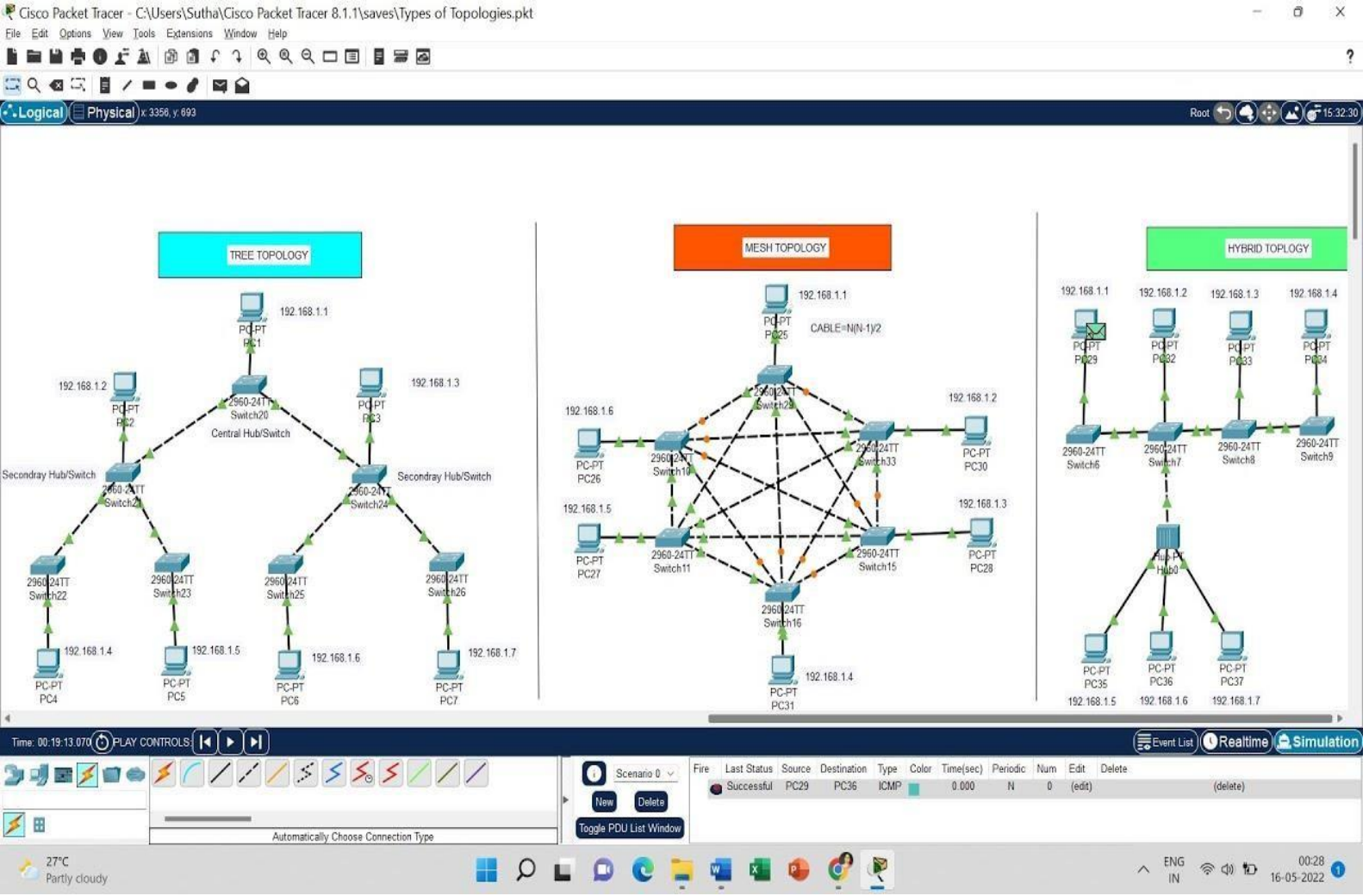
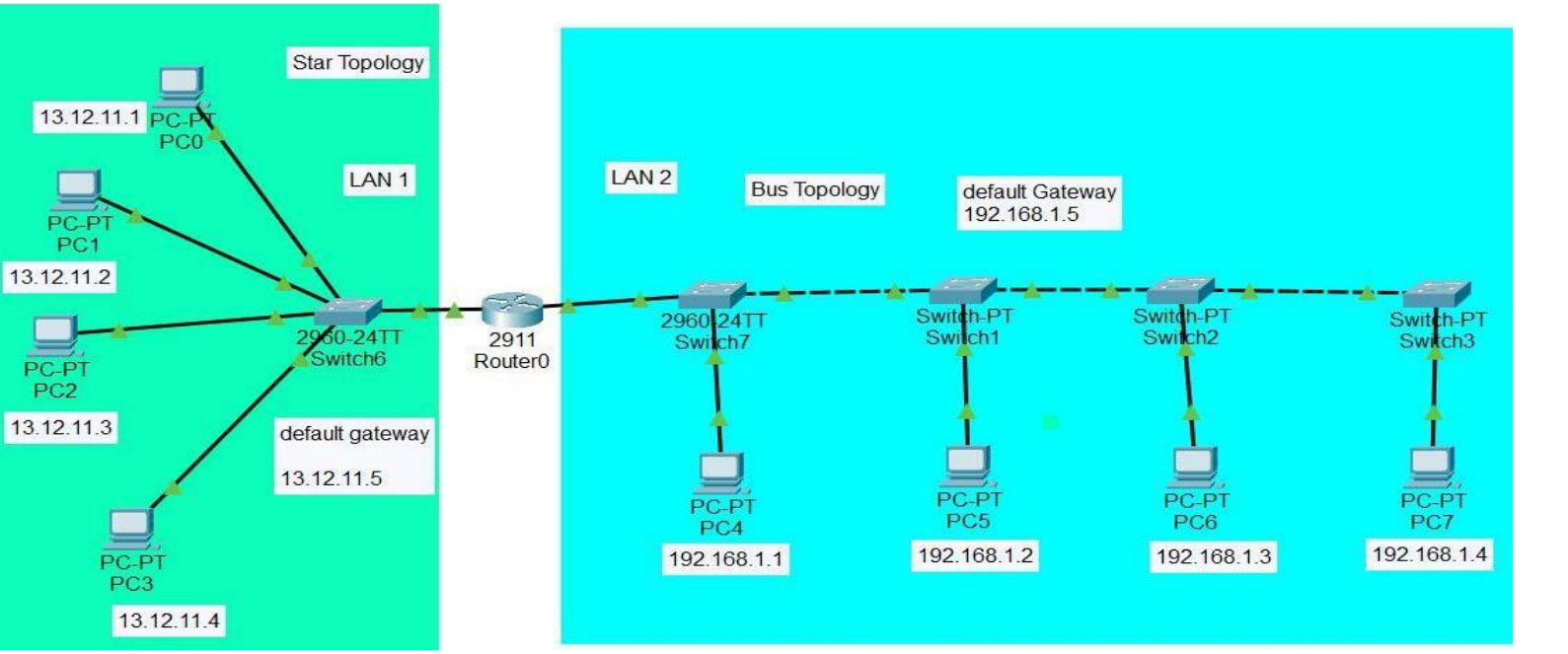
Practical No. 01

Name: Tavhare Ruchita Sharad

Roll No.:58





Steps to Run:

1. Star Topology

- Step 1: Open Packet Tracer.
- Step 2: Drag a Switch into the workspace.
- Step 3: Drag PCs (at least 3) into the workspace.
- Step 4: Use Copper Straight-Through cables to connect each PC to the switch.

---

2. Bus Topology

- Step 1: Place a Hub in the workspace.
- Step 2: Drag multiple PCs into the workspace.
- Step 3: Use Copper Straight-Through cables to connect each PC to the hub.

---

3. Ring Topology

- Step 1: Place multiple Switches in a circular arrangement.
- Step 2: Connect each switch using Copper Cross-Over cables in a ring.
- Step 3: Connect PCs to each switch.

---

4. Mesh Topology

- Step 1: Place several PCs and Routers in the workspace.
- Step 2: Use Copper Cross-Over or Fiber Optic cables to connect each PC to multiple other PCs.

---

5. Hybrid Topology (Star-Bus)

- Step 1: Create multiple star topologies using Switches and PCs.
- Step 2: Use Copper Cross-Over or Fiber Optic cables to connect the switches.

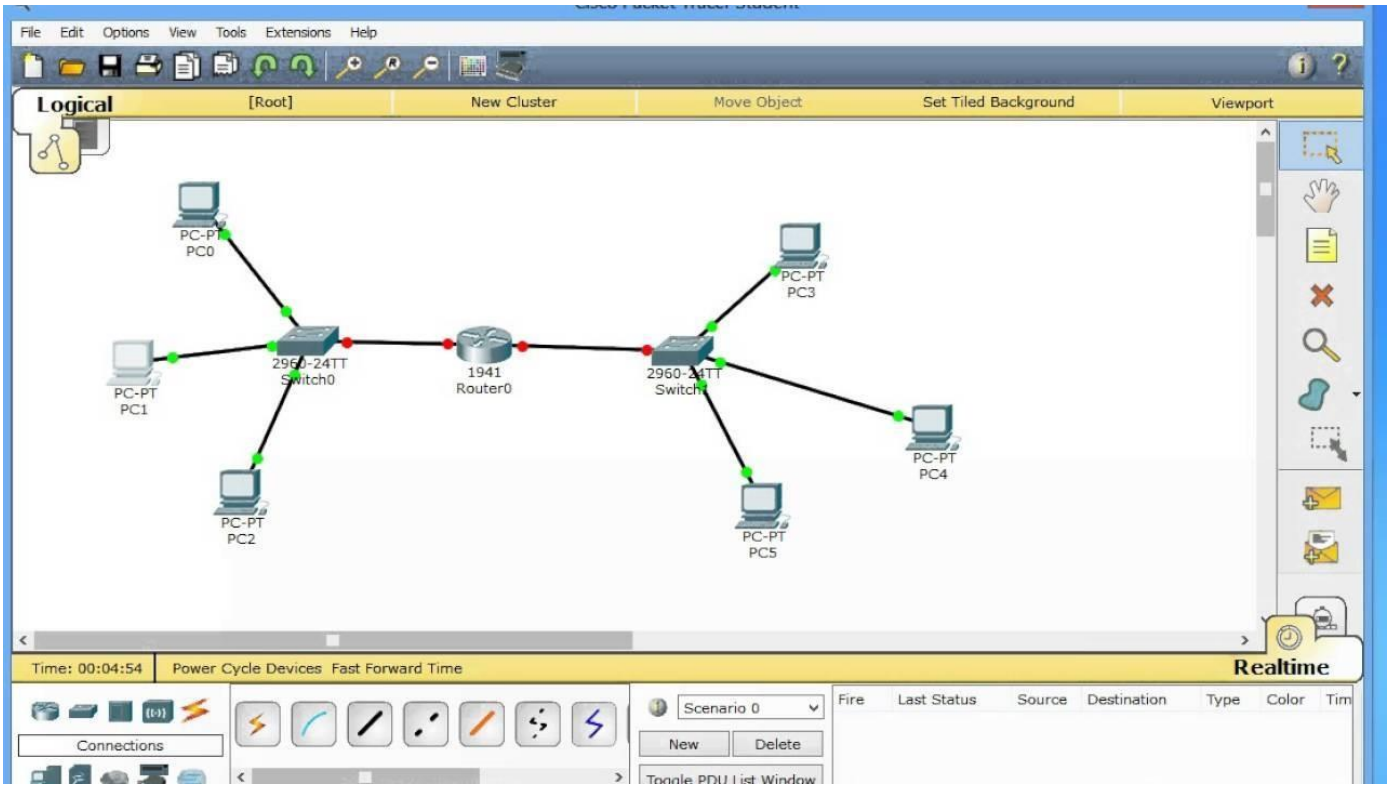
Transmission Media

- Step 1: Use Copper Straight-Through for PC to switch or switch to router.
- Step 2: Use Copper Cross-Over for switch to switch or router to router.
- Step 3: Use Fiber Optic cables for high-speed connections between switches or routers.
- Step 4: Use Wireless Devices (Access Points) for wireless connections between PCs and access points.

Practical No. 02

Name: Tavhare Ruchita Sharad

Roll No.:58



Steps To Run:

Step 1: Create Network Topology in Packet Tracer

- Step 1.1: Open Cisco Packet Tracer.
- Step 1.2: Drag a Layer 2 Switch (e.g., 2960) into the workspace.
- Step 1.3: Drag 2 PCs into the workspace.
- Step 1.4: Use Copper Straight-Through cables to connect each PC to the switch:
    - Connect PC1's FastEthernet0 to Switch0's FastEthernet0/1.
    - Connect PC2's FastEthernet0 to Switch0's FastEthernet0/2.

Step 2: Assign IP Addresses to PCs

- Step 2.1: Click on PC1, go to the Desktop tab, and select IP Configuration.
- Step 2.2: Assign the following IP details:
    - IP Address: 192.168.1.2
    - Subnet Mask: 255.255.255.0
- Step 2.3: Do the same for PC2 and assign:
    - IP Address: 192.168.1.3

- Subnet Mask: 255.255.255.0

Step 3: Test Connectivity Using PING Utility

- Step 3.1: Click on PC1, open the Command Prompt from the Desktop tab.
- Step 3.2: Test the connection by typing:

Code:

```
ping 192.168.1.3
```

- You should receive Reply from 192.168.1.3 confirming successful communication.

Step 4: Capture Ping Packets Using Simulation Mode in Packet Tracer

- Step 4.1: Switch to Simulation Mode by clicking the Stopwatch icon in the bottom-right corner.
- Step 4.2: Click on PC1 and go to the Command Prompt. Again, type:

Code:

```
ping 192.168.1.3
```

- Step 4.3: Observe the packets being sent across the network in the simulation window.
- Step 4.4: Click on any ICMP packet to view details such as Source, Destination, and the packet journey.

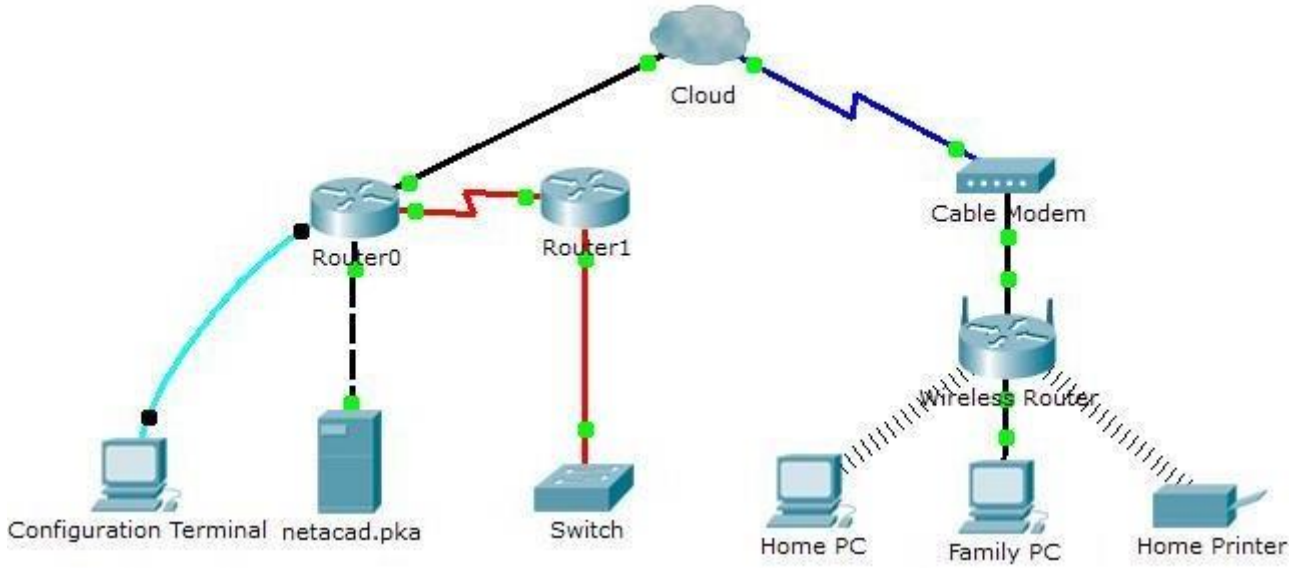
Step 5: Analysing Packet Details

- Step 5.1: After the packet is captured, click on the Event List in the Simulation panel.
- Step 5.2: Click on any of the packets in the list to see the detailed information, including:
  - ICMP Echo Request
  - ICMP Echo Reply
  - Source and Destination IPs

Practical No. 03

Name: Tavhare Ruchita Sharad

Roll No.:58



Steps to run:

Step 1: Create Wired LAN (LAN1)

- Step 1.1: Open Cisco Packet Tracer.
- Step 1.2: Drag a Layer 2 Switch (2960) into the workspace.
- Step 1.3: Drag 2 PCs into the workspace.
- Step 1.4: Use Copper Straight-Through cables to connect each PC to the switch:
    - Connect PC1's FastEthernet0 to Switch0's FastEthernet0/1.
    - Connect PC2's FastEthernet0 to Switch0's FastEthernet0/2.

Step 2: Create Wireless LAN (LAN2)

- Step 2.1: Drag a Wireless Router into the workspace (e.g., Linksys WRT300N).
- Step 2.2: Drag 2 Laptops into the workspace.
- Step 2.3: Configure the wireless router:
    - Click the Wireless Router, go to GUI or Config tab, and set SSID (e.g., WirelessLAN).
    - Enable DHCP to assign IP addresses to wireless clients automatically.
    - Set the Wireless Security (e.g., WPA2 Personal) and define a Password.
- Step 2.4: Configure the laptops:
    - Click on Laptop1, go to the Desktop tab, and select PC Wireless.
    - Click on Connect to select the WirelessLAN network and enter the password.

- Repeat this for Laptop2.

Step 3: Interconnect Wired and Wireless LAN

- Step 3.1: Drag a Router (e.g., 1841 or 2911) into the workspace.
- Step 3.2: Connect the Wired LAN (Switch) to the Router:
  - Use a Copper Straight-Through cable to connect the Switch's FastEthernet0/24 to the Router's FastEthernet0/0.
- Step 3.3: Connect the Wireless Router to the Router:
  - Use a Copper Straight-Through cable to connect the Wireless Router's Internet port to the Router's FastEthernet0/1.

Step 4: Configure IP Addresses

- Step 4.1: Configure the Router interfaces:
  - Click on the Router, go to the Config tab.
  - Assign IP Addresses to the Router interfaces:
    - FastEthernet0/0: 192.168.1.1/24 (for Wired LAN)
    - FastEthernet0/1: 192.168.2.1/24 (for Wireless LAN)
- Step 4.2: Assign Static IP addresses to the PCs in LAN1 (Wired):
  - On PC1: 192.168.1.2, Subnet Mask 255.255.255.0, Default Gateway: 192.168.1.1.
  - On PC2: 192.168.1.3, Subnet Mask 255.255.255.0, Default Gateway: 192.168.1.1.
- Step 4.3: The Wireless Router will automatically assign IP addresses to the Laptops in LAN2 (Wireless) using DHCP.
  - Laptops will have IP addresses in the range 192.168.2.x with the Default Gateway as 192.168.2.1.

Step 5: Configure Routing on the Router

- Step 5.1: Go to the Router's CLI.
- Step 5.2: Enable routing between the wired and wireless LANs by configuring static routes:
  - Type the following commands:

Code:

```
enable
configure terminal
interface FastEthernet0/0
ip address 192.168.1.1 255.255.255.0
no shutdown
```

exit

interface FastEthernet0/1

ip address 192.168.2.1 255.255.255.0

no shutdown

exit

end

Step 6: Test Connectivity Using PING Utility

- Step 6.1: On PC1 (Wired LAN), open the Command Prompt and ping Laptop1 in Wireless LAN:

Code:

ping 192.168.2.x # (Laptop1's IP address)

- Step 6.2: On Laptop1 (Wireless LAN), open the Command Prompt and ping PC1 in Wired LAN:

Code:

ping 192.168.1.2 # (PC1's IP address)

You should receive a successful reply in both cases, demonstrating communication between the two LANs.

Step 7: Demonstrate Packet Transfer in Simulation Mode

- Step 7.1: Switch to Simulation Mode in Packet Tracer by clicking the Stopwatch icon in the bottom-right corner.
- Step 7.2: Send a ping from PC1 to Laptop1 and observe the packet flow.
- Step 7.3: View the detailed packet journey as it travels from the Wired LAN1 (Switch) through the Router to Wireless LAN2 (Wireless Router).

Practical No. 04

Name: Tavhare Ruchita Sharad

Roll No.:58

Code:

```
import ipaddress
import math

def calculate_subnet_mask(ip_address, num_subnets):
    # Convert IP address to an IPv4Network object
    ip_network = ipaddress.IPv4Network(ip_address, strict=False)

    # Get the base subnet mask in CIDR notation
    base_mask = ip_network.prefixlen

    # Calculate the new subnet mask by finding how many additional bits are needed
    # log2(num_subnets) gives the number of bits to borrow for subnetting
    bits_to_borrow = math.ceil(math.log2(num_subnets))
    new_mask = base_mask + bits_to_borrow

    # Calculate the new subnet mask in dotted decimal notation
    subnet_mask = ipaddress.IPv4Network(f'0.0.0.0/{new_mask}').netmask

    # Calculate the number of subnets and hosts per subnet
    num_hosts_per_subnet = 2**(32 - new_mask) - 2  # Minus 2 for network and broadcast address

    return new_mask, subnet_mask, num_hosts_per_subnet

def main():
    print("Subnetting Demonstration Program")

    # Input IP address and the number of subnets
    ip_address = input("Enter the IP address with CIDR (e.g., 192.168.1.0/24): ")
    num_subnets = int(input("Enter the number of subnets required: "))
```

```
    # Calculate subnet mask and related information
    new_mask, subnet_mask, num_hosts_per_subnet = calculate_subnet_mask(ip_address, num_subnets)

    # Display the results
    print(f"\nCalculated Subnet Mask Information:")
    print(f"Original IP Address and CIDR: {ip_address}")
    print(f"New Subnet Mask (CIDR Notation): /{new_mask}")
    print(f"New Subnet Mask (Dotted Decimal Notation): {subnet_mask}")
    print(f"Number of Hosts per Subnet: {num_hosts_per_subnet}")

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

Output:

```
Subnetting Demonstration Program
Enter the IP address with CIDR (e.g., 192.168.1.0/24): 192.168.1.0/24
Enter the number of subnets required: 4
Calculated Subnet Mask Information:
Original IP Address and CIDR: 192.168.1.0/24
New Subnet Mask (CIDR Notation): /26
New Subnet Mask (Dotted Decimal Notation): 255.255.255.192
Number of Hosts per Subnet: 62
```

Practical No. 05

Name: Tavhare Ruchita Sharad

Roll No.:58

Steps :
A. Say Hello to Each Other

Server Code:

1. Socket Creation:
    - socket.socket(socket.AF_INET, socket.SOCK_STREAM) creates a TCP socket.
2. Binding:
    - server_socket.bind(('0.0.0.0', 12345)) binds the server to any available network interface on port 12345.
3. Listening:
    - server_socket.listen(1) puts the server in listening mode, waiting for incoming connections.
4. Accept Connection:
    - client_socket, addr = server_socket.accept() waits for a client to connect and accepts the connection.
5. Message Exchange:
    - The server sends a greeting message to the client using client_socket.send().
    - It then receives a message from the client using client_socket.recv() and prints it.
6. Cleanup:
    - The sockets are closed after the communication is done.

Client Code:

1. Socket Creation:
    - Similar to the server, it creates a TCP socket.
2. Connecting:
    - client_socket.connect(('127.0.0.1', 12345)) connects to the server at the specified address and port.
3. Message Exchange:
    - It receives a message from the server and prints it.
    - The client sends a greeting back to the server.
4. Cleanup:
    - The client socket is closed after the exchange.

B. File Transfer

Server Code (File Sender):

1. Socket Setup:
    - Similar setup as the "Say Hello" example.
2. File Sending:
    - The server opens a file (file_to_send.txt) in binary read mode ('rb').
    - It reads data from the file in chunks of 1024 bytes and sends it to the client in a loop until all data is sent.
3. Cleanup:
    - The server closes the file and sockets after sending the file.

Client Code (File Receiver):

1. Socket Setup:
    - Connects to the server as in the previous example.
2. File Receiving:
    - It opens a new file (received_file.txt) in binary write mode ('wb').
    - The client receives data in chunks of 1024 bytes in a loop and writes it to the new file until there’s no more data.
3. Cleanup:
    - The client socket is closed after the file transfer is complete.

Output :
A. Say Hello to Each Other

Server Output:

Server is listening...

Connection established with ('127.0.0.1', 54321) # The client's IP and port

Received from client: Hello from Client!

Client Output:

Received from server: Hello from Server!

B. File Transfer

Server Output:

Waiting for connection...

Connected to ('127.0.0.1', 54321)

File sent successfully!

Client Output:

File received successfully!

Practical No. 06

Name: Tavhare Ruchita Sharad

Roll No.:58

Steps :
Server Code Explanation

1. **Import Libraries**:
    - The socket library is imported to facilitate network communication.
2. **Create UDP Socket**:
    - A UDP socket is created using socket.socket(socket.AF_INET, socket.SOCK_DGRAM).
    - The server binds to all available interfaces on port 12345 with server_socket.bind(('0.0.0.0', 12345)).
3. **Receive File**:
    - A file named received_file is opened in binary write mode ('wb').
    - The server enters a loop to continuously receive data using server_socket.recvfrom(1024), which reads data in chunks of 1024 bytes.
    - If the received data is b"END", the loop breaks, indicating the end of transmission.
    - Otherwise, the received data is written to the opened file.
4. **Cleanup**:
    - After receiving all the data, a success message is printed, and the socket is closed with server_socket.close().

Client Code Explanation

1. **Import Libraries**:
    - The socket library is imported to facilitate network communication.
2. **Create UDP Socket**:
    - A UDP socket is created similarly to the server. The server's IP address and port 12345 are specified for the connection.
3. **Send File**:
    - The filename variable is set to the name of the file to be sent (e.g., file_to_send.ext).
    - The file is opened in binary read mode ('rb').
    - The client reads the file in chunks of 1024 bytes and sends each chunk to the server using client_socket.sendto(data, server_address).
    - After sending all data, the client sends a special message b"END" to signal the end of the file transfer.

4. **Cleanup**:
    - A success message is printed after the file transfer is complete, and the socket is closed with client_socket.close().

Output:

Server Output

When you run the server code, the output will be:

Server is ready to receive files...

File received successfully!

Client Output

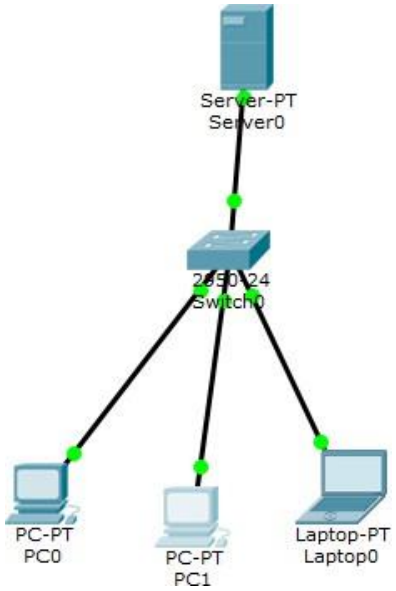
When you run the client code, the output will be:

File sent successfully!

Practical No. 07

Name: Tavhare Ruchita Sharad

Roll No.:58



Steps to run :

1. **Set Up Network**:
    - Open Cisco Packet Tracer.
    - Add a **Router**, **Switch**, multiple **PCs**, and an **HTTP/FTP Server**.
2. **Configure IP Addresses**:
    - Assign unique IP addresses to each PC and the server in the same subnet.
3. **Enable Services on Server**:
    - **HTTP**: Enable in the **Services** tab.
    - **FTP**: Enable in the **Services** tab and set up credentials.
    - (HTTPS might be theoretical in Packet Tracer.)
4. **Test Connectivity**:
    - Use the **Command Prompt** on each PC to ping the server.
5. **Analyze HTTP Performance**:
    - Open a PC's web browser and access the server via its IP.
    - Note response times.
6. **Analyze HTTPS Performance**:
    - Use the web browser on a different PC and access the server with HTTPS.

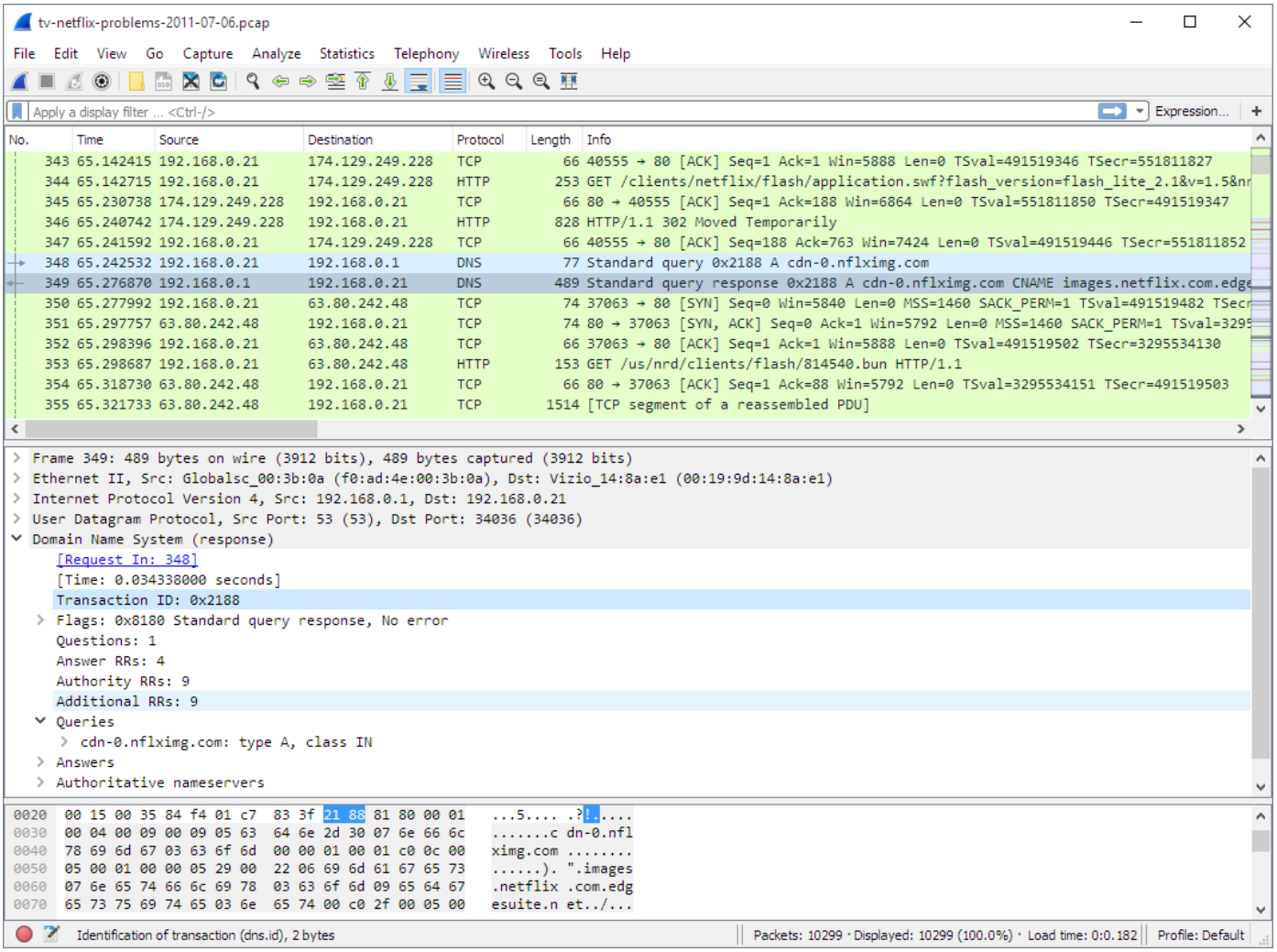
- Note response times.

7. **Analyze FTP Performance**:
   - From a PC, open **Command Prompt** and use the command ftp <server IP>.
   - Log in and transfer a test file; measure transfer times.
8. **Capture Traffic**:
   - Switch to **Simulation Mode** in Packet Tracer.
   - Capture packets during HTTP, HTTPS, and FTP tests for analysis.
9. **Document Findings**:
   - Record performance metrics (response times, transfer rates) for all protocols.
10. **Save Your Work**:
    - Save the Packet Tracer file for future reference.

Practical No. 08

Name: Tavhare Ruchita Sharad

Roll No.:58



Steps to Capture SSL Packets Using Wireshark

1. Install Wireshark:
    - Download and install Wireshark if you haven't already.
2. Open Wireshark:
    - Launch the Wireshark application.
3. Select Network Interface:
    - Choose the appropriate network interface (e.g., Wi-Fi or Ethernet) to capture packets.
    - Click the Start Capturing Packets button.
4. Visit SSL-Secured Website:
    - Open a web browser and navigate to an SSL-secured website (e.g., your bank or an e-commerce site).
    - Ensure the URL starts with https://.
5. Stop Packet Capture:

- After the website loads, return to Wireshark and click the Stop Capturing Packets button.

6. Filter SSL/TLS Traffic:
    - In the Wireshark filter bar, enter ssl or tls to filter only the SSL/TLS packets.
7. Analyze SSL Packets:
    - Click on a packet to view details in the packet details pane.
    - Look for the Client Hello and Server Hello messages, which initiate the SSL handshake.
    - Analyze other relevant packets, such as those containing encrypted data.
8. Inspect SSL Handshake:
    - Identify the SSL handshake process, including key exchange, cipher negotiation, and session establishment.
9. Save Capture:
    - Optionally, save the captured packets for further analysis by clicking on File > Save As.
10. Exit Wireshark:
    - Close Wireshark once you have completed your analysis.

Practical No. 09

Name: Tavhare Ruchita Sharad

Roll No.:58

Steps for Implementing S/MIME in Outlook

Step 1: Obtain a Digital Certificate

1. Choose a Certificate Authority (CA):
    - Select a trusted CA (e.g., DigiCert, GlobalSign, or Comodo) to purchase a digital certificate.
2. Generate a Certificate Request:
    - Follow the CA's process to generate a Certificate Signing Request (CSR) if needed.
3. Receive and Install the Certificate:
    - After verification, the CA will issue your digital certificate.
    - Download the certificate file, usually in .pfx or .p12 format.
4. Install the Certificate:
    - Double-click the downloaded certificate file.
    - Follow the prompts to install it, entering the password if required.

Step 2: Configure Outlook to Use the Certificate

1. Open Outlook:
    - Launch Microsoft Outlook.
2. Access Trust Center:
    - Go to File > Options.
    - Select Trust Center and click on Trust Center Settings.
3. Select Email Security:
    - In the Trust Center, choose Email Security.
4. Choose S/MIME Settings:
    - Under Encrypted email, click on Settings.
5. Select Your Certificate:
    - In the Security Settings dialog, click on Choose next to the Certificates and Algorithms section.
    - Select your digital certificate from the list and click OK.
6. Set Encryption and Signing Options:
    - Enable the options for Encrypt contents and attachments for outgoing messages and Add digital signature to outgoing messages.
7. Click OK to save the settings and exit the Trust Center.

Step 3: Sending Encrypted and Signed Emails

1. Compose a New Email:
    - Click on New Email to compose a message.
2. Set Security Options:
    - In the message window, go to the Options tab.
    - Click on Encrypt to encrypt the message.
    - Click on Sign to add a digital signature.
3. Send the Email:
    - After composing your email, click Send.

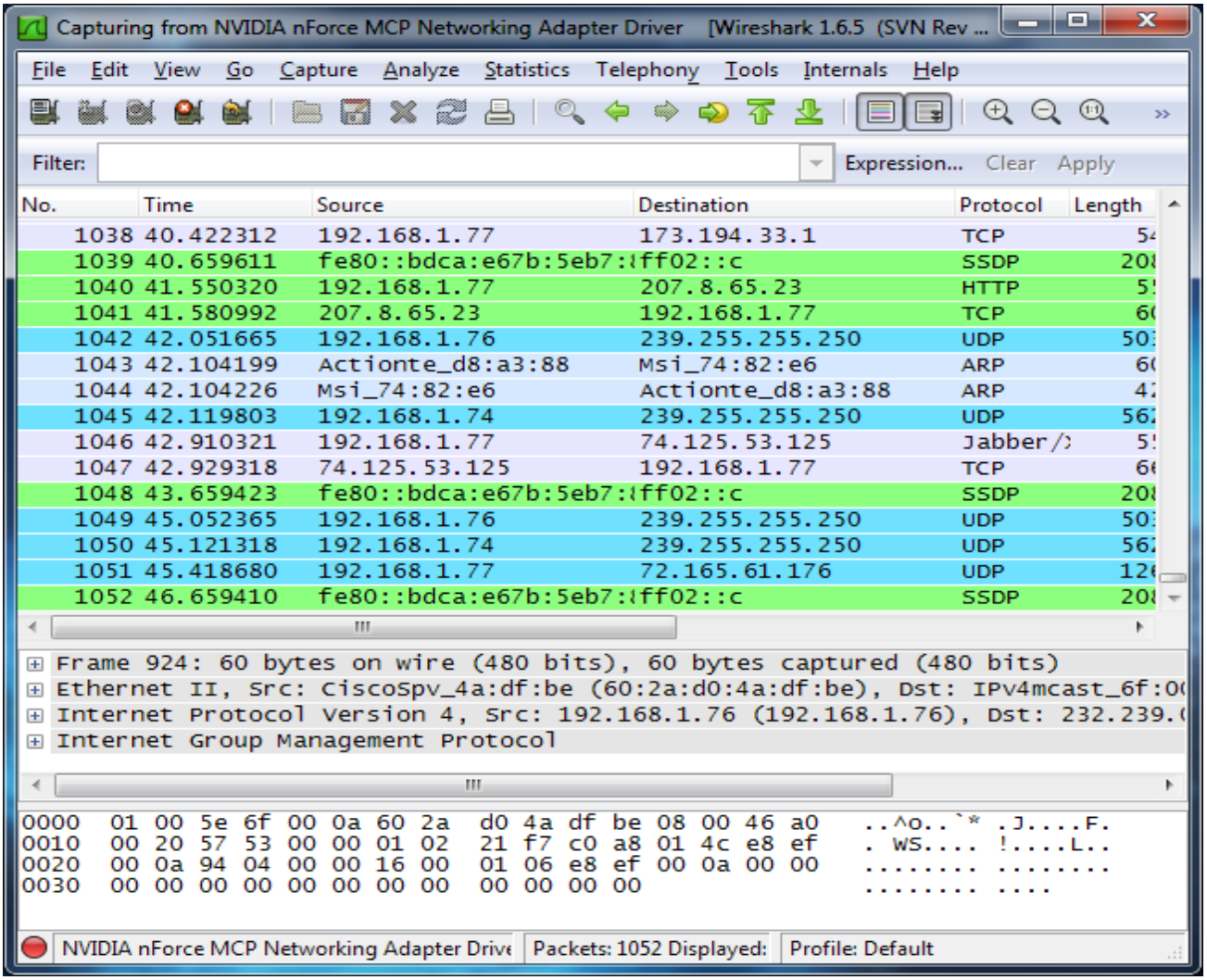
Step 4: Receiving and Verifying S/MIME Emails

1. Open an Encrypted Email:
    - When you receive an S/MIME email, it will be indicated as encrypted.
2. Verify the Signature:
    - Click on the signature icon (often appears in the reading pane) to verify the sender's signature and confirm the message's integrity.
3. Decrypt the Message:
    - Outlook will automatically decrypt the message if you have the sender's public key and your certificate installed.

Practical No. 10

Name: Tavhare Ruchita Sharad

Roll No.:58



Steps to Capture IPSec Packets Using Wireshark

Step 1: Set Up Your Environment

1. Install Wireshark:
    - Download and install Wireshark if you haven't already.
2. Configure IPSec:
    - Set up an IPSec VPN connection using devices or software that supports IPSec, such as a VPN client or a router configured with IPSec. Ensure both AH and ESP are enabled.

Step 2: Open Wireshark

1. Launch Wireshark:
    - Open the Wireshark application on the machine where the IPSec traffic will be generated.
2. Select Network Interface:
    - Choose the appropriate network interface (e.g., Wi-Fi or Ethernet) to capture packets.

Step 3: Start Packet Capture

1. Begin Capture:
    - Click the Start Capturing Packets button to start capturing traffic.

Step 4: Generate IPSec Traffic

1. Establish VPN Connection:
    - Connect to your IPSec VPN to generate traffic. This can involve accessing resources over the VPN, such as a remote server or intranet.

Step 5: Stop Packet Capture

1. End Capture:
    - After sufficient traffic has been generated, return to Wireshark and click the Stop Capturing Packets button.

Step 6: Filter IPSec Traffic

1. Use Filters:
    - In the Wireshark filter bar, enter the following filters to isolate IPSec traffic:
        - For ESP packets: ip.proto == 50
        - For AH packets: ip.proto == 51
    - This will display only the relevant packets for each protocol.

Step 7: Analyze ESP and AH Packets

1. Examine Packet Details:
    - Click on an ESP packet to view its details in the packet details pane. Look for:
        - Security Parameters Index (SPI)
        - Sequence Number
        - Encrypted Payload
    - Click on an AH packet to analyze its details, focusing on:
        - Integrity Check Value (ICV)
        - Sequence Number
2. Interpret the Information:
    - Analyze how ESP provides confidentiality through encryption and how AH provides integrity and authentication.

Step 8: Save Capture (Optional)

1. Save Your Work:
    - If you want to keep the captured packets for further analysis, click on File > Save As to save your capture file.

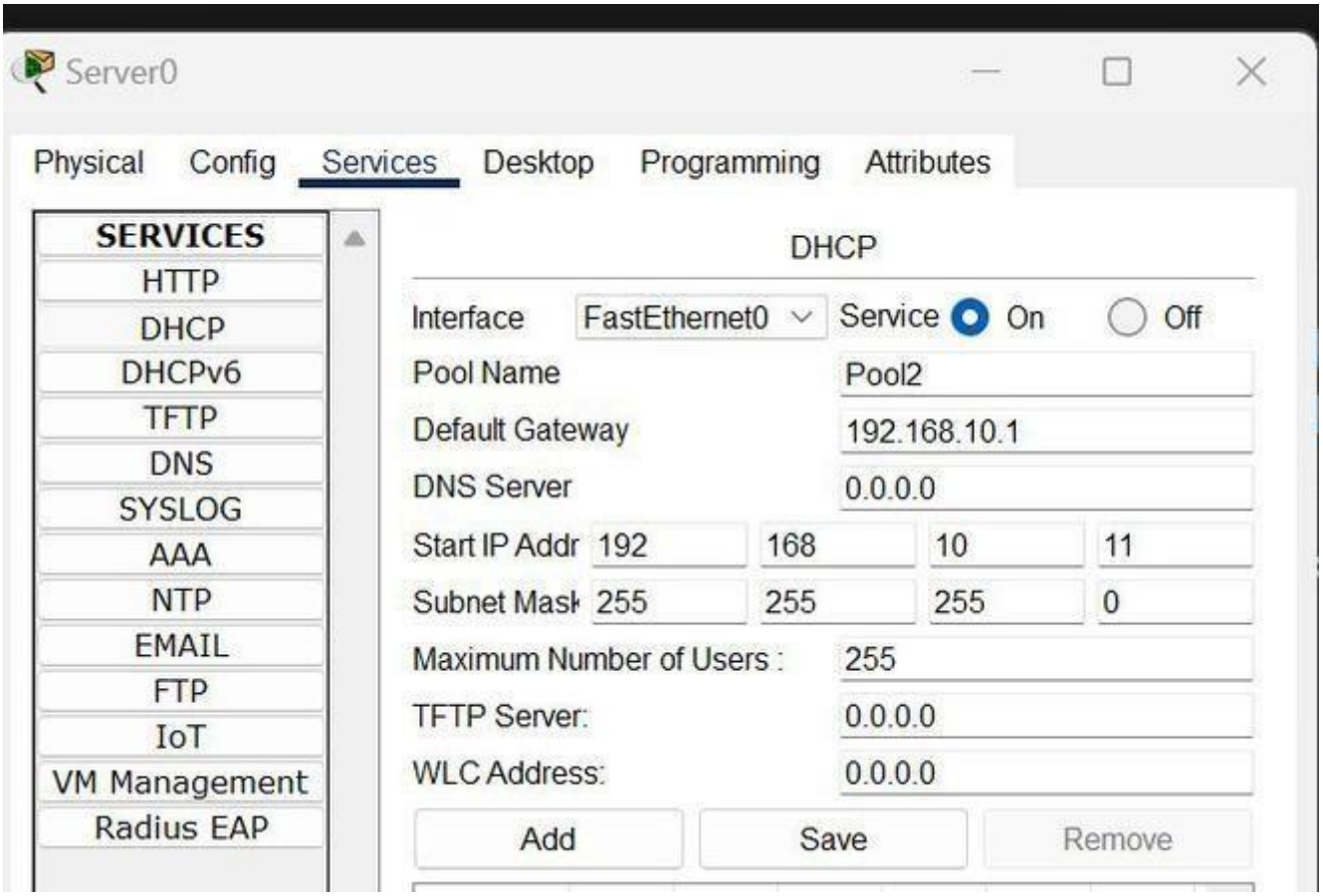
Step 9: Exit Wireshark

1. Close Wireshark:
    - After your analysis is complete, you can close Wireshark.

Practical No. 11

Name: Tavhare Ruchita Sharad

Roll No.:58



Steps to Configure a DHCP Server in Cisco Packet Tracer

Step 1: Open Cisco Packet Tracer

1. Launch Cisco Packet Tracer:
    - Open the application on your computer.

Step 2: Set Up the Network

1. Add Devices:
    - Drag and drop the following devices onto the workspace:
        - 1 Router
        - 1 Switch
        - Multiple PCs (clients)
2. Connect Devices:
    - Use the copper straight-through cable to connect:
        - The router’s LAN port to the switch.
        - The switch to the PCs.

Step 3: Configure the Router

1. Access the Router Configuration:
    - Click on the router and go to the CLI tab.
2. Enter Global Configuration Mode:

code

enable

configure terminal

3. Set the Router Interface:
    - Enter the interface configuration mode for the interface connected to the switch (e.g., FastEthernet0/0):

Code interface FastEthernet0/0

ip address 192.168.1.1 255.255.255.0

no shutdown

exit

4. Configure the DHCP Server:
    - Define the DHCP pool:

plaintext

ip dhcp pool MY_POOL

network 192.168.1.0 255.255.255.0

default-router 192.168.1.1

dns-server 8.8.8.8

5. Exclude Addresses (Optional):
    - If you want to exclude certain addresses (like the router's IP):

code

ip dhcp excluded-address 192.168.1.1

Step 4: Configure Client PCs

1. Select Each PC:
    - Click on each PC to open the configuration window.
2. Set IP Configuration:
    - Go to the Desktop tab, select IP Configuration, and choose DHCP.
    - Each PC will automatically request an IP address from the DHCP server.

Step 5: Verify DHCP Configuration

1. Check IP Assignment on PCs:
    - After setting the PCs to DHCP, you can check if they received the correct IP addresses.
    - On each PC, go to the Desktop tab, select Command Prompt, and type:

Practical No. 12

Name: Tavhare Ruchita Sharad

Roll No.:58

Code:

```
import socket


def dns_lookup():
    choice = input("Enter '1' for URL to IP address or '2' for IP address to URL: ")

    if choice == '1':
        # URL to IP Address
        url = input("Enter the URL (e.g., www.example.com): ")
        try:
            ip_address = socket.gethostbyname(url)
            print(f"The IP address for {url} is: {ip_address}")
        except socket.gaierror:
            print("Could not resolve the URL. Please check the URL and try again.")
    elif choice == '2':
        # IP Address to URL
        ip_address = input("Enter the IP address (e.g., 192.0.2.1): ")
        try:
            url = socket.gethostbyaddr(ip_address)
            print(f"The URL for {ip_address} is: {url[0]}")
        except socket.herror:
            print("Could not resolve the IP address. Please check the IP and try again.")
    else:
        print("Invalid choice. Please enter '1' or '2'.")

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

Output:

URL to IP Address:

Enter '1' for URL to IP address or '2' for IP address to URL: 1

Enter the URL (e.g., www.example.com): www.google.com

The IP address for www.google.com is: 142.250.64.206

IP Address to URL:

Enter '1' for URL to IP address or '2' for IP address to URL: 2

Enter the IP address (e.g., 192.0.2.1): 142.250.64.206

The URL for 142.250.64.206 is: www.google.com