Lab 4: Introduction to Packet Tracer

Theory:

Cisco Packet Tracer is a widely used network simulation tool designed to teach networking concepts, ranging from basic to advanced levels. It plays a crucial role in helping learners understand and design network topologies, configurations, and simulations in a controlled, virtual environment. By offering a hands-on approach, Packet Tracer allows users to experiment with various networking scenarios, making it an essential resource for developing real-world networking skills and preparing for certifications like the Cisco Certified Network Associate (CCNA).

Key Concepts of Packet Tracer

- Cisco Packet Tracer is a powerful tool that allows users to create and visualize network topologies.
- It configures networking devices such as routers, switches, and PCs, allowing users to practice setting up and managing networks.
- Packet Tracer also includes both real-time and simulation modes, which let users test and observe network configurations and behavior.

Interface of Cisco Packet Tracer

I. Work Space Details

Explain the workspace where users can design and simulate networks. It consists of a large area for building networks, where devices and connections are placed.

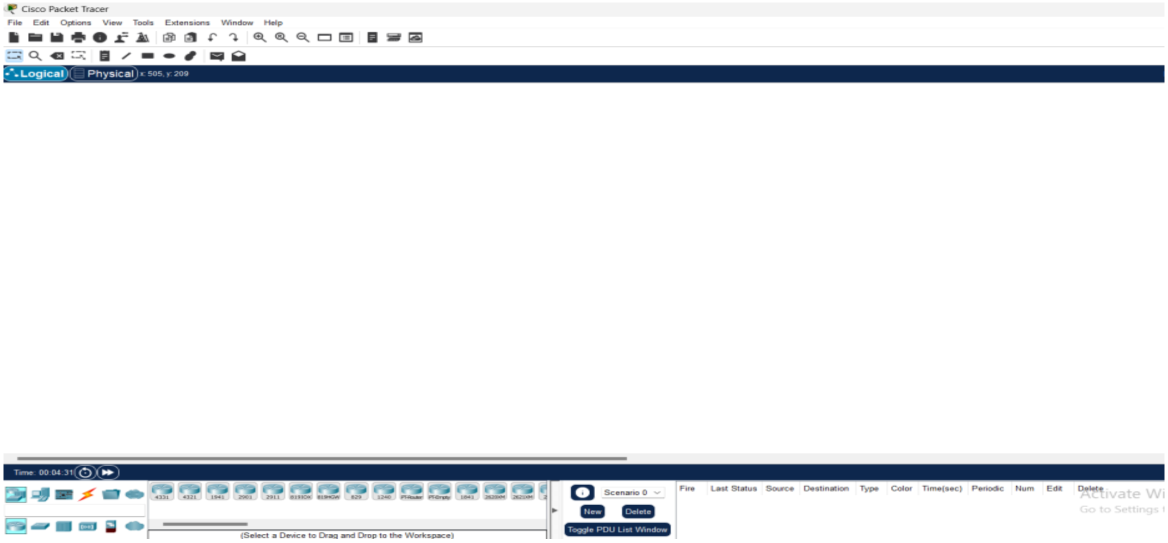

Cisco Packet Tracer
File Edit Options View Tools Extensions Window Help
Logical
Physical
x: 505, y: 209
Time: 00:04:31
(Select a Device to Drag and Drop to the Workspace)
Scenario 0
New
Delete
Toggle PDU List Window
Fire Last Status Source Destination Type Color Time(sec) Periodic Num Edit Delete
Activate Windows
Go to Settings

Fig 1.0: Workspace Details

II. Toolbar

Describes the toolbar that contains icons for essential tools, such as selecting, connecting devices, zooming, etc.

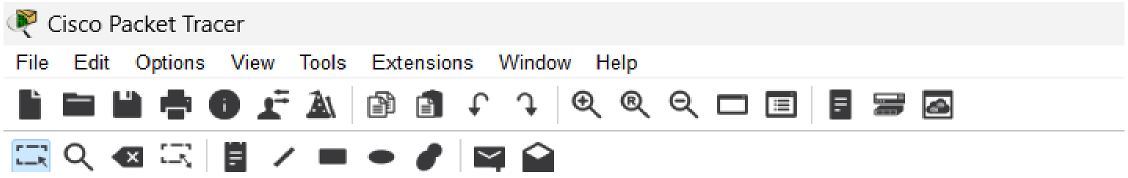


Fig 1.1: Toolbar

III. Device-Type Selection Panel

This section allows you to choose the types of devices (routers, switches, end devices) and connections you want to use in your network.

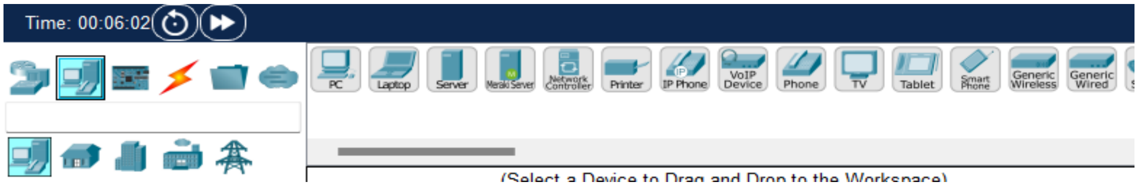


Fig 1.2: Device-Type Selection Panel

IV. Device Configurations

Explain how you configure each device (by clicking on it) and setting parameters such as IP addresses, routing protocols, and security settings.

V. Real-Time and Simulation Mode

- **Real-Time Mode:** Explains how networks behave in real time.

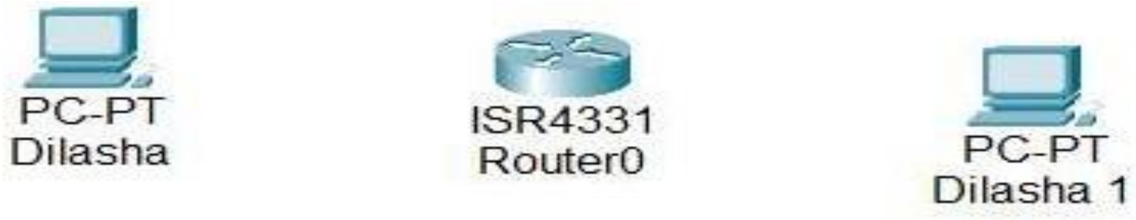
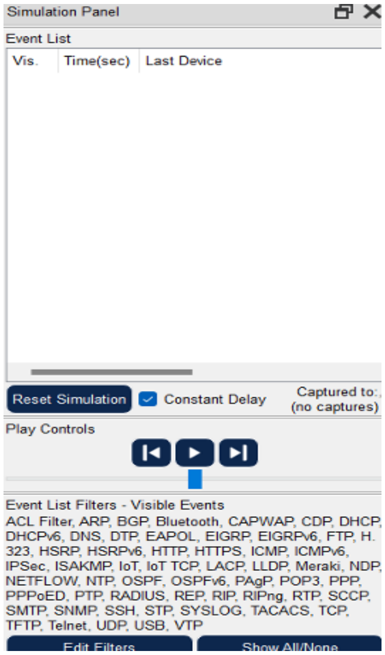


Fig 1.3: Real-Time Model

- **Simulation Mode:** Allows users to pause and analyze the flow of data, making it easier to troubleshoot and understand packet movement. It is also useful for understanding protocol behavior.



☑ Constant Delay

Captured to:
(no captures)

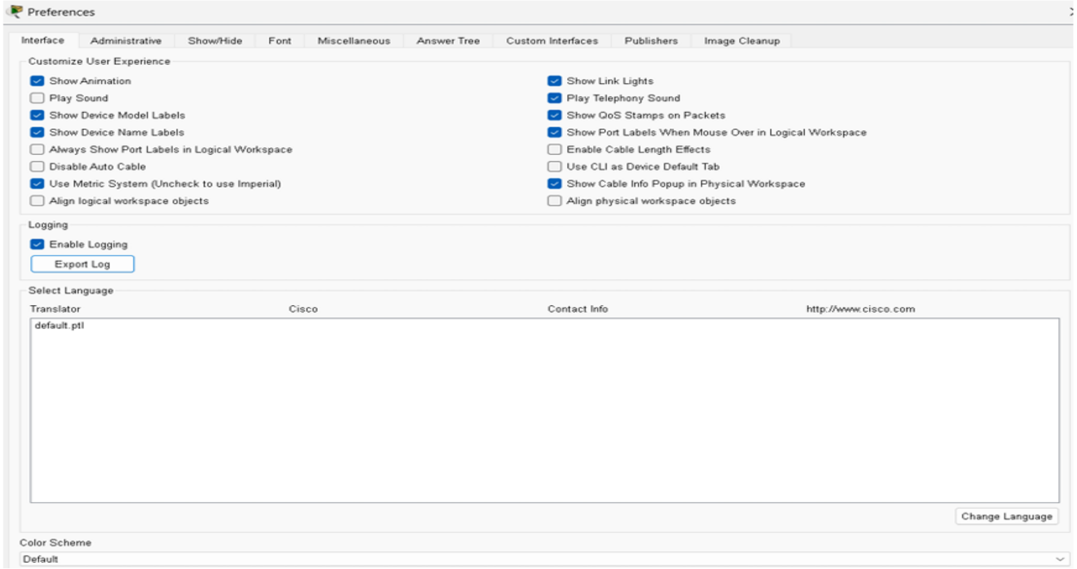
Event List Filters - Visible Events

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, Meraki, 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

Fig 1.4: Simulation Mode

VI. Options and Preferences

Highlight the customizable settings available, such as language preferences, interface themes, and default device settings. This customization helps tailor the software to your personal preferences and working style, enhancing your overall experience.



Customize User Experience

☑ Show Animation
□ Play Sound
☑ Show Device Model Labels
☑ Show Device Name Labels
□ Always Show Port Labels in Logical Workspace
□ Disable Auto Cable
☑ Use Metric System (Uncheck to use Imperial)
□ Align logical workspace objects

☑ Show Link Lights
☑ Play Telephony Sound
☑ Show QoS Stamps on Packets
☑ Show Port Labels When Mouse Over in Logical Workspace
□ Enable Cable Length Effects
□ Use CLI as Device Default Tab
☑ Show Cable Info Popup in Physical Workspace
□ Align physical workspace objects

Logging

☑ Enable Logging

Select Language

Translator	Cisco	Contact Info	http://www.cisco.com
default.ptl			

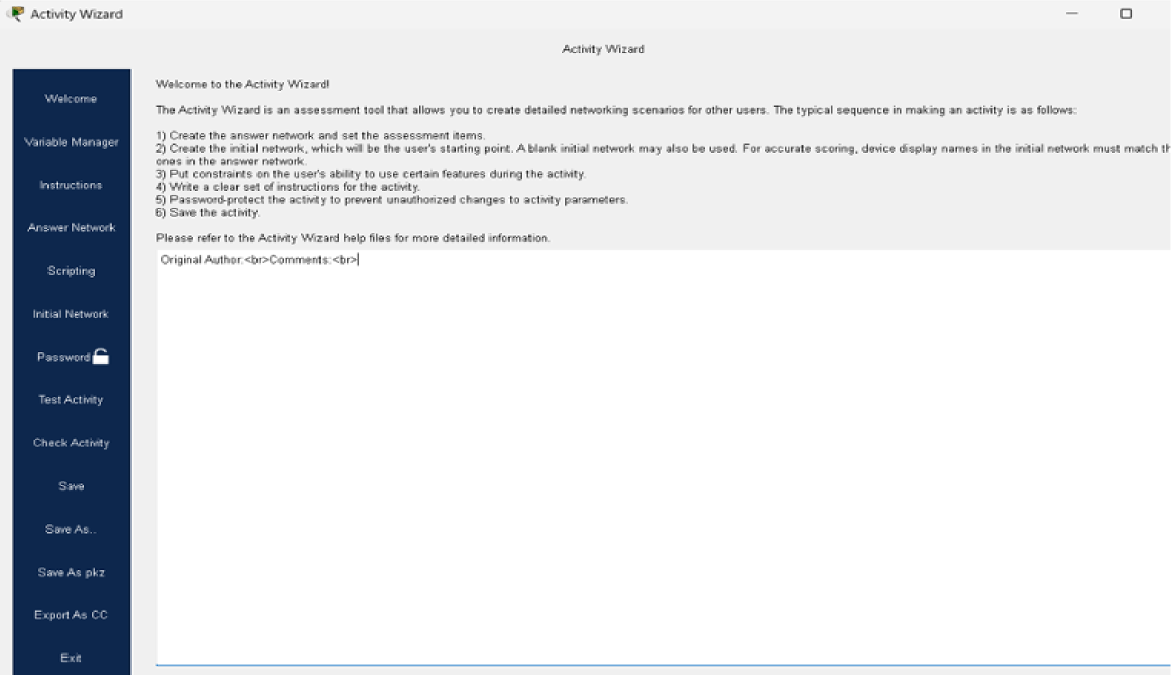
Color Scheme

Default

Fig 1.5: Options and Preferences

VII. Activity Wizard

The Activity Wizard is a feature designed for creating interactive learning activities within Packet Tracer. Instructors can use it to design tasks, provide step-by-step instructions, and set up assessments that students can follow within the software. This tool is particularly useful for educational purposes, allowing students to practice and test their networking skills in a guided and structured environment.



Welcome to the Activity Wizard!

The Activity Wizard is an assessment tool that allows you to create detailed networking scenarios for other users. The typical sequence in making an activity is as follows:

1) Create the answer network and set the assessment items.
2) Create the initial network, which will be the user's starting point. A blank initial network may also be used. For accurate scoring, device display names in the initial network must match th
ones in the answer network.
3) Put constraints on the user's ability to use certain features during the activity.
4) Write a clear set of instructions for the activity.
5) Password-protect the activity to prevent unauthorized changes to activity parameters.
6) Save the activity.

Please refer to the Activity Wizard help files for more detailed information.

Original Author:
Comments:

Fig 1.6: Activity Wizard

Conclusion:

In this lab, we have built a solid understanding of network simulation and configuration using Cisco Packet Tracer. By exploring its interface and main features, we have gained the confidence to set up devices, simulate different network designs, and fix issues. This hands-on experience has shown how useful the tool is for developing important networking skills and has sparked our interest to dive into more advanced topics in network design and management as we continue learning.

Lab 5: Creating a LAN and testing the connectivity using Packet Tracer

Theory:

A **Local Area Network (LAN)** is a network that links computers and devices within a confined geographic area, such as a home, office, or building. LANs facilitate the sharing of resources like files, printers, and internet connections among multiple devices. Due to their limited coverage, LANs typically offer higher data transfer speeds and lower latency compared to larger networks like **Wide Area Networks (WANs)**.

LAN Architecture
LAN architecture defines the structure, components, and communication protocols of a LAN. It involves several key elements:

Ethernet Cables: Commonly used cables include Cat5e, Cat6, and fiber optics.

Network Interface Cards (NICs): NICs are hardware components that allow devices to connect to the Ethernet.

Topologies
The physical arrangement of devices in a LAN is called topology. Common topologies include:
1. Star Topology
2. Bus Topology
3. Ring Topology
4. Mesh Topology etc.

Protocols
LAN's use protocols to ensure smooth communication. Some common protocols include:
Ethernet: The most common protocol for wired LAN's.

b. Components Used

- Router
- Switch
- End devices (PCs, Laptops)
- Cables (Ethernet)
- Network Interface Cards (NICs)

Network Diagram:

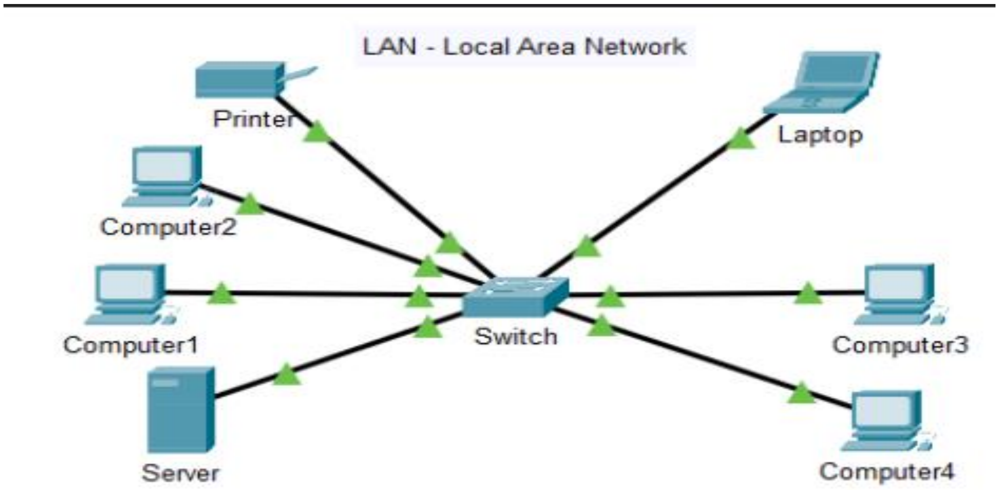


Fig 1.0: Network diagram of LAN

3. Implementation Sequence:

The following steps outline how the LAN was configured in Cisco Packet Tracer:

Step 1: Add the nodes and network devices:

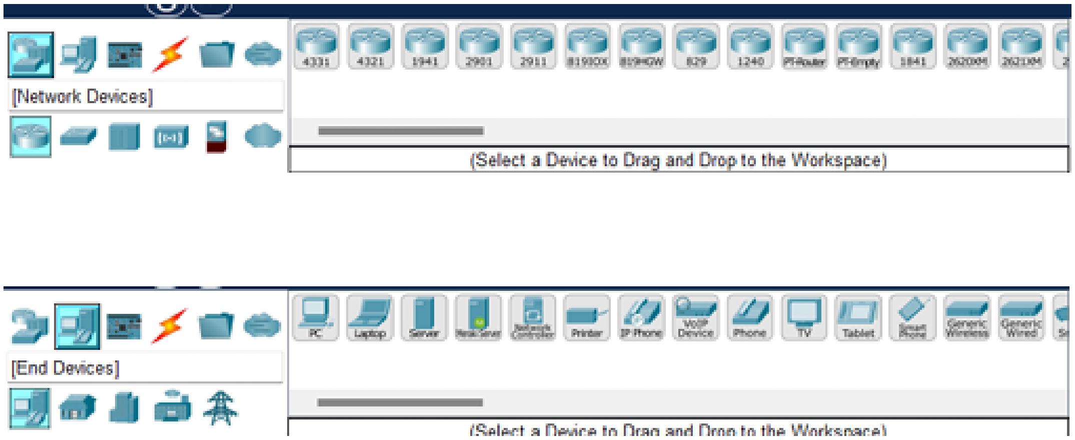


Fig 1.1: Adding Nodes and Switch

Step 2: Place PCs and a switch in the workspace

- Drag and drop a switch onto the workspace.
- Drag and drop four PCs onto the workspace labeled PC0, PC1, PC2, PC3.

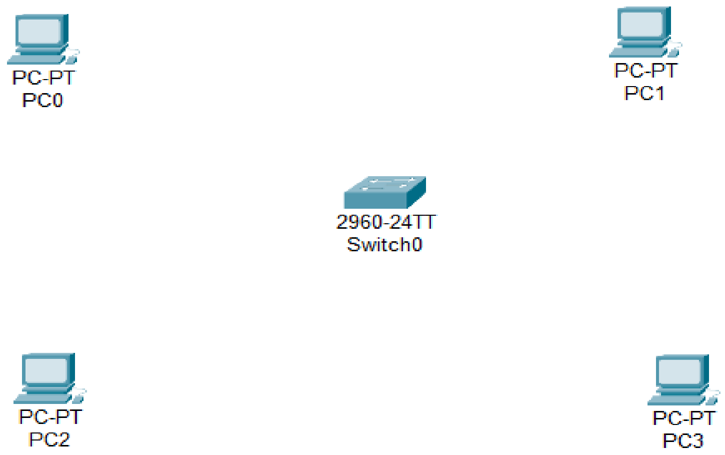


Fig 1.2: Switch an PC in the workspace

Step 3: Connect the Devices

- Choose the **“Copper Straight-Through”** cable.
- Click on the first PC and select the **“FastEthernet”** port.
- Click on the switch and select one of the available **“FastEthernet”** ports.
- Also, Renamed the PC as Dilasha1, Dilasha2, Dilasha3, Dilasha4

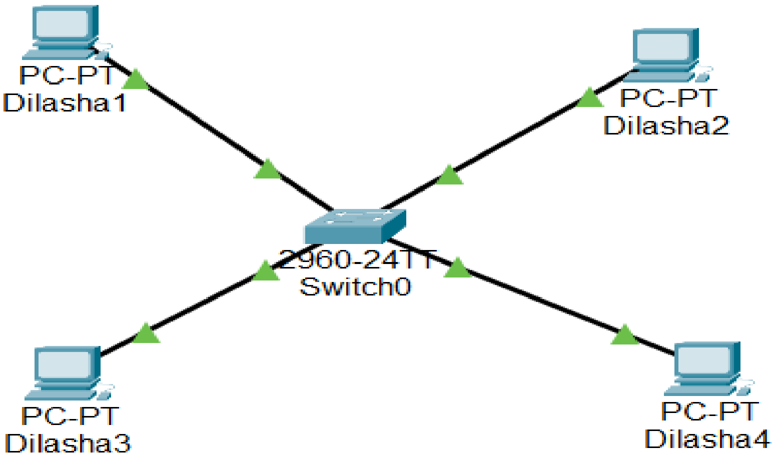
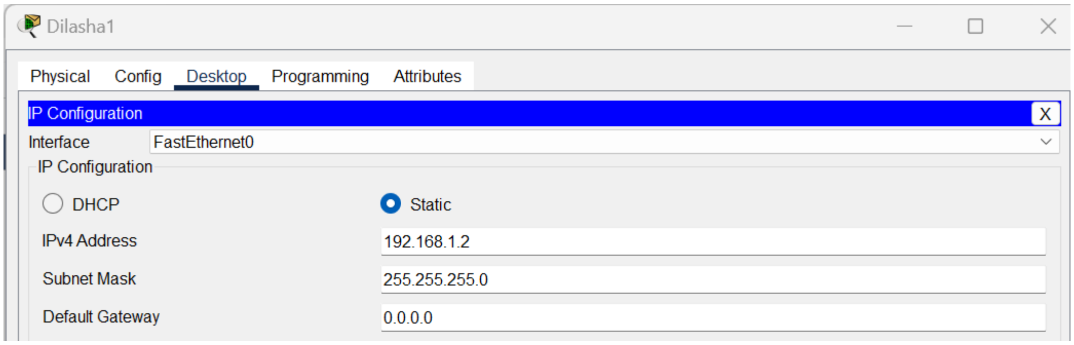


Fig 1.3: Connection between switch and PCs

Step 4: Configure IP addresses:

Click on each PC and assign different IPs to each PCs.

- Dilasha1-192.168.1.2
- Dilasha2-192.168.1.3
- Dilasha3-192.168.1.4
- Dilasha4-192.168.1.5



Interface: FastEthernet0

IP Configuration

○ DHCP ● Static

IPv4 Address: 192.168.1.2

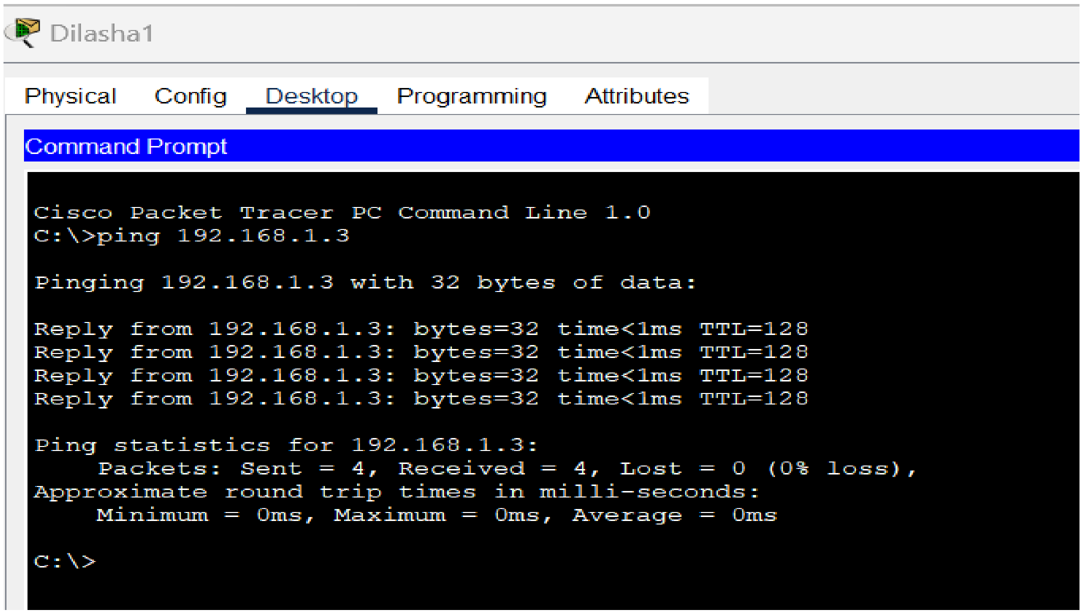
Subnet Mask: 255.255.255.0

Default Gateway: 0.0.0.0

Fig 1.3: IP configuration

Step 5: Test the connectivity between the PCs

- To test whether the network is working, you can ping other devices on the network from each PC.
- To ping another device, open a command prompt on the PC and type "ping <IP address of the other device>."
- If the ping is successful, you should see replies from the other device.



```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.1.3

Pinging 192.168.1.3 with 32 bytes of data:

Reply from 192.168.1.3: bytes=32 time<1ms TTL=128
Reply from 192.168.1.3: bytes=32 time<1ms TTL=128
Reply from 192.168.1.3: bytes=32 time<1ms TTL=128
Reply from 192.168.1.3: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.1.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>
```

Fig 1.4: Testing connectivity between Dilasha1 and Dilasha3

In the above figure, we test the connectivity of network by pinging PC with IP Address 192.168.1.3 from PC with IP Address 192.168.1.1. The ping was successful as we received replies from the destination PC with 0% Loss.

Conclusion:

In conclusion, we have successfully completed this project by designing and configuring a fully operational LAN network using Cisco Packet Tracer. From setting up and interconnecting devices to configuring IP addresses and testing network connectivity, we gained practical, hands-on experience in establishing a basic network. This project not only strengthened our understanding of LAN architecture and device configuration but also emphasized the significance of each network component in ensuring smooth and efficient communication.

Lab 6: Creating network topologies using Packet Tracer

Theory

Network topology refers to the arrangement or layout of various elements (nodes, links, devices) in a computer network. It defines how different devices are connected and how data flows through the network. Network topology plays a crucial role in determining the performance, reliability, and scalability of a network. The choice of topology depends on factors such as the size of the network, its purpose, and cost considerations.

Different types of Network Topologies:

1. Bus Topology
2. Ring Topology
3. Star Topology
4. Mesh Topology
5. Tree Topology
6. Hybrid Topology

Ring Topology

Ring Topology connects devices in a circular loop, where data travels in one direction, offering equal access but risking network disruption if one device fails.

Component Used

Hardware: Switches (4), Ethernet cables, End devices (4).
Software: Cisco Packet Tracer

Network Diagram

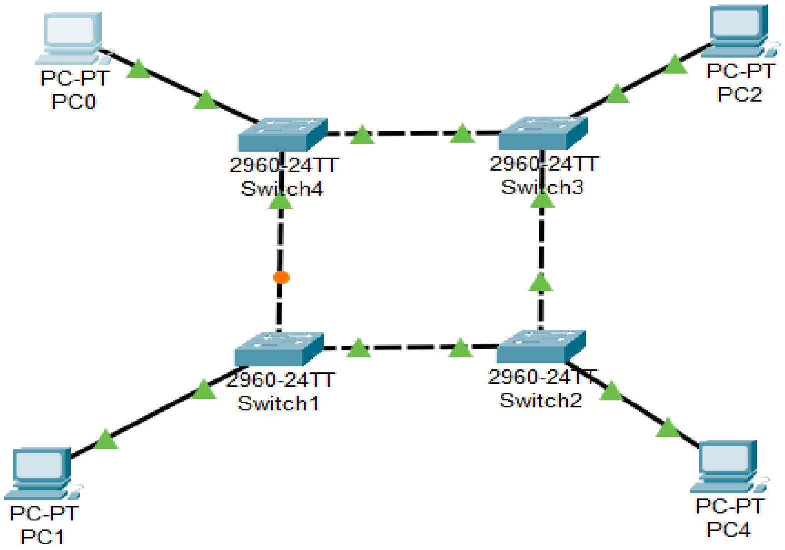


Fig 1.0: Network diagram for Ring Topology

Implementation Sequence

Here is the implementation sequence for creating the Ring Topology shown in the image using Cisco Packet Tracer:

Step 1: Launch Cisco Packet Tracer

Open Cisco Packet Tracer in a computer device.

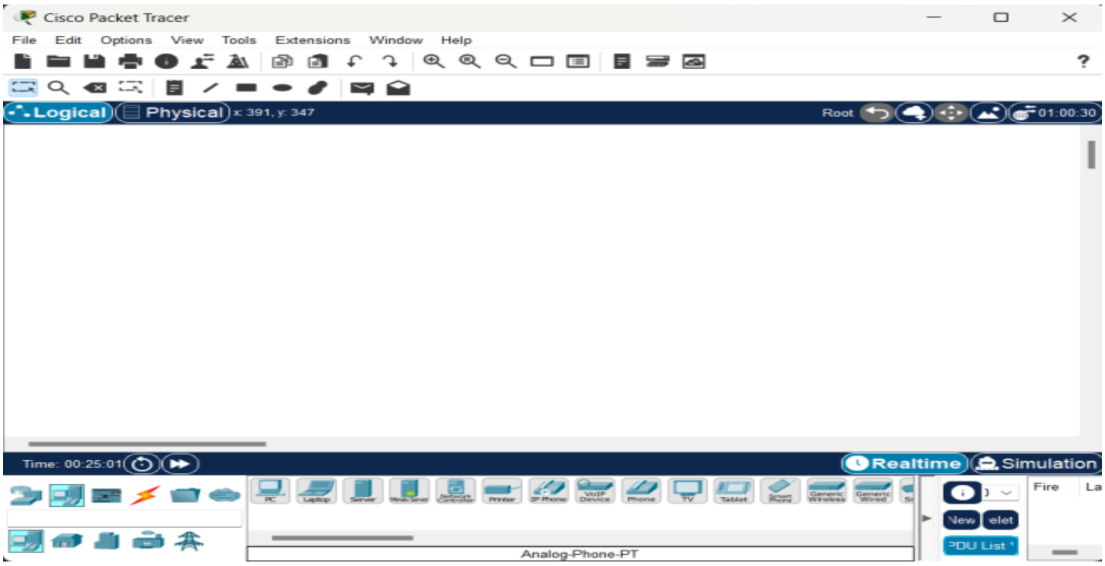


Fig 1.1: Workspace for network design

Step 2: Add the network devices to the workspace

- From the Device-Type Selection box, four switches and add them to the workspace:
- Four PCs labeled as (PC0, PC1, PC2, and PC3)

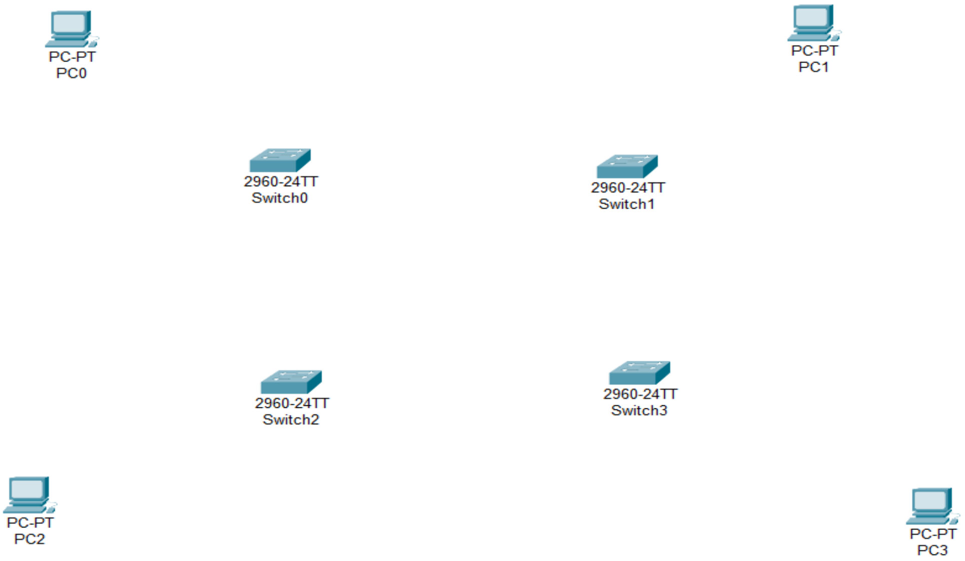


Fig 1.2: Switches and PCs in the workspace

Step 3: Connect the devices

Use the copper straight-through cable to connect each PC to one of the available ports on each switch and copper cross-over cable to connect between each adjacent switch.

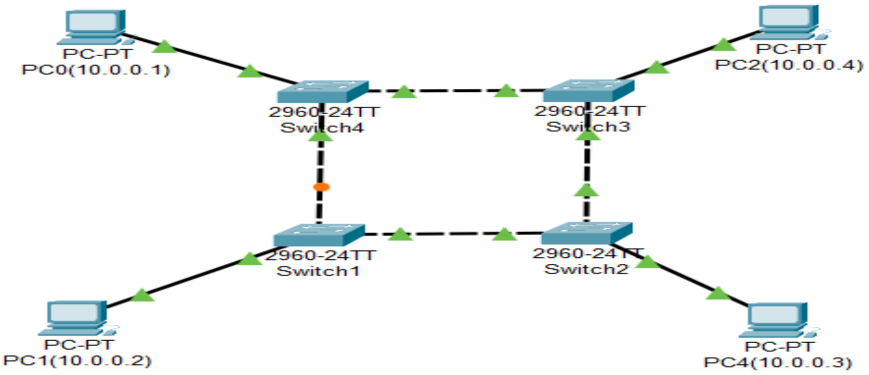
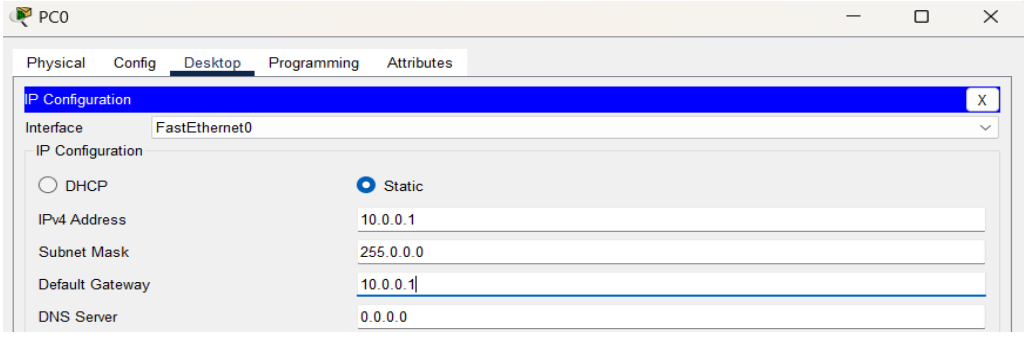


Fig 1.3: Connection between Switch and PC

Step 4: Configure IP addresses

- Right-click on each PC and select "IP Configuration."
- In the IP Configuration window, enter the IP address as 10.0.0.1 for PC1,10.0.0.2 for PC2,10.0.0.3 for PC3 and 10.0.0.4 for PC4, subnet mask, and default gateway.



IP Configuration

Interface FastEthernet0

IP Configuration

○ DHCP ● Static

IPv4 Address 10.0.0.1

Subnet Mask 255.0.0.0

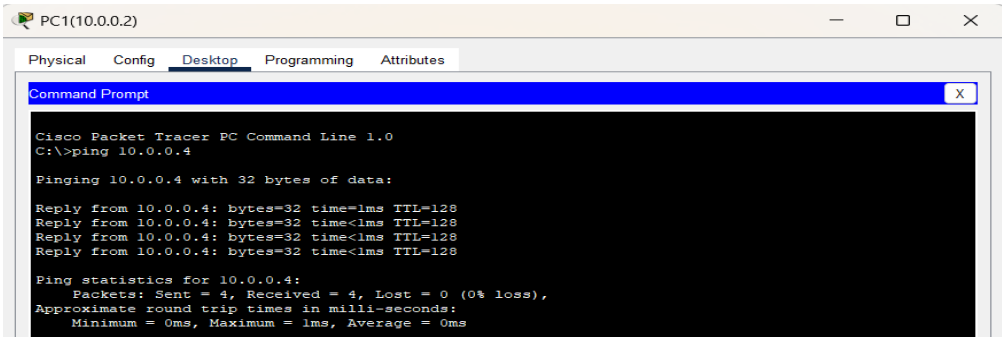
Default Gateway 10.0.0.1

DNS Server 0.0.0.0

Fig 1.4: IP configuration

Step 5: Verify connectivity:

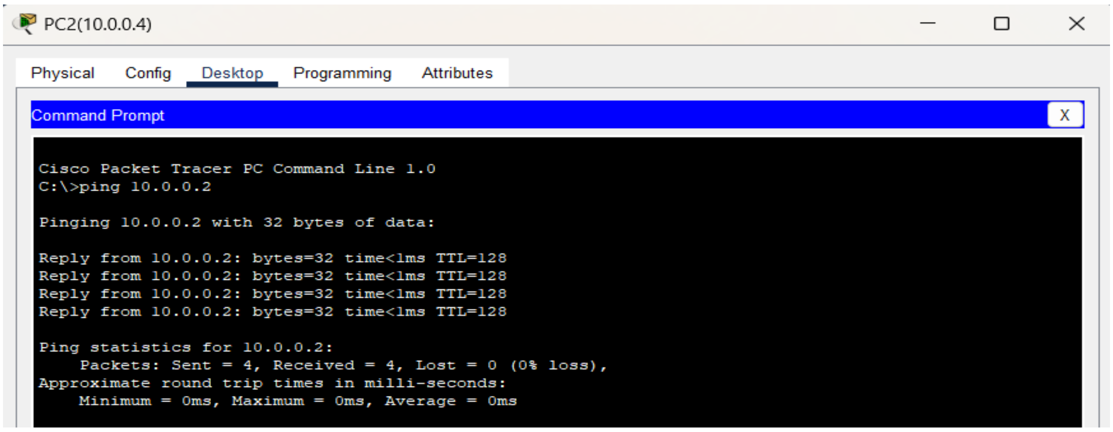
- To check if the network is functioning, ping other devices on the network from each PC.
- Open a command prompt on the PC and enter: ping <IP address of the other device>.
- If the ping is successful, you will receive replies from the other device.



```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 10.0.0.4

Pinging 10.0.0.4 with 32 bytes of data:

Reply from 10.0.0.4: bytes=32 time=1ms TTL=128
Reply from 10.0.0.4: bytes=32 time<1ms TTL=128
Reply from 10.0.0.4: bytes=32 time<1ms TTL=128
Reply from 10.0.0.4: bytes=32 time<1ms TTL=128

Ping statistics for 10.0.0.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

Fig 1.5: Connectivity test between PC1 and PC2

In the above figure, we test the connectivity of network by pinging PC with IP Address 10.0.0.3 from PC with IP Address 10.0.0.2. The ping was successful as we received replies from the destination PC with 0% Loss.



```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 10.0.0.2

Pinging 10.0.0.2 with 32 bytes of data:

Reply from 10.0.0.2: bytes=32 time<1ms TTL=128
Reply from 10.0.0.2: bytes=32 time<1ms TTL=128
Reply from 10.0.0.2: bytes=32 time<1ms TTL=128
Reply from 10.0.0.2: bytes=32 time<1ms TTL=128

Ping statistics for 10.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Fig 1.6: Connectivity test between PC2 and PC1

In the above figure, we test the connectivity of network by pinging PC with IP Address 10.0.0.2 from PC with IP Address 10.0.0.3. The ping was successful as we received replies from the destination PC with 0% Loss.

Star Topology

Star Topology connects all devices to a central hub, which makes it easy to manage, though the entire network depends on the hub's functioning.

Component Used

Hardware: Switches (1), Ethernet cables, End devices(5).
Software: Cisco Packet Tracer

Network Diagram

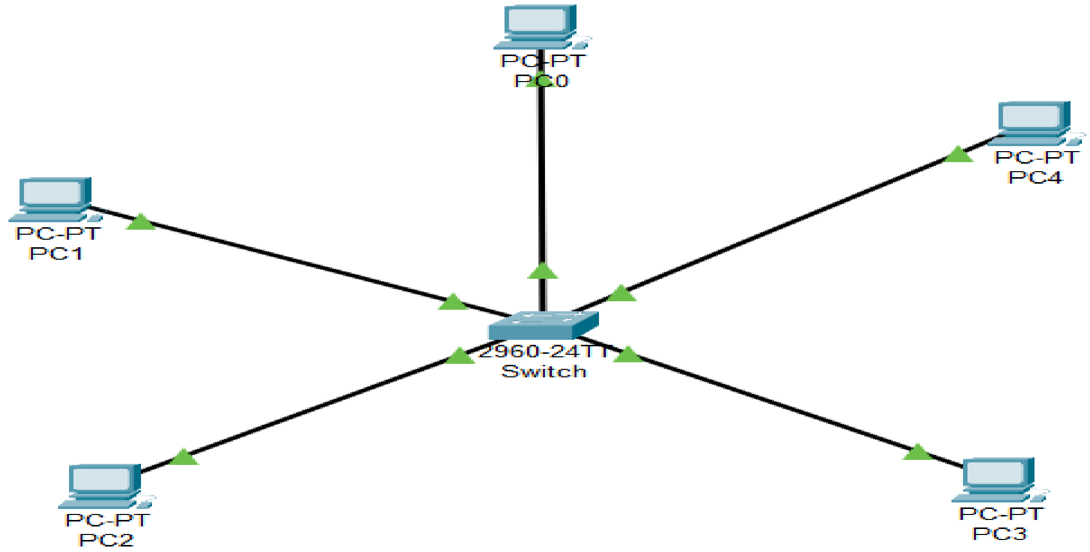


Fig 1.7: Network Map for Star Topology

Implementation Sequence

Here is the implementation sequence for creating the Star Topology shown in the image using Cisco Packet Tracer:

Step 1: Launch Cisco Packet Tracer

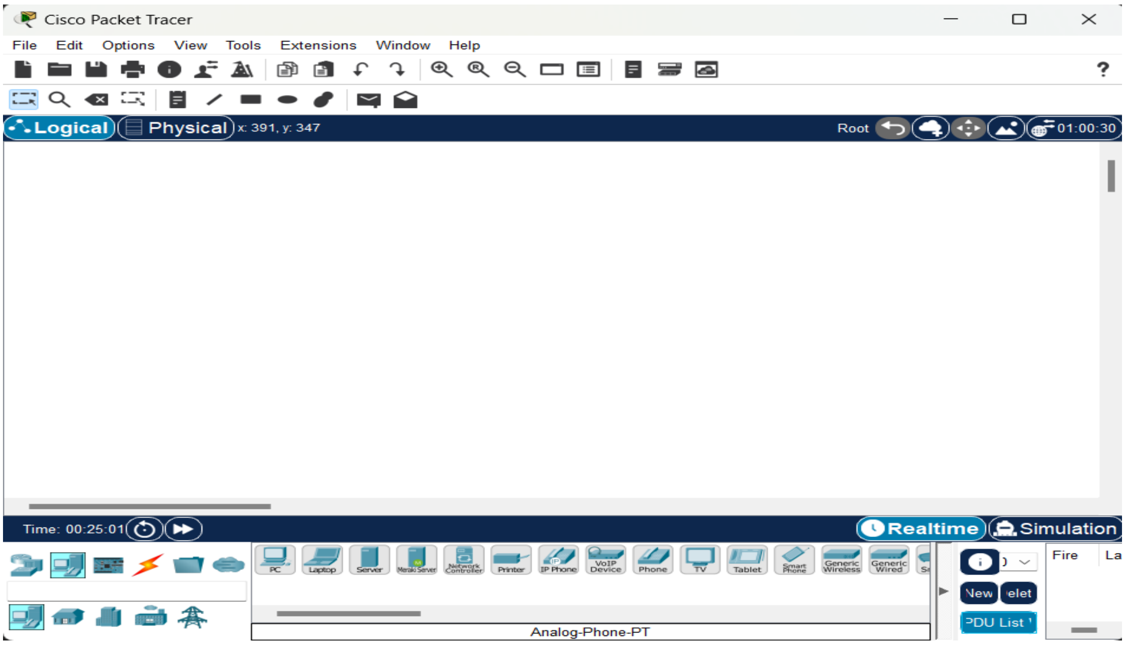


Fig 1.8: Workspace for network design

Step 2: Add the network devices to the workspace

- From the Device-Type Selection box, choose the following devices and add them to the workspace:
- One 2960-24TT Switch
- Five PCs (labeled PC0, PC1, PC2, PC3and PC4)

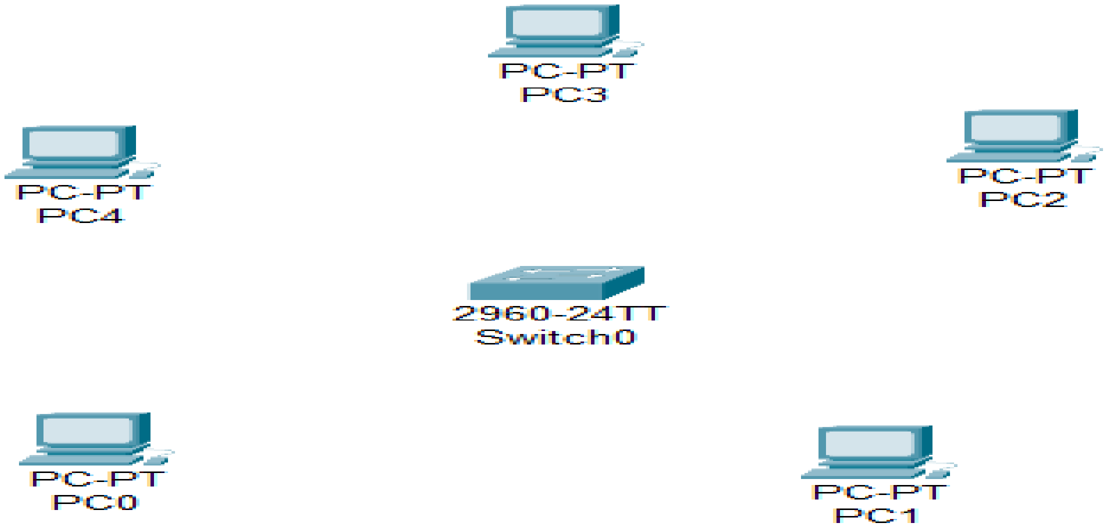


Fig 1.9: Switches and PCs for Star Topology

Step 3: Connect the devices

- Use the copper straight-through cable to connect each PC to one of the available ports on the switch.
- Ensure that each connection is made properly.

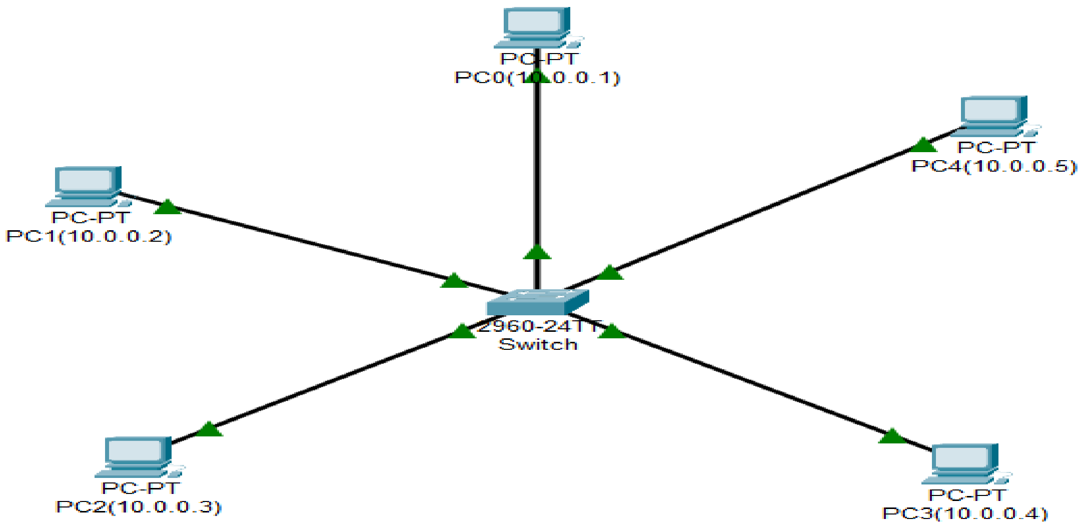
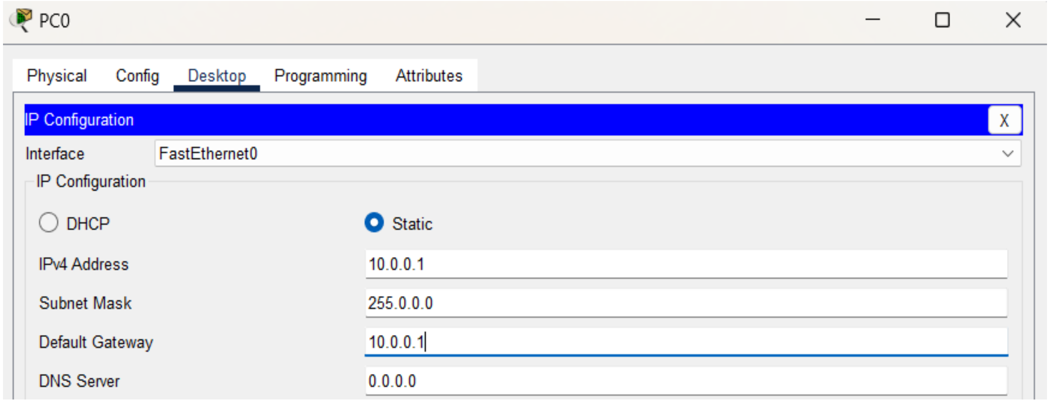


Fig 2.0: Connection between Switch and PC's

Step 4: Configure IP addresses

4.1 Right-click on each PC and select "IP Configuration."

4.2 In the IP Configuration window, enter the IP address as 10.0.0.1 for PC0,10.0.0.2 for PC1,10.0.0.3 for PC2,10.0.0.4 for PC3and 10.0.0.5 for PC4, subnet mask, and default gateway.



IP Configuration

Interface FastEthernet0

IP Configuration

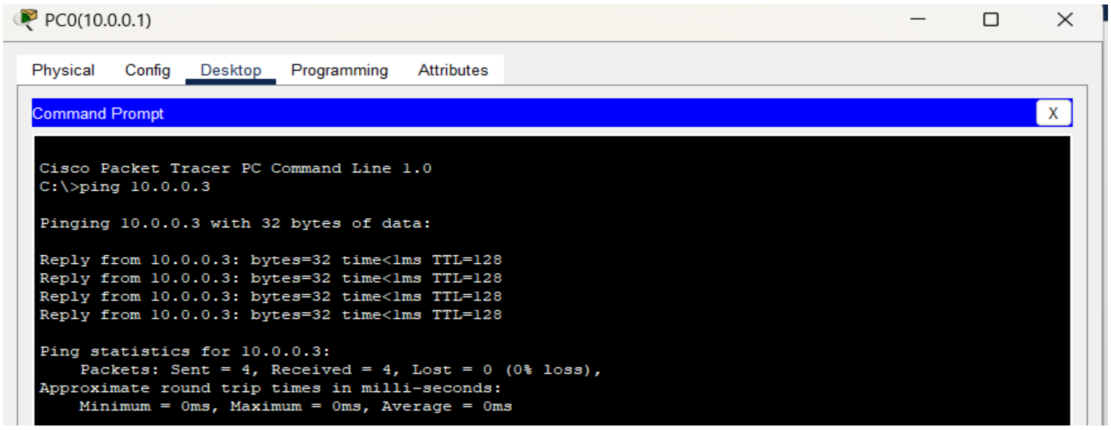
○ DHCP ● Static

IPv4 Address	10.0.0.1
Subnet Mask	255.0.0.0
Default Gateway	10.0.0.1
DNS Server	0.0.0.0

Fig 2.1: IP configuration

Step 5: Verify connectivity:

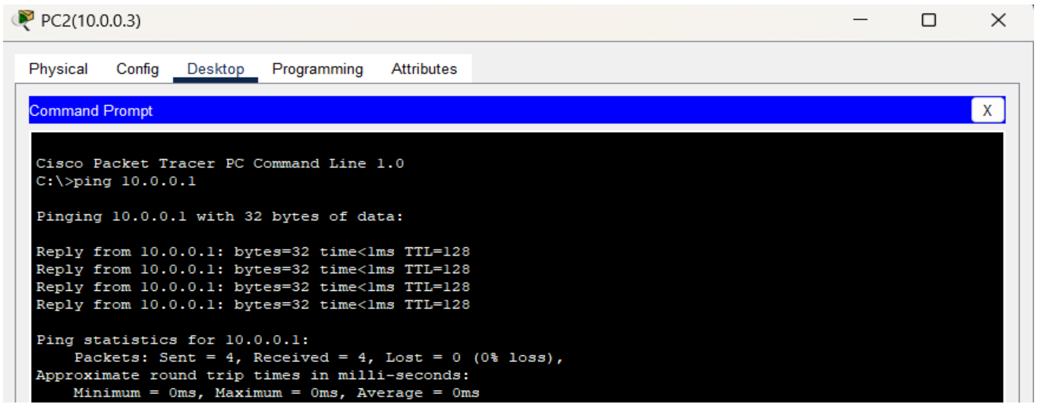
- To test whether the network is working, you can ping other devices on the network from each PC.
- To ping another device, open a command prompt on the PC and type "ping <IP address of the other device>."
- If the ping is successful, you should see replies from the other device.



```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 10.0.0.3

Pinging 10.0.0.3 with 32 bytes of data:

Reply from 10.0.0.3: bytes=32 time<1ms TTL=128
Reply from 10.0.0.3: bytes=32 time<1ms TTL=128
Reply from 10.0.0.3: bytes=32 time<1ms TTL=128
Reply from 10.0.0.3: bytes=32 time<1ms TTL=128

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

Fig 2.2: Connectivity test between PC0 and PC2

In the above figure, we test the connectivity of network by pinging PC with IP Address 10.0.0.3 from PC with IP Address 10.0.0.1. The ping was successful as we received replies from the destination PC with 0% Loss.



```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 10.0.0.1

Pinging 10.0.0.1 with 32 bytes of data:

Reply from 10.0.0.1: bytes=32 time<1ms TTL=128
Reply from 10.0.0.1: bytes=32 time<1ms TTL=128
Reply from 10.0.0.1: bytes=32 time<1ms TTL=128
Reply from 10.0.0.1: bytes=32 time<1ms TTL=128

Ping statistics for 10.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Fig 2.3: Connectivity test between PC2 and PC0

In the above figure, we test the connectivity of network by pinging PC with IP Address 10.0.0.1 from PC with IP Address 10.0.0.3. The ping was successful as we received replies from the destination PC with 0% Loss.

Mesh Topology

Mesh Topology provides high reliability by connecting each device to every other device, allowing multiple data paths, but it is costly and complex to implement.

Component Used

Hardware: Switches (4), Ethernet cables, End devices (4).
Software: Cisco Packet Tracer

Network Diagram

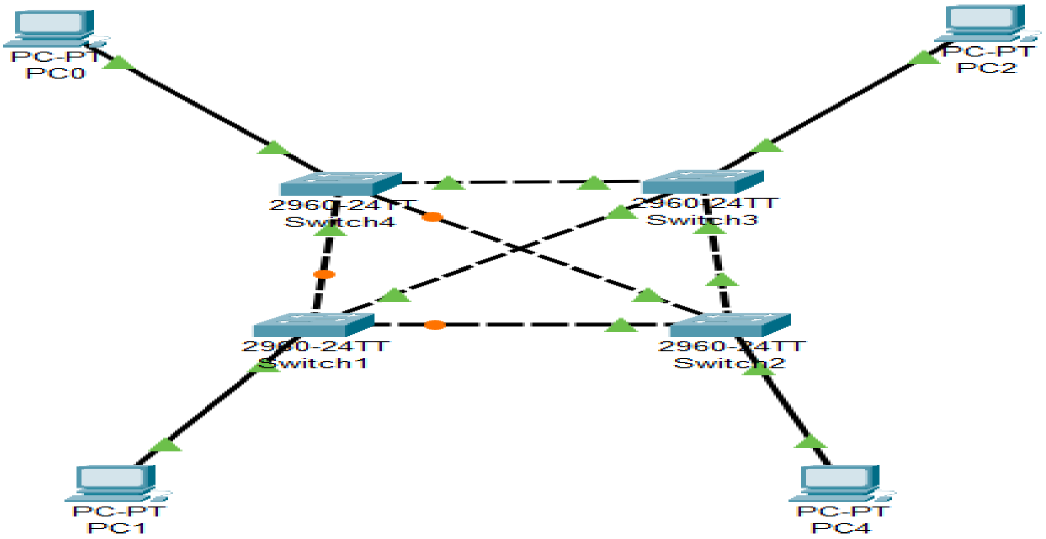


Fig 2.4: Network Map for Mesh Topology

Implementation Sequence

Here is the implementation sequence for creating the Mesh Topology shown in the image using Cisco Packet Tracer:

Step 1: Launch Cisco Packet Tracer

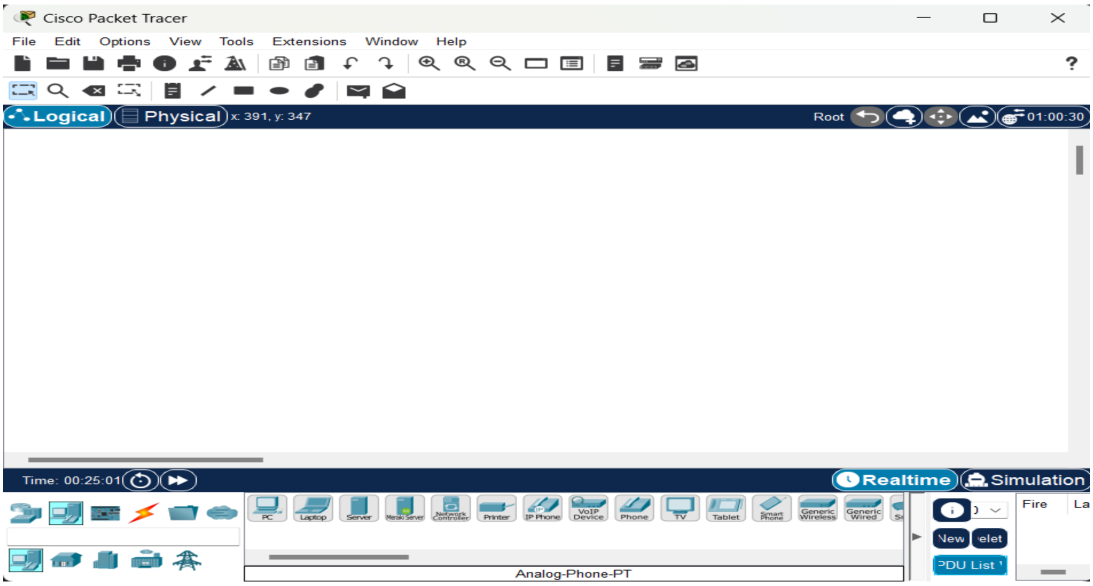


Fig 2.5: Workspace for network design

Step 2: Add the network devices to the workspace

2.1 From the Device-Type Selection box, choose the following devices and add them to the workspace:

2.2 Four 2960-24TT Switch

2.3 Four PCs (labeled PC0, PC1, PC2, and PC4)

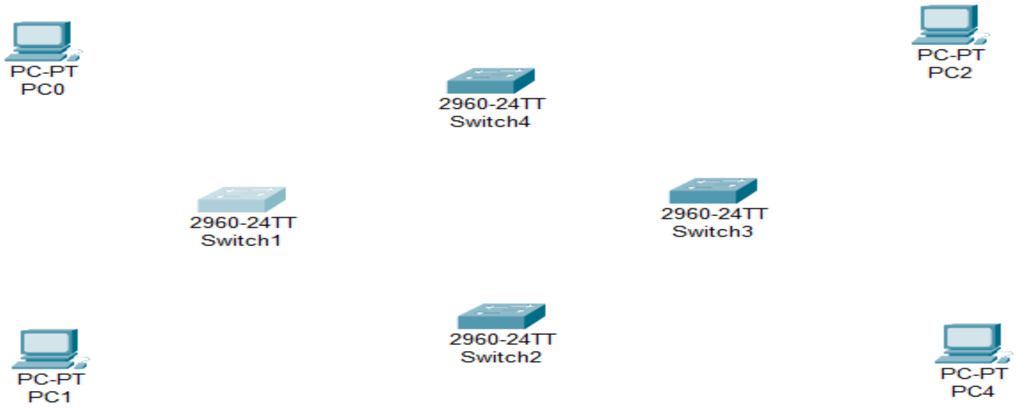


Fig 2.6: Switches and PCs for Mesh Topology

Step 3: Connect the devices

- Use the copper straight-through cable to connect each PC to one of the available ports on each switch and copper cross-over cable to connect between each adjacent and diagonal switch.
- Ensure that each connection is made properly.

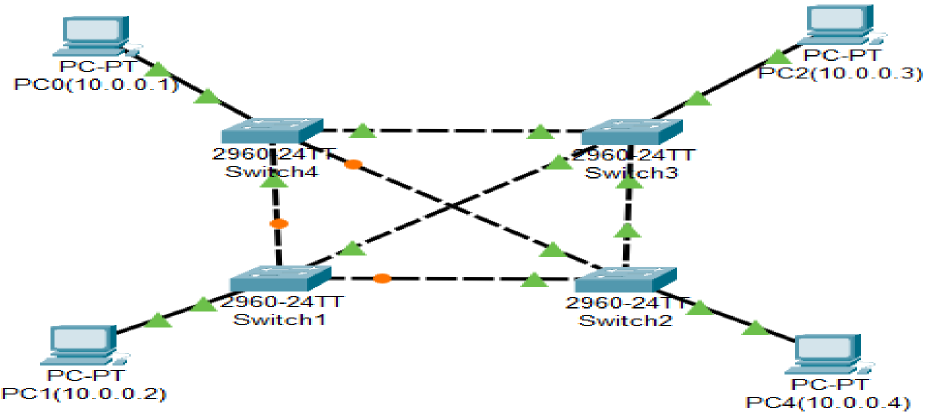
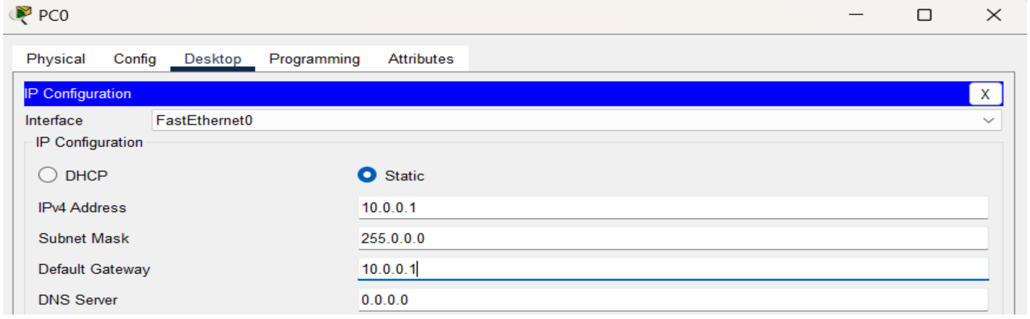


Fig 2.7: Connection between Switch and PC

Step 4: Configure IP addresses

- Right-click on each PC and select "IP Configuration."
- In the IP Configuration window, enter the IP address as (10.0.0.1 to 10.0.0.4), subnet mask, and default gateway for each PC.



IP Configuration

Interface FastEthernet0

IP Configuration

○ DHCP ● Static

IPv4 Address	10.0.0.1
Subnet Mask	255.0.0.0
Default Gateway	10.0.0.1
DNS Server	0.0.0.0

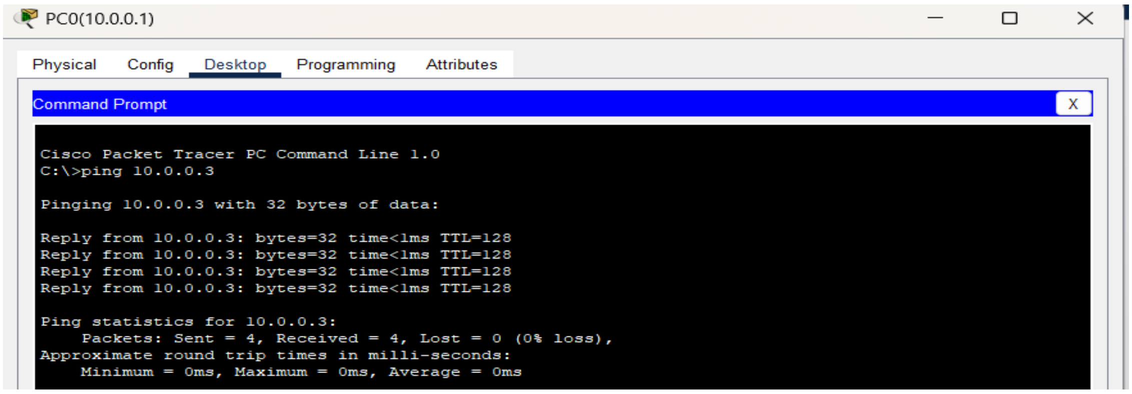
Fig 2.8: IP Configuration

Step 5: Verify connectivity:

5.1 To test whether the network is working, you can ping other devices on the network from each PC.

5.2 To ping another device, open a command prompt on the PC and type "ping <IP address of the other device>."

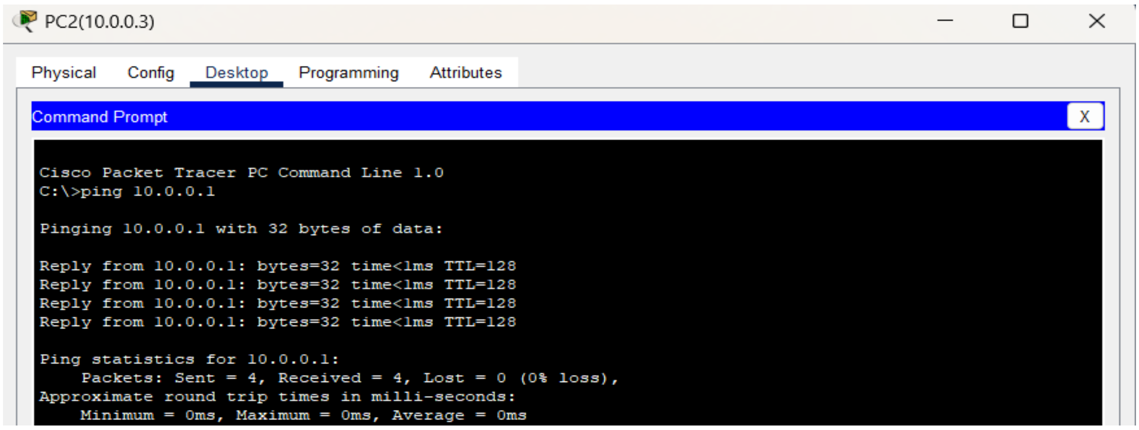
5.3 If the ping is successful, you should see replies from the other device.



```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 10.0.0.3

Pinging 10.0.0.3 with 32 bytes of data:

Reply from 10.0.0.3: bytes=32 time<1ms TTL=128
Reply from 10.0.0.3: bytes=32 time<1ms TTL=128
Reply from 10.0.0.3: bytes=32 time<1ms TTL=128
Reply from 10.0.0.3: bytes=32 time<1ms TTL=128

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

Fig 2.9: Connectivity test between PC0 and PC2

In the above figure, we test the connectivity of network by pinging PC with IP Address 10.0.0.3 from PC with IP Address 10.0.0.2. The ping was successful as we received replies from the destination PC with 0% Loss.



```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 10.0.0.1

Pinging 10.0.0.1 with 32 bytes of data:

Reply from 10.0.0.1: bytes=32 time<1ms TTL=128
Reply from 10.0.0.1: bytes=32 time<1ms TTL=128
Reply from 10.0.0.1: bytes=32 time<1ms TTL=128
Reply from 10.0.0.1: bytes=32 time<1ms TTL=128

Ping statistics for 10.0.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Fig 3.0: Connectivity test between PC2 and PC0

In the above figure, we test the connectivity of network by pinging PC with IP Address 10.0.0.2 from PC with IP Address 10.0.0.3. The ping was successful as we received replies from the destination PC with 0% Loss.

Conclusion

In this lab, we designed various network topologies using **Cisco Packet Tracer**, providing practical experience in understanding how devices within a network are interconnected and communicate. This simulation allowed us to experiment with the design and configuration of different topologies, including the strategic placement and connection of key components like switches and end devices.

Lab 7: Creating VLAN and VLAN Trunking using Packet Tracer

Theory

A VLAN (Virtual Local Area Network) is a network configuration that partitions a single physical network into multiple logical networks. Each VLAN functions as an independent network, despite sharing the same physical infrastructure with other VLANs. By isolating network segments, VLANs improve security, reduce unnecessary broadcast traffic, and allow for logical traffic segmentation based on factors like department or function within an organization.

VLAN Trunking

VLAN trunking is a method used to allow traffic from multiple VLANs to traverse a single network link between switches or other network devices. This is achieved by tagging Ethernet frames with a VLAN identifier, commonly through IEEE 802.1Q tagging. Trunking enables the extension of VLANs across network devices, supporting greater flexibility in network design and allowing VLANs to span across different physical locations.

VLAN Architecture

VLAN architecture is designed to logically group devices across different network segments, creating multiple broadcast domains on a single network infrastructure. Each VLAN typically corresponds to a different logical network, isolating traffic between VLANs unless explicitly allowed through routing or firewall rules. The architecture includes components like access ports (where devices are connected to the VLAN), trunk ports (which carry traffic for multiple VLANs), and VLAN-aware network devices that manage traffic across various segments. This modular design enhances scalability, security, and performance in modern networks.

Component Used

Hardware: Switches (2), Ethernet cables, End devices (4).
Software: Cisco Packet Tracer

Network Diagram

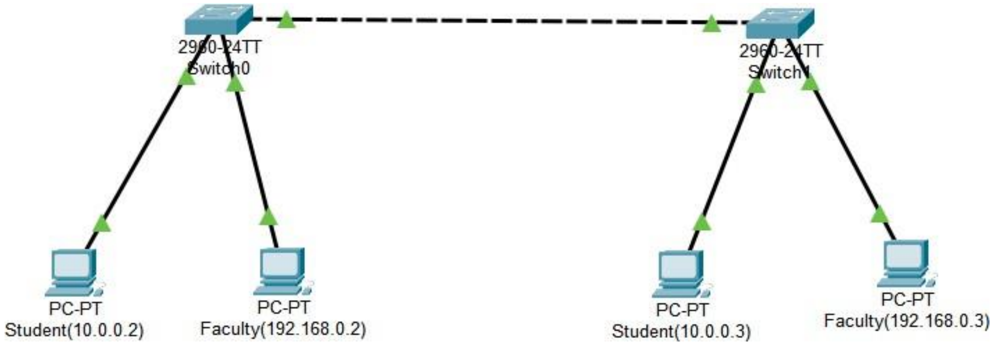


Fig 1.0: Network map for VLAN

Procedure

Here is the procedure for creating the LAN network shown in the image using Cisco Packet Tracer:

Step1: Launch Cisco Packet Tracer

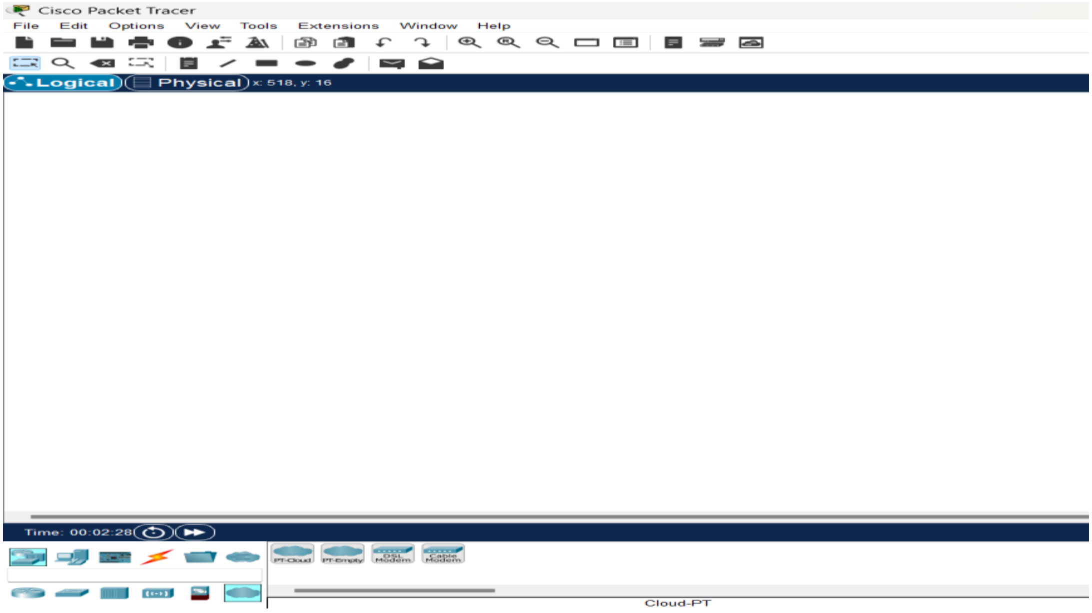


Fig 1.1: Workspace for network design

Step2: Add the network devices to the workspace and connecting devices:

2.1 From the Device-Type Selection box, choose the following devices and add them to the workspace:

2.2 One 2960-24TT Switch and four PC's

2.3 Use the copper straight-through cable to connect each PC to one of the available ports on the switch.

2.4 Ensure that each connection is made properly.

2.5 Also rename the PC's as Dilasha0(10.0.0.2), Faculty (192.168.0.2), Dilasha1(10.0.0.3), and Faculty (192.168.0.3).

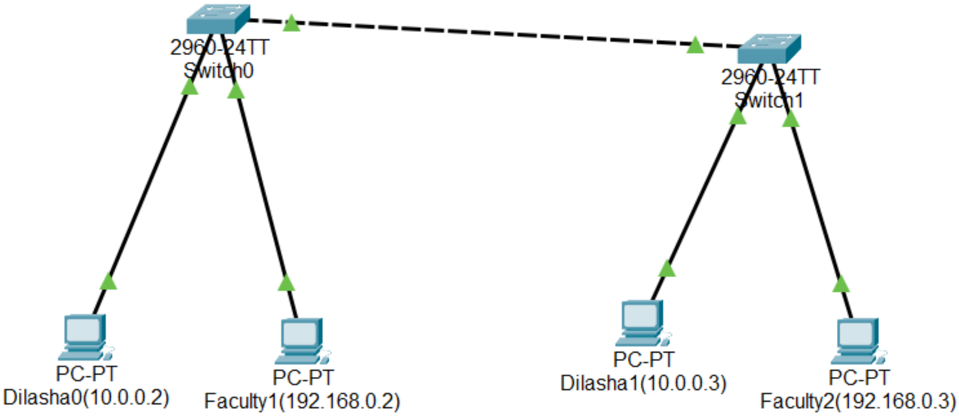
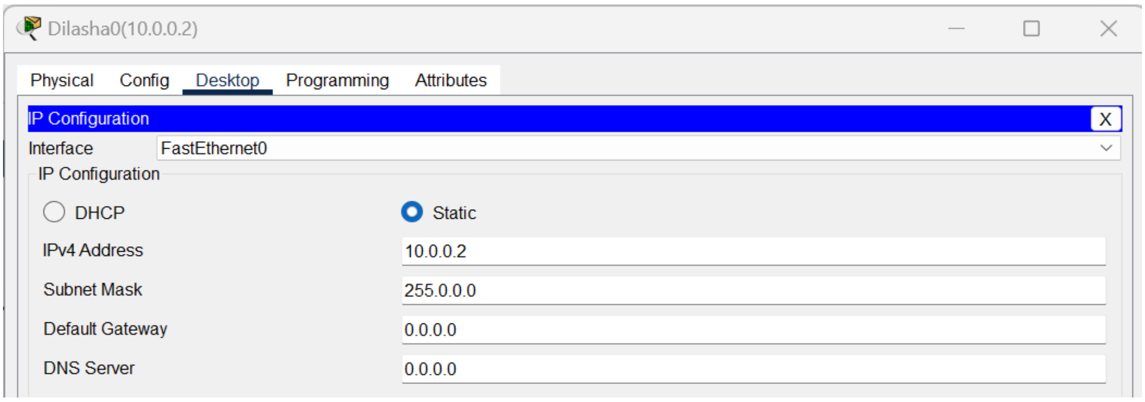


Fig 1.2: Connection between all devices in VLAN

Step3: Configure IP addresses:

3.1 Right-click on each PC and select "IP Configuration."

3.2 In the IP Configuration window, assign IP addresses as follows: Dilasha devices- 10.0.0.2 and 10.0.0.3; Faculty devices - 192.168.0.2 and 192.168.0.3. Dilasha devices will connect via port Fa 0/1, while Faculty devices will connect via port Fa 0/2.



Interface: FastEthernet0

IP Configuration

○ DHCP ● Static

IPv4 Address: 10.0.0.2

Subnet Mask: 255.0.0.0

Default Gateway: 0.0.0.0

DNS Server: 0.0.0.0

Fig 1.3: IP Configuration

Step 4: Configuring VLANs:

4.1Create VLAN on Both Switches &Assign Port to Both Switches:

To create VLANs on both switches, enter the configuration mode on each switch and use the vlan command to create separate VLANs for Student and Faculty devices. After creating the VLANs, assign ports to the VLANs by selecting the specific interfaces (e.g., Fa 0/1 for Dilasha and Fa 0/2 for Faculty) and using the switch port access vlan <VLAN_ID> command to associate the ports with the correct VLAN.

4.2Create Trunking on Both Switches:

To enable trunking, configure the interfaces between the two switches using the switchport mode trunk command. This allows multiple VLANs to pass through the same link, making communication between devices in the same VLAN but connected to different switches. Trunking ensures that tagged traffic is carried across the switches while maintaining the VLAN distinctions.

Code for VLAN configurations:

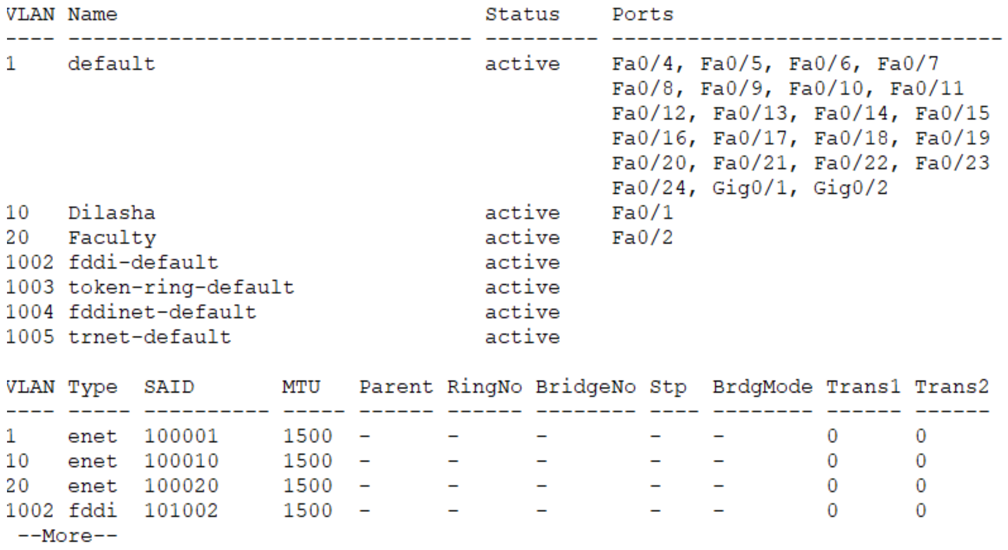
```
Switch(config)#vlan 10
Switch(config-vlan)#name
Dilasha

Switch(config-vlan)#vlan 20
Switch(config-vlan)#namefaculty
Switch(config-vlan)#exit
Switch(config)#exit
```

Code for Assigning ports:

```
Switch#config t
Switch(config)#int fa0/1
Switch(config-if)#switchport mode access
Switch(config-if)#switchport access vlan10
Switch(config-if)#int fa 0/2
Switch(config-if)#switchport mode access
Switch(config-if)#switchport access  vlan20
Switch(config-if)#exit
Switch(config)#exit
```

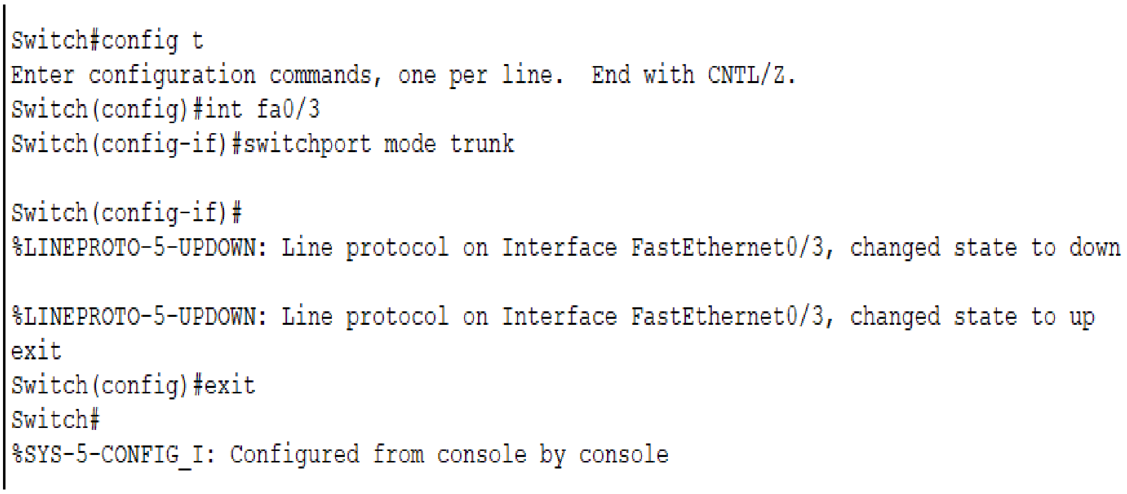
Code for Trunking Switches:

```
Switch#config t
Switch(config)#intfa0/3
Switch(config-if)#switchport mode trunk
Switch(config-if)#exit
Switch(config)#exit
```



```
VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Fa0/4, Fa0/5, Fa0/6, Fa0/7
                                                Fa0/8, Fa0/9, Fa0/10, Fa0/11
                                                Fa0/12, Fa0/13, Fa0/14, Fa0/15
                                                Fa0/16, Fa0/17, Fa0/18, Fa0/19
                                                Fa0/20, Fa0/21, Fa0/22, Fa0/23
                                                Fa0/24, Gig0/1, Gig0/2
10   Dilasha                          active    Fa0/1
20   Faculty                          active    Fa0/2
1002 fddi-default                     active
1003 token-ring-default               active
1004 fddinet-default                  active
1005 trnet-default                    active

VLAN Type  SAID       MTU   Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ -------- ---- -------- ------ ------
1    enet  100001     1500  -      -      -        -    -        0      0
10   enet  100010     1500  -      -      -        -    -        0      0
20   enet  100020     1500  -      -      -        -    -        0      0
1002 fddi  101002     1500  -      -      -        -    -        0      0
 --More--
```

Fig 1.4: Assigning ports to VLAN



```
Switch#config t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#int fa0/3
Switch(config-if)#switchport mode trunk

Switch(config-if)#
%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
exit
Switch(config)#exit
Switch#
%SYS-5-CONFIG_I: Configured from console by console
```

Fig 1.5: Configuring trunking between switches

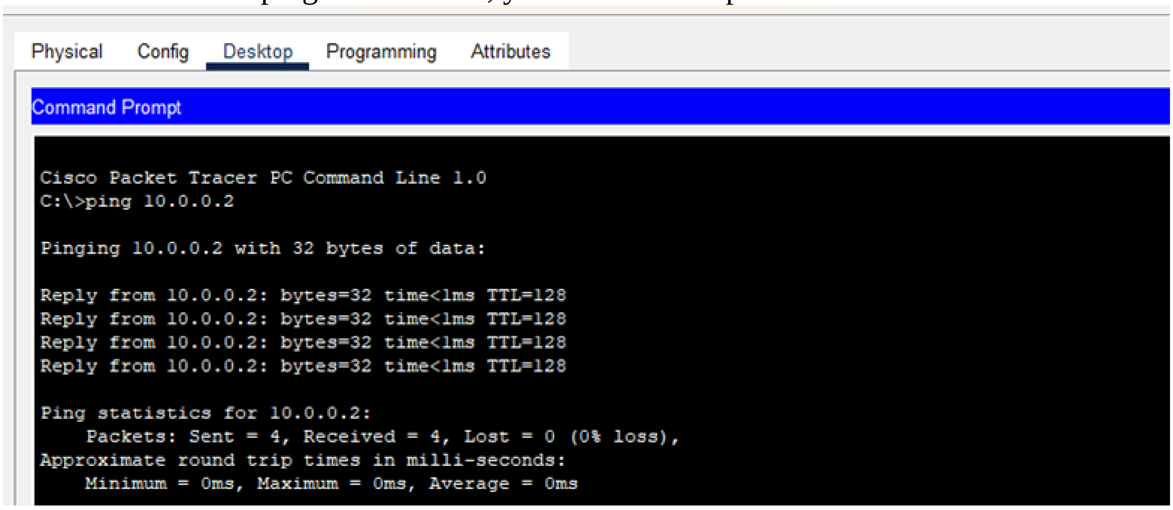
Step5: Verify Connectivity

5.1 To test whether the network is working, you can ping other devices on the network from each PC.

5.2 Now ping Dilasha (10.0.0.2) from Dilasha (10.0.0.3) and vice-versa.

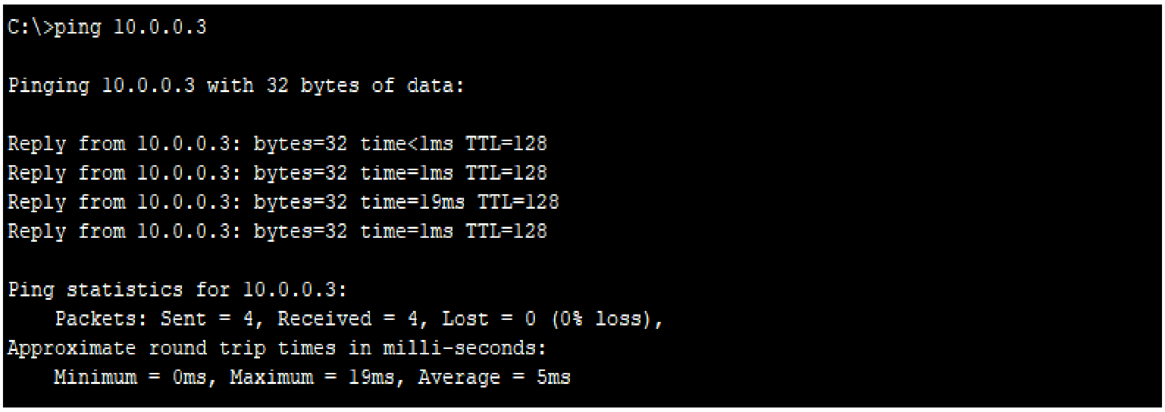
5.3 Also ping Dilasha (10.0.0.2) from faculty (198.68.0.2) to check there is no connection between student and faculty.

5.3 If the ping is successful, you should see replies from the other device.



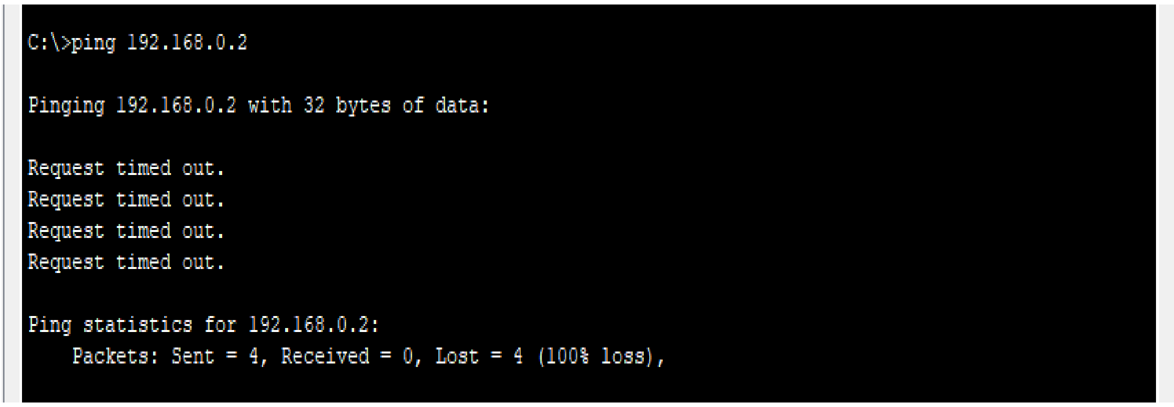
```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 10.0.0.2

Pinging 10.0.0.2 with 32 bytes of data:

Reply from 10.0.0.2: bytes=32 time<1ms TTL=128
Reply from 10.0.0.2: bytes=32 time<1ms TTL=128
Reply from 10.0.0.2: bytes=32 time<1ms TTL=128
Reply from 10.0.0.2: bytes=32 time<1ms TTL=128

Ping statistics for 10.0.0.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Fig 1.5: Connectivity test from Dilasha (10.0.0.3) to Dilasha (10.0.0.2)



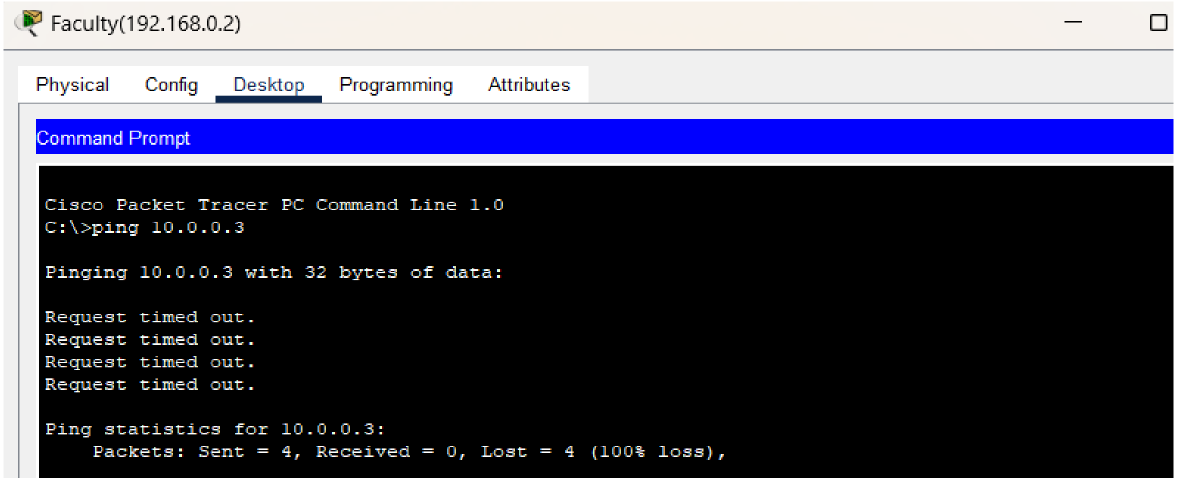
```
C:\>ping 10.0.0.3

Pinging 10.0.0.3 with 32 bytes of data:

Reply from 10.0.0.3: bytes=32 time<1ms TTL=128
Reply from 10.0.0.3: bytes=32 time=1ms TTL=128
Reply from 10.0.0.3: bytes=32 time=19ms TTL=128
Reply from 10.0.0.3: bytes=32 time=1ms TTL=128

Ping statistics for 10.0.0.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 19ms, Average = 5ms
```

Fig 1.6: Connectivity test from Dilasha (10.0.0.2) to Dilasha (10.0.0.3)



```
C:\>ping 192.168.0.2

Pinging 192.168.0.2 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.0.2:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

Fig 1.7: Connectivity test from Dilasha (10.0.0.2) to faculty (192.168.0.2)



```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 10.0.0.3

Pinging 10.0.0.3 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 10.0.0.3:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

Fig 1.8: Connectivity test from faculty (192.168.0.2) to Dilasha (10.0.0.3)

Conclusion

In this lab, we configure VLANs and implement VLAN trunking in Cisco Packet Tracer to significantly enhance network segmentation and management. By organizing devices into distinct VLANs, we reduce broadcast domains, improve security, and optimize network performance. VLAN trunking allows the transfer of multiple VLANs over a single link, facilitating seamless communication between VLANs across different switches. This demonstrates the importance of organized network design, which reduces broadcast traffic, simplifies management, and improves scalability and efficiency in modern networks.

Lab 8: Basic router configuration and static routing in Packet Tracer

Theory

A router is a device used in networking to forward data packets between different computer networks, helping to direct traffic across the Internet. Information sent over the Internet, like a web page or an email, is broken down into data packets. The router, which is connected to at least two networks, determines the best path for each packet based on its knowledge of the connected networks.

Network Diagram

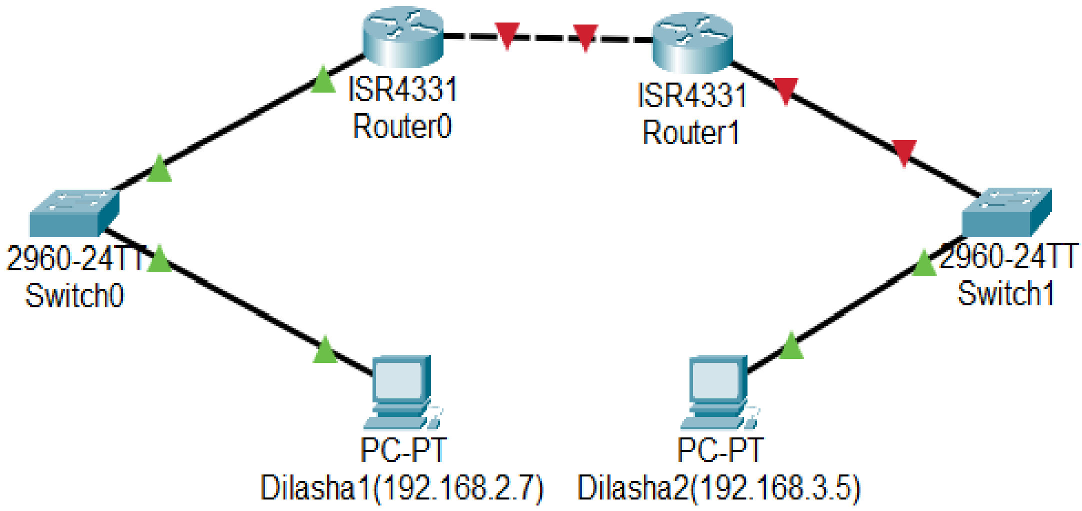


Fig: Network diagram

Basic Router Configuration

Configuring Global Parameters

The initial configuration of the router involves setting global parameters such as hostname, passwords, and interface descriptions.

Steps:

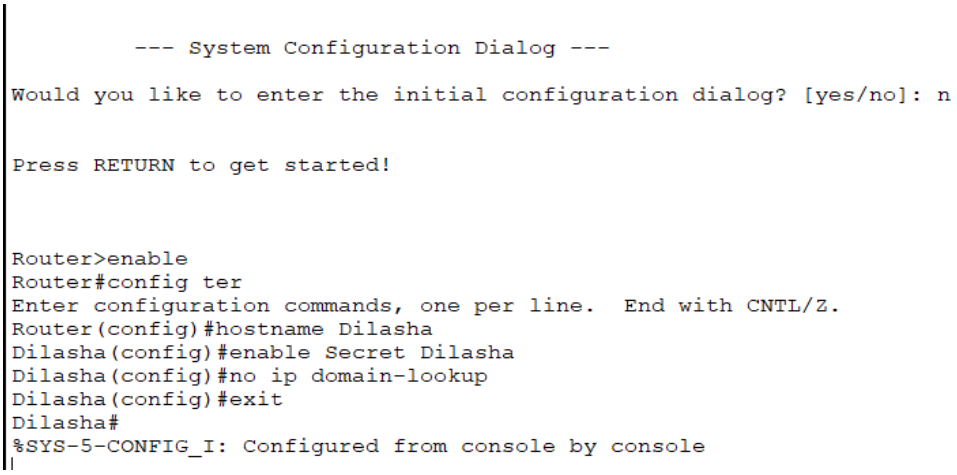
1.Enter privileged EXEC mode.

2.Configure the hostname using the hostname command.

3.Set passwords for privileged EXEC mode and console access.

4.Configure banners if necessary.

5.Repeat same for another router as well.



```
         --- System Configuration Dialog ---

Would you like to enter the initial configuration dialog? [yes/no]: n


Press RETURN to get started!



Router>enable
Router#config ter
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#hostname Dilasha
Dilasha(config)#enable Secret Dilasha
Dilasha(config)#no ip domain-lookup
Dilasha(config)#exit
Dilasha#
%SYS-5-CONFIG_I: Configured from console by console
```

Fig: Router configuration

Configuring Gigabit Ethernet

Once global parameters are set, configure the Gigabit Ethernet interfaces of the router to enable communication between different networks.

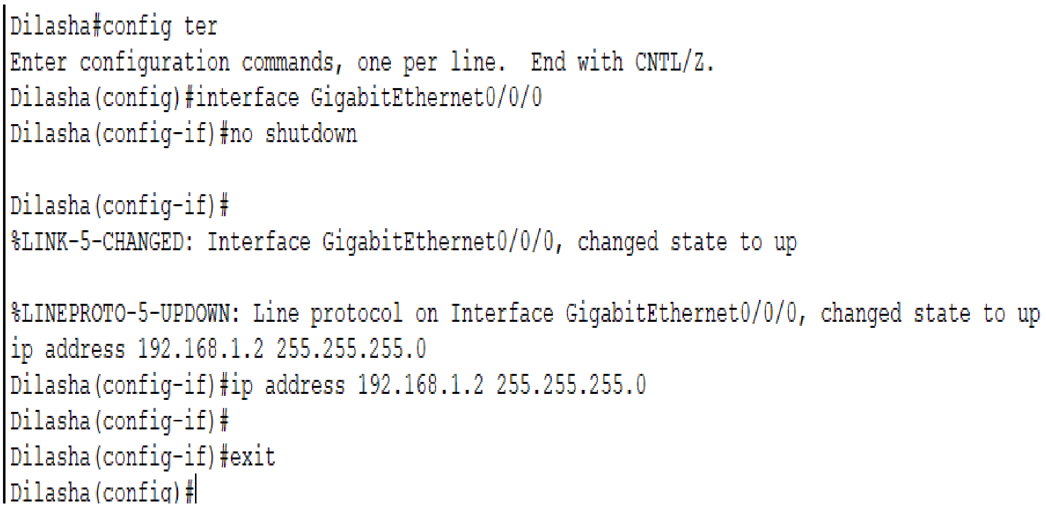
Steps:

1.Access the interface using the interface gig0/0 command.

2.Set the IP address and subnet mask for the interface.

3.Enable the interface using the no shutdown command.

4.Repeat same for another router as well.



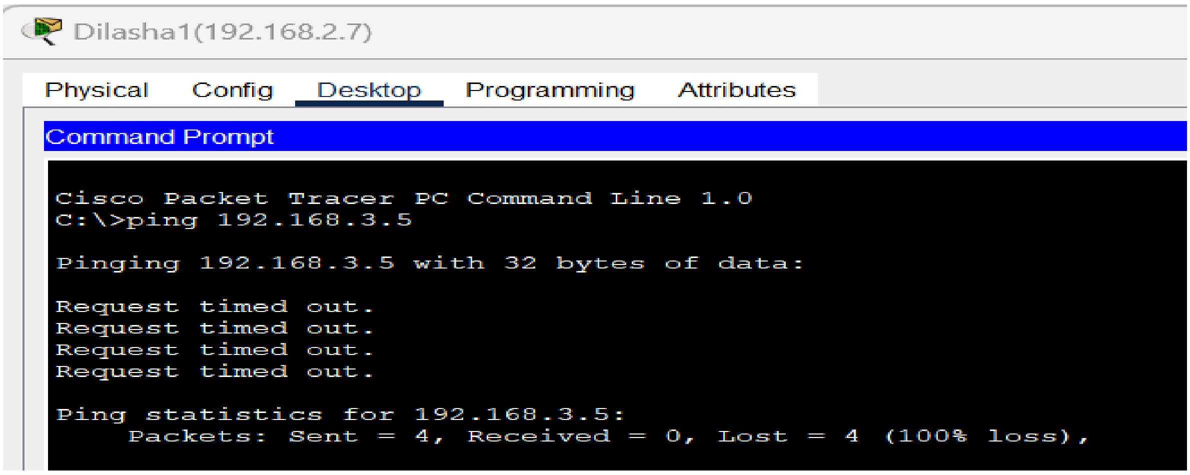
```
Dilasha#config ter
Enter configuration commands, one per line.  End with CNTL/Z.
Dilasha(config)#interface GigabitEthernet0/0/0
Dilasha(config-if)#no shutdown

Dilasha(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0/0, changed state to up
ip address 192.168.1.2 255.255.255.0
Dilasha(config-if)#ip address 192.168.1.2 255.255.255.0
Dilasha(config-if)#
Dilasha(config-if)#exit
Dilasha(config)#
```

Fig: Gigabit Ethernet configuration

Connection Testing Before Static Routing Configuration

Steps:

1.Pinging PC (Dilasha 2(192.168.3.5)) from PC (Dilasha 1(192.168.2.7)) to verify connection if exists.



```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.3.5

Pinging 192.168.3.5 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.3.5:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

Fig: Connectivity test from PC (Dilasha 1(192.168.2.7)) to PC (Dilasha2(192.168.3.5))

Here we can see there is not any connection in the network. So to establish connection in the network, we need to statically configure the router through CLI.

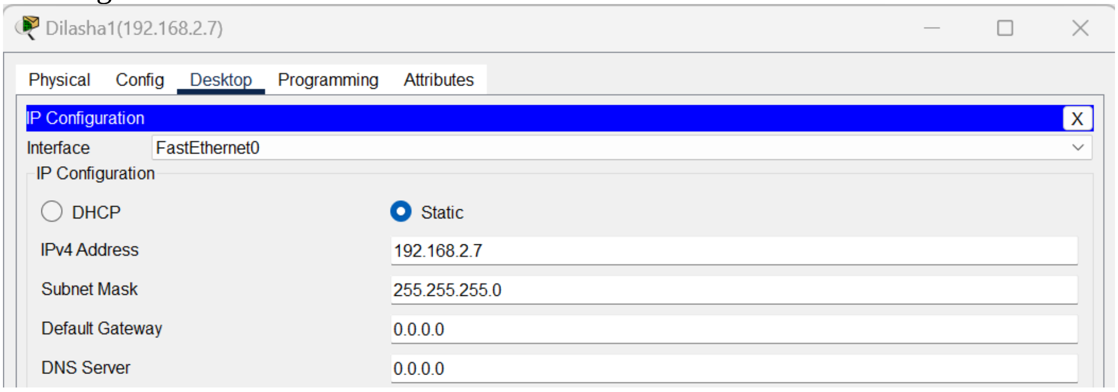
Static Routing Configuration

Configuring Network (PCs and Routers)

Set up static routes to allow the routers to communicate with networks beyond their directly connected networks.

Steps:

1.Configure the IP addresses of PCs connected to each network.



Interface: FastEthernet0

IP Configuration

○ DHCP ● Static

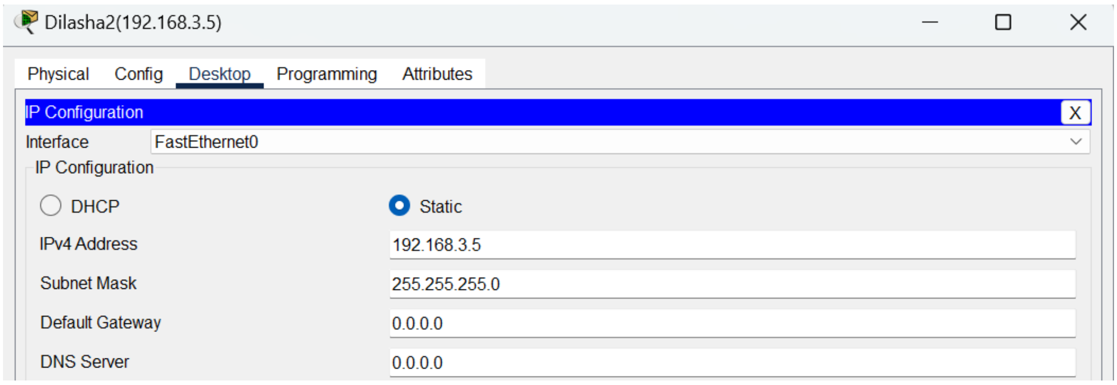
IPv4 Address: 192.168.2.7

Subnet Mask: 255.255.255.0

Default Gateway: 0.0.0.0

DNS Server: 0.0.0.0

Fig: IP configuration PC (Dilasha 1(192.168.2.7))



Interface: FastEthernet0

IP Configuration

○ DHCP ● Static

IPv4 Address: 192.168.3.5

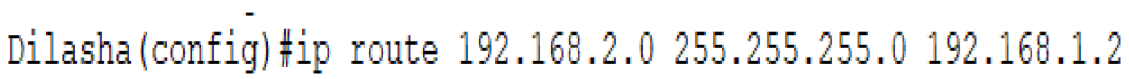
Subnet Mask: 255.255.255.0

Default Gateway: 0.0.0.0

DNS Server: 0.0.0.0

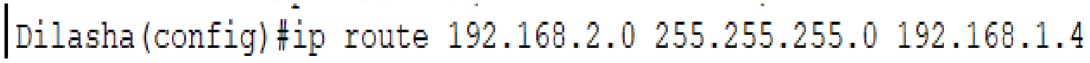
Fig: IP configuration on PC (Dilasha 2(192.168.3.5))

2. On each router, configure static routes using the ip route command to manually specify the next hop for network traffic.



```
Dilasha(config)#ip route 192.168.2.0 255.255.255.0 192.168.1.2
```

Fig: Configuring IP route on router (Dilasha 1)



```
Dilasha(config)#ip route 192.168.2.0 255.255.255.0 192.168.1.4
```

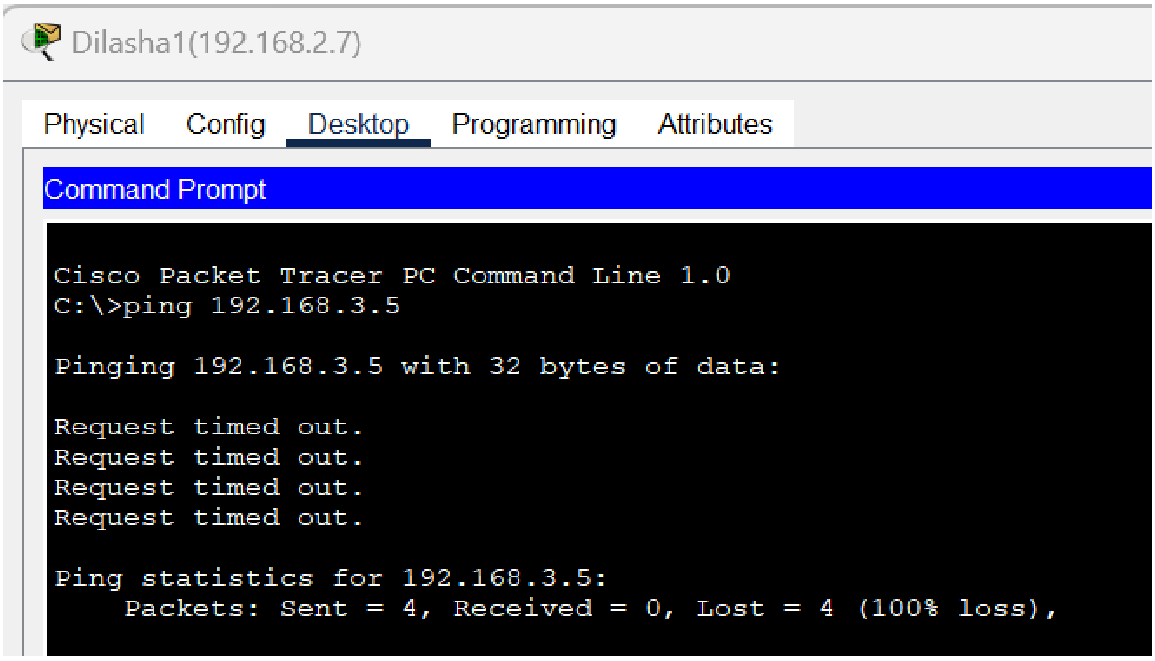
Fig: Configuring IP route on router (Dilasha 2)

Testing and Validation

To test whether the network is working, you can ping other devices on the network from each PC.

Steps:

1. Ping PC (Dilasha 2(192.168.3.5)) from PC (Dilasha 1(192.168.2.7).
2. If the ping is successful, you should see replies from the other device.



```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.3.5

Pinging 192.168.3.5 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.3.5:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

Fig: Connectivity test from PC (Dilasha 1(192.168.2.7)) to PC (Dilasha 2(192.168.3.5))

Conclusion

In this lab, we configured basic router settings and applied static routing in Cisco Packet Tracer. This hands-on exercise helped us practice setting up global parameters, configuring Ethernet interfaces, and creating static routes between routers. By manually setting these routes, we directed network traffic, ensuring that different network segments could communicate smoothly and efficiently. This reinforced the importance of static routing in managing traffic flow across a network.

Lab 9: Implementation of Dynamic interior/ Exterior Routing (RIP, OSPF, BGP)

Theory

Dynamic Interior/Exterior Routing

Interior Routing: Dynamic Interior Routing refers to routing protocols used within a single autonomous system (AS). An autonomous system is a collection of IP networks and routers under the control of a single organization. Dynamic Interior Gateway Protocols (IGPs) automatically update the routing table in response to changes in network topology, making it easier to maintain large networks. Interior routing protocols help ensure that data packets find the most efficient path within the AS.

The most common Interior Gateway Protocols (IGPs) include:
1.RIP (Routing Information Protocol)
2.OSPF (Open Shortest Path First)
3.EIGRP (Enhanced Interior Gateway Routing Protocol)

Exterior Routing: Dynamic Exterior Routing is used for routing between different autonomous systems, typically over the internet. Exterior Gateway Protocols (EGPs) are designed to exchange routing information between different organizations, ISPs, or large networks. The most common protocol in this category is BGP (Border Gateway Protocol). BGP ensures that data packets can travel across the internet by finding the best path between ASes, taking into consideration policies, path attributes, and network stability. Unlike interior protocols, BGP focuses on scalability, security, and policy-based routing to manage traffic between different autonomous systems.

RIP, OSPF, BGP

RIP (Routing Information Protocol): RIP is a distance-vector routing protocol that uses hop count as the metric to determine the best route. It has a maximum hop limit of 15, making it suitable for small networks. While easy to configure, it has slow convergence and is not ideal for large or complex networks due to its simplicity and limited scalability.

OSPF (Open Shortest Path First): OSPF is a link-state routing protocol that calculates the shortest path using the Dijkstra algorithm. It’s highly scalable and supports large networks by dividing them into areas. OSPF converges quickly and allows for advanced features like route summarization and variable-length subnet masks, but it is more complex to configure than RIP.

BGP (Border Gateway Protocol): BGP is a path-vector protocol used for routing between different autonomous systems, primarily on the internet. It prioritizes policy-based routing, making it essential for controlling traffic between ISPs and large networks. BGP is highly scalable but requires careful configuration due to its complexity and slower convergence compared to interior protocols like OSPF.

Network Diagram

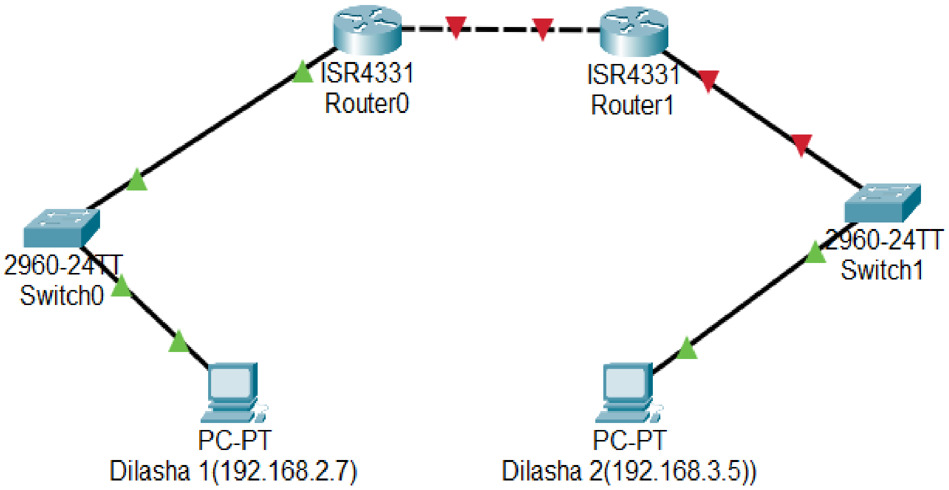


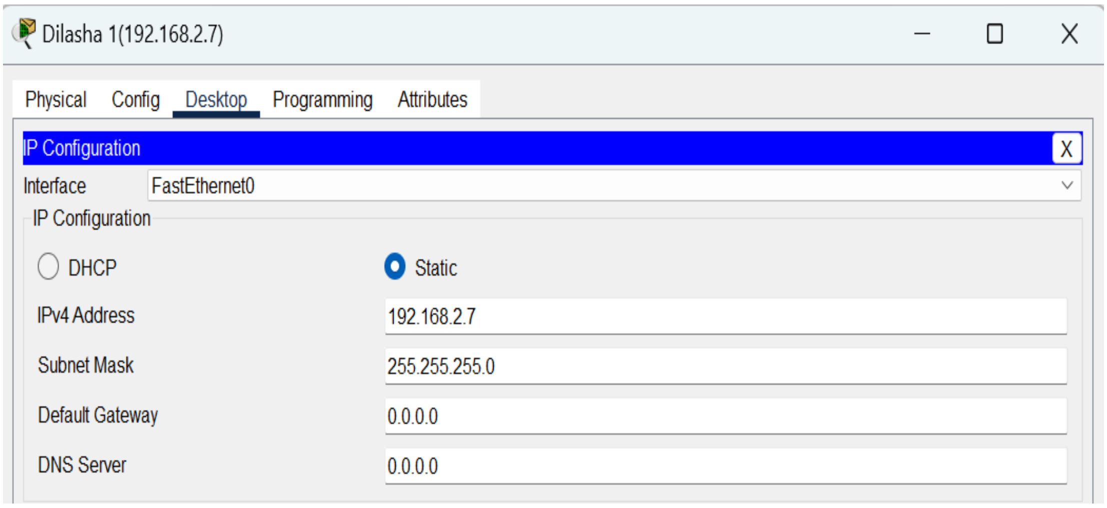
Fig: Network Diagram

Configuring Network

Configure network for PCs and Routers

Steps:

1.Configure the IP addresses of PCs connected to each network.



Interface: FastEthernet0

IP Configuration

○ DHCP ● Static

IPv4 Address: 192.168.2.7

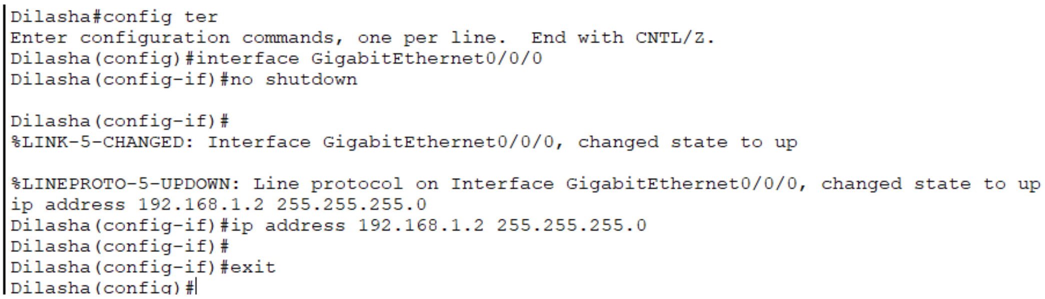
Subnet Mask: 255.255.255.0

Default Gateway: 0.0.0.0

DNS Server: 0.0.0.0

Fig: IP configuration

2.Configure the Gigabit Ethernet interfaces of the router to enable communication between different networks.



```
Dilasha#config ter
Enter configuration commands, one per line.  End with CNTL/Z.
Dilasha(config)#interface GigabitEthernet0/0/0
Dilasha(config-if)#no shutdown

Dilasha(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0/0, changed state to up
ip address 192.168.1.2 255.255.255.0
Dilasha(config-if)#ip address 192.168.1.2 255.255.255.0
Dilasha(config-if)#
Dilasha(config-if)#exit
Dilasha(config)#
```

Fig: Gigabit Ethernet configuration

3.Repeat same for another PC’s and router as well.

Implementation & Need for Dynamic Routing

Implementation

Dynamic routing protocols like RIP, OSPF, and BGP are implemented to allow routers to automatically adjust and exchange routing information in response to network changes. We configure routers to use these protocols to dynamically update their routing table4

Network Configuration: Set up the routers and PCs, assigning IP addresses to each device.

Dynamic Routing Setup: Activate RIP, OSPF, or BGP on each router, depending on the network's size and complexity, to automatically share routing information.

Testing: Validate the configurations using tools like ping to ensure network connectivity across the routers.

Need for Dynamic Routing

Dynamic routing is essential in modern networks due to its ability to automatically adjust to changes in the network topology without requiring manual intervention. The key reasons for needing dynamic routing are:
Automatic Route Updates: When a network changes, such as when a link fails or new devices are added, dynamic routing protocols automatically update routing tables across routers. This ensures continuous connectivity without manual reconfiguration.4

Scalability: In large or frequently changing networks, manually configuring static routes becomes unmanageable. Dynamic routing protocols efficiently handle the routing of traffic as the network grows, making it scalable.

Efficient Path Selection: Dynamic protocols continuously monitor network conditions and select the best possible path for data transmission. This helps optimize network performance and reduces delays.

Redundancy and Fault Tolerance: Dynamic routing enhances network reliability by quickly adapting to network failures, rerouting traffic through alternate paths, which minimizes downtime.

Overall, dynamic routing is necessary for reducing administrative overhead and ensuring the network remains efficient and responsive to changes.

Dynamic Routing Configuration

Using RIP Command

Network Diagram

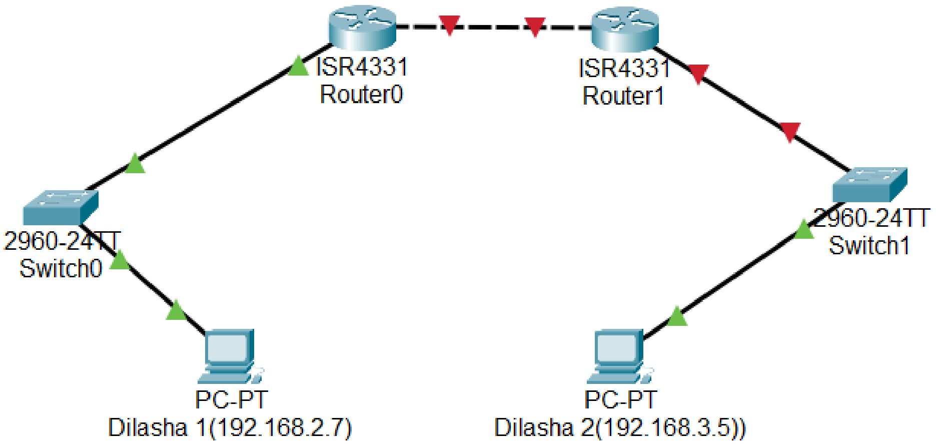
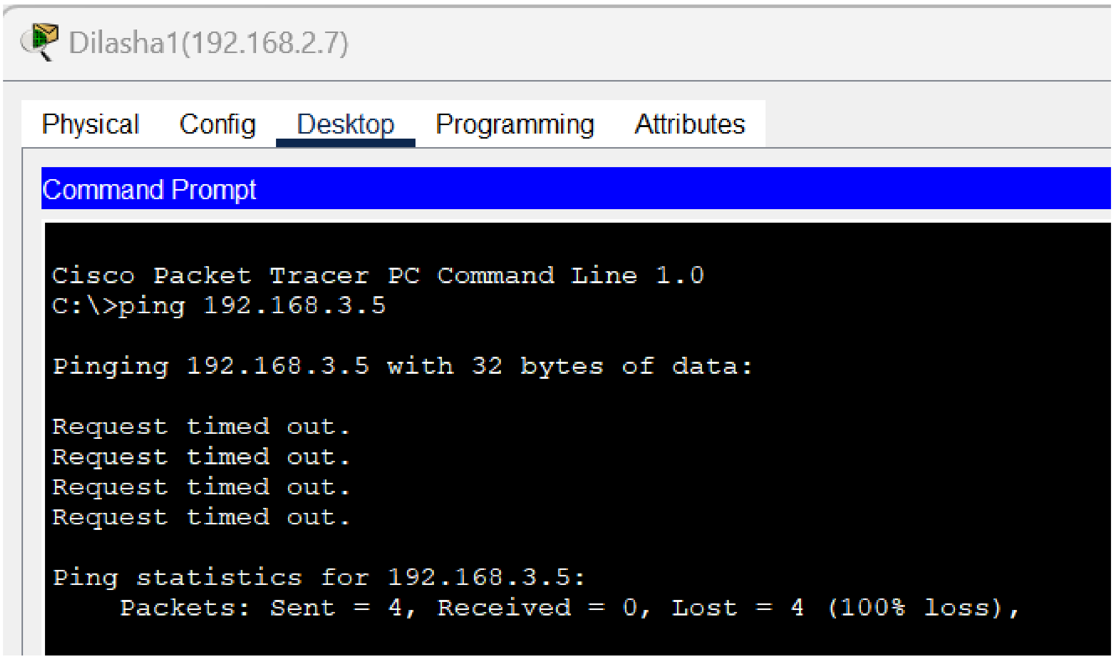


Fig: Network Diagram

Connection Testing Before Dynamic Routing Configuration using RIP command
Steps:
1.Pinging PC(Dilasha2(192.168.3.5)) from PC(Dilasha1(192.168.2.7)) to verify connection if exists.



```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.3.5

Pinging 192.168.3.5 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.3.5:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

Fig: Connectivity test from PC(Dilasha1(192.168.2.7)) to PC(Dilasha2(192.168.3.5))

Here we can see there is not any connection in the network. So, to establish connection in the network, we need to dynamically configure the router through CLI using RIP command.

Code for Dynamic Routing Configuration Using RIP Command

For Router 0:

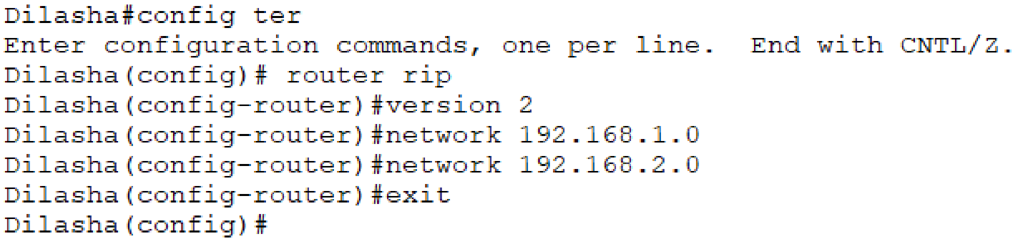
Router0> enable
Router0# configure terminal
Router0(config)# router rip
Router0(config-router)# version 2
Router0(config-router)# network 192.168.1.0
Router0(config-router)# network 192.168.2.0
Router0(config-router)# exit

For Router 1:

Router1> enable
Router1# configure terminal
Router1(config)# router rip
Router1(config-router)# version 2
Router1(config-router)# network 192.168.1.0
Router1(config-router)# network 192.168.3.0
Router1(config-router)# exit

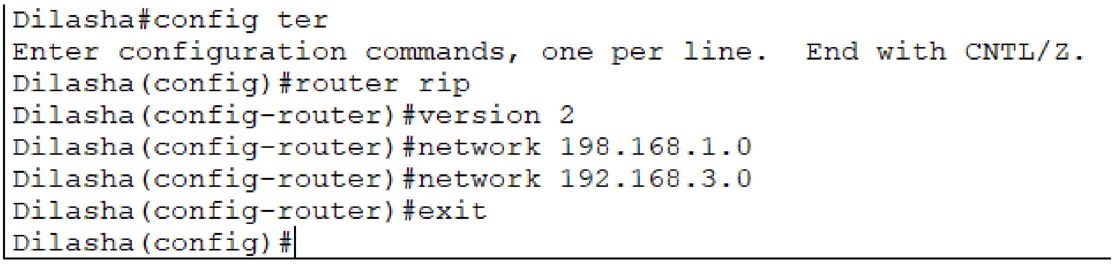
Steps for Dynamic Routing Configuration Using RIP Command

1.Access Router.
2.Enable RIP on Router.
3.Specify RIP version 2 (for more efficiency and subnet support).
4.Advertise the networks connected to Router1 (LAN and WAN).
5.Exit RIP configuration.
6.Repeat same steps for another Router.



```
Dilasha#config ter
Enter configuration commands, one per line.  End with CNTL/Z.
Dilasha(config)# router rip
Dilasha(config-router)#version 2
Dilasha(config-router)#network 192.168.1.0
Dilasha(config-router)#network 192.168.2.0
Dilasha(config-router)#exit
Dilasha(config)#
```

Fig: Router configuration on router (Dilasha) using RIP command.



```
Dilasha#config ter
Enter configuration commands, one per line.  End with CNTL/Z.
Dilasha(config)#router rip
Dilasha(config-router)#version 2
Dilasha(config-router)#network 198.168.1.0
Dilasha(config-router)#network 192.168.3.0
Dilasha(config-router)#exit
Dilasha(config)#
```

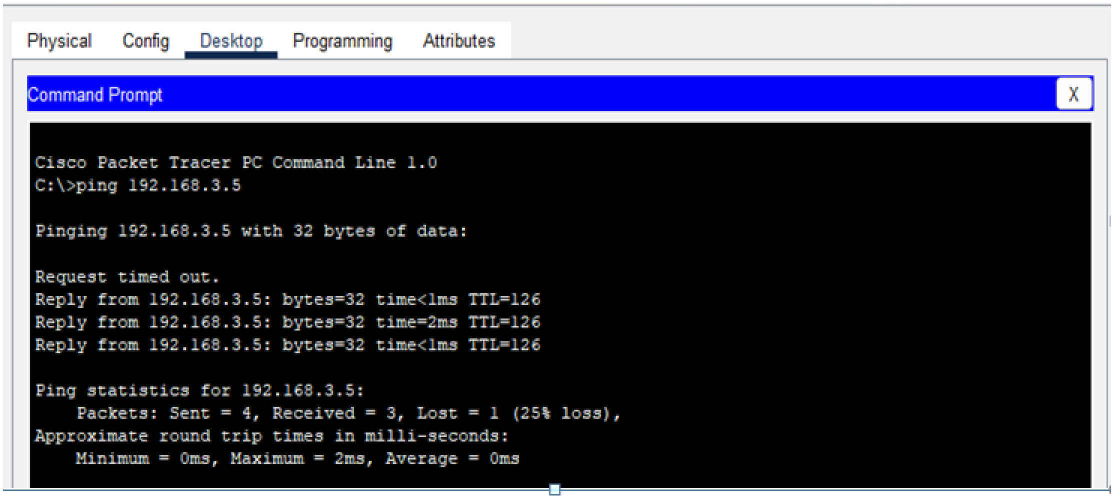
Fig: Router configuration router (Dilasha1) using RIP command.

Testing and Validation

To test whether the network is working, you can ping other devices on the network from each PC.

Steps:

1. Ping PC(Dilasha2(192.168.3.5)) from PC(Dilasha1(192.168.2.7)).
2. If the ping is successful, you should see replies from the other device.



```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.3.5

Pinging 192.168.3.5 with 32 bytes of data:

Request timed out.
Reply from 192.168.3.5: bytes=32 time<1ms TTL=126
Reply from 192.168.3.5: bytes=32 time=2ms TTL=126
Reply from 192.168.3.5: bytes=32 time<1ms TTL=126

Ping statistics for 192.168.3.5:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 2ms, Average = 0ms
```

Fig: Connectivity test from PC(Dilasha1(192.168.2.7)) to PC(Dilasha2(192.168.3.5)

Using OSPF Command

Network Diagram

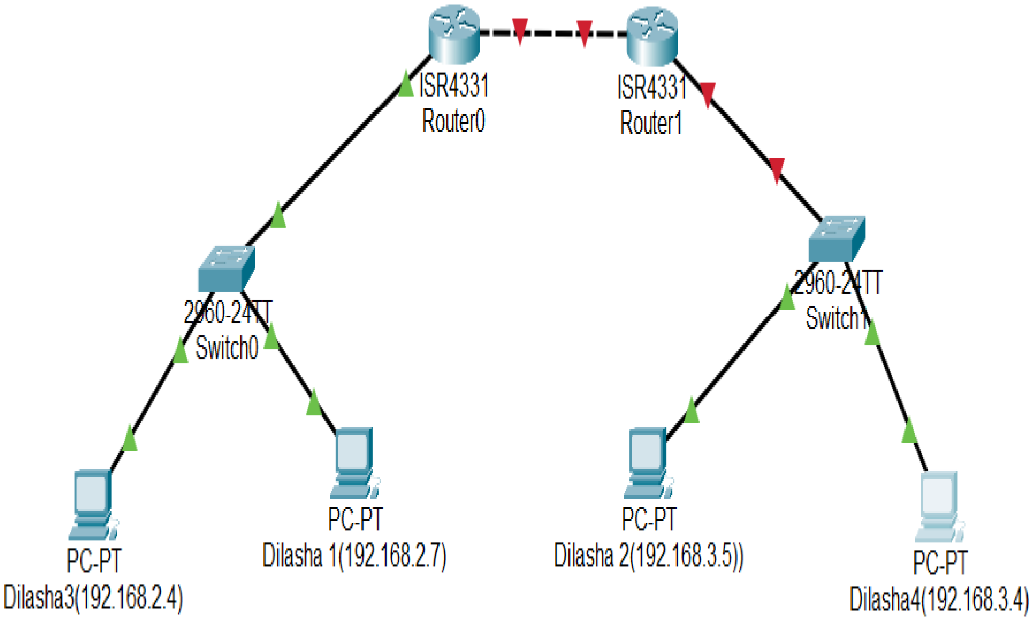
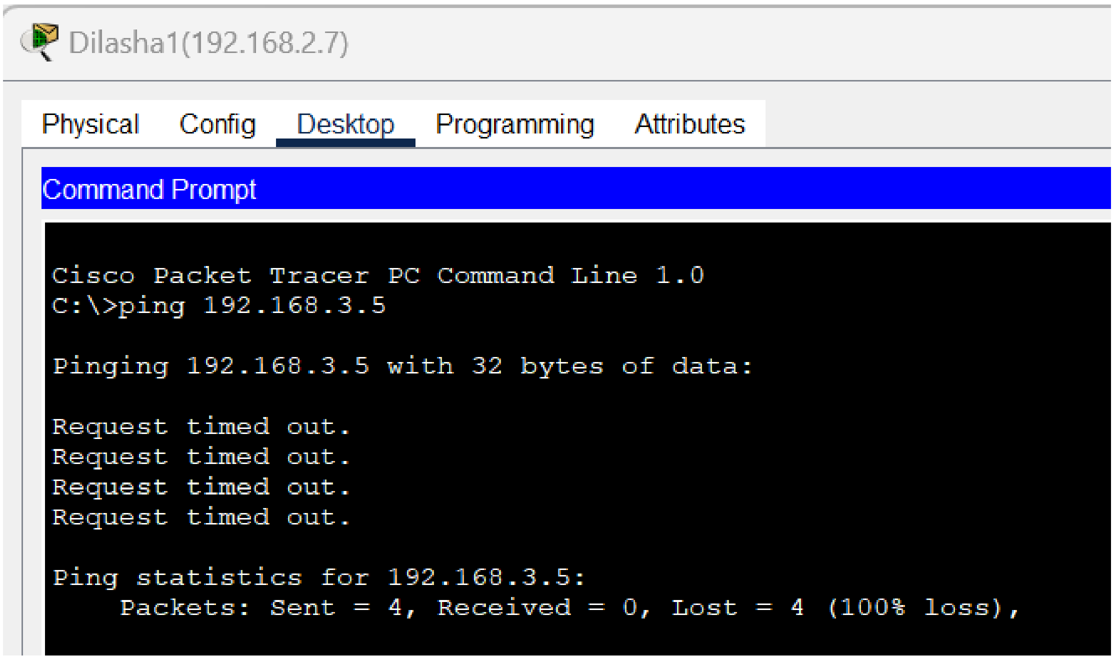


Fig: Network Diagram

Connection Testing Before Dynamic Routing Configuration using OSPF command

Steps:

1.Pinging PC(Dilasha2(192.168.3.5)) from PC(Dilasha1(192.168.2.7)) to verify connection if exists.



```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.3.5

Pinging 192.168.3.5 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.3.5:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

Fig: Connectivity test from PC(Dilasha1(192.168.2.7)) to PC(Dilasha2(192.168.3.5))

Here we can see there is not any connection in the network. So to establish connection in the network, we need to dynamically configure the router through CLI using OSPF command.

Code for Dynamic Routing Configuration Using OSPF Command

For Router 0:

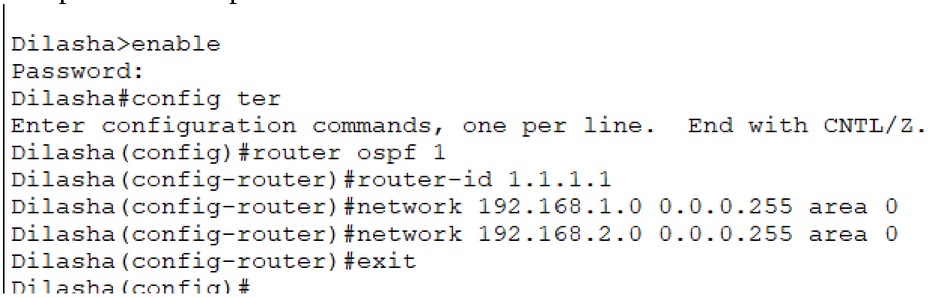
```
Router0> enable
Router0# configure terminal
Router0(config)# router ospf 1
Router0(config-router)# router-id 1.1.1.1
Router0(config-router)# network 192.168.1.0 0.0.0.255 area 0
Router0(config-router)# network 192.168.2.0 0.0 0.255 area 0
Router0(config-router)# exit
```

For Router 1:

```
Router1> enable
Router1# configure terminal
Router1(config)# router ospf 1
Router1(config-router)# router-id 2.2.2.2
Router1(config-router)# network 192.168.1.0 0.0.0.255 area 0
Router1(config-router)# network 192.168.3.0 0.0 0.255 area 0
Router1(config-router)# exit
```

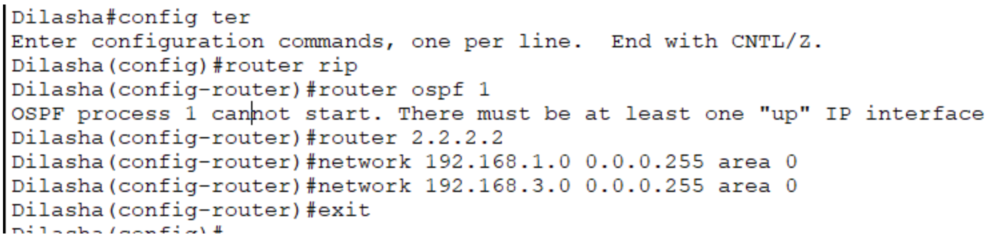
Steps for Dynamic Routing Configuration Using OSPF Command

1.Access Router.
2.Start the OSPF process and assign it a process ID (use 1 in this case)
3.Assign a router ID (OSPF will choose automatically if omitted)
4.Specify the networks connected to Router1, and define the areas.
5.Exit OSPF configuration.
6.Repeat same steps for another Router.



```
Dilasha>enable
Password:
Dilasha#config ter
Enter configuration commands, one per line.  End with CNTL/Z.
Dilasha(config)#router ospf 1
Dilasha(config-router)#router-id 1.1.1.1
Dilasha(config-router)#network 192.168.1.0 0.0.0.255 area 0
Dilasha(config-router)#network 192.168.2.0 0.0.0.255 area 0
Dilasha(config-router)#exit
Dilasha(config)#
```

Fig: Router configuration on router (Dilasha) using OSPF command.



```
Dilasha#config ter
Enter configuration commands, one per line.  End with CNTL/Z.
Dilasha(config)#router rip
Dilasha(config-router)#router ospf 1
OSPF process 1 cannot start. There must be at least one "up" IP interface
Dilasha(config-router)#router 2.2.2.2
Dilasha(config-router)#network 192.168.1.0 0.0.0.255 area 0
Dilasha(config-router)#network 192.168.3.0 0.0.0.255 area 0
Dilasha(config-router)#exit
Dilasha(config)#
```

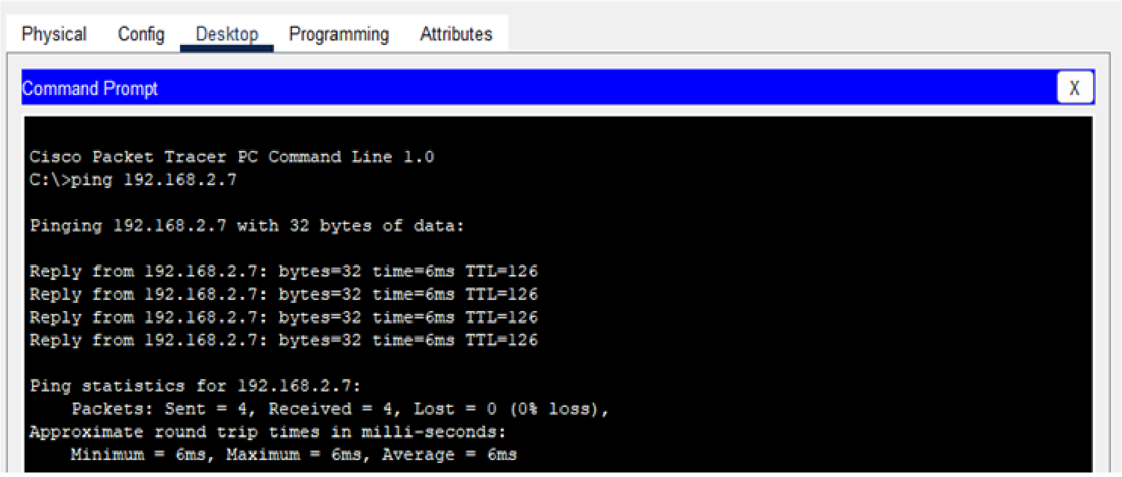
Fig: Router configuration on router (Dilasha1) using OSPF command.

Testing and Validation

To test whether the network is working, you can ping other devices on the network from each PC.

Steps:

1. Ping PC(Dilasha2(192.168.3.5)) from PC(Dilasha1(192.168.2.7)).
2. If the ping is successful, you should see replies from the other device.



```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.2.7

Pinging 192.168.2.7 with 32 bytes of data:

Reply from 192.168.2.7: bytes=32 time=6ms TTL=126
Reply from 192.168.2.7: bytes=32 time=6ms TTL=126
Reply from 192.168.2.7: bytes=32 time=6ms TTL=126
Reply from 192.168.2.7: bytes=32 time=6ms TTL=126

Ping statistics for 192.168.2.7:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 6ms, Maximum = 6ms, Average = 6ms
```

Fig: Connectivity test from PC(Dilasha1(192.168.2.7)) to PC(Dilasha2(192.168.3.5))

Using BGP Command

Network Diagram

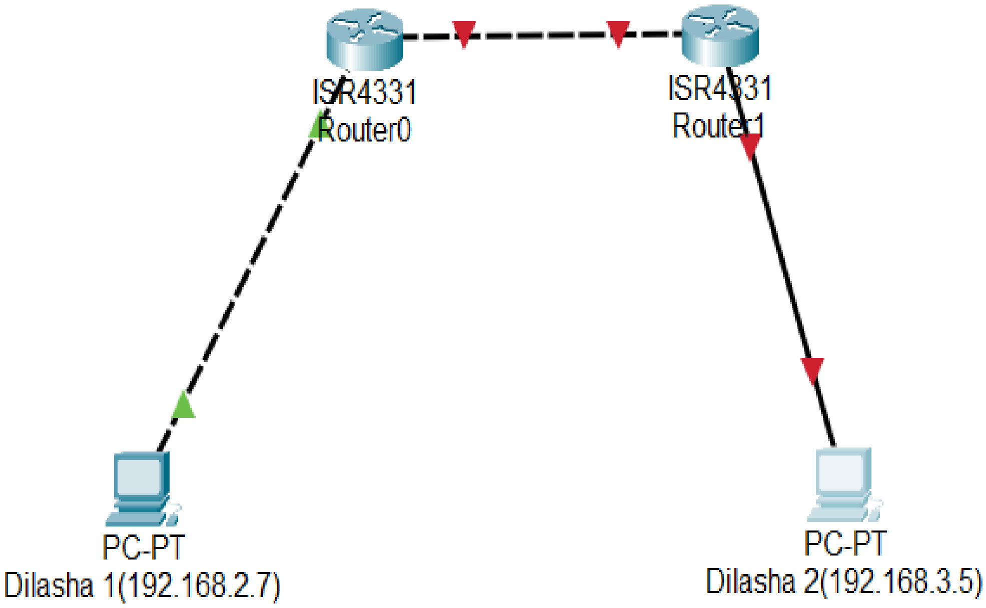
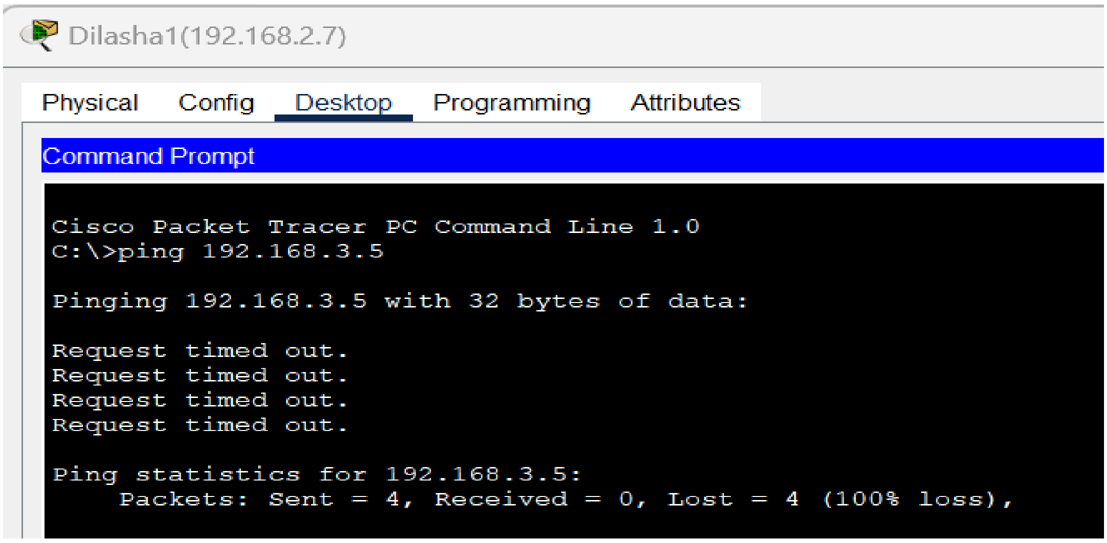


Fig: Network Diagram

Connection Testing Before Dynamic Routing Configuration using BGP Command

Steps:

1.Pinging PC(Dilasha2(192.168.3.5)) from PC(Dilasha1(192.168.2.7)) to verify connection if exists.



```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.3.5

Pinging 192.168.3.5 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.3.5:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

Fig: Connectivity test from PC(Dilasha1(192.168.2.7)) to PC(Dilasha2(192.168.3.5))

Here we can see there is not any connection in the network. So to establish connection in the network ,we need to dynamically configure the router through CLI using BGP command.

Code for Dynamic Routing Configuration Using BGP Command

For Router 0:

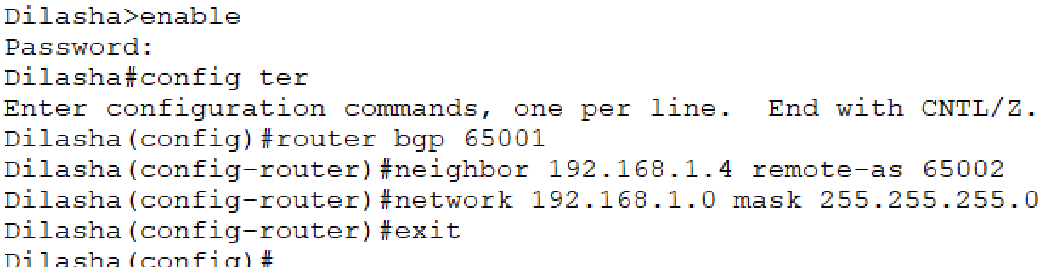
Router0> enable
Router0# configure terminal
Router0(config)# router bgp 65001
Router0(config-router)# neighbor 192.168.1.4 remote-as 65002
Router0(config-router)# network 192.168.1.0 mask 255.255.255.0
Router0(config-router)# exit

For Router 1:

Router1> enable
Router1# configure terminal
Router1(config)# router bgp 65002
Router1(config-router)# neighbor 192.168.1.2 remote-as 65001
Router1(config-router)# network 192.168.2.0 mask 255.255.255.0
Router1(config-router)# exit

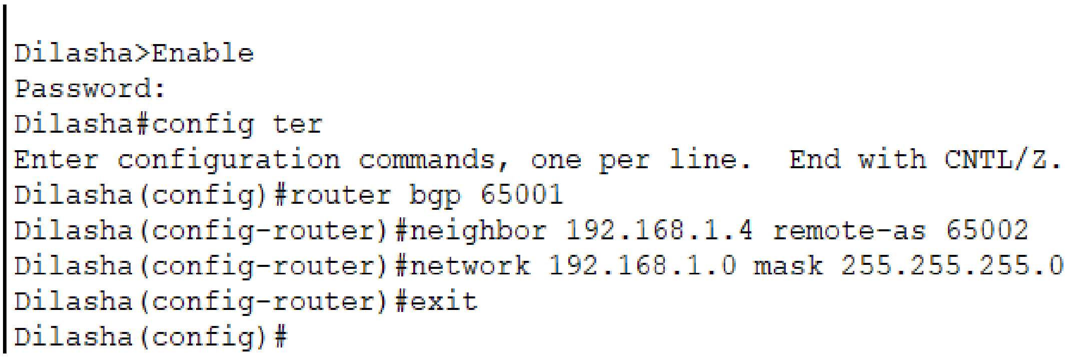
Steps for Dynamic Routing Configuration Using BGP Command

1.Access Router.
2.Start the BGP process and specify the AS number (65001)
3.Specify Router1 as a neighbor and provide its AS number (65002)
4.Advertise the LAN network behind Router1
5.Exit BGP configuration.
6.Repeat same steps for another Router.



```
Dilasha>enable
Password:
Dilasha#config ter
Enter configuration commands, one per line.  End with CNTL/Z.
Dilasha(config)#router bgp 65001
Dilasha(config-router)#neighbor 192.168.1.4 remote-as 65002
Dilasha(config-router)#network 192.168.1.0 mask 255.255.255.0
Dilasha(config-router)#exit
Dilasha(config)#
```

Fig: Router configuration on router (Dilasha1) using BGP command.



```
Dilasha>Enable
Password:
Dilasha#config ter
Enter configuration commands, one per line.  End with CNTL/Z.
Dilasha(config)#router bgp 65001
Dilasha(config-router)#neighbor 192.168.1.4 remote-as 65002
Dilasha(config-router)#network 192.168.1.0 mask 255.255.255.0
Dilasha(config-router)#exit
Dilasha(config)#
```

Fig: Router configuration on router (Dilasha) using BGP command.

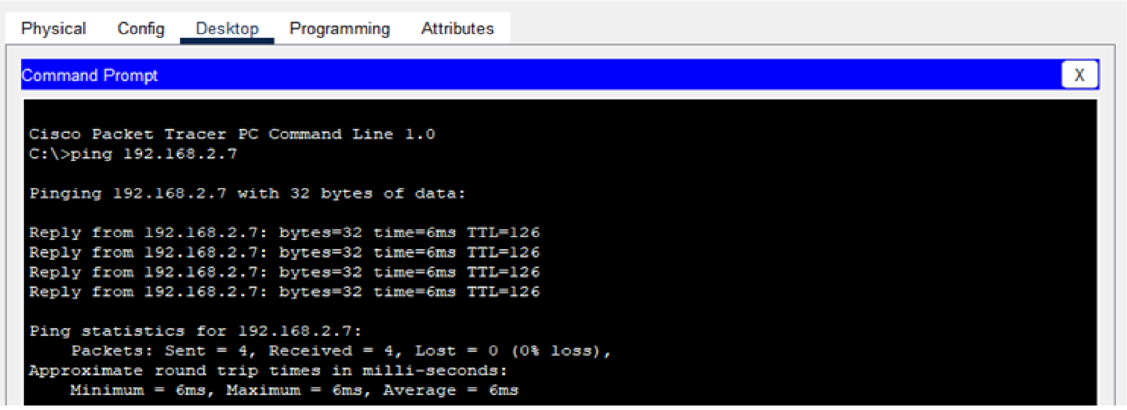
Testing and Validation

To test whether the network is working, you can ping other devices on the network from each PC.

Steps:

1. Ping PC(Dilasha1(192.168.2.7)) from PC(Dilasha2(192.168.3.5)).

2. If the ping is successful, you should see replies from the other device.



```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.2.7

Pinging 192.168.2.7 with 32 bytes of data:

Reply from 192.168.2.7: bytes=32 time=6ms TTL=126
Reply from 192.168.2.7: bytes=32 time=6ms TTL=126
Reply from 192.168.2.7: bytes=32 time=6ms TTL=126
Reply from 192.168.2.7: bytes=32 time=6ms TTL=126

Ping statistics for 192.168.2.7:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 6ms, Maximum = 6ms, Average = 6ms
```

Fig: Connectivity test from PC(Dilasha2(192.168.3.5)) to PC(Dilasha1(192.168.2.7))

Conclusion

In this lab, we successfully implemented three key dynamic routing protocols—RIP, OSPF, and BGP demonstrating their functionality in both interior and exterior routing contexts. Each protocol serves distinct purposes based on the network's size, complexity, and requirements. Through testing and verification, we observed how dynamic routing protocols automatically adjusted to network changes and maintained efficient data routing.

Lab 10: Implementing ACL in Packet Tracer

Theory

An Access Control List (ACL) is a set of guidelines used on router interfaces to manage the traffic passing through a network. ACLs can be configured to either allow or block traffic based on several factors, including the source and destination IP addresses, protocols (such as TCP, UDP, or ICMP), and port numbers.
ACLs are classified into two types:

Standard ACL: Filters traffic based only on the source IP address.
Extended ACL: Filters traffic based on both source and destination IPs, protocols, and ports.

The primary functions of ACLs are:

1. Enhancing network security by limiting access to certain resources.
2. Controlling traffic flow by permitting or denying packets based on specific criteria.
3. Improving network performance by reducing unwanted traffic.

Network Diagram

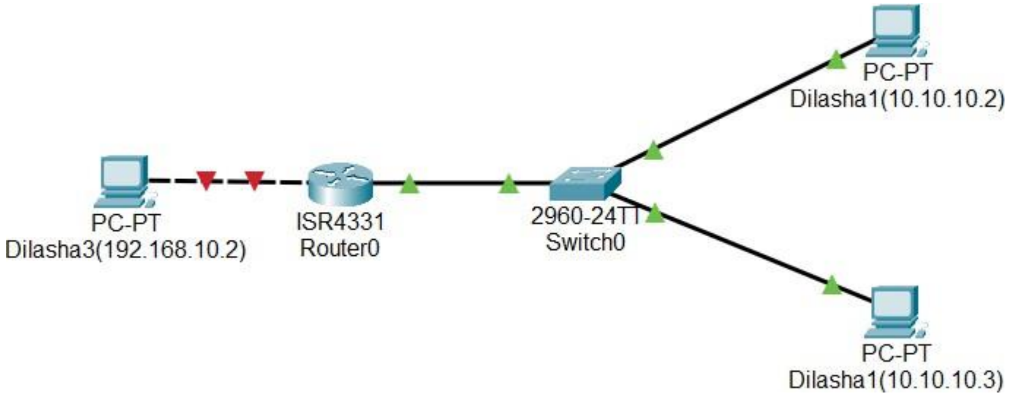
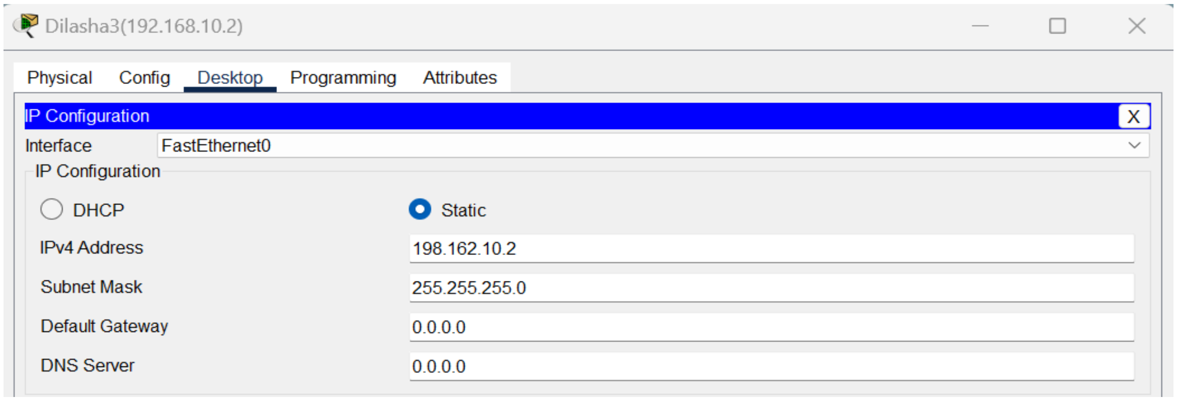


Fig: Network Diagram

Configure network for PCs and Routers

1. Assign IP addresses to the PCs according to the network plan like PC(Dilasha3(192.168.10.2)), PC(Dilasha2(10.10.10.2)) and PC(Dilasha2(10.10.10.3)).



Dilasha3(192.168.10.2)

IP Configuration

Interface: FastEthernet0

IP Configuration

○ DHCP ● Static

IPv4 Address: 198.162.10.2

Subnet Mask: 255.255.255.0

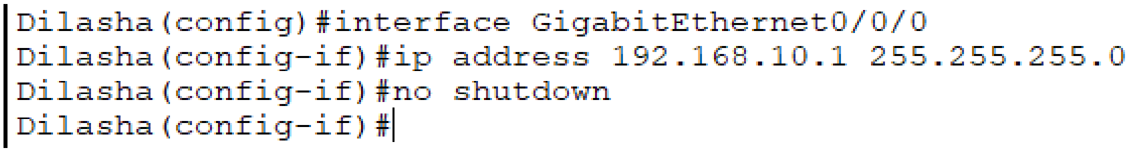
Default Gateway: 0.0.0.0

DNS Server: 0.0.0.0

Fig: IP configuration on PC's

Configure Routers

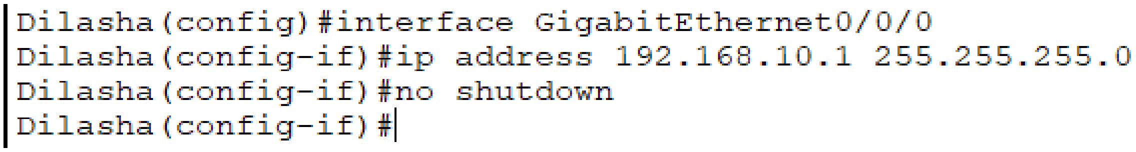
1. Access the router's command-line interface.
2. Assign IP addresses to router interfaces that connect to the PCs.



```
Dilasha(config)#interface GigabitEthernet0/0/0
Dilasha(config-if)#ip address 192.168.10.1 255.255.255.0
Dilasha(config-if)#no shutdown
Dilasha(config-if)#
```

Configuring Access List

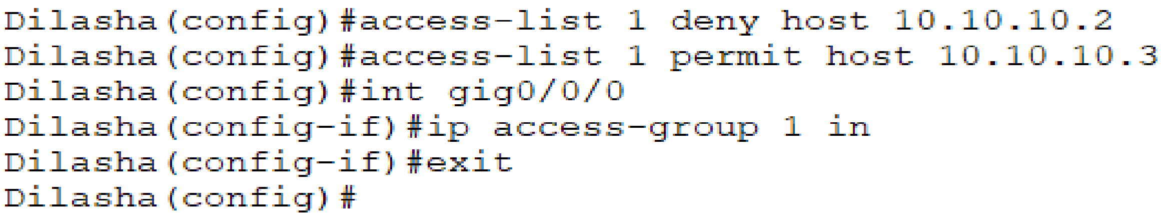
1. Access the router's command-line interface.
2. Assign IP addresses to router interfaces that connect to the PCs.



```
Dilasha(config)#interface GigabitEthernet0/0/0
Dilasha(config-if)#ip address 192.168.10.1 255.255.255.0
Dilasha(config-if)#no shutdown
Dilasha(config-if)#
```

Configure DENY and PERMIT list

1. Access global configuration mode
2. Apply the ACL to an interface (e.g., blocking PC1's access to the network):



```
Dilasha(config)#access-list 1 deny host 10.10.10.2
Dilasha(config)#access-list 1 permit host 10.10.10.3
Dilasha(config)#int gig0/0/0
Dilasha(config-if)#ip access-group 1 in
Dilasha(config-if)#exit
Dilasha(config)#
```

Fig: Configuring DENY and PERMIT list

Code for Configuring DENY and PERMIT list

Router(config)# access-list 1 deny host 10.10.10.2
Router(config)# access-list 1 permit host 10.10.10.3 Router(config)#
interface gig0/0
Router(config-if)# ip access-group 1 in
Router(config-if)# exit

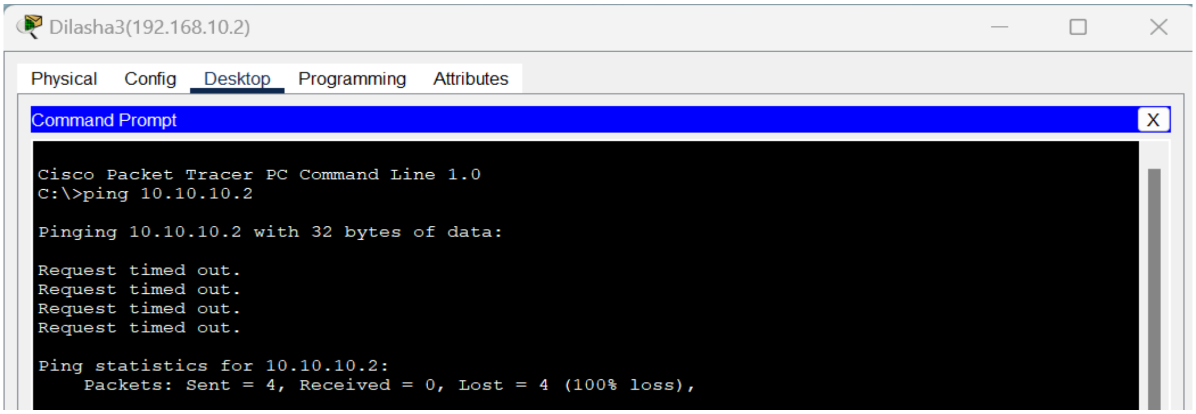
Implementation and Testing

To test whether the network is working, you can ping other devices on the network from each PC.

Steps:

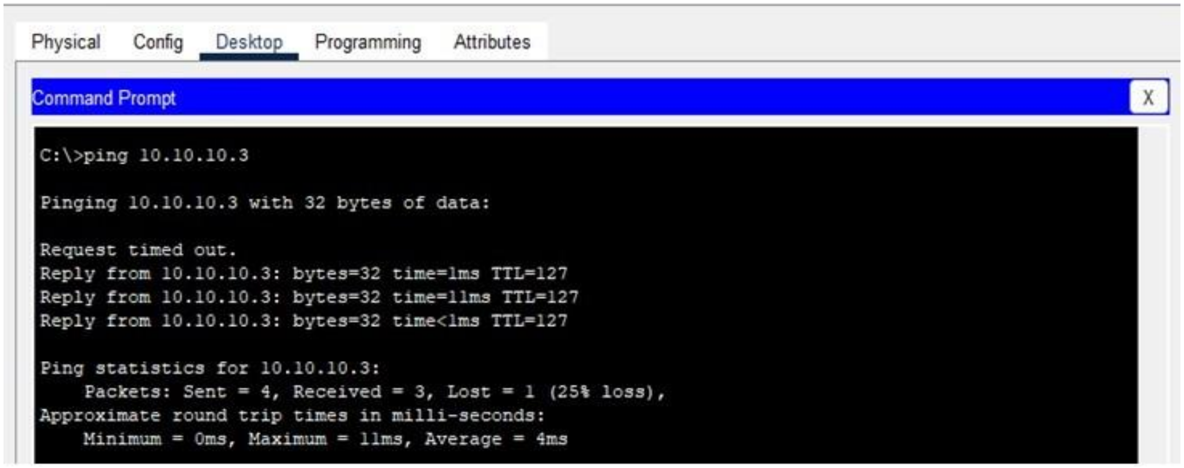
1. Ping PC(Dilasha1(10.10.10.2)) from PC(Dilasha3(192.168.10.2)) to verify that the connection is denied.
2. Ping PC(Dilasha1(10.10.10.3)) from PC(Dilasha3(192.168.10.2)) to verify that the connection is permitted.

If the ping is successful, you should see replies from the other device.



```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 10.10.10.2

Pinging 10.10.10.2 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 10.10.10.2:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

Fig: Connectivity test from PC(Dilasha3(192.168.10.2)) to PC(Dilasha1(10.10.10.2))



```
C:\>ping 10.10.10.3

Pinging 10.10.10.3 with 32 bytes of data:

Request timed out.
Reply from 10.10.10.3: bytes=32 time=1ms TTL=127
Reply from 10.10.10.3: bytes=32 time=11ms TTL=127
Reply from 10.10.10.3: bytes=32 time<1ms TTL=127

Ping statistics for 10.10.10.3:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 11ms, Average = 4ms
```

Fig: Connectivity test from PC(Dilasha3(192.168.10.2)) to PC(Dilasha1(10.10.10.3)

Conclusion

In this lab, we utilized Cisco Packet Tracer to implement Access Control Lists (ACLs) for regulating network traffic. Both standard and extended ACLs were applied to allow or block traffic based on IP addresses, protocols, and ports. This exercise emphasized the role of ACLs in strengthening network security by controlling access and preventing unauthorized traffic. Testing verified the correct operation and effectiveness of the ACL configurations.

Lab 11: DNS and Web Server Configuration using Packet Tracer

Theory

DNS (Domain Name System): DNS is a system that converts human-readable domain names (such as google.com) into numerical IP addresses (like 192.168.1.1) that computers use to communicate. By mapping these domain names to IP addresses, DNS makes it easier for users to browse the internet, removing the need to memorize complicated number sequences. When a user enters a domain name in their browser, DNS operates behind the scenes to find the associated IP address, enabling fast and easy access to the desired web page.

Web Server: A web server is a computer system responsible for hosting websites and delivering web pages to users upon request. When a user enters a URL or clicks a link, the web server processes the request, retrieves the appropriate files (such as HTML, images, or scripts), and sends them to the user's browser. Web servers manage these requests through HTTP or HTTPS protocols. They handle both static content (like basic web pages) and dynamic content generated by server-side applications. In short, web servers ensure that web content is available and delivered efficiently to users.

Network Diagram

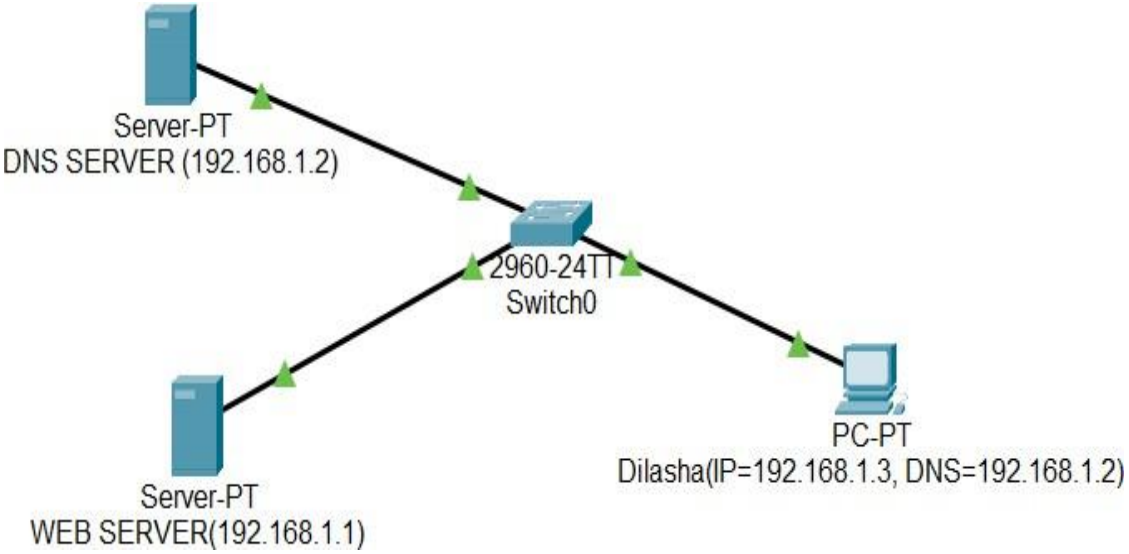


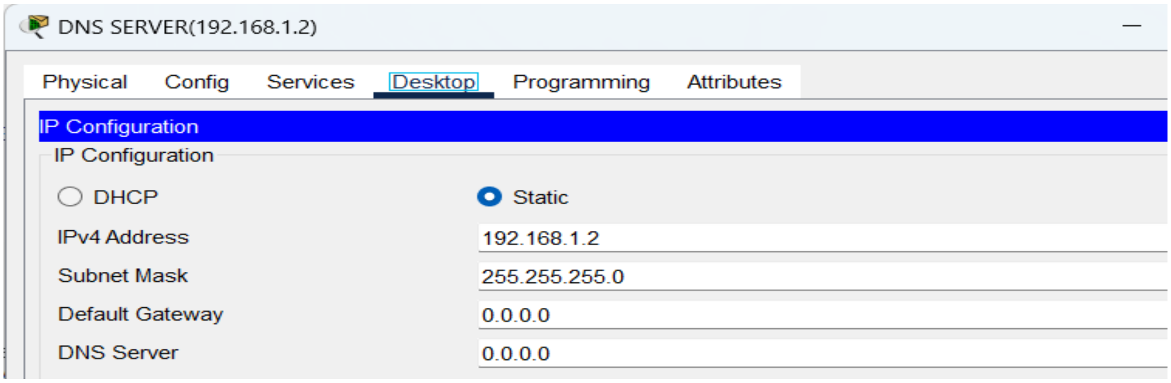
Fig: Network Diagram

1.Configuring DNS Server and Web Server

DNS Server Configuration:

Step 1: Go to the DNS Server and click on the Desktop tab.

Step 2: Set the IP Address to 192.168.1.2 and the Subnet Mask to 255.255.255.0.



IP Configuration

○ DHCP ● Static

IPv4 Address: 192.168.1.2

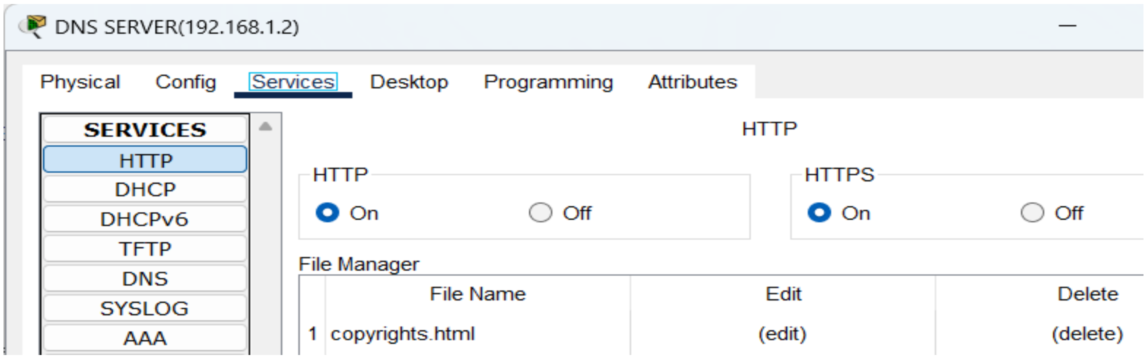
Subnet Mask: 255.255.255.0

Default Gateway: 0.0.0.0

DNS Server: 0.0.0.0

Fig: Configuring DNS server

Step 3: Click on the Services tab, then navigate to HTTP and turn it ON.



HTTP

HTTP: ● On ○ Off

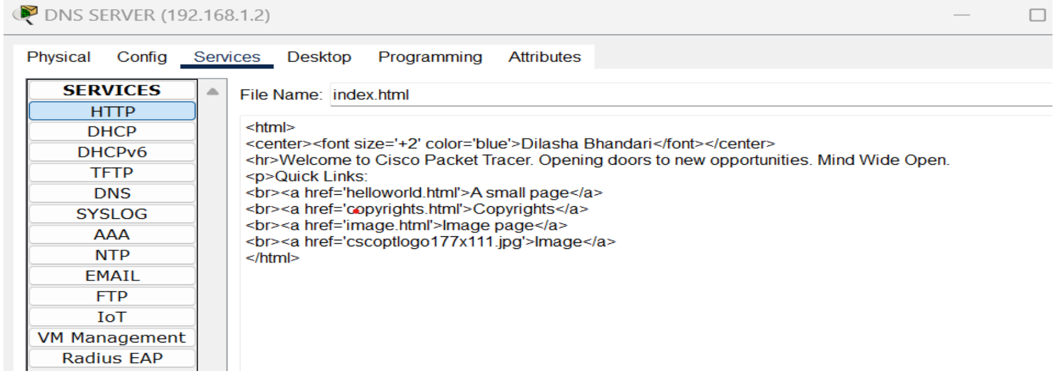
HTTPS: ● On ○ Off

File Manager

	File Name	Edit	Delete
1	copyrights.html	(edit)	(delete)

Fig: Turning ON HTTP

Step 4: And Go to index.html, click edit, make the necessary changes to the file, and click Save.



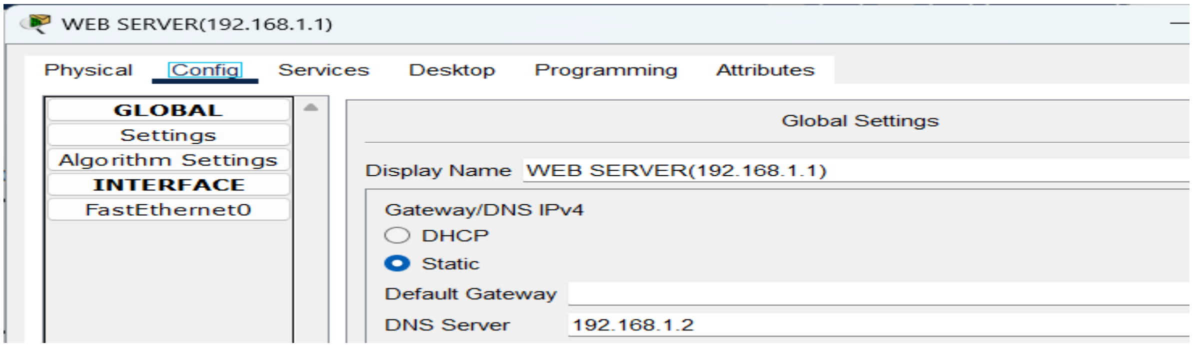
File Name: index.html

```
<html>
<center><font size='+2' color='blue'>Dilasha Bhandari</font></center>
<hr>Welcome to Cisco Packet Tracer. Opening doors to new opportunities. Mind Wide Open.
<p>Quick Links:
<br><a href='helloworld.html'>A small page</a>
<br><a href='copyrights.html'>Copyrights</a>
<br><a href='image.html'>Image page</a>
<br><a href='cscoptlogo177x111.jpg'>Image</a>
</html>
```

Fig: Configuring index.html file

Web Server Configuration:

Step 1: Click on the Web Server, then go to the Config tab.
Step 2: Set the DNS Server to 192.168.1.2.



WEB SERVER(192.168.1.1)

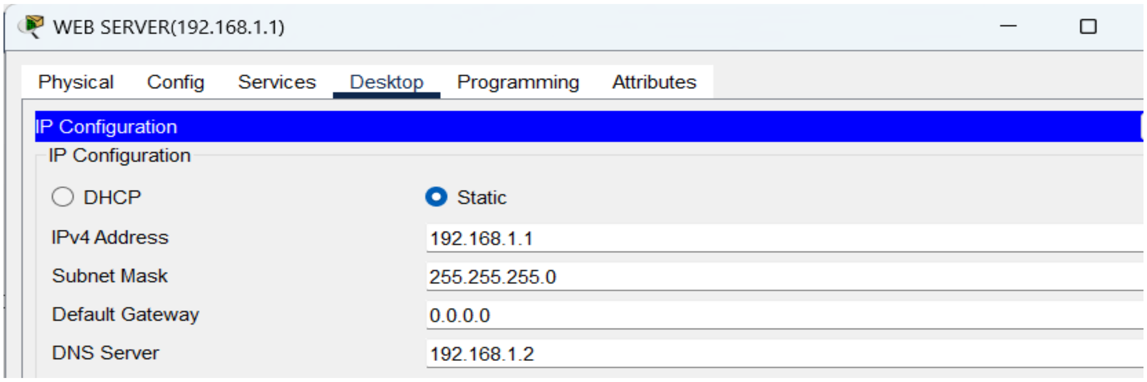
Global Settings

Display Name WEB SERVER(192.168.1.1)

Gateway/DNS IPv4
○ DHCP
● Static
Default Gateway
DNS Server 192.168.1.2

Fig: Configuring DNS on WEB server

Step 3: Go to the Desktop tab and set the IP Address to 192.168.1.1, the Subnet Mask to 255.255.255.0, and the DNS Server to 192.168.1.2.



WEB SERVER(192.168.1.1)

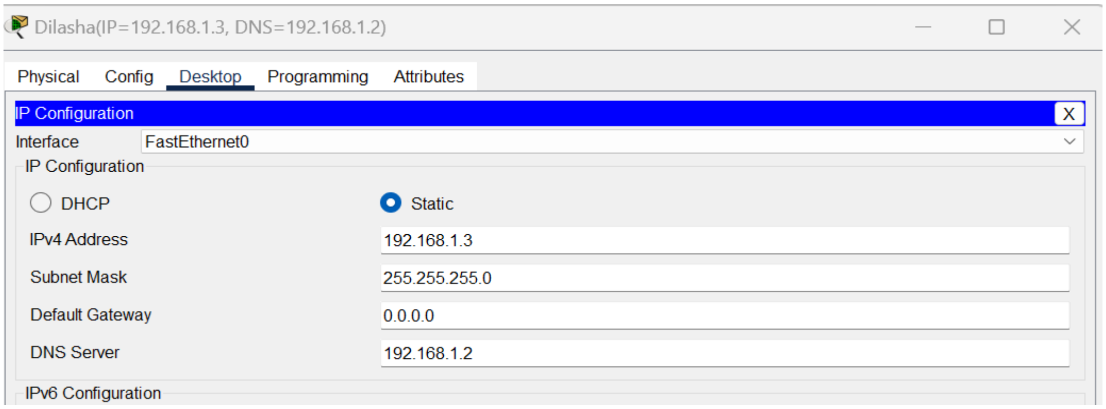
IP Configuration
IP Configuration
○ DHCP ● Static
IPv4 Address 192.168.1.1
Subnet Mask 255.255.255.0
Default Gateway 0.0.0.0
DNS Server 192.168.1.2

Fig: Configuring WEB server

PC Configuration:

Step 1: Go to the PC, then navigate to the Desktop tab.

Step 2: Set the IP Address to 192.168.1.3, the Subnet Mask to 255.255.255.0, and the DNS Server to 192.168.1.2.



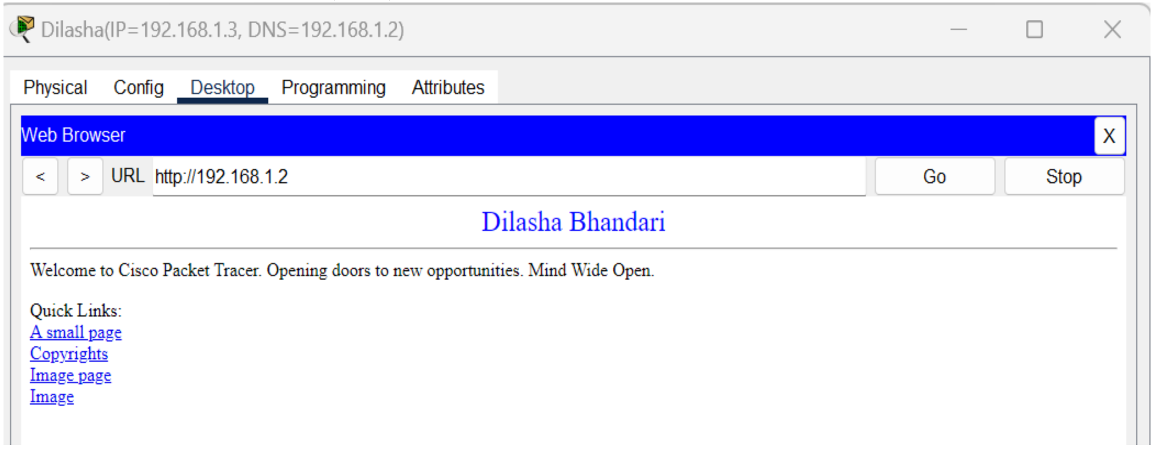
Dilasha(IP=192.168.1.3, DNS=192.168.1.2)

IP Configuration
Interface FastEthernet0
IP Configuration
○ DHCP ● Static
IPv4 Address 192.168.1.3
Subnet Mask 255.255.255.0
Default Gateway 0.0.0.0
DNS Server 192.168.1.2
IPv6 Configuration

Fig: IP configuration on PC

Testing Web Server on PC

Step 1: Go to the Desktop, click on Web Browser, and in the URL tab, enter the IP address of the Web Server, i.e., 192.168.1.1.



URL http://192.168.1.2

Dilasha Bhandari

Welcome to Cisco Packet Tracer. Opening doors to new opportunities. Mind Wide Open.

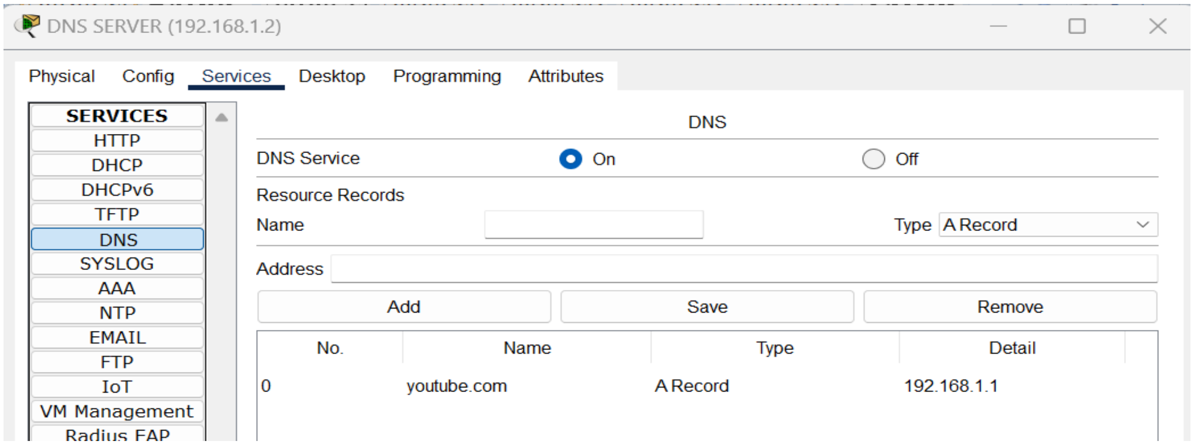
Quick Links:
A small page
Copyrights
Image page
Image

Fig: Checking WEB server on PC

Web Server in DNS Server Configuration

Step 1: Go to the DNS Server then click on Services, then select DNS.

Step 2: In the Name field, type google.com, enter the IP address of the Web Server in Address field, i.e., 192.168.1.1 then click add and then save.



DNS Service ● On ○ Off

Resource Records

Name

Type A Record

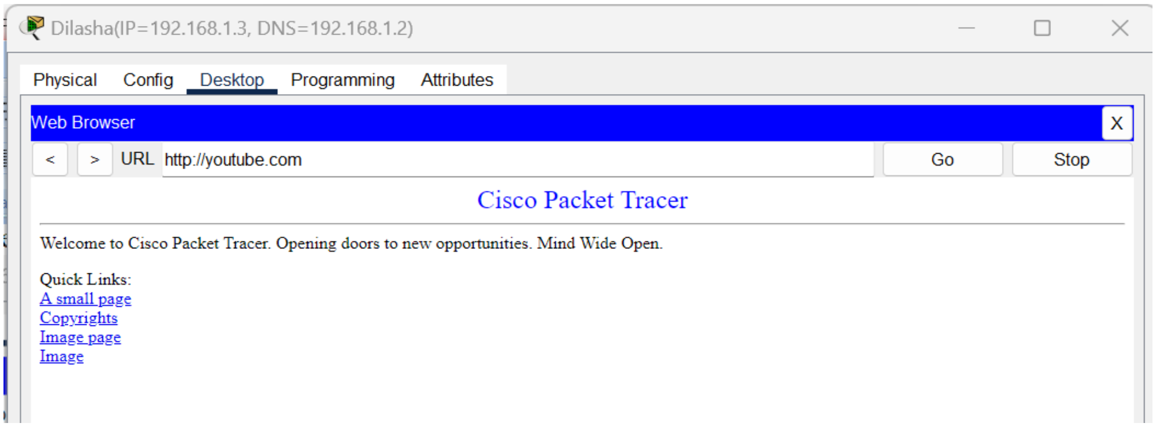
Address

No.	Name	Type	Detail
0	youtube.com	A Record	192.168.1.1

Fig: Adding WEB server on DNS server

Testing Web Server on PC after configuring in DNS

Step 1: Go to the Desktop, click on Web Browser, and in the URL tab, type google.com to access the web page served by the Web Server.



Dilasha(IP=192.168.1.3, DNS=192.168.1.2)

URL http://youtube.com

Cisco Packet Tracer

Welcome to Cisco Packet Tracer. Opening doors to new opportunities. Mind Wide Open.

Quick Links:
A small page
Copyrights
Image page
Image

Fig: Checking Web Server after configuration in DNS

From above picture, we can verify that whether we type Name or IP address we will get the same web page.

Conclusion

In this lab, we demonstrate how DNS simplifies the process of accessing web services by translating human-readable domain names into machine-friendly IP addresses. The configuration in Packet Tracer highlights how DNS and web servers are set up and operate within a network, working together to make it more convenient for users to access web resources through familiar domain names, rather than needing to remember complex numerical IP addresses.

Lab 12: FTP Configuration and Implementation using Packet Tracer

Theory

FTP (File Transfer Protocol) is a widely-used network protocol for transferring files between a client and a server over a TCP/IP network. It utilizes port 21 for control commands and port 20 for data transfer in active mode. Clients can connect to FTP servers either through anonymous access or with user authentication. FTP operates in two modes: active and passive, depending on which side, the server or the client, initiates the data connection. For enhanced security, protocols like FTPS or SFTP are used to encrypt the data during transfer. FTP supports various file operations such as uploading, downloading, and file management through commands like RETR, STOR, and DELE. In ASCII mode, it adjusts text files by converting line endings to match different operating systems, while in binary mode, it preserves the original byte sequence of files. However, FTP lacks built-in encryption, making it less secure than modern alternatives like FTPS and SFTP, which are better suited for transferring sensitive information.

Key Concepts of FTP

1. **Client-Server Model**: FTP follows a client-server architecture where the client initiates requests for file operations and the server responds. The client communicates with the server to request files, upload files, or perform other file management tasks.

2. **Ports:** FTP uses port 21 for the control connection, where commands and responses are exchanged. Port 20 is used for the data connection in active mode, while passive mode uses a dynamically assigned port by the server for data transfer.

3. **Active and Passive Modes:** In active mode, the client opens a port for data transfer and the server connects to it. In passive mode, the server opens a port and the client connects to it, which helps navigate firewalls and NAT issues.

4. **Authentication:** FTP can operate in anonymous mode, allowing users to access files without a password, or in authenticated mode, requiring a username and password for access. This distinction helps manage access and security.

5. **FTP Commands:** Common FTP commands include LIST to view files, RETR to download files, STOR to upload files, DELE to delete files, and MKD to create directories. These commands manage file operations on the server.

6. **Data Types:** FTP supports ASCII mode for text files, converting line endings between systems, and binary mode for non-text files, preserving exact byte sequences. Choosing the correct mode ensures proper file integrity.

7. **FTP Security:** FTPS and SFTP are used to secure FTP connections. FTPS adds SSL/TLS encryption to FTP, while SFTP uses SSH for secure file transfer, protecting data from interception and unauthorized access.

Network Diagram

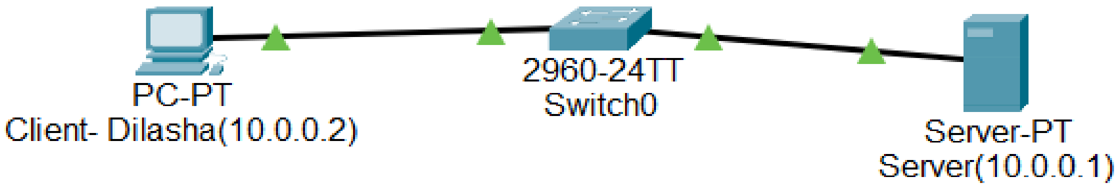
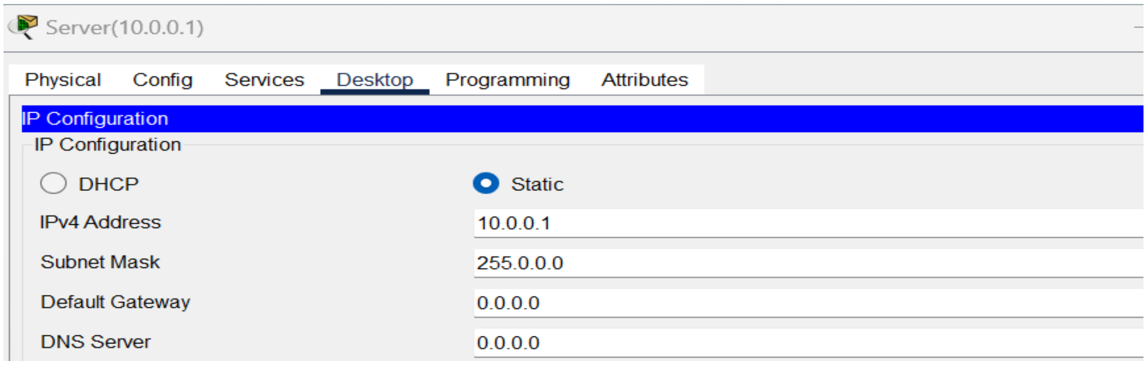


Fig: Network diagram

Configuring FTP Server and FTP Client

FTP Server Configuration

Step1: Click on server and go to ip configuration and set ip address and subnet mask.



IP Configuration

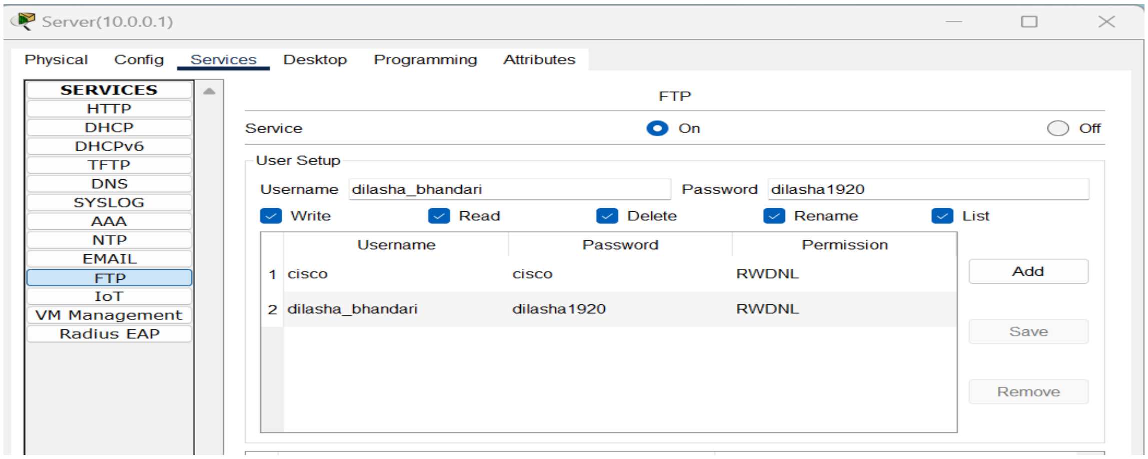
○ DHCP ● Static

IPv4 Address	10.0.0.1
Subnet Mask	255.0.0.0
Default Gateway	0.0.0.0
DNS Server	0.0.0.0

Fig: IP configuration on Sever

Step 2: Click on server, go to service, click ftp and click on ON button.

Step 3: Set username, password, and tick write, read, delete, rename, and list. Click add.



Service ● On ○ Off

User Setup

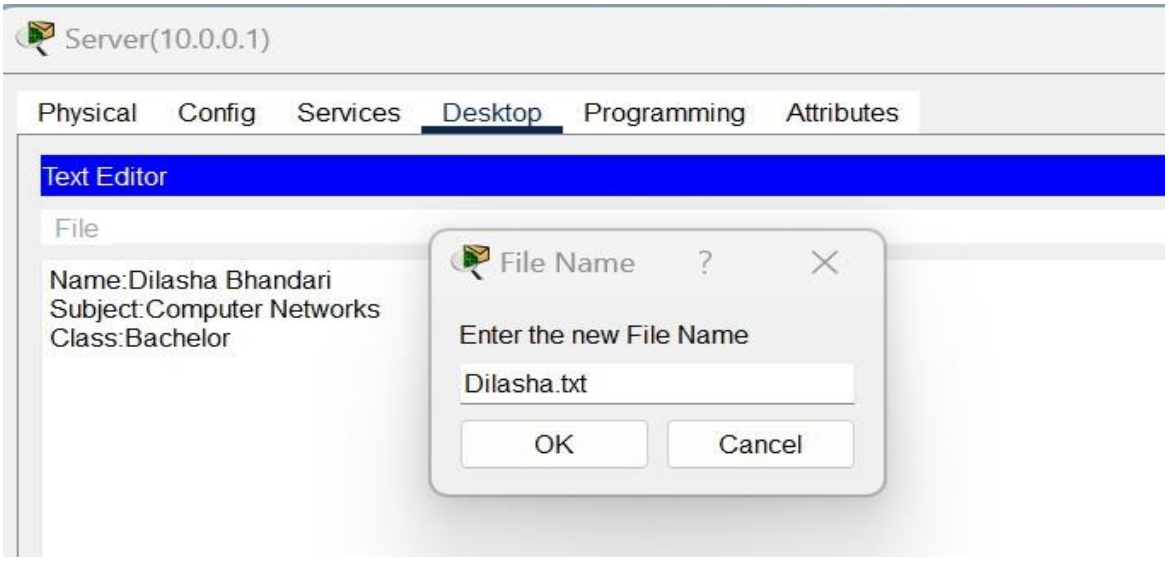
Username dilasha_bhandari Password dilasha1920

☑ Write ☑ Read ☑ Delete ☑ Rename ☑ List

	Username	Password	Permission
1	cisco	cisco	RWDNL
2	dilasha_bhandari	dilasha1920	RWDNL

Fig: Server Configuration

Step 4: Go to desktop, click text editor write something and save the file as Dilasha.txt.

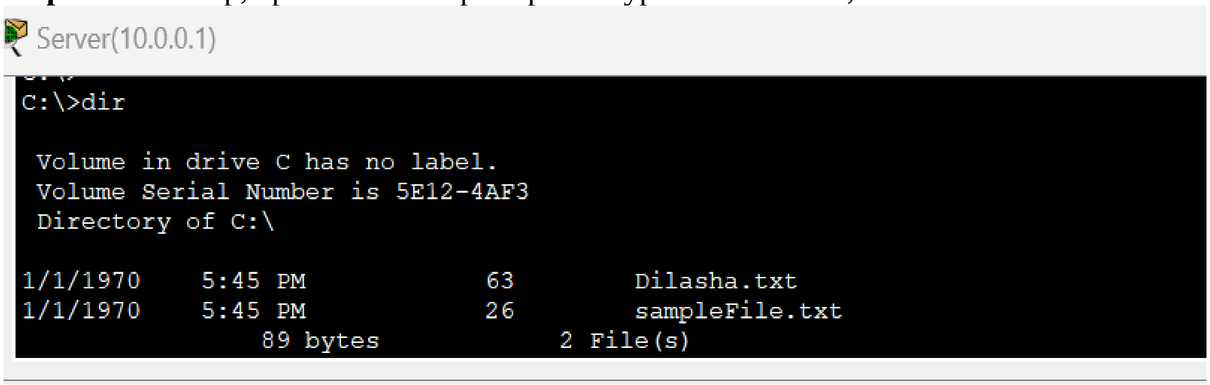


Name:Dilasha Bhandari
Subject:Computer Networks
Class:Bachelor

Enter the new File Name: Dilasha.txt

Fig: Creating a file name Dilasha.txt

Step 5: In desktop, open command prompt and type dir command, we can see the file.

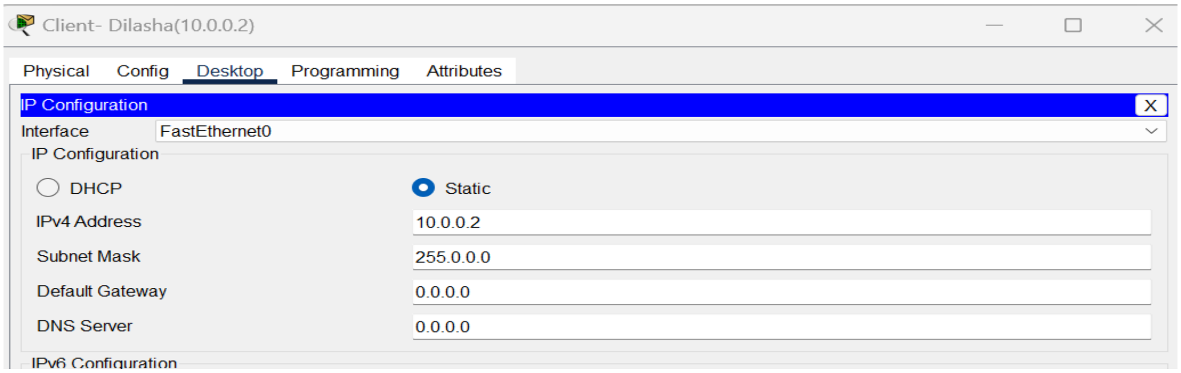


```
C:\>dir

 Volume in drive C has no label.
 Volume Serial Number is 5E12-4AF3
 Directory of C:\

1/1/1970    5:45 PM            63        Dilasha.txt
1/1/1970    5:45 PM            26        sampleFile.txt
                89 bytes            2 File(s)
```

Fig: Using dir command to see file

FTP Client Configuration

Step 1: Click on pc and goto ip configuration and set ip address and subnet mask.



Interface: FastEthernet0

IP Configuration

○ DHCP ● Static

IPv4 Address: 10.0.0.2

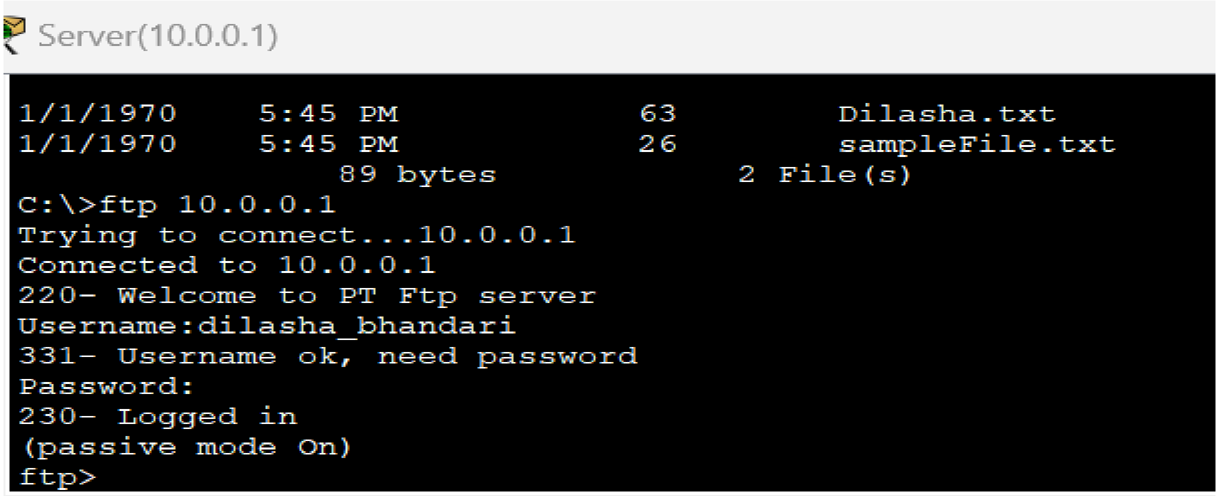
Subnet Mask: 255.0.0.0

Default Gateway: 0.0.0.0

DNS Server: 0.0.0.0

Fig: IP Configuration in Client

In command prompt type command ‘ftp 10.0.0.1’then insert username and password, we will be connected to ftp server.

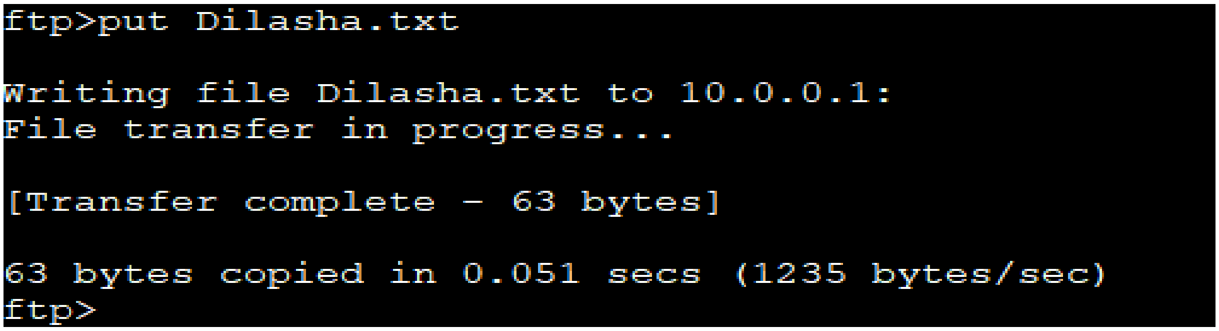


```
1/1/1970    5:45 PM            63        Dilasha.txt
1/1/1970    5:45 PM            26        sampleFile.txt
                89 bytes            2 File(s)
C:\>ftp 10.0.0.1
Trying to connect...10.0.0.1
Connected to 10.0.0.1
220- Welcome to PT Ftp server
Username:dilasha_bhandari
331- Username ok, need password
Password:
230- Logged in
(passive mode On)
ftp>
```

Fig: FTP Server Connection

TRANSFERRING FILE USING PUT COMMAND

Command: Ftp > put Dilasha.txt

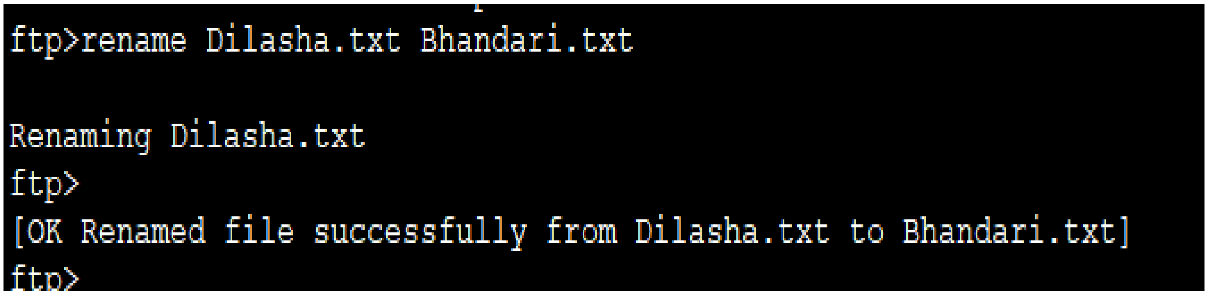


```
ftp>put Dilasha.txt

Writing file Dilasha.txt to 10.0.0.1:
File transfer in progress...

[Transfer complete - 63 bytes]

63 bytes copied in 0.051 secs (1235 bytes/sec)
ftp>
```

Fig: Transferring file using PUT command

RENAME FILE

Command: Ftp > rename Dilasha.txt temp.txt



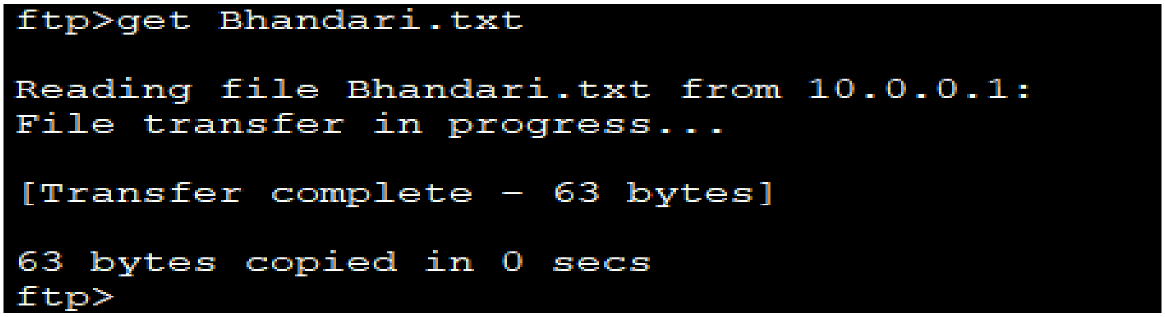
```
ftp>rename Dilasha.txt Bhandari.txt

Renaming Dilasha.txt
ftp>
[OK Renamed file successfully from Dilasha.txt to Bhandari.txt]
ftp>
```

Fig: Renaming file

GET THE FILE AND SAVE THE COPY ON OUR MACHINE

Command: Ftp > get temp.txt

FTP Server Connection

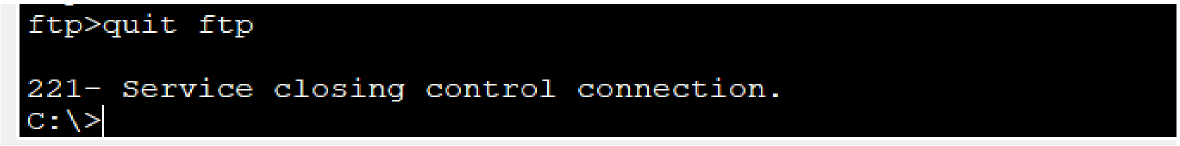


```
ftp>get Bhandari.txt

Reading file Bhandari.txt from 10.0.0.1:
File transfer in progress...

[Transfer complete - 63 bytes]

63 bytes copied in 0 secs
ftp>
```

Fig: Saving copy of file in PC

GO TO PC

Command: Ftp> quit ftp

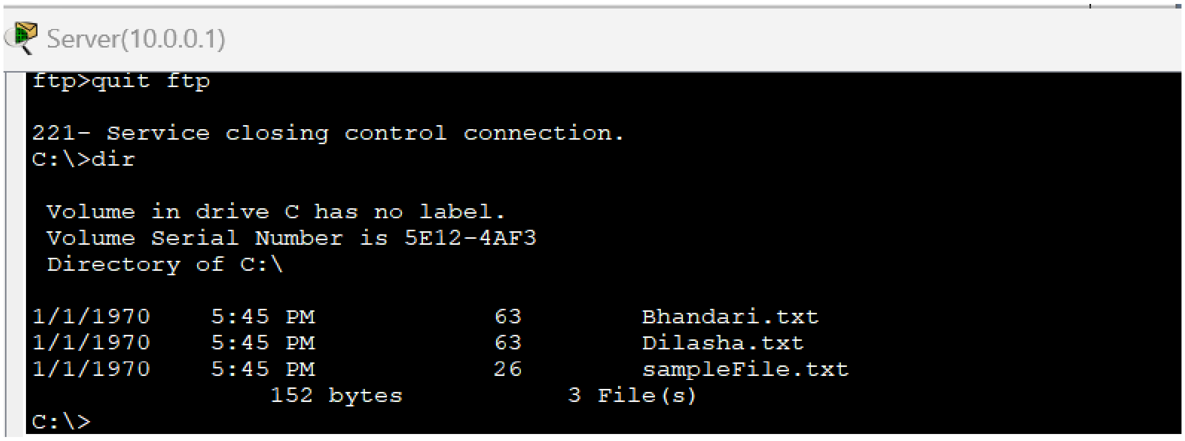


```
ftp>quit ftp

221- Service closing control connection.
C:\>
```

Fig: Quitting FTP

DISPLAYING THE FILES

Command: PC > dir



```
ftp>quit ftp

221- Service closing control connection.
C:\>dir

 Volume in drive C has no label.
 Volume Serial Number is 5E12-4AF3
 Directory of C:\

1/1/1970    5:45 PM            63        Bhandari.txt
1/1/1970    5:45 PM            63        Dilasha.txt
1/1/1970    5:45 PM            26        sampleFile.txt
                152 bytes           3 File(s)
C:\>
```

Fig: Displaying the files

Conclusion

In this lab, we effectively set up an FTP server and client using Packet Tracer. Through a step-by-step approach, we learned how to configure IP addresses, activate the FTP service, and establish user accounts with suitable permissions. We also created a file, confirmed its presence using command-line commands, and showcased file management through FTP operations. This hands-on exercise provided insight into the FTP client-server model, the significance of IP addressing, and the crucial role of secure file transfer within network environments.

Lab 13: Introduction to Network Traffic Analysis using Wireshark

Theory

Wireshark is a popular open-source tool used to analyze network protocols by capturing and inspecting real-time network traffic. It enables users to examine data packets in detail, making it an effective tool for diagnosing network problems, monitoring performance, and detecting security risks. With an intuitive graphical interface and support for various protocols, Wireshark offers a comprehensive view of network activity. Its versatility makes it an essential tool for both network administrators and cybersecurity professionals.

Key Concepts of Wireshark

Packet Capture:

Wireshark captures real-time data packets traveling across a network, allowing detailed analysis of both headers and payloads. This information helps diagnose network performance issues and detect potential security threats. Its packet-level insights are vital for effective network monitoring.

Protocols:

Wireshark supports thousands of protocols, including TCP, UDP, HTTP, and DNS, enabling comprehensive analysis of traffic across various network layers. This broad protocol support allows users to examine data from transport, application, and other layers. It's a powerful tool for in-depth network diagnostics.

Filters:

Wireshark filters allow users to focus on specific data, such as IP addresses, port numbers, or protocol types. **Display filters** show only relevant packets for analysis, while **capture filters** restrict the data being recorded. These filters streamline traffic inspection and improve troubleshooting efficiency.

Frames and Layers:

Wireshark organizes packet data into different layers, such as Ethernet, IP, and TCP, based on the OSI model. This layered view helps break down and analyze packet content at various levels of the network. It provides a clear, structured approach to understanding network traffic.

Hexadecimal and ASCII Views:

Wireshark displays packet data in both hexadecimal and ASCII formats, allowing users to inspect the raw content of each packet. This dual representation enhances understanding of the packet's structure and composition. Users can effectively analyze data integrity and identify anomalies.

Packet Dissection:

Wireshark breaks down packet contents into readable components, showing fields, flags, and values related to the specific protocol in use.

Statistics and Graphs:
Wireshark includes statistical tools like flow graphs and I/O graphs to visualize network traffic and performance metrics. It can capture packets from multiple interfaces, such as Ethernet, Wi-Fi, and loopback, based on the system's available options. This capability enables thorough analysis across diverse network environments.

Real-Time and Offline Analysis:
Wireshark can analyze both live network traffic and saved capture files (PCAP), making it versatile for real-time troubleshooting and post-event analysis. This flexibility allows users to diagnose issues as they occur or review historical data for in-depth examination.

Security:
Wireshark aids in detecting network security issues, such as malware, unauthorized access, and vulnerabilities, by analyzing unusual patterns in packet flows. Its detailed traffic analysis helps identify potential threats and anomalies. This capability is crucial for proactive security monitoring and incident response.

Interface of Wireshark

Main Toolbar:
The toolbar provides quick access to common functions such as opening, saving, or closing captures. It also includes buttons for starting and stopping packet captures, restarting them, and filtering the packet display. Key features include a field for applying display filters, navigating through packets, and zooming in on packet views.

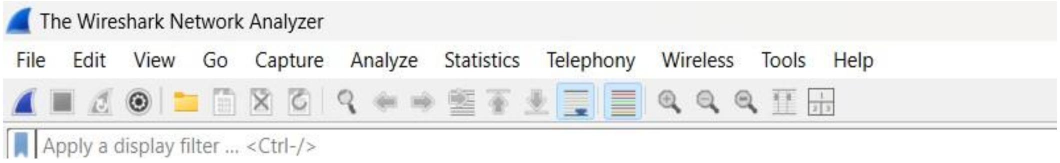

The Wireshark Network Analyzer
File
Edit
View
Go
Capture
Analyze
Statistics
Telephony
Wireless
Tools
Help
Apply a display filter ... <Ctrl-/>

Fig: Toolbar

Packet List Pane:
This pane shows all the captured packets in real time. Each row represents a packet, and columns display key details like packet number, timestamp, source, destination, protocol, length, and info. Clicking on a packet here allows you to view its detailed information in the other panes.

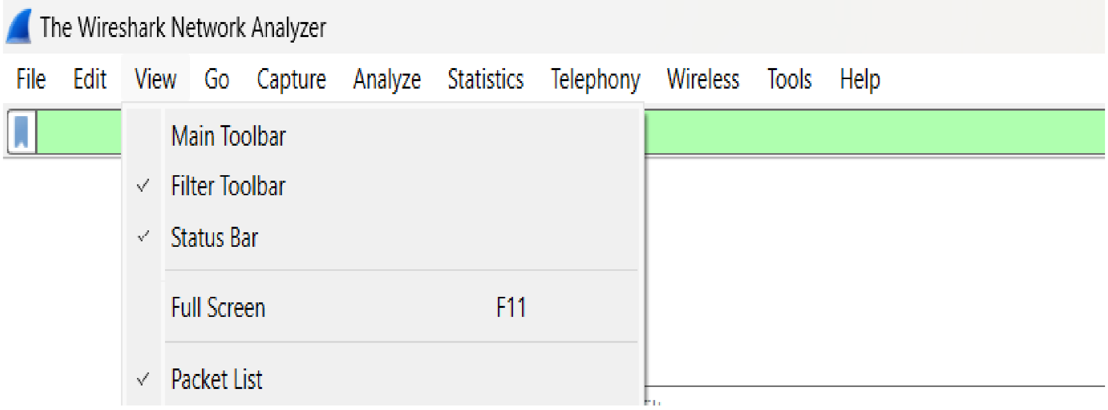


Fig: Packet list pane

Packet Details Pane:

When you select a packet from the Packet List, this pane provides an expandable tree view of the packet's structure. It breaks down the packet into its various protocol layers, such as Ethernet, IP, TCP/UDP, etc., offering detailed protocol information for analysis.

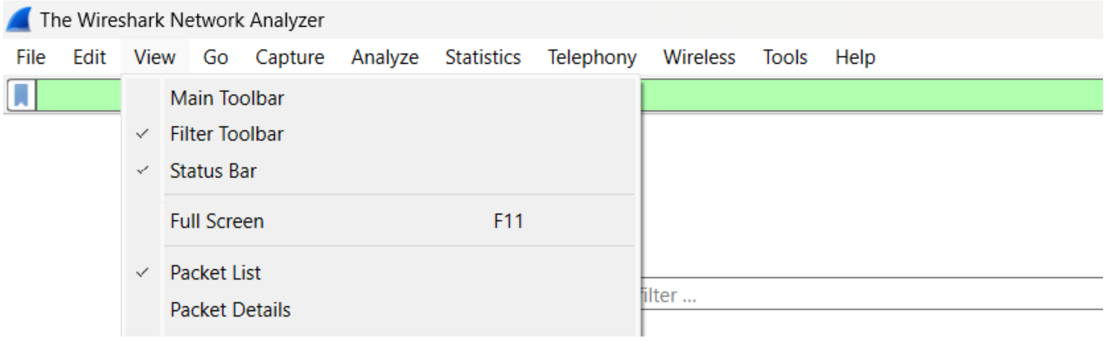


Fig: Packet Details pane

Packet Bytes Pane:

This pane displays the raw data of the selected packet in hexadecimal and ASCII format. It allows users to view the actual bytes transmitted, which can be useful for low-level protocol analysis or detecting anomalies at the byte level.

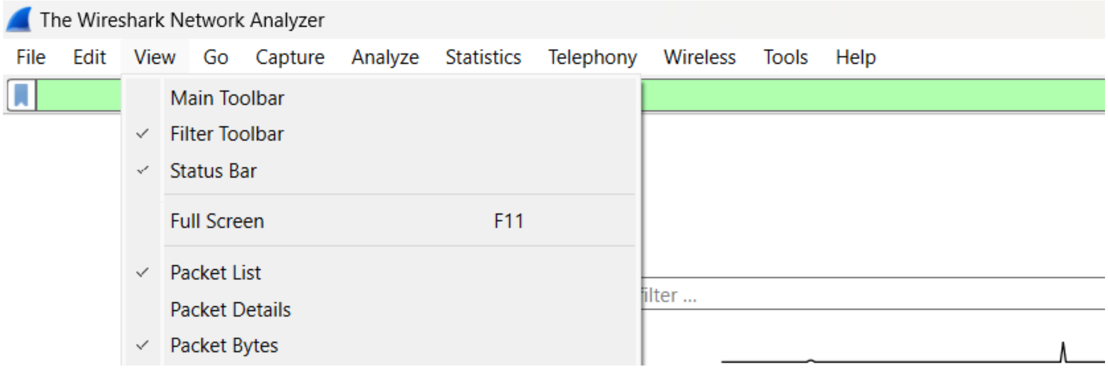


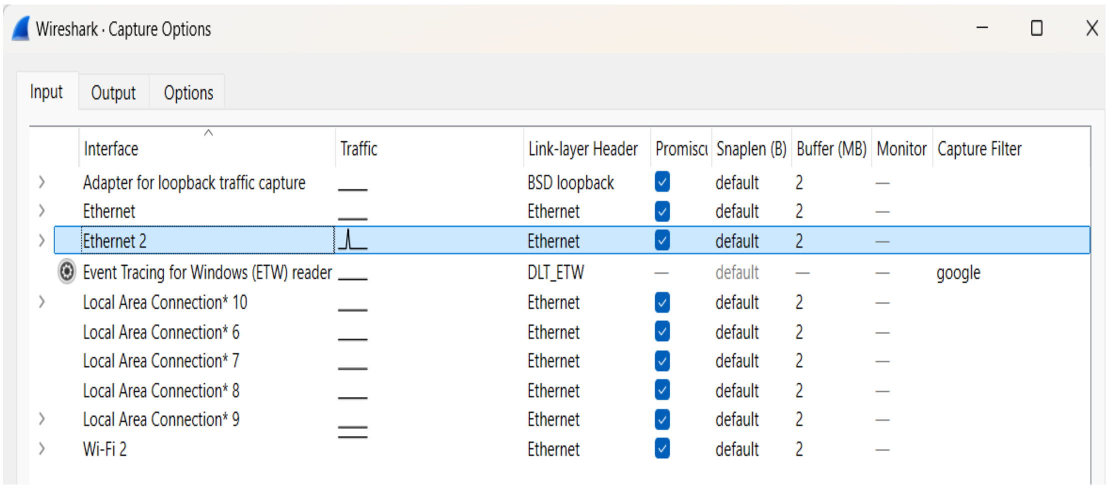
Fig: Packet Bytes pane

Basic Network Capture and Analysis

Selecting a Network Interface

Steps:

1. Open Wireshark.
2. You’ll see a list of available network interfaces (Wi-Fi, Ethernet, etc.).
3. Look for the interface capturing active traffic (typically with a moving graph or traffic count).
4. Select the desired interface by clicking on it.



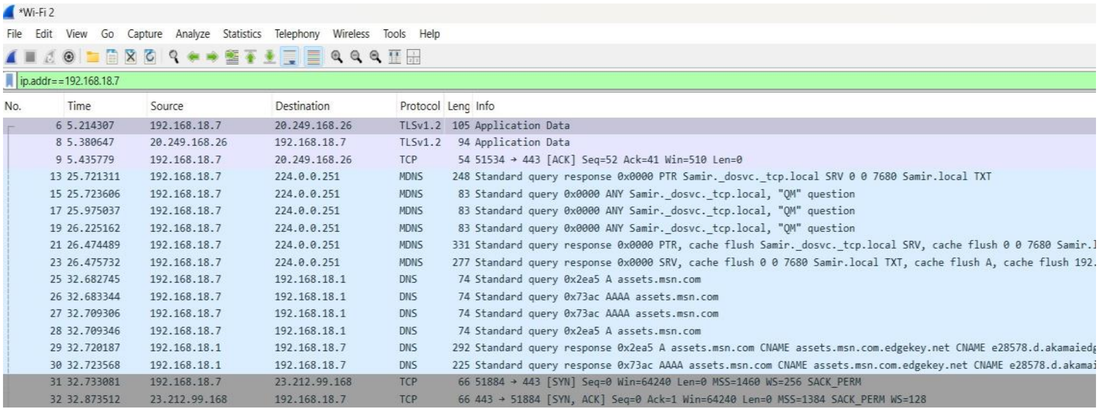
Interface	Traffic	Link-layer Header	Promiscu	Snaplen (B)	Buffer (MB)	Monitor	Capture Filter
Adapter for loopback traffic capture		BSD loopback	☑	default	2	—	
Ethernet		Ethernet	☑	default	2	—	
Ethernet 2		Ethernet	☑	default	2	—	
Event Tracing for Windows (ETW) reader		DLT_ETW	—	default	—	—	google
Local Area Connection* 10		Ethernet	☑	default	2	—	
Local Area Connection* 6		Ethernet	☑	default	2	—	
Local Area Connection* 7		Ethernet	☑	default	2	—	
Local Area Connection* 8		Ethernet	☑	default	2	—	
Local Area Connection* 9		Ethernet	☑	default	2	—	
Wi-Fi 2		Ethernet	☑	default	2	—	

Fig: Selecting network interface

Starting Packet Capture

Steps:

1. After selecting the network interface, click the blue shark fin icon in the toolbar to start the packet capture,
2. Open your browser and navigate to any like (example.com),
3. Wireshark will start capturing all network traffic on the selected interface,
4. To filter the capture for packets related to example.com, you can use a display filter,
5. In the filter bar, type ip.address == <ip address of src>,
6. Press Enter to apply the filter.



ip.addr==192.168.18.7

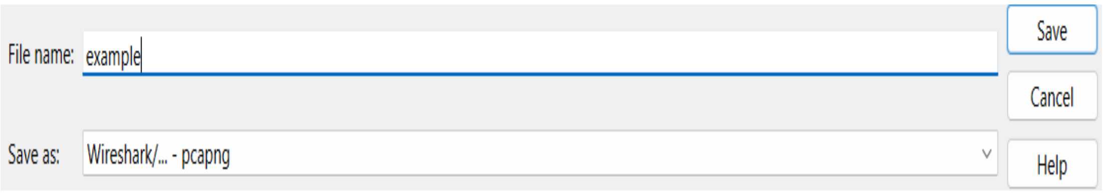
No.	Time	Source	Destination	Protocol	Leng	Info
6	5.214307	192.168.18.7	20.249.168.26	TLSv1.2	105	Application Data
8	5.380647	20.249.168.26	192.168.18.7	TLSv1.2	94	Application Data
9	5.435779	192.168.18.7	20.249.168.26	TCP	54	51534 → 443 [ACK] Seq=52 Ack=41 Win=510 Len=0
13	25.721311	192.168.18.7	224.0.0.251	MDNS	248	Standard query response 0x0000 PTR Samir._dosvc._tcp.local SRV 0 0 7680 Samir.local TXT
15	25.723606	192.168.18.7	224.0.0.251	MDNS	83	Standard query 0x0000 ANY Samir._dosvc._tcp.local, "QM" question
17	25.975037	192.168.18.7	224.0.0.251	MDNS	83	Standard query 0x0000 ANY Samir._dosvc._tcp.local, "QM" question
19	26.225162	192.168.18.7	224.0.0.251	MDNS	83	Standard query 0x0000 ANY Samir._dosvc._tcp.local, "QM" question
21	26.474489	192.168.18.7	224.0.0.251	MDNS	331	Standard query response 0x0000 PTR, cache flush Samir._dosvc._tcp.local SRV, cache flush 0 0 7680 Samir.l
23	26.475732	192.168.18.7	224.0.0.251	MDNS	277	Standard query response 0x0000 SRV, cache flush 0 0 7680 Samir.local TXT, cache flush A, cache flush 192.
25	32.682745	192.168.18.7	192.168.18.1	DNS	74	Standard query 0x2ea5 A assets.msn.com
26	32.683344	192.168.18.7	192.168.18.1	DNS	74	Standard query 0x73ac AAAA assets.msn.com
27	32.709306	192.168.18.7	192.168.18.1	DNS	74	Standard query 0x73ac AAAA assets.msn.com
28	32.709346	192.168.18.7	192.168.18.1	DNS	74	Standard query 0x2ea5 A assets.msn.com
29	32.720187	192.168.18.1	192.168.18.7	DNS	292	Standard query response 0x2ea5 A assets.msn.com CNAME assets.msn.com.edgekey.net CNAME e28578.d.akamaiedg
30	32.723568	192.168.18.1	192.168.18.7	DNS	225	Standard query response 0x73ac AAAA assets.msn.com CNAME assets.msn.com.edgekey.net CNAME e28578.d.akamai
31	32.733081	192.168.18.7	23.212.99.168	TCP	66	51884 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
32	32.873512	23.212.99.168	192.168.18.7	TCP	66	443 → 51884 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1384 SACK_PERM WS=128

Fig: Packet Capture

Stopping and Saving Captures

Steps:

1. Once you’ve captured the desired traffic, click the red square icon in the toolbar to stop the capture,
2. To save the capture: Go to File > SaveAs,
3. Choose a file name and location,
4. Select a format (default is .pcapng), then click Save.



File name: example

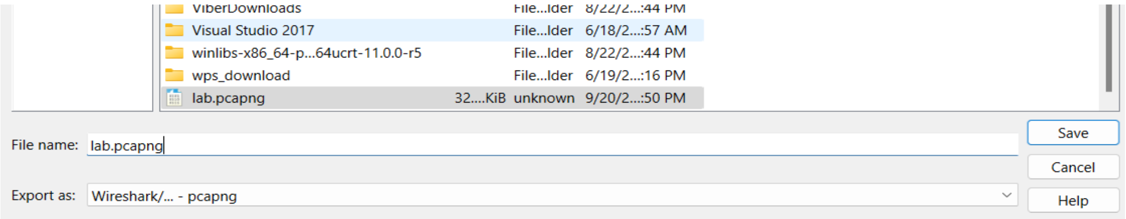
Save as: Wireshark/... - pcapng

Fig: Saving a file

Exporting the Captured Data

Steps:

1. To export specific packets or data: Go to File > Export SpecifiedPackets.
2. Choose the range or filter for the packets you want to export.



ViberDownloads		File...lder	8/22/2...:44 PM
Visual Studio 2017		File...lder	6/18/2...:57 AM
winlibs-x86_64-p...64ucrt-11.0.0-r5		File...lder	8/22/2...:44 PM
wps_download		File...lder	6/19/2...:16 PM
lab.pcapng	32....KiB	unknown	9/20/2...:50 PM

File name: lab.pcapng

Export as: Wireshark/... - pcapng

Fig: Exporting Capture Data

Conclusion:

In this lab, we worked with Wireshark, a powerful open-source network protocol analyzer, to capture, analyze, and troubleshoot network traffic. By using its filtering features and examining packet details, we gained a clear understanding of how protocols like TCP, UDP, and IP function. This hands-on experience helped us identify network performance issues and security risks, reinforcing Wireshark's value as an essential tool for network administrators and cybersecurity professionals in ensuring efficiency.

Lab 14: Packet Capture and Header Analysis by Wireshark (TCP, UDP, IP)

Theory

Wireshark is a popular open-source tool used to capture and analyze network traffic in real time. It allows users to examine data packets, making it highly effective for troubleshooting, monitoring network performance, and identifying security vulnerabilities. With its easy-to-use graphical interface and support for a wide variety of protocols, Wireshark provides in-depth insights into network communications. Its ability to dissect packets across multiple layers helps users better understand network behavior. Whether for resolving network issues or improving security, Wireshark is an essential resource for network administrators and cybersecurity experts, playing a key role in maintaining network efficiency.

TCP (Transmission Control Protocol):
TCP (Transmission Control Protocol) is a connection-oriented protocol that ensures reliable data transmission between devices. It establishes a connection before data transfer, providing error checking and ensuring packets are delivered in order and without duplication. This reliability makes TCP ideal for applications requiring consistent data delivery, such as web browsing (HTTP/HTTPS) and email, making it a fundamental protocol for internet communications.

UDP (User Datagram Protocol):
UDP (User Datagram Protocol) is a connectionless protocol that prioritizes speed over reliability. It transmits data packets without establishing a connection or guaranteeing delivery, making it ideal for applications like video streaming, online gaming, and VoIP. In these scenarios, low latency is crucial, and occasional data loss is acceptable.

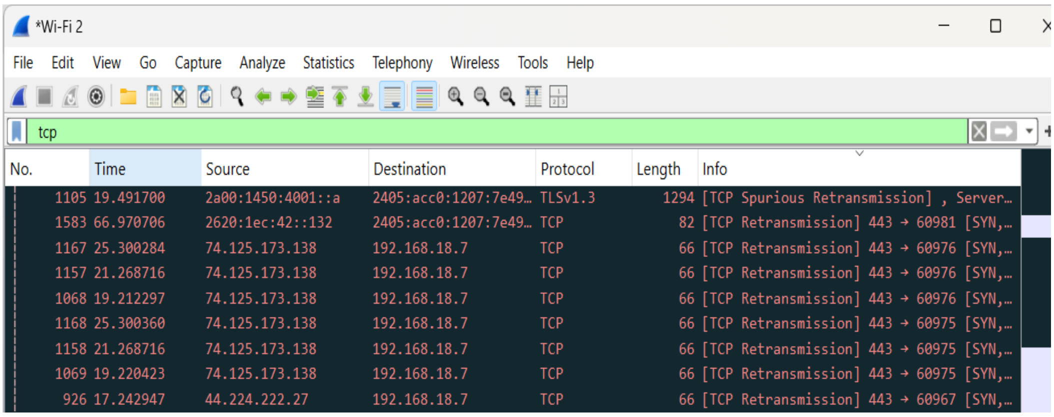
IP (Internet Protocol):
IP (Internet Protocol) is the primary protocol for routing packets across networks at the network layer. It assigns IP addresses to devices and ensures data packets are forwarded to the correct destinations through routing. Working in tandem with TCP and UDP, IP facilitates end-to-end communication over the internet, making it essential for network connectivity.

Network Interface Selection and Traffic Filtering

Steps:

1. Open Wireshark and select the network interface (Wi-Fi or Ethernet) where traffic is to be captured,
2. Click the start button to begin capturing live traffic,

3. Apply a filter to focus on specific traffic, such as tcp for TCP traffic, udp for UDP, or IP for general IP traffic.



```
tcp
No.   Time       Source            Destination         Protocol  Length Info
1105 19.491700   2a00:1450:4001::a 2405:acc0:1207:7e49… TLSv1.3   1294 [TCP Spurious Retransmission] , Server…
1583 66.970706   2620:1ec:42::132  2405:acc0:1207:7e49… TCP         82 [TCP Retransmission] 443 → 60981 [SYN,…
1167 25.300284   74.125.173.138    192.168.18.7         TCP         66 [TCP Retransmission] 443 → 60976 [SYN,…
1157 21.268716   74.125.173.138    192.168.18.7         TCP         66 [TCP Retransmission] 443 → 60976 [SYN,…
1068 19.212297   74.125.173.138    192.168.18.7         TCP         66 [TCP Retransmission] 443 → 60976 [SYN,…
1168 25.300360   74.125.173.138    192.168.18.7         TCP         66 [TCP Retransmission] 443 → 60975 [SYN,…
1158 21.268716   74.125.173.138    192.168.18.7         TCP         66 [TCP Retransmission] 443 → 60975 [SYN,…
1069 19.220423   74.125.173.138    192.168.18.7         TCP         66 [TCP Retransmission] 443 → 60975 [SYN,…
 926 17.242947   44.224.222.27     192.168.18.7         TCP         66 [TCP Retransmission] 443 → 60967 [SYN,…
```

Fig: Traffic filtering

TCP Header Analysis

After capturing TCP traffic, select a TCP packet to view its header details, which include:

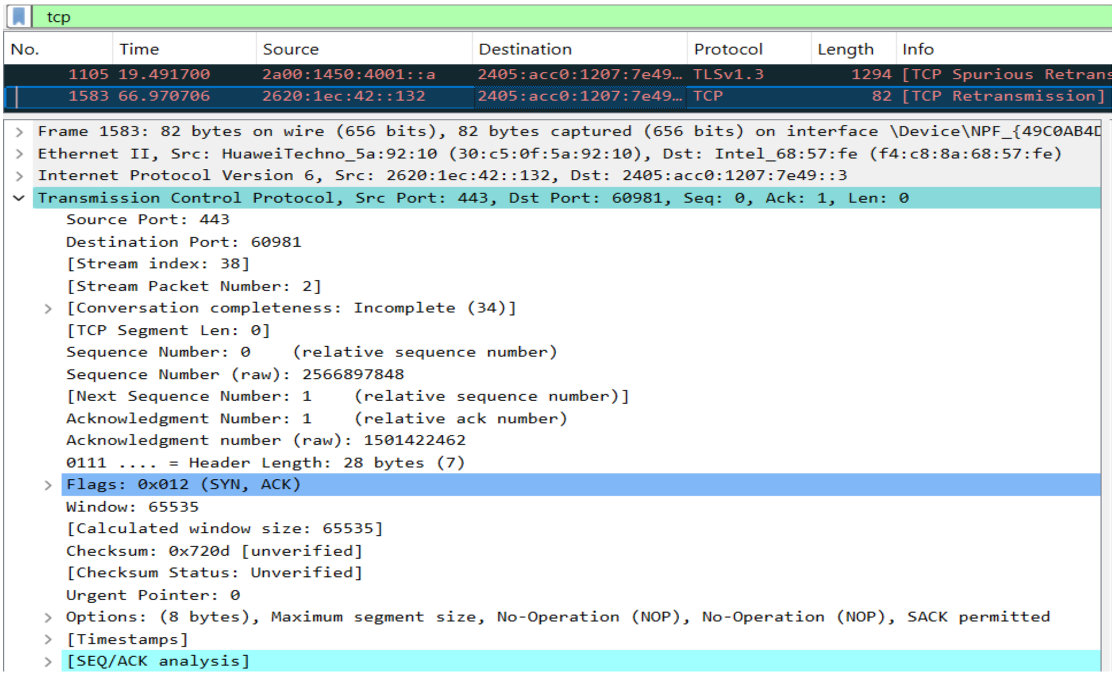
Source Port: Identifies the port on the sender's machine (e.g., port 443 for HTTPS).

Destination Port: Specifies the port on the recipient's machine.

Sequence Number: Keeps track of the packet's position in the communication stream.

Acknowledgment Number: Confirms the receipt of previous packets.

Flags: Control bits (e.g., SYN, ACK, FIN) used to manage the connection's state.



```
tcp
No.   Time       Source            Destination          Protocol  Length Info
1105 19.491700   2a00:1450:4001::a 2405:acc0:1207:7e49… TLSv1.3   1294 [TCP Spurious Retrans
1583 66.970706   2620:1ec:42::132  2405:acc0:1207:7e49… TCP         82 [TCP Retransmission]

> Frame 1583: 82 bytes on wire (656 bits), 82 bytes captured (656 bits) on interface \Device\NPF_{49C0AB4D
> Ethernet II, Src: HuaweiTechno_5a:92:10 (30:c5:0f:5a:92:10), Dst: Intel_68:57:fe (f4:c8:8a:68:57:fe)
> Internet Protocol Version 6, Src: 2620:1ec:42::132, Dst: 2405:acc0:1207:7e49::3
v Transmission Control Protocol, Src Port: 443, Dst Port: 60981, Seq: 0, Ack: 1, Len: 0
    Source Port: 443
    Destination Port: 60981
    [Stream index: 38]
    [Stream Packet Number: 2]
  > [Conversation completeness: Incomplete (34)]
    [TCP Segment Len: 0]
    Sequence Number: 0    (relative sequence number)
    Sequence Number (raw): 2566897848
    [Next Sequence Number: 1    (relative sequence number)]
    Acknowledgment Number: 1    (relative ack number)
    Acknowledgment number (raw): 1501422462
    0111 .... = Header Length: 28 bytes (7)
  > Flags: 0x012 (SYN, ACK)
    Window: 65535
    [Calculated window size: 65535]
    Checksum: 0x720d [unverified]
    [Checksum Status: Unverified]
    Urgent Pointer: 0
  > Options: (8 bytes), Maximum segment size, No-Operation (NOP), No-Operation (NOP), SACK permitted
  > [Timestamps]
  > [SEQ/ACK analysis]
```

Fig: TCP header analysis of selected packet

TCP Header Analysis Result:
From above figure of TCP header analysis, we can deduce the following details for the website youtube.com

SN	Parameters	Details
1	Source Port	443
2	Destination Port	60981
3	Sequence Number	0
4	Acknowledgment Number	1
5	Flags	SYN, ACK

Fig: TCP header analysis details table

UDP Header Analysis

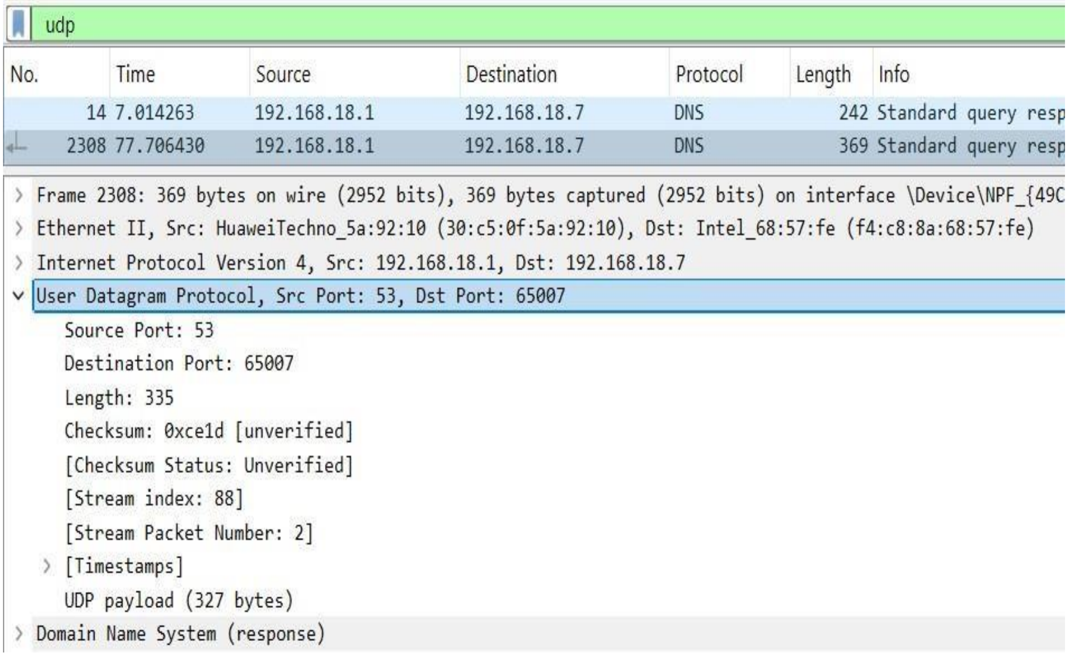
Select a UDP packet and analyze its header:

Source Port: The port on the sender's side.

Destination Port: The port on the receiver's side.

Length: Indicates the size of the UDP packet, including the header and data.

Checksum: A verification field for ensuring data integer.



```
udp

No.     Time        Source         Destination    Protocol  Length  Info
     14 7.014263    192.168.18.1   192.168.18.7   DNS          242  Standard query resp
   2308 77.706430   192.168.18.1   192.168.18.7   DNS          369  Standard query resp

> Frame 2308: 369 bytes on wire (2952 bits), 369 bytes captured (2952 bits) on interface \Device\NPF_{49C
> Ethernet II, Src: HuaweiTechno_5a:92:10 (30:c5:0f:5a:92:10), Dst: Intel_68:57:fe (f4:c8:8a:68:57:fe)
> Internet Protocol Version 4, Src: 192.168.18.1, Dst: 192.168.18.7
v User Datagram Protocol, Src Port: 53, Dst Port: 65007
    Source Port: 53
    Destination Port: 65007
    Length: 335
    Checksum: 0xce1d [unverified]
    [Checksum Status: Unverified]
    [Stream index: 88]
    [Stream Packet Number: 2]
  > [Timestamps]
    UDP payload (327 bytes)
> Domain Name System (response)
```

Fig: UDP header analysis of selected packet

UDP Header Analysis Result:
From above figure of UDP header analysis, we can deduce the following details for the website youtube.com

SN	Parameters	Details
1	Source Port	53
2	Destination Port	65007
3	Length	335
4	Checksum	0xce1d
5	Stream Index	88
6	Stream Packet Number	2

Fig: UDP header analysis details table

IP Header Analysis

For IP packet analysis, the following fields are important:

Source IP: The sender's IP address.

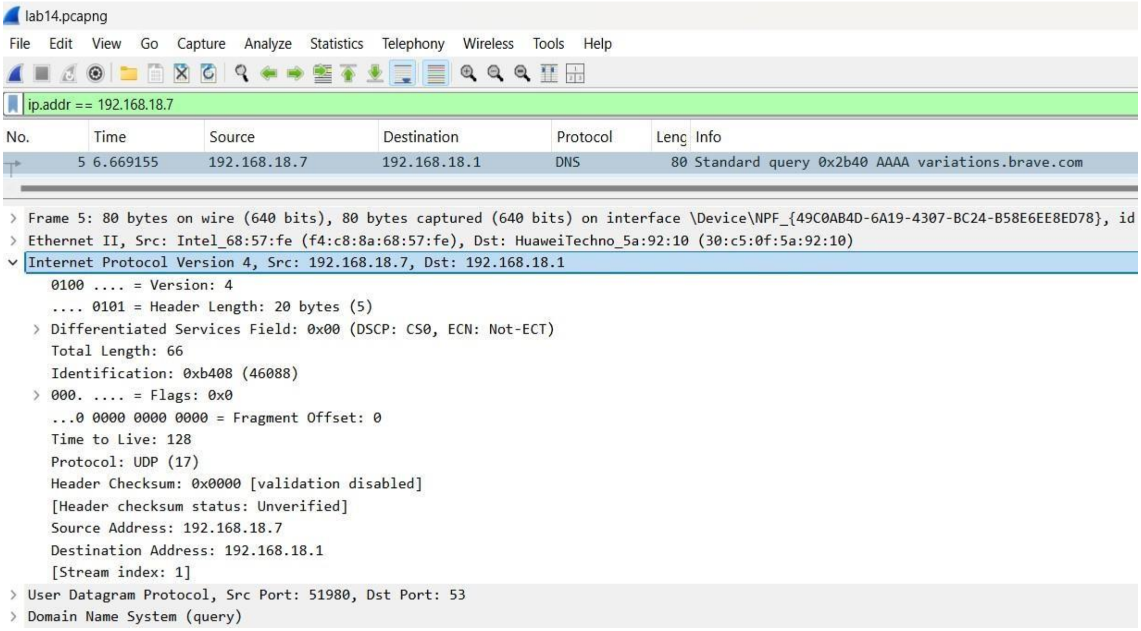
Destination IP: The receiver's IP address.

Header Length: Indicates the size of the IP header.

TTL (Time to Live): Limits the lifespan of the packet, decremented by each router.

Protocol: Specifies whether TCP, UDP, or another protocol is being used.

Fragmentation: If the packet is fragmented, this field shows relevant information.


lab14.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

ip.addr == 192.168.18.7

```
No.   Time       Source          Destination     Protocol  Leng Info
    5 6.669155   192.168.18.7    192.168.18.1    DNS         80 Standard query 0x2b40 AAAA variations.brave.com
```

```
> Frame 5: 80 bytes on wire (640 bits), 80 bytes captured (640 bits) on interface \Device\NPF_{49C0AB4D-6A19-4307-BC24-B58E6EE8ED78}, id
> Ethernet II, Src: Intel_68:57:fe (f4:c8:8a:68:57:fe), Dst: HuaweiTechno_5a:92:10 (30:c5:0f:5a:92:10)
v Internet Protocol Version 4, Src: 192.168.18.7, Dst: 192.168.18.1
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 66
    Identification: 0xb408 (46088)
  > 000. .... = Flags: 0x0
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 128
    Protocol: UDP (17)
    Header Checksum: 0x0000 [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 192.168.18.7
    Destination Address: 192.168.18.1
    [Stream index: 1]
> User Datagram Protocol, Src Port: 51980, Dst Port: 53
> Domain Name System (query)
```

Fig: IP header analysis of selected packet

IP Header Analysis Result:

From above figure of IP header analysis, we can deduce the following details for the website youtube.com

SN	Parameters	Details
1	Source address	192.168.18.7
2	Destination address	192.168.18.1
3	Length	66
4	Time to Live	128
5	Header Checksum	0x0000
6	Stream Index	1
7	Protocol	UDP (17)

Fig: IP header analysis details table

Conclusion:

In this lab, we effectively used Wireshark to capture packets and analyze the headers of TCP, UDP, and IP protocols. By closely examining packet headers, we gained important insights into data flow and communication within networks. This analysis deepened our understanding of network behavior and plays a crucial role in troubleshooting and maintaining secure, efficient network operations. Wireshark proved to be an essential tool for both network administrators and cybersecurity professionals.