive/folders/1ps1OS8kWsComBI4n6erG2VDeZSZanCar

Google Drive Link:

https://drive.google.com/drive/folders/1ps1OS8kWsComBI4n6erG2VDeZSZanCar

Rubrics for Wireshark labs: (To be Filled by the Class Teacher)

Rubrics	Excellent	Fair	Poor	Marks
Understanding (2)	Understand the Concept very well. (2)	Understand the Concept (1)	Poor Understand the Concept (0)	
Usage of filters (3)	Identified and applied the filter correctly (3)	Identified the filter, but not applied correctly (2-1)	Couldn't identify and apply the filter. Just captured the packets (1)	
Attach relevant Screenshots (3)	clearly Highlighted the answers and attached the screenshots (3)	attached the screenshots, but not highlighted. (2-1)	Did not attach the screenshots (0)	
On time Submission (2)	Early or on time submission (2)	Submitted after deadline (1)	Did not Submit (0)	
			Total	

Conclusion: The above experiment has been executed and successfully implememted.

Ex.No:13	Capture and Anlyze ICMP packet
Date :	

Objective(s):

To capture and analyse ICMP Request Response packet using Wireshark.

Introduction:

The **I**nternet **C**ontrol **M**essage **P**rotocol (**ICMP**) is a supporting protocol in the Internet protocol suite. It is used by network devices, including routers, to send error messages and operational information which indicates that a requested service is not available or that a host or router could not be reached.

It is layer 3 i.e. network layer protocol used by the ping command for sending a message through ICMP payload which is encapsulated with IP Header Packet. According to MTU the size of the ICMP packet cannot be greater than 1500 bytes.

ICMP packet at Network layer

IP header	ICMP header	ICMP payload size	MTU (1500)
20 bytes	8 bytes	1472 bytes (maximum)	20 + 8 + 1472 = 1500

ICMP packet at Data Link layer

Ethernet header	IP header	ICMP header	ICMP payload size	MTU (1514)
14	20 bytes	8 bytes	1472 bytes (maximum)	14 + 20 + 8 + 1472 = 1514

ICMP Message code & Packet description with Wireshark

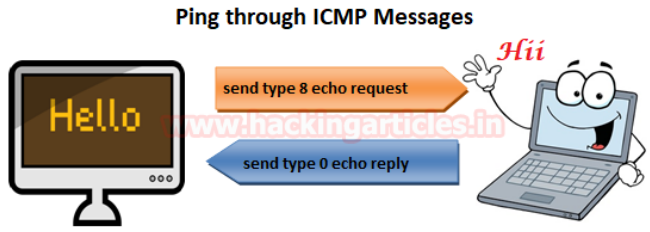
ICMP message contains two types of codes i.e. query and error.

Query: The query messages are the information we get from a router or another destination host.

For example, given below message types are some ICMP query codes:

- Type 0 = Echo Reply
- Type 8 = Echo Request
- Type 9 = Router Advertisement
- Type 10 = Router Solicitation
- Type 13 = Timestamp Request
- Type 14 = Timestamp Reply

A ping command sends an ICMP **echo request** to the target host. The target host responds with an **echo Reply** which means the target host is alive.

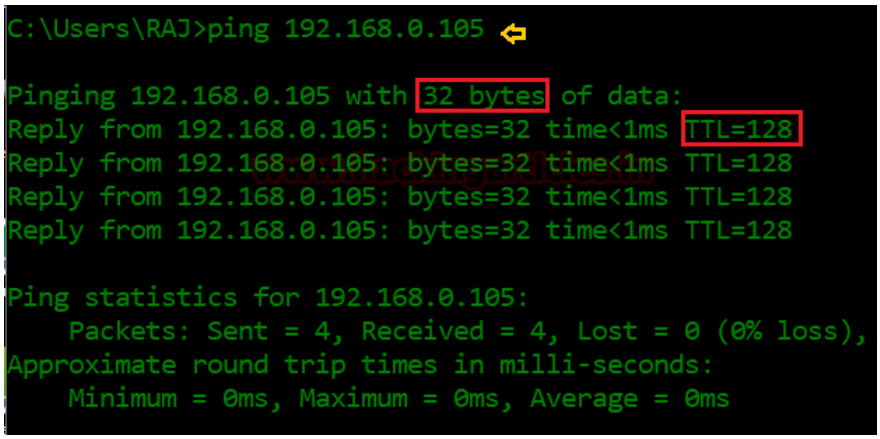


Here we are going to test how ping command helps in identifying an alive host by Pinging host IP.

```
ping 192.168.0.105
```

From the given below image you can see a reply from the host; now notice a few more things as given below:

- The default size of payload sent by source machine is 32 bytes (**request**)
- The same size of payload received by source machine is 32 bytes from Destination machine (**reply**)
- **TTL = 128** which means host machine is windows system.
- Total packets are **8**, 4 packets of the request and 4 of reply.

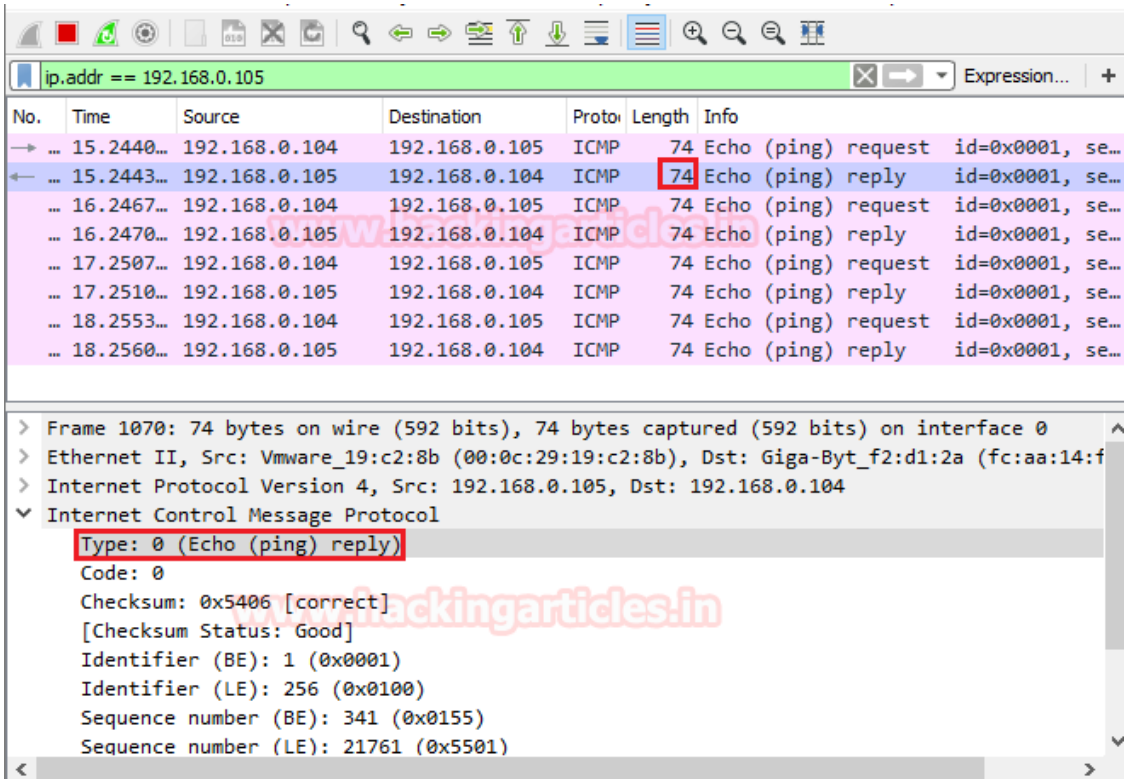


Look over the sequence of packet transfer between source and destination captured through Wireshark.

Total numbers of packet captured are 8, 4 for request and 4 for reply between the source and destination machine.

The 1st packet is sent by source machine is ICMP echo request and if you look by the given below image, you will observe highlighted text is showing ICMP query code: **type 8 echo ping request.**

Similarly given below image is showing details of **2nd** packet i.e. Echo reply, you can observe that the highlighted text is showing ICMP query code: **type 0 echo ping reply.**

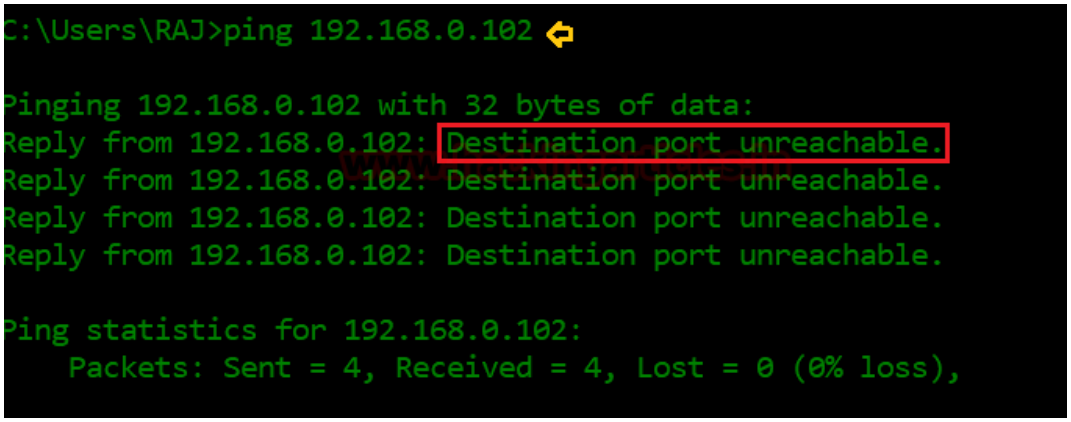


Error: The error statement messages reports problem which a router or a destination host may generate.

For example: given below message types are some of the ICMP error codes:

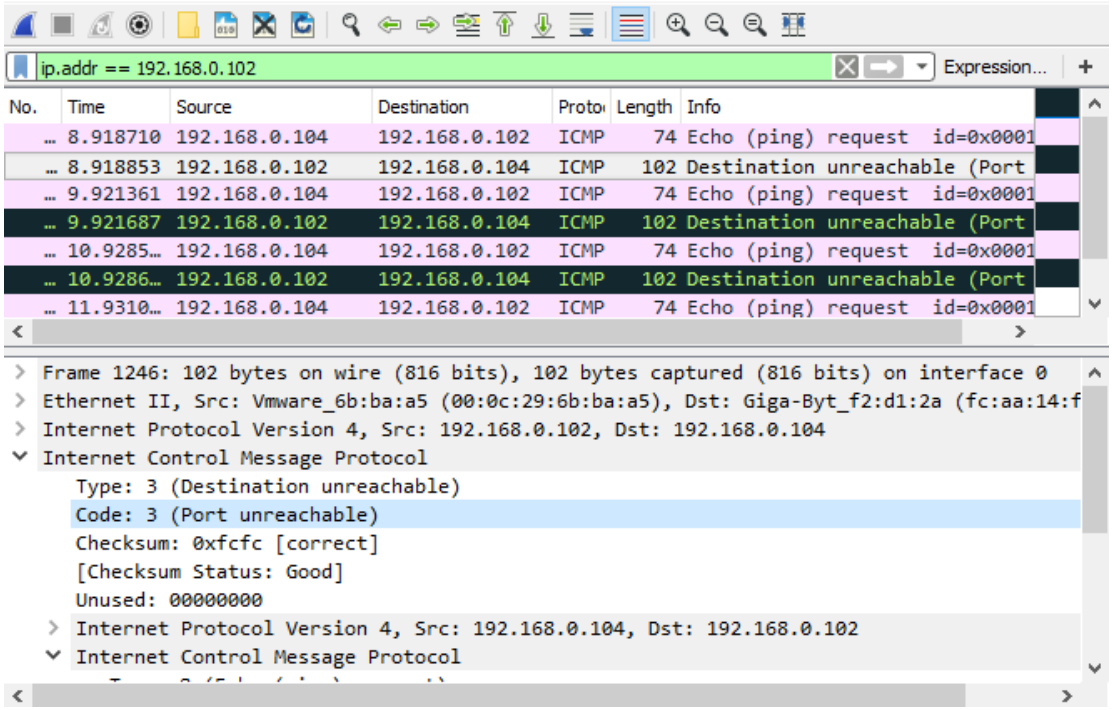
- Type 3 = Destination Unreachable
- Type 4 = Source Quench
- Type 5 = Redirect
- Type 11 = Time Exceeded
- Type 12 = Parameter Problems

When we ping an IP sometimes we don't get echo ping reply from the host machine, instead of that, we get some reply such as **destination unreachable** or **time exceeded** this is known as **ICMP error reporting message**. There are so many reasons behind such kind of error message, possibly a host in a network is down or firewall is blocking your ping request.



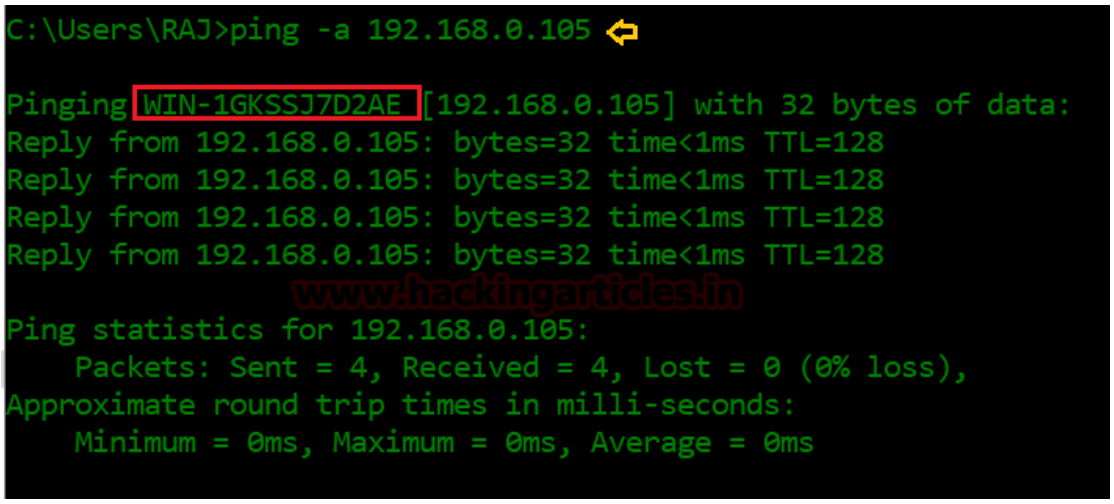
The **1st** packet sends by source machine is ICMP echo request and if you observe by the given below image the highlighted text is showing ICMP query code: **type 8 echo ping request.**

Similarly given below image is showing the detail of **2nd** packet i.e. Destination unreachable, you can observe that it is showing ICMP error code: **type 3.**

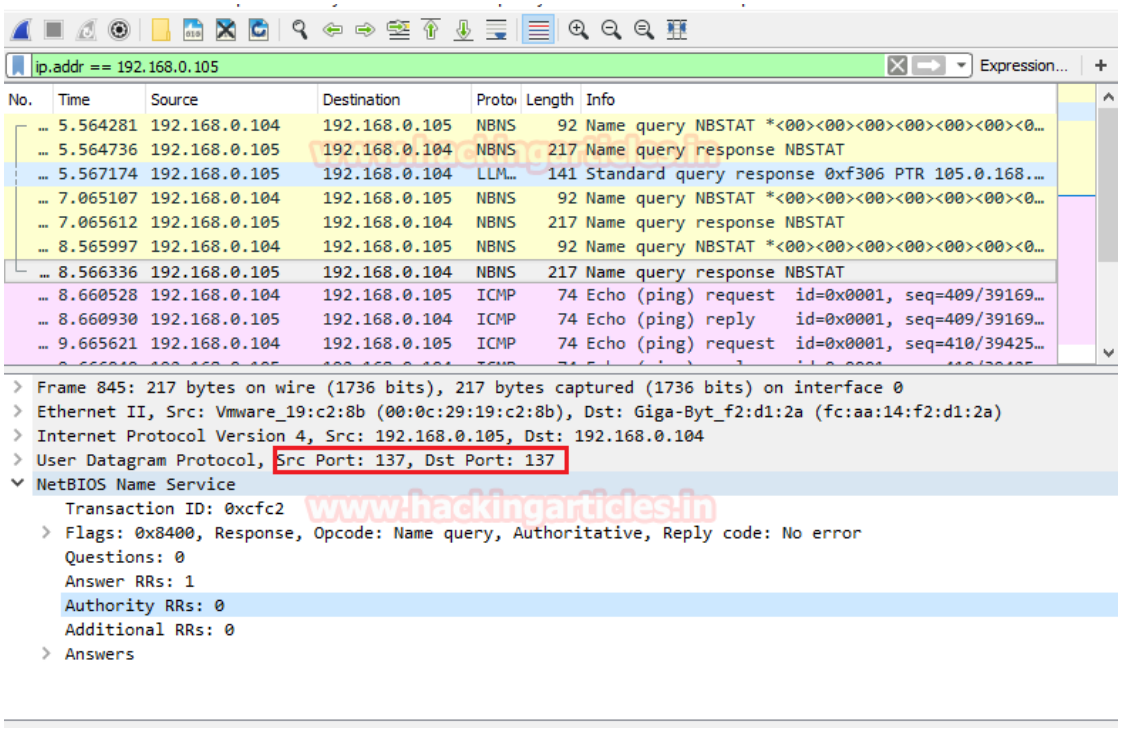



```
ping –a 192.168.0.105
```

-a: Resolve IP addresses to host-name, identify's that reverse name resolution is carried out on the host IP address. If it is successful, ping shows the matching hostname.



From the given below image, you can observe that instead of ICMP protocol the ping request has been sent through NBNS (NetBIOS Name Service) protocol through port 137 which is a UDP port.

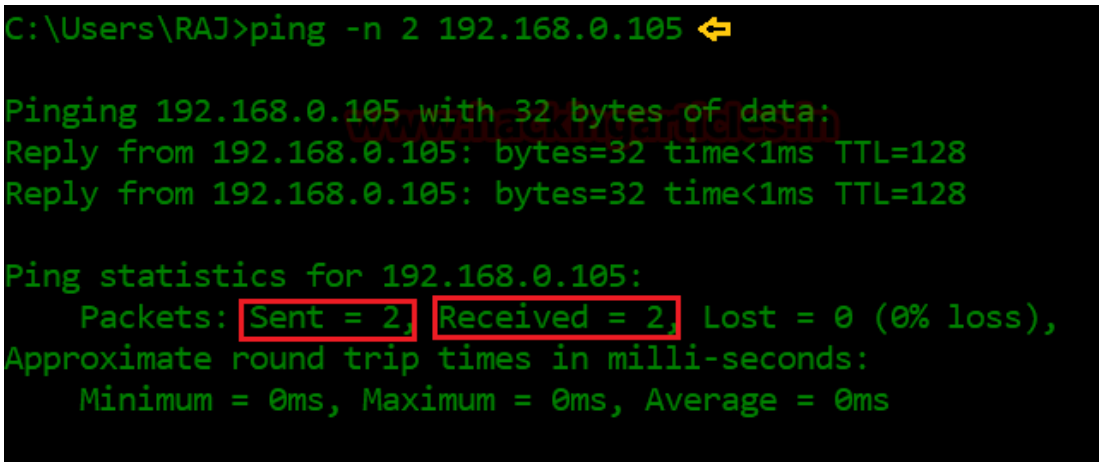


By default, a ping sends 4 packets of the request and receives the same number of the packet as a reply from the host. You can increase or decrease this number of the packet by using given below command.

```
ping –n 2 192.168.0.105
```

-n: Number of echo requests to send

As we had set -n as 2 packets of request hence we got two packets as a reply.



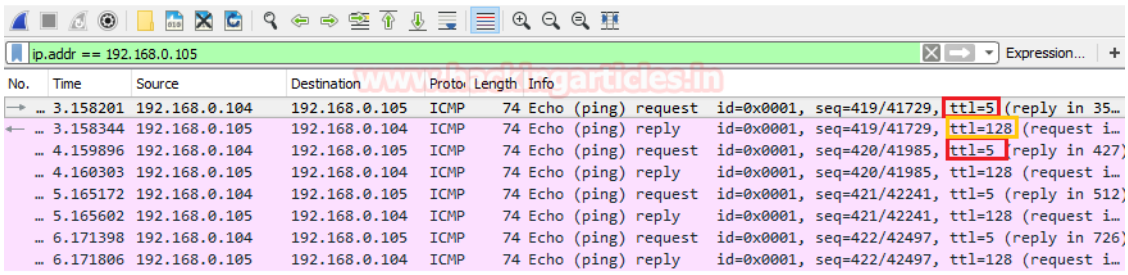
Similarly, we can also set TTL (Time to Live) for echo request packet, by default 4 packet of request query are sent from source machine at the rate of 1 millisecond per packet. Suppose we want to give TTL between two packets, set -i as 5ms so that after the first packet is delivered the second packet is sent after 5ms.

```
ping –i 5 192.168.0.105
```

-i TTL: Time To Live

```
C:\Users\RAJ>ping -i 5 192.168.0.105

Pinging 192.168.0.105 with 32 bytes of data:
Reply from 192.168.0.105: bytes=32 time<1ms TTL=128
Reply from 192.168.0.105: bytes=32 time<1ms TTL=128
Reply from 192.168.0.105: bytes=32 time<1ms TTL=128
Reply from 192.168.0.105: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.0.105:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Let's verify TTL for a packet sent from source to destination through Wireshark. Now if you observe by the given below image you will notice that every echo ping request packet has TTL 5 but every echo reply has default TTL value i.e.128.



ICMP payload description through Wireshark

As we have discussed above default size of ICMP payload is 32 bytes and the maximum is 1472 if the size of the payload packet is greater than 1472 then packet gets fragmented into small packets.

From the given below image, you can observe source has pinged the host which carries default 32 bytes size payload.

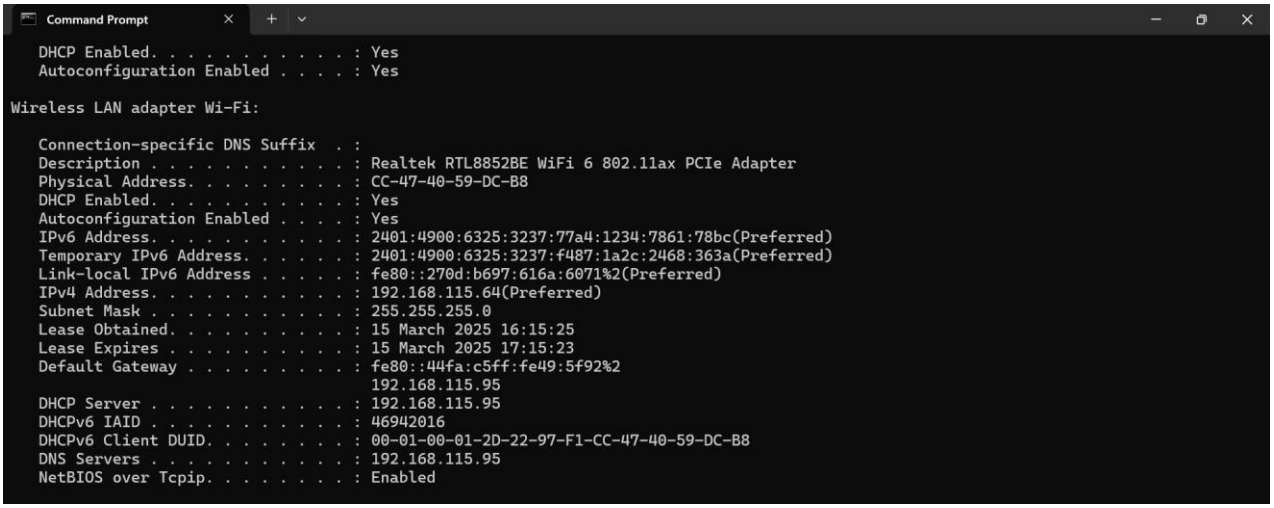
```
C:\Users\RAJ>ping 192.168.0.105

Pinging 192.168.0.105 with 32 bytes of data:
Reply from 192.168.0.105: bytes=32 time<1ms TTL=128
Reply from 192.168.0.105: bytes=32 time<1ms TTL=128
Reply from 192.168.0.105: bytes=32 time<1ms TTL=128
Reply from 192.168.0.105: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.0.105:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

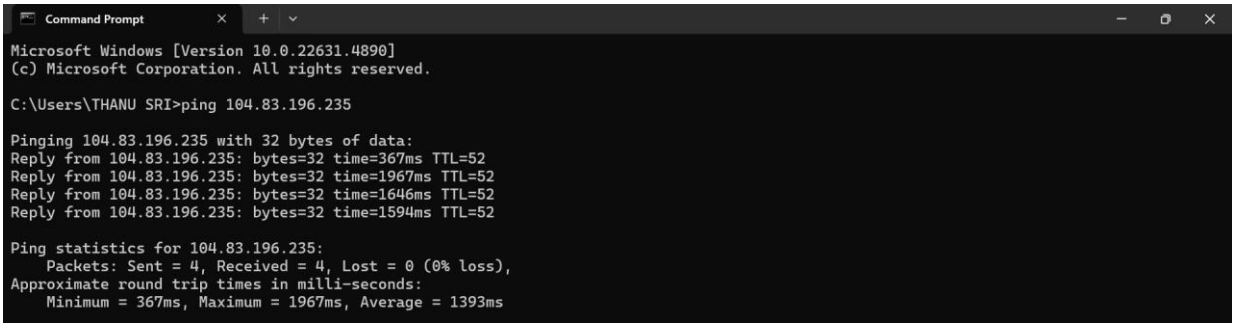
Now let check the information payload carries from source to destination using Wireshark. From the given below image, you can read that highlighted texts are alphabets that have been used as 32 bytes payload.

Exercise:

1. Use command prompt and fill the following details using ipconfig /all command. (highlight and paste screenshot for each of the output).

Parameter	Value
Your Machine IP Address.	192.168.115.64
Your Machine MAC Address	CC-47-40-59-DC-B8
Default Gateway address	192.168.115.95
DNS Server IP Address	192.168.115.95

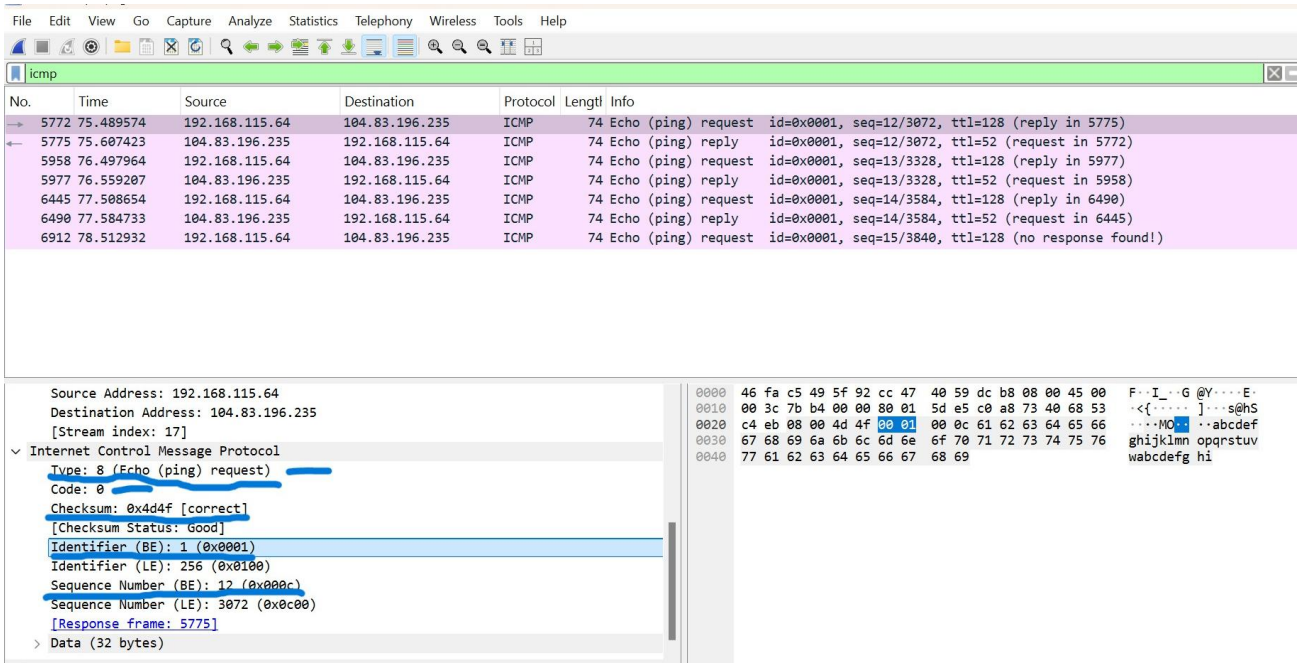




2. Ping any website through command prompt and Fill the following details by applying the filter as **ICMP**: (highlight and paste screenshot for each of the output).

ICMP Request message:

Field Name	Field Length (no of bits)	Field value
Type	8	8
Code	8	0
Checksum	16	0x4d4f
Identifier	16	1(0x0001)
Sequence Number	16	12(0x000c)



Rubrics	Excellent	Fair	Poor	Marks
Understanding (2)	Understand the Concept very well. (2)	Understand the Concept (1)	Poor Understand the Concept (0)	
Usage of filters (3)	Identified and applied the filter correctly (3)	Identified the filter, but not applied correctly (2-1)	Couldn't identify and apply the filter. Just captured the packets (1)	
Attach relevant Screenshots (3)	clearly Highlighted the answers and attached the screenshots (3)	attached the screenshots, but not highlighted. (2-1)	Did not attach the screenshots (0)	
On time Submission (2)	Early or on time submission (2)	Submitted after deadline (1)	Did not Submit (0)	
			Total	

Conclusion: The above experiment has been executed and implemented successfully.

Ex.No: 14	
Date :	**Capture and Analyze DNS packet**

Objective(s):

To capture and analyse DNS Query Response packet using Wireshark.

Introduction:

What is DNS?

The Domain Name System (DNS) is a hierarchical and decentralized naming system for computers, services, or other resources connected to the internet or a private network.

To do DNS analysis in Wireshark, the nslookup command must be used.

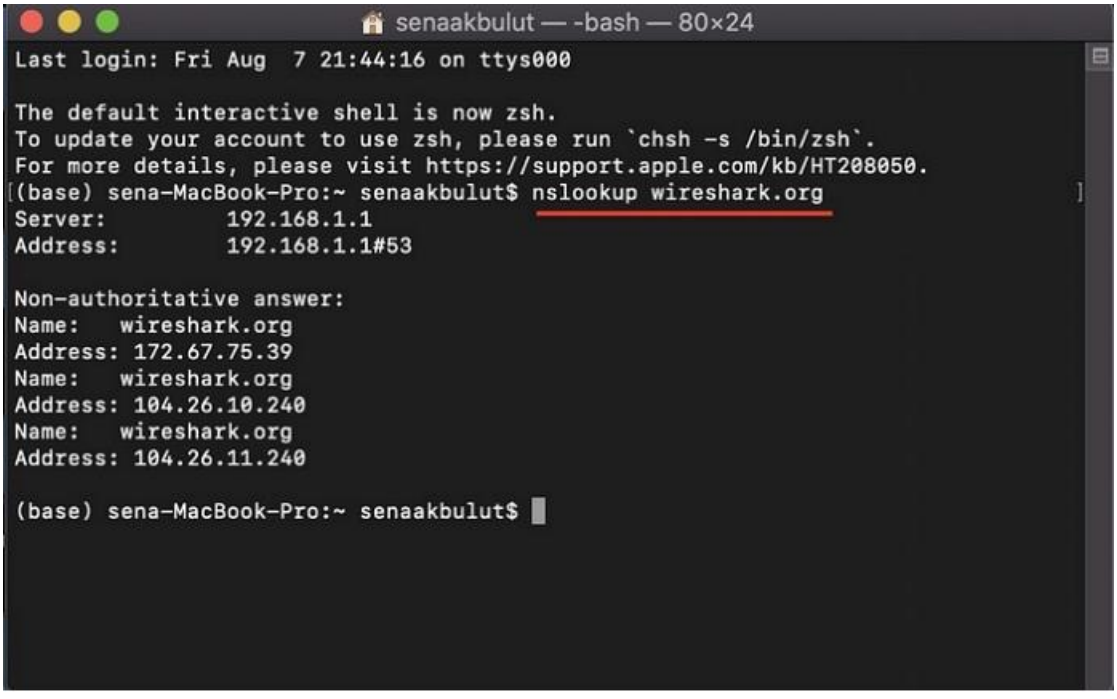
What is nslookup?

nslookup is a network administration command-line tool available in many computer operating system for querying the Domain Name System (DNS) to obtain a domain name or IP address mapping, or other DNS records.

Now that we have learned the meanings of these terms, let's examine the analysis steps in Wireshark.

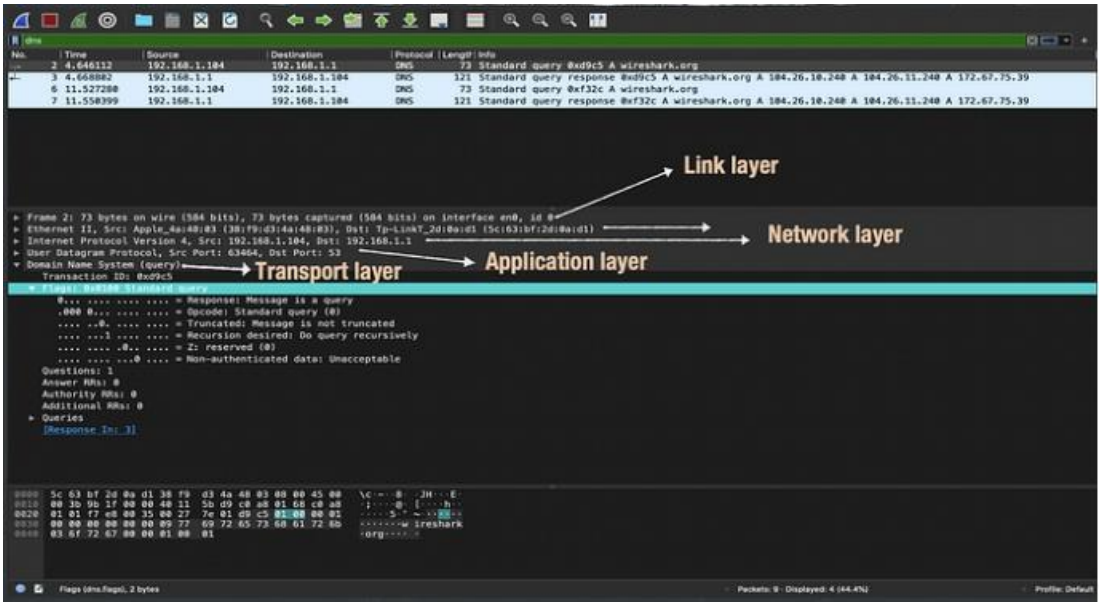
- To analyze it, I first ran the nslookup command for wireshark.org in the terminal and viewed the site's IP address and non-authoritative replies with the nslookup command.

nslookup wireshark.org



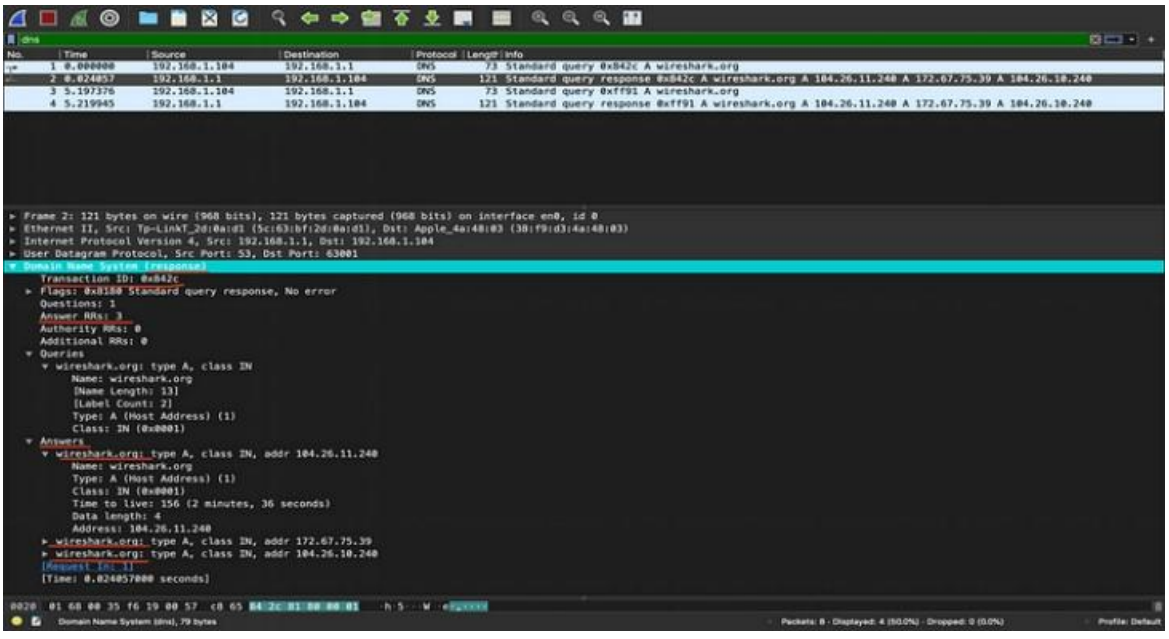
- Then when I ran the Wireshark traffic capture application and applied the DNS filter, the traffic I made in the terminal was displayed as follows.

When I looked at the first query, a small screen with information about the query appeared. The first feature here is below the link layer, the second and third is below the network layer, the fourth is below the transport layer, and the last feature is below the application layer.



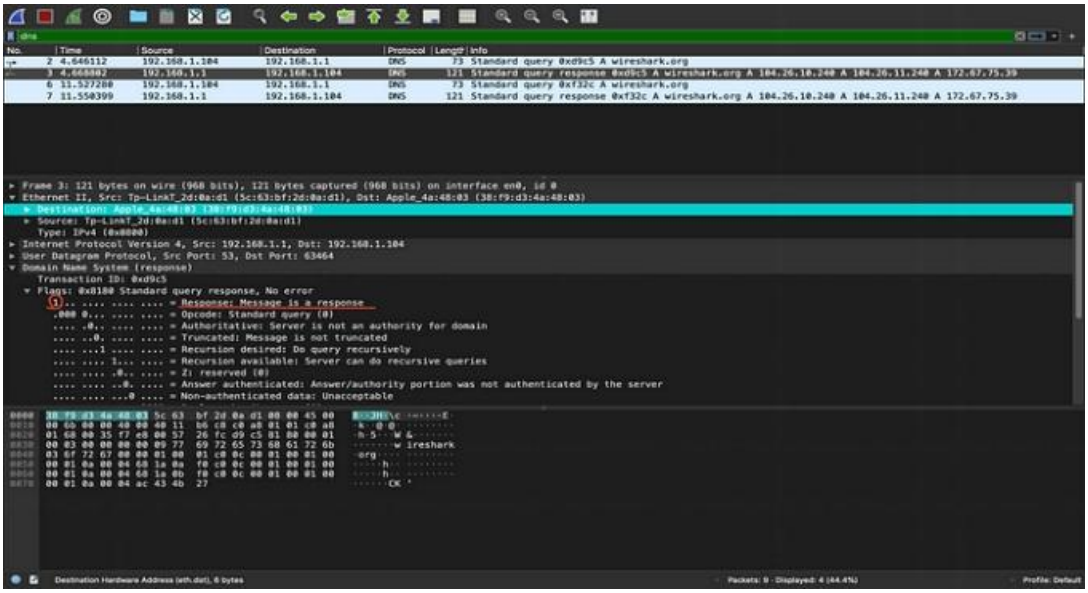
- When I came to response and when I opened the section that says domain name system, I saw sub-features named transaction id, flags and answers.

The Answer RRs part in the response section is as many as the answers we see in the terminal, so 3. The characteristics of the answers can also be examined in the lower part. The Answer RRs part is 0 in the query part because there is no answer yet.



- When we open the flags section, we see that it says 0 in query and 1 in response. This first flag bit indicates whether it is a query or a response.

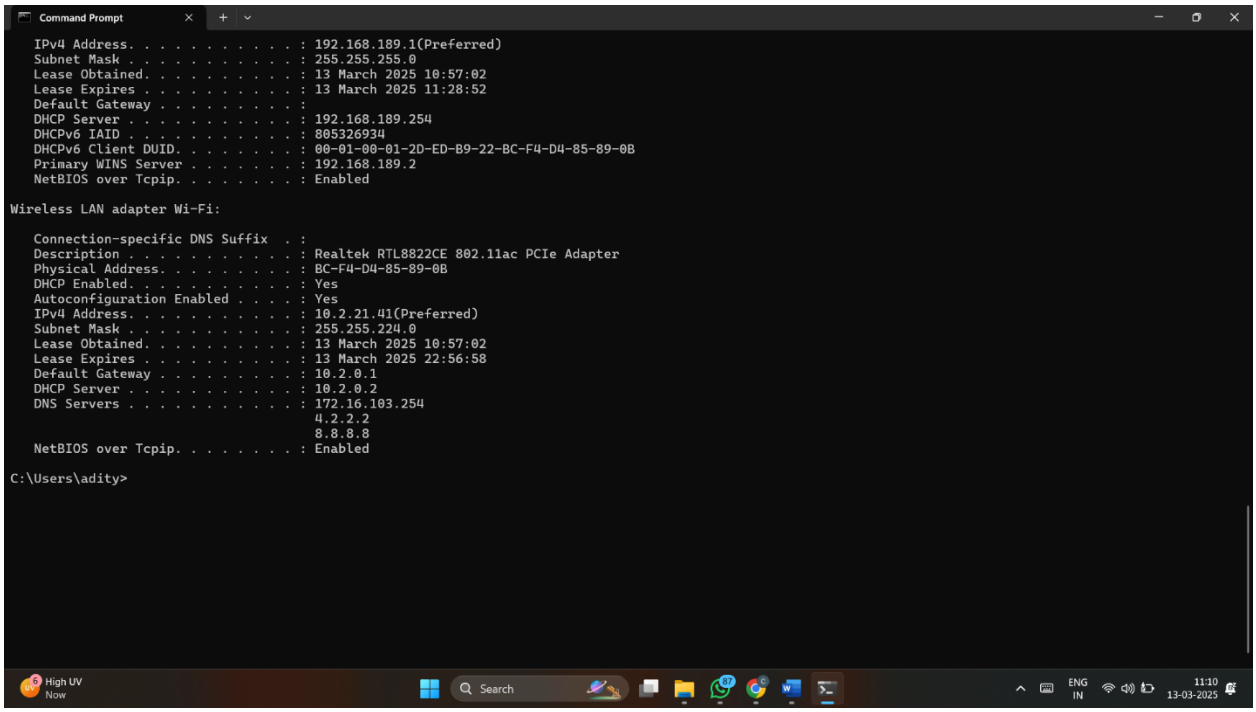
It also displays hexadecimal equivalents of destinations and sources. The first set of bits represents destination and the second set of bits represents source.



Exercise:

1. Use command prompt and fill the following details using ipconfig /all command. (highlight and paste screenshot for each of the output).

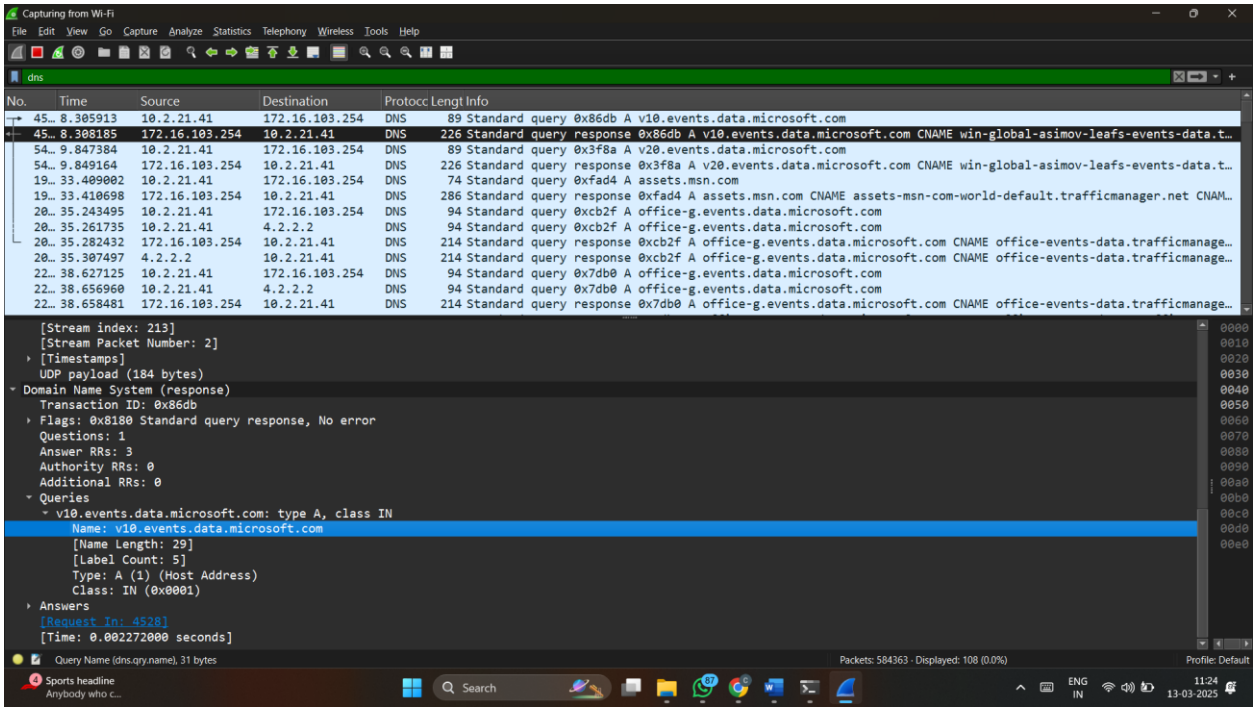
Parameter	**Value**
Your Machine IP Address.	10.2.21.41
Your Machine MAC Address	BC-F4-D4-85-89-0B
Default Gateway address	10.2.0.1
DNS Server IP Address	172.16.103.254



2. Ping any website through command prompt and Fill the following details by applying the filter as DNS: (highlight and paste screenshot for each of the output).

DNS Query message:

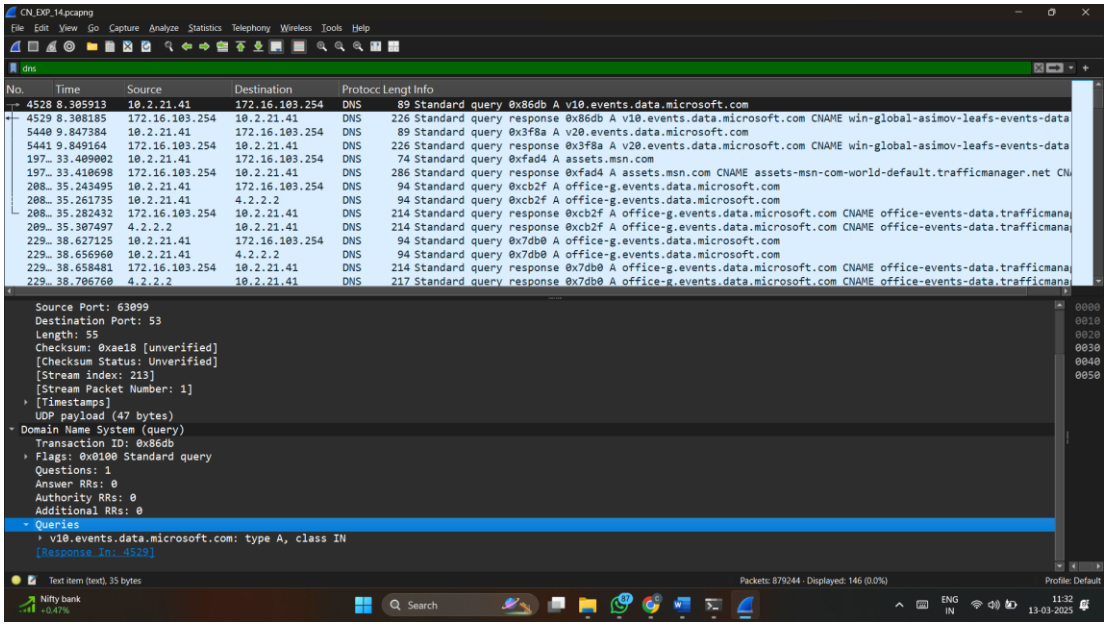
Field Name	**Field Length (no of bits)**	**Field value**
Destination MAC Address	48	C8:4f:86:fc:00:10
Source MAC Address	48	Bc:f4:d4:85:89:0b
Destination IP Address	32	74.6.231.20
Source IP Address	32	10.2.21.41
Destination UDP port	16	53
Source UDP port	16	63099
DNS Tx id	16	0x86db
DNS Flags	16	0x0100
DNS Questions	16	1
DNS Queries	variable	events.data.microsoft.com



3. DNS Response message: (highlight and paste screenshot for each of the output).

DNS Response message:

Field Name	**Field Length (no of bits)**	**Field value**
Destination MAC Address	48	C8:4f:86:fc:00:10
Source MAC Address	48	Bc:f4:d4:85:89:0b
Destination IP Address	32	74.6.231.20
Source IP Address	32	10.2.21.41
Destination UDP port	16	53
Source UDP port	16	63099
DNS Tx id	16	0x0xcb2f
DNS Flags	16	0x0100
DNS Questions	16	1
DNS Queries	Variable	Office-g.microsoft.com



Rubrics for Wireshark labs: (To be Filled by the Class Teacher)

Rubrics	Excellent	Fair	Poor	Marks
Understanding (2)	Understand the Concept very well. (2)	Understand the Concept (1)	Poor Understand the Concept (0)	
Usage of filters (3)	Identified and applied the filter correctly (3)	Identified the filter, but not applied correctly (2-1)	Couldn't identify and apply the filter. Just captured the packets (1)	
Attach relevant Screenshots (3)	clearly Highlighted the answers and attached the screenshots (3)	attached the screenshots, but not highlighted. (2-1)	Did not attach the screenshots (0)	
On time Submission (2)	Early or on time submission (2)	Submitted after deadline (1)	Did not Submit (0)	
Total				

Result: Thus the implementation of the DNS packet server has been successfully executed using cisco packet racer.

Ex.No:15	FTP server Configuration
Date :	

Objective(s):

To design and implement FTP server configuration using packet tracer

Introduction:

The File Transfer Protocol is a standard network protocol used for the transfer of computer files between a client and server on a computer network. FTP is built on a client-server model architecture using separate control and data connections between the client and the server.

Let's now do FTP configuration in Packet Tracer

1) Open Cisco Packet Tracer and select 2 End Devices (PC device), 1 Switch, 1 Router, 1 Server.

2) Now Connect all the devices using the auto connection.

3) Then configure the IP addresses as per the diagram.

4) Now just wait for some time to let all the connection status turns green.

5) Now we have achieved a connection where a class C IP address is being translated to class A IP Address.

6) Go to one of the PC devices and on Desktop tab select CMD.

7) Now we need to check the connection to the server by `C:\>ping 10.10.10.2`

8) If reply is coming then it means the server is properly configured and connected.

9) Go to the Server → Services → FTP.

10) Put on the FTP service and give username and password and click on ADD.

11) Come back to PC device and open the CMD and type `C:\>ftp 10.10.10.2`

12) It will ask for username and password. Provide the username and password configured earlier.

13) Once the connection is established exit rom the CMD and go to Text Editor and make a new text file.

14) Save the new text file and return to cmd and type `ftp>put filenme.txt`

15) This will send the text file from the PC device (192.168.0.2) to Server (10.10.10.2).

16) Now to verify that the file has been transferred to the server, so type

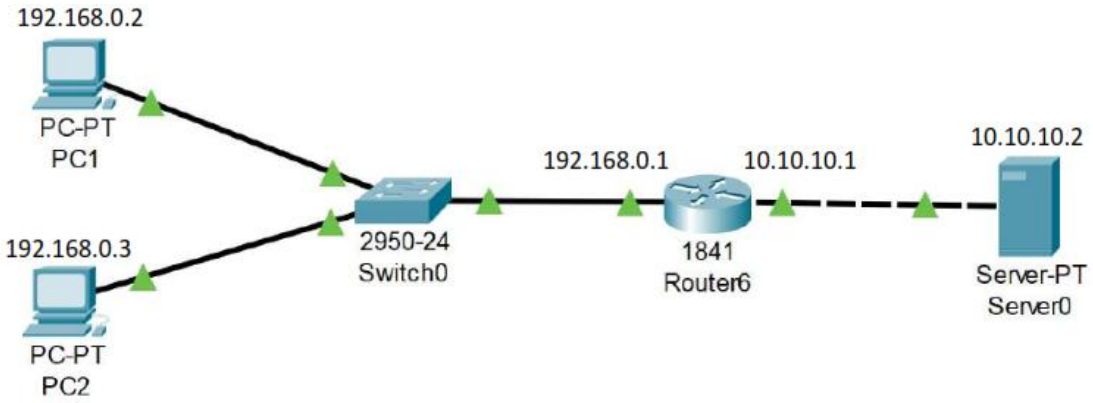
17) You will see your Filename in the list.

18) Now to get a file from server to PC type `ftp>get filename.txt`

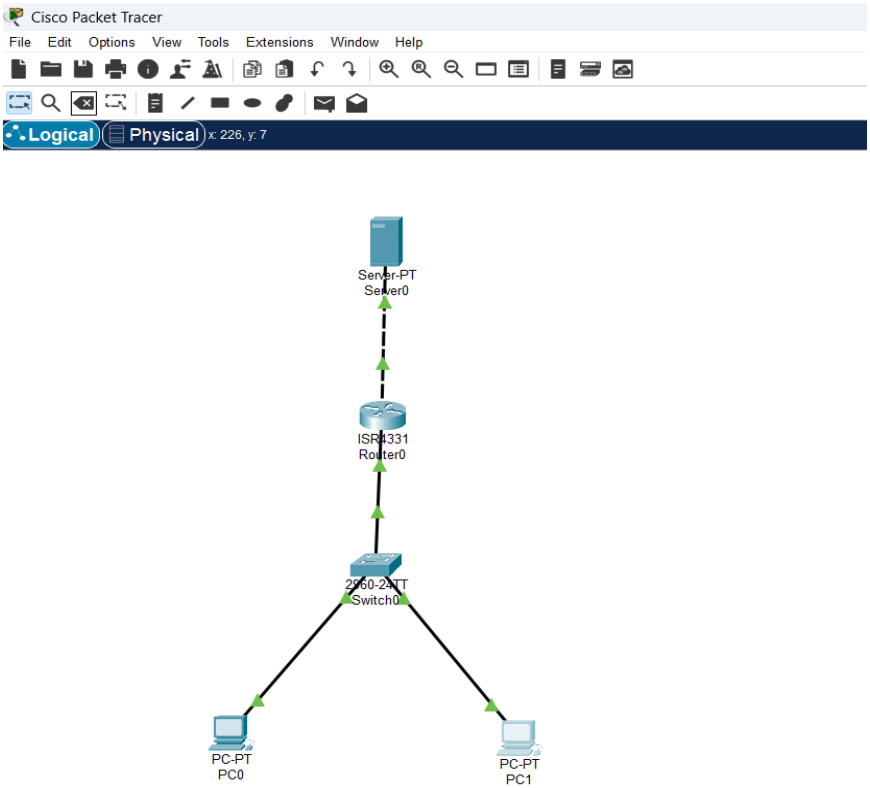
19) Now exit from FTP type ctrl+C, then type dir to check that the file is there in the PC or not.

20) So we have successfully send and got a file from a server using FTP protocol.

1. **Device Requirements:**
   1. Router
   2. Server
   3. PC0
   4. PC1
   5. Switch 0
2. **Network Diagram for your experiment (draw the diagram either hand drawing/ms paint or any other drawing tools)**



3. **Network Diagram (Packet tracer diagram before configuration):**



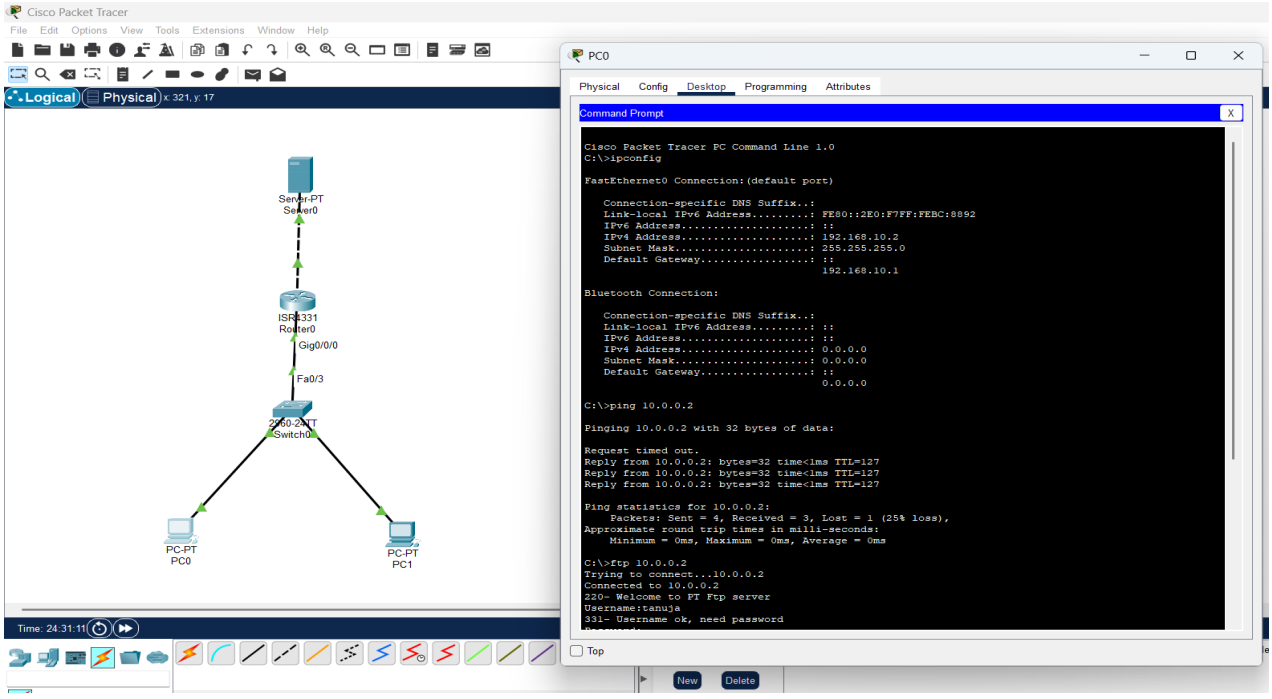
4. Configuration details:

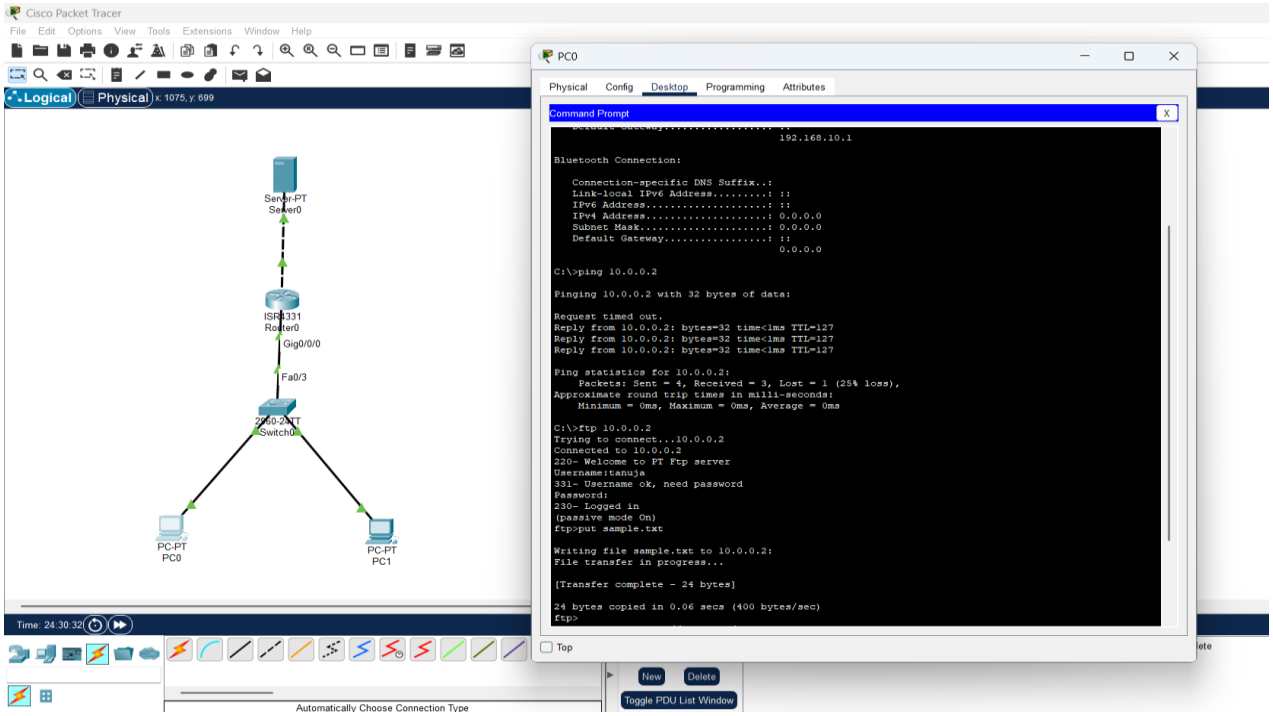
Device Name	Interface Name	IP Address	Subnet mask	Default Gateway
PC 0	Fa 0 – fa0/1	192.168.10.2	255.255.255.0	192.168.10.1
PC 1	Fa 0 – fa0/2	192.168.10.3	255.255.255.0	192.168.10.1

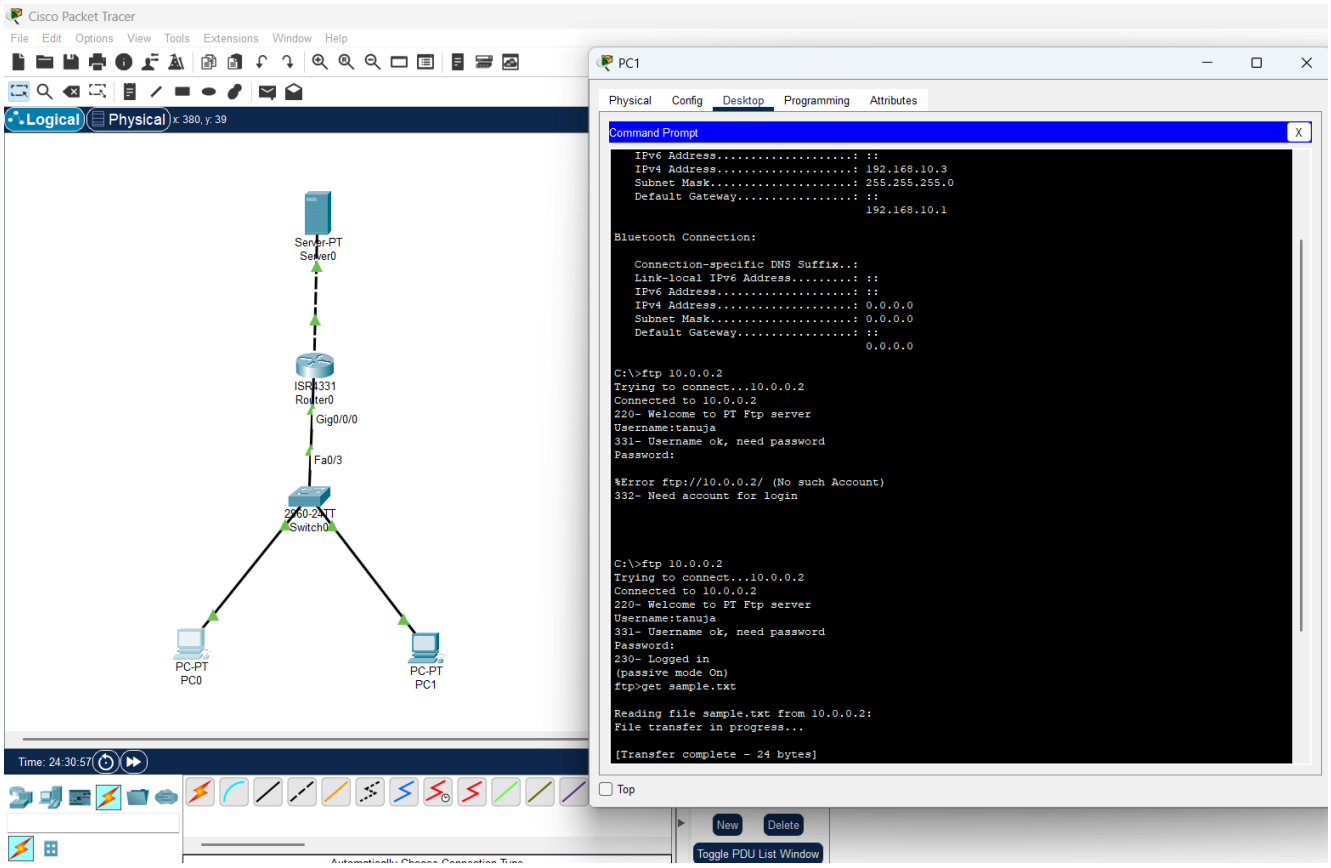
5. Describe step by step configuration steps properly (you may copy the commands used in the configuration tab and paste it.)

- ipconfig
- ping
- ftp
- put (to import)
- get (to download)

6. Output Diagram (Minimum 3 screenshot):




Cisco Packet Tracer
File Edit Options View Tools Extensions Window Help
Logical Physical x: 1075, y: 699
Server-PT Server0
ISR4331 Router0
Gig0/0/0
Fa0/3
2960-24TT Switch0
PC-PT PC0
PC-PT PC1
Time: 24:30:32
Automatically Choose Connection Type
PC0
Physical Config Desktop Programming Attributes
Command Prompt
192.168.10.1
Bluetooth Connection:
Connection-specific DNS Suffix..:
Link-local IPv6 Address.....: ::
IPv6 Address.....: ::
IPv4 Address.....: 0.0.0.0
Subnet Mask.....: 0.0.0.0
Default Gateway.....: ::
0.0.0.0
C:\>ping 10.0.0.2
Pinging 10.0.0.2 with 32 bytes of data:
Request timed out.
Reply from 10.0.0.2: bytes=32 time<1ms TTL=127
Reply from 10.0.0.2: bytes=32 time<1ms TTL=127
Reply from 10.0.0.2: bytes=32 time<1ms TTL=127
Ping statistics for 10.0.0.2:
Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\>ftp 10.0.0.2
Trying to connect...10.0.0.2
Connected to 10.0.0.2
220- Welcome to PT Ftp server
Username:tanuja
331- Username ok, need password
Password:
230- Logged in
(passive mode On)
ftp>put sample.txt
Writing file sample.txt to 10.0.0.2:
File transfer in progress...
[Transfer complete - 24 bytes]
24 bytes copied in 0.06 secs (400 bytes/sec)
ftp>
Top
New
Delete
Toggle PDU List Window


Cisco Packet Tracer
File Edit Options View Tools Extensions Window Help
Logical Physical x: 380, y: 39
Server-PT Server0
ISR4331 Router0
Gig0/0/0
Fa0/3
2960-24TT Switch0
PC-PT PC0
PC-PT PC1
Time: 24:30:57
PC1
Physical Config Desktop Programming Attributes
Command Prompt
IPv6 Address.....: ::
IPv4 Address.....: 192.168.10.3
Subnet Mask.....: 255.255.255.0
Default Gateway.....: ::
192.168.10.1
Bluetooth Connection:
Connection-specific DNS Suffix..:
Link-local IPv6 Address.....: ::
IPv6 Address.....: ::
IPv4 Address.....: 0.0.0.0
Subnet Mask.....: 0.0.0.0
Default Gateway.....: ::
0.0.0.0
C:\>ftp 10.0.0.2
Trying to connect...10.0.0.2
Connected to 10.0.0.2
220- Welcome to PT Ftp server
Username:tanuja
331- Username ok, need password
Password:
%Error ftp://10.0.0.2/ (No such Account)
332- Need account for login
C:\>ftp 10.0.0.2
Trying to connect...10.0.0.2
Connected to 10.0.0.2
220- Welcome to PT Ftp server
Username:tanuja
331- Username ok, need password
Password:
230- Logged in
(passive mode On)
ftp>get sample.txt
Reading file sample.txt from 10.0.0.2:
File transfer in progress...
[Transfer complete - 24 bytes]
Top
New
Delete
Toggle PDU List Window

Rubrics for Experiment Assessment:

Rubrics	Good	Normal	Poor	Marks
Creation of Topology (4)	Created the topology, Identify the proper devices and making the connections **(4)**	Created the topology, Identify the proper devices, making the connections But missing some features **(3)**	Created wrong topology, Failed to Identify the proper devices and making connections **(1)**	
Verify the connectivity (4)	Verified the connectivity in all the levels **(4)**	Verified the connectivity at some levels (only some nodes) **(2)**	Verified the connectivity is not done. **(1)**	
Timely Completion (2)	Completed the lab before the allotted time **(2)**	Completed the lab after the deadline **(1)**	Did not submitted before grading **(0)**	
			Total	

Result: Thus the implement of configuration of FTP service has been executed successfully in Cisco Packet

Ex.No:16	E-mail server Configuration
Date :	

Objective(s):

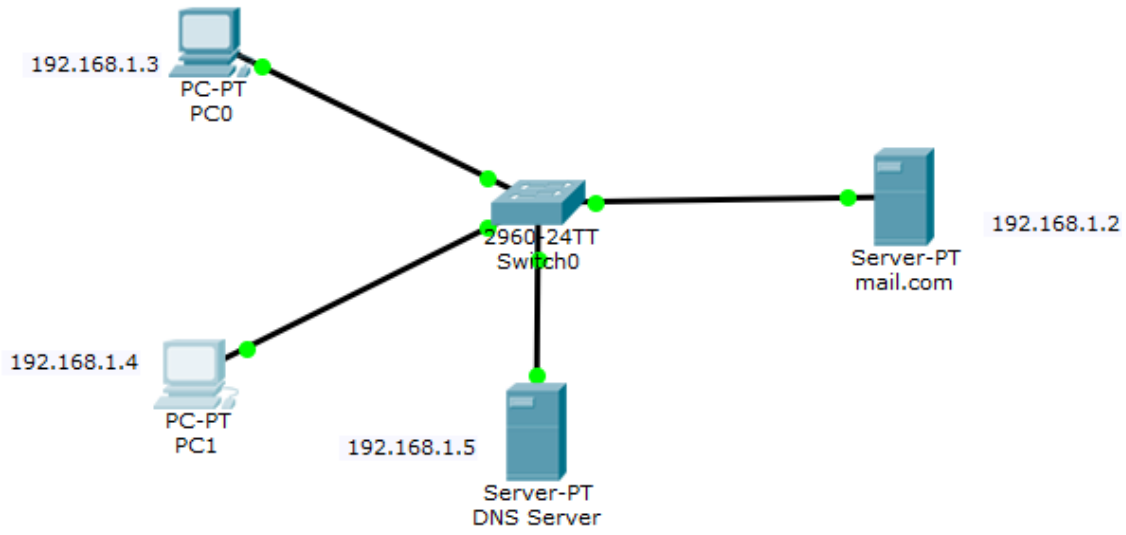
To design and implement Email server configuration using packet tracer

Introduction:

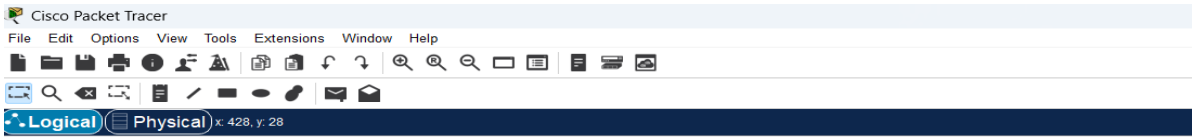
An email server, such as Gmail stores and sends email messages to email clients on request. We often send and receive emails on our mobile devices or computers. Have you ever imagined how this happens? Well, whenever you compose and send an email to another person, the message you send first goes to a mail server. It's the mail server which then sends the email when it is requested from the email client (e.g Gmail App) of the recipient's device.

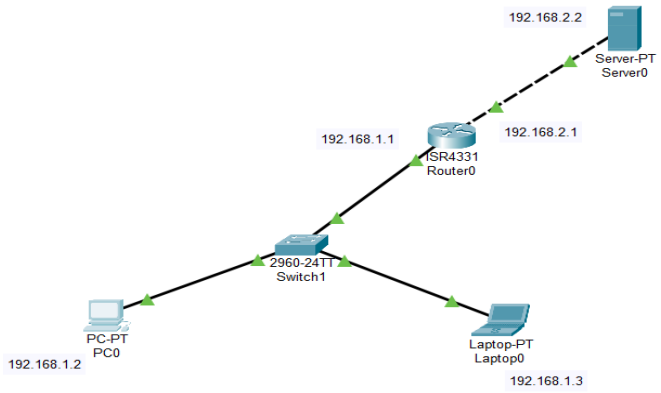
So now, lets configure a mail server in Packet Tracer. And have in mind that although our main focus is configuring an email server, we'll still need services of a DNS server at one point.

1. **Device Requirements:**
   1. Server
   2. Router
   3. Switch
   4. PC0
2. **Network Diagram for your experiment (draw the diagram either hand drawing/ms paint or any other drawing tools)**



3. Network Diagram (Packet tracer diagram before configuration):





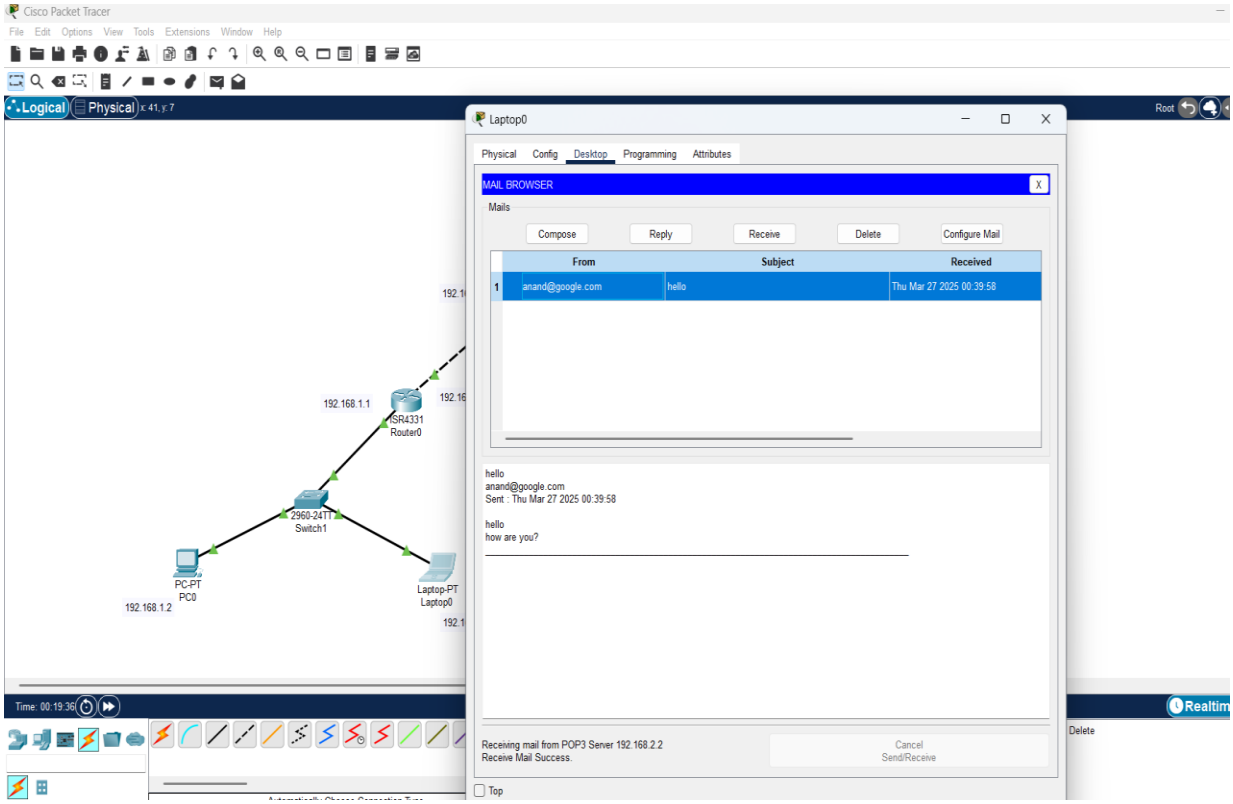
4. Configuration details:

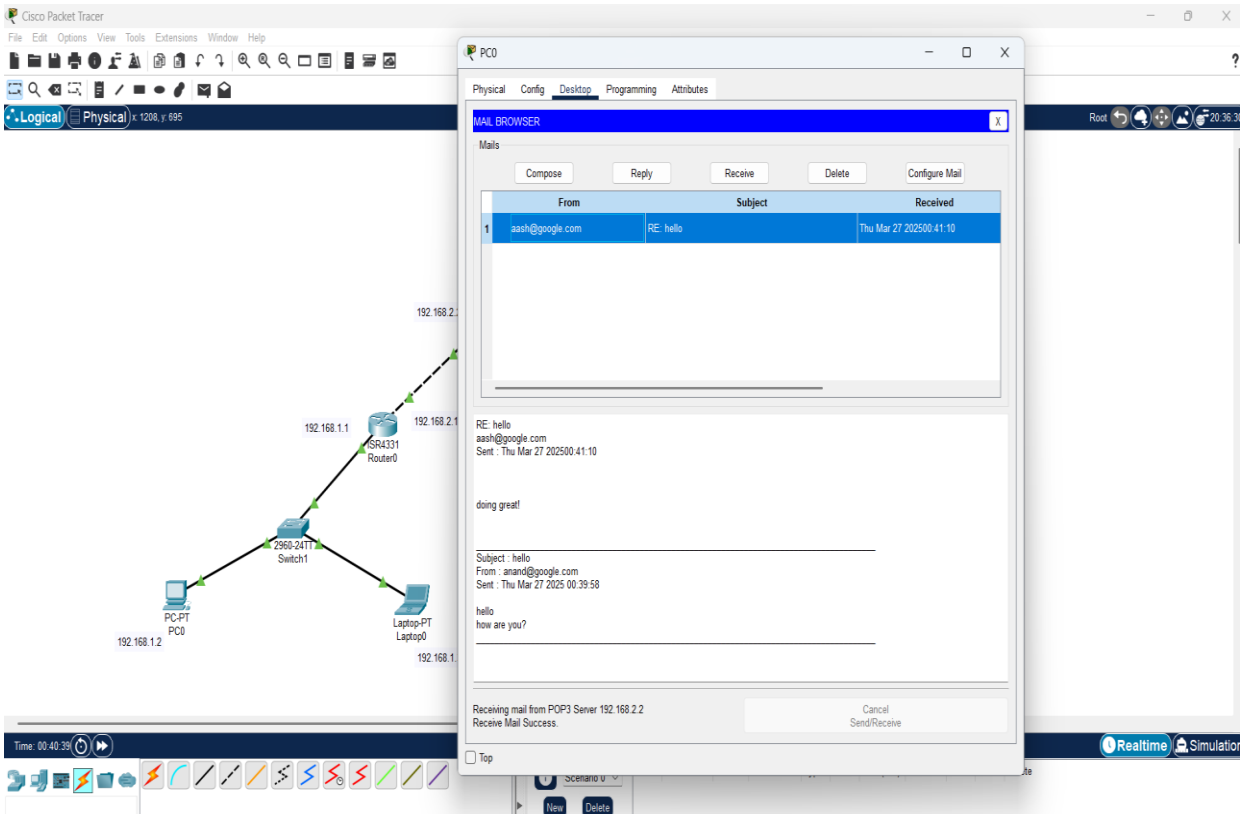
Device Name	Interface Name	IP Address	Subnet mask	Default Gateway
Router	Fa- 0 fa0/0	192.168.1.1	255.255.255.0	
Server	Fast Ethernet0	192.168.2.2	255.255.255.0	192.168.2.1
Switch				
Pc0	Fast Ethernet0	192.168.1.2	255.255.255.0	192.168.1.1
Laptop 0	Fast Ethernet0	192.168.1.3	255.255.255.0	192.168.1.1

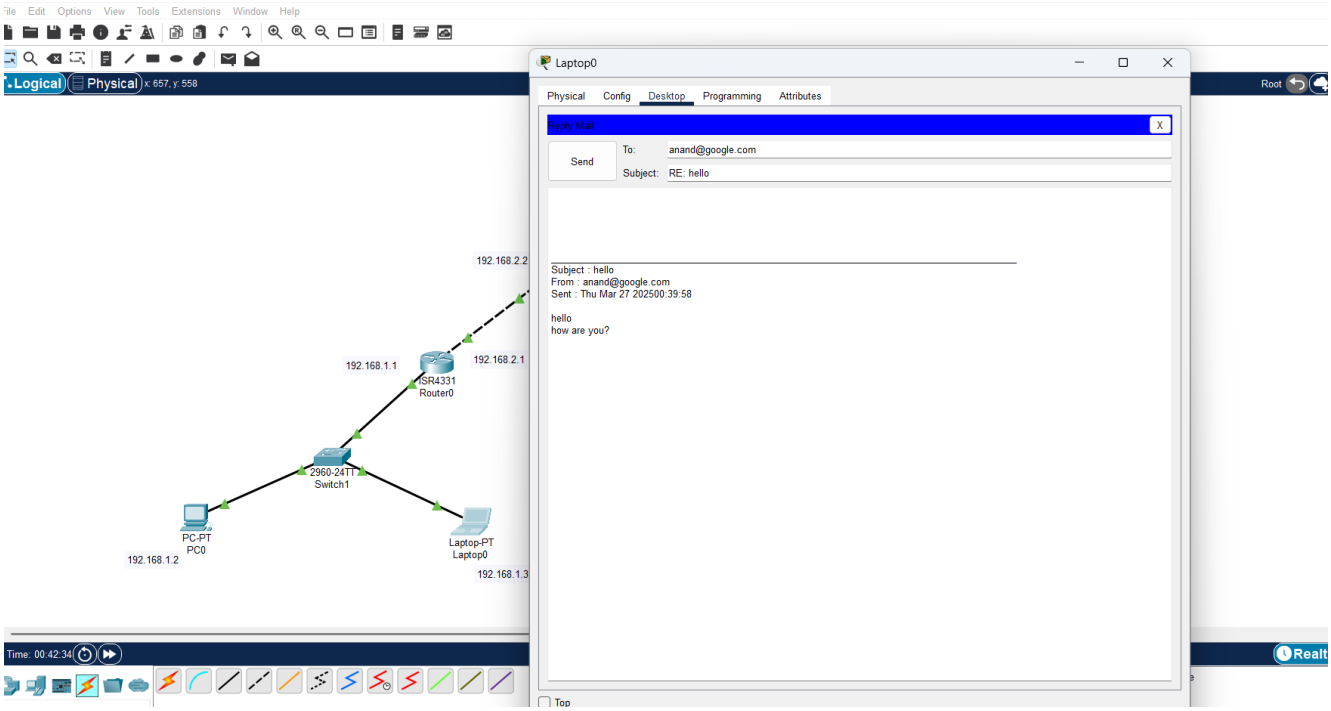
5. Describe step by step configuration steps properly (you may copy the commands used in the configuration tab and paste it.)

- **Create IP address**
- **Compose email through pc0**
- **Receive email through laptop**
- **Reply back for email received**

6. Output Diagram (Minimum 3 screenshot):







Rubrics for Experiment Assessment:

Rubrics	Good	Normal	Poor	Marks
Creation of Topology (4)	Created the topology, Identify the proper devices and making the connections **(4)**	Created the topology, Identify the proper devices, making the connections But missing some features **(3)**	Created wrong topology, Failed to Identify the proper devices and making connections **(1)**	
Verify the connectivity (4)	Verified the connectivity in all the levels **(4)**	Verified the connectivity at some levels (only some nodes) **(2)**	Verified the connectivity is not done. **(1)**	
Timely Completion (2)	Completed the lab before the allotted time **(2)**	Completed the lab after the deadline **(1)**	Did not submitted before grading **(0)**	
			Total	

Result: Thus the implement of configuration of Email service has been executed successfully in Cisco Packet