

# AMITY SCHOOL OF ENGINEERING AND TECHNOLOGY



Title: Exploring the Networks Lab File

Subject: Exploring the Networks

Course Code: IT307

Submitted by:

Name: RIJU KUMARI

Enrollment no.: A41105223228

Course: B. Tech. CSE (5th Semester)

Under the guidance of: Prof. Dr. Tejaswi Khanna

Amity University, Plot no. 48A, Knowledge Park III, Greater Noida, Uttar Pradesh-201308

## INDEX

S.no.	Practical
1.	To observe TCP and UDP connections and ports using the netstat command in Cisco Packet Tracer.
2.	To examine and compare TCP and UDP packets using Wireshark.
3.	To examine and verify a device's default gateway in a network.
4.	To study and verify network connectivity using <b>Ping</b> and <b>Traceroute</b> using <i>ping</i> and <i>tracert</i> commands, and to display routing information using the <i>show ip route</i> command in Cisco Packet Tracer.
5.	To examine <b>ICMP (Internet Control Message Protocol) packets</b> by generating ping traffic and analyzing packet flow in Cisco Packet Tracer using the simulation mode
6.	To understand and perform <b>IPv4 Address Subnetting</b> by dividing a large network into smaller subnetworks for efficient utilization of IP addresses.
7.	To perform IPv4 subnetting and configure a router to enable communication between different subnets.

## **Practical - 1**

**Aim:** To observe TCP and UDP connections and ports using the netstat command in Cisco Packet Tracer.

### **Network Topology:**

We will use:

- 1 Server
- 2 PCs
- 1 Switch

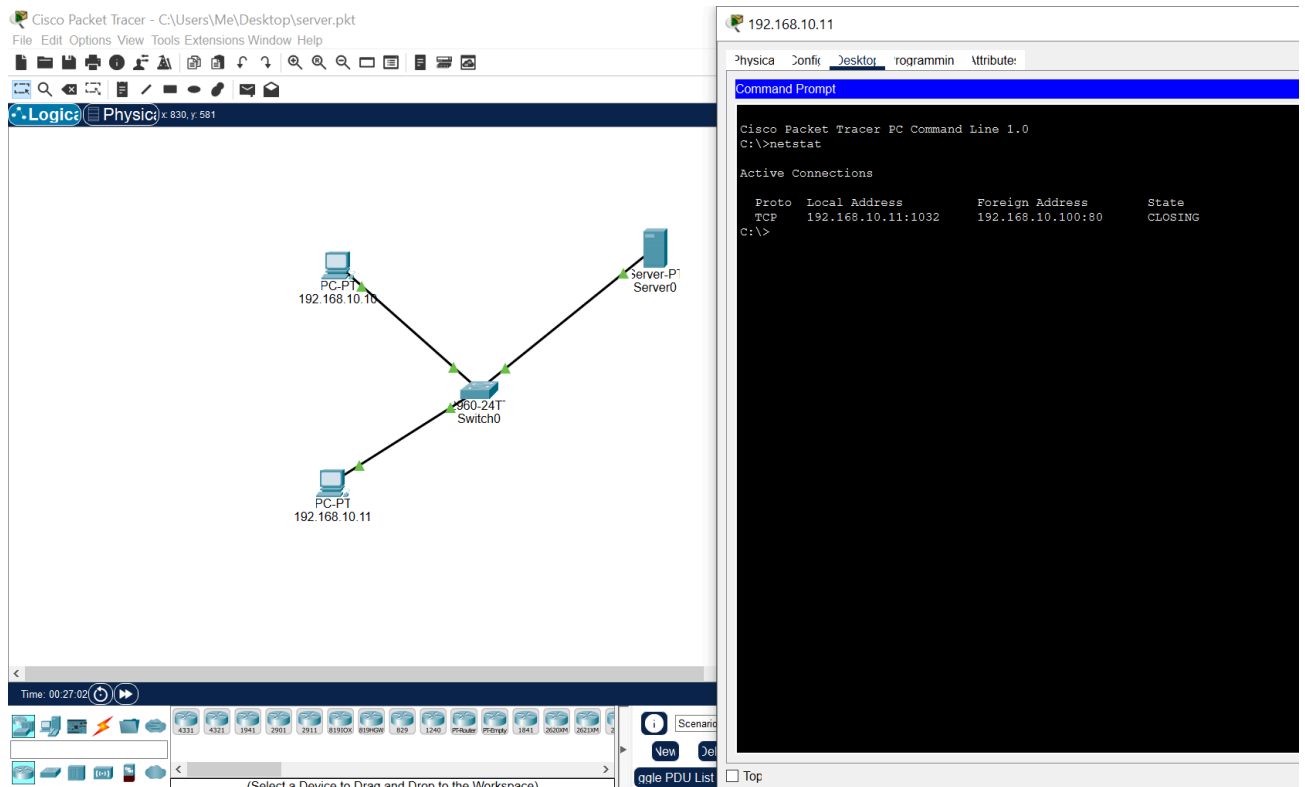
### **Network Setup**

<b>Device</b>	<b>IP Address</b>	<b>Subnet Mask</b>
PC0	192.168.10.10	255.255.255.0
PC1	192.168.10.11	255.255.255.0
Server	192.168.10.100	255.255.255.0

### **Procedure**

1. Open **Cisco Packet Tracer** and create a network with **2 PCs**, **1 Server**, and **1 Switch**.
2. Connect devices using **Copper Straight-Through** cables.
3. Assign IP addresses as shown above in network setup.
4. On the Server, enable **HTTP** service (TCP) and **DNS** service (UDP).
5. From PC0, open a web browser and go to `http://192.168.10.100` to create TCP traffic.
6. From PC0, open **Command Prompt** and type `netstat` to view TCP and UDP connections.
7. Note the TCP connection to port **80** (HTTP).

## Network setup and Output:



### Result:

netstat displayed an active TCP connection to port 80 (HTTP), confirming TCP is connection-oriented.

## **Practical - 2**

**Aim:** To examine and compare TCP and UDP packets using Wireshark.

### **Apparatus / Tools**

- Wireshark (installed on PC)
- Internet connection / Local server

### **Theory**

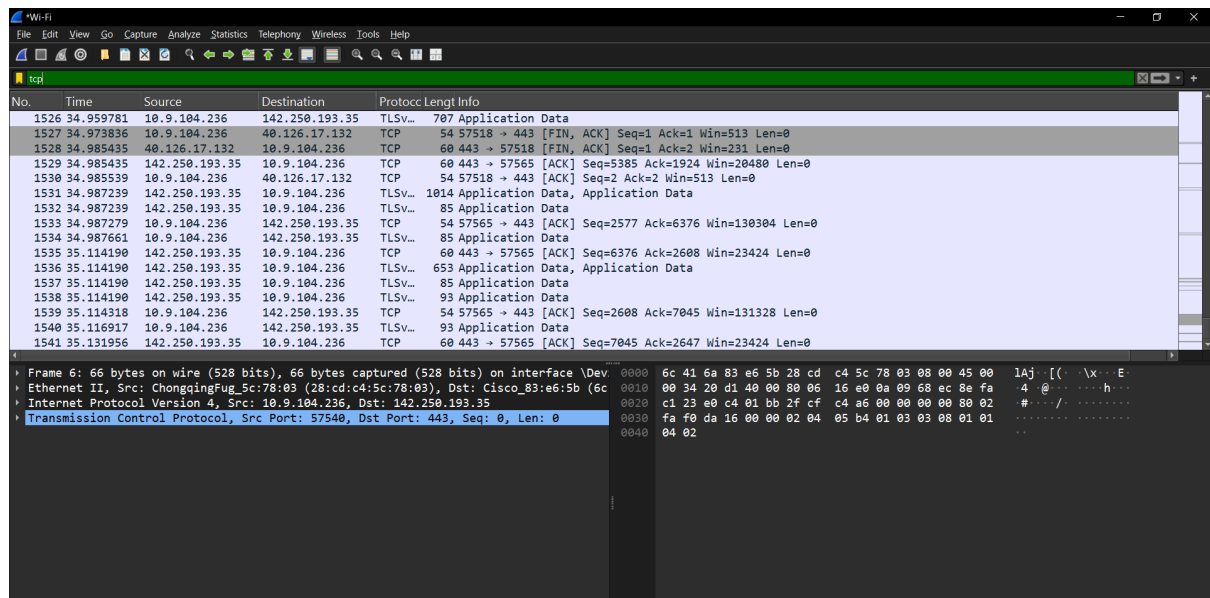
- **TCP (Transmission Control Protocol):** Connection-oriented, reliable protocol that uses acknowledgments, sequencing, and retransmissions.
- **UDP (User Datagram Protocol):** Connectionless, fast, lightweight protocol with no delivery guarantee.

Wireshark is a packet analyzer that captures and displays the details of TCP and UDP headers in real time.

### **Procedure**

1. Open **Wireshark** and start a capture on your active network interface.
2. **Generate TCP traffic:**
  - Open a web browser and visit any website (HTTP/HTTPS).
  - In Wireshark, apply the filter: 'tcp'
  - Observe fields like **Source Port, Destination Port, Sequence Number, Acknowledgment, Flags (SYN, ACK, FIN)**.
3. **Generate UDP traffic:**
  - Open Command Prompt and use a DNS query: 'nslookup [www.google.com](http://www.google.com)'
  - In Wireshark, apply the filter: 'udp'
  - Observe fields like **Source Port, Destination Port, Length, Checksum**.
4. Stop the capture and compare the structure of TCP and UDP packets.

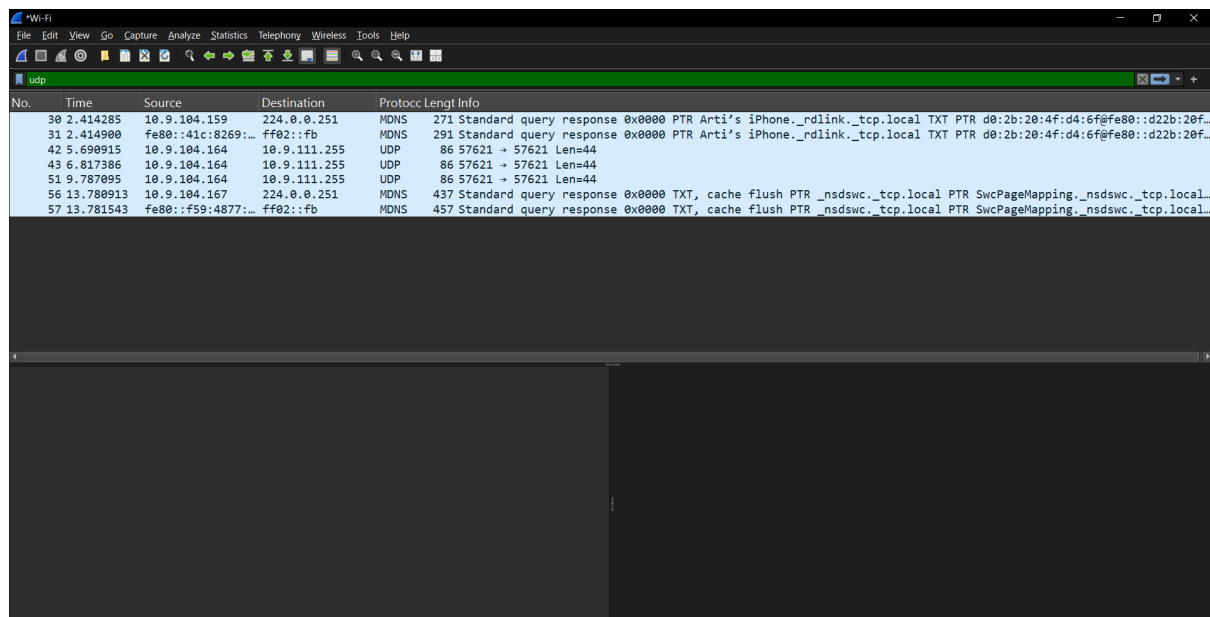
## Observation in Wireshark:



No.	Time	Source	Destination	Protocol	Length	Info
1526	34.959781	10.9.104.236	142.250.193.35	TLSv...	707	Application Data
1527	34.973836	10.9.104.236	40.126.17.132	TCP	54	57518 → 443 [FIN, ACK] Seq=1 Ack=1 Win=513 Len=0
1528	34.985435	40.126.17.132	10.9.104.236	TCP	60	443 → 57518 [FIN, ACK] Seq=1 Ack=2 Win=231 Len=0
1529	34.985435	142.250.193.35	10.9.104.236	TCP	60	443 → 57565 [ACK] Seq=5385 Ack=1924 Win=20480 Len=0
1530	34.985539	10.9.104.236	40.126.17.132	TCP	54	57518 → 443 [ACK] Seq=2 Ack=2 Win=513 Len=0
1531	34.987239	142.250.193.35	10.9.104.236	TLSv...	1014	Application Data, Application Data
1532	34.987239	142.250.193.35	10.9.104.236	TLSv...	85	Application Data
1533	34.987279	10.9.104.236	142.250.193.35	TCP	54	57565 → 443 [ACK] Seq=2577 Ack=6376 Win=130304 Len=0
1534	34.987661	10.9.104.236	142.250.193.35	TLSv...	85	Application Data
1535	35.114190	142.250.193.35	10.9.104.236	TCP	60	443 → 57565 [ACK] Seq=6376 Ack=2608 Win=23424 Len=0
1536	35.114190	142.250.193.35	10.9.104.236	TLSv...	653	Application Data, Application Data
1537	35.114190	142.250.193.35	10.9.104.236	TLSv...	85	Application Data
1538	35.114190	142.250.193.35	10.9.104.236	TLSv...	93	Application Data
1539	35.114318	10.9.104.236	142.250.193.35	TCP	54	57565 → 443 [ACK] Seq=2608 Ack=7045 Win=131328 Len=0
1540	35.116917	10.9.104.236	142.250.193.35	TLSv...	93	Application Data
1541	35.131956	142.250.193.35	10.9.104.236	TCP	60	443 → 57565 [ACK] Seq=7045 Ack=2647 Win=23424 Len=0

Frame 6: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface \Device\NPF{...} Ethernet II, Src: ChongqingFug\_5c:78:03 (28:cd:c4:5c:78:03), Dst: Cisco\_83:e6:5b (6c:00:00:00:00:00) Internet Protocol Version 4, Src: 10.9.104.236, Dst: 142.250.193.35 Transmission Control Protocol, Src Port: 57540, Dst Port: 443, Seq: 0, Len: 0

## TCP



No.	Time	Source	Destination	Protocol	Length	Info
30	2.414285	10.9.104.159	224.0.0.251	MDNS	271	Standard query response 0x0000 PTR Arti's iPhone._rdlink._tcp.local TXT d0:2b:20:4f:d4:6f@fe80::d22b:20f...
31	2.414900	fe80::41c:8269::...	ff02::fb	MDNS	291	Standard query response 0x0000 PTR Arti's iPhone._rdlink._tcp.local TXT PTR d0:2b:20:4f:d4:6f@fe80::d22b:20f...
42	5.690915	10.9.104.164	10.9.111.255	UDP	86	57621 → 57621 Len=44
43	6.817386	10.9.104.164	10.9.111.255	UDP	86	57621 → 57621 Len=44
51	9.787095	10.9.104.164	10.9.111.255	UDP	86	57621 → 57621 Len=44
56	13.780913	10.9.104.167	224.0.0.251	MDNS	437	Standard query response 0x0000 TXT, cache flush PTR _nsdswc._tcp.local PTR SwcPageMapping._nsdswc._tcp.local...
57	13.781543	fe80::f59:4877::...	ff02::fb	MDNS	457	Standard query response 0x0000 TXT, cache flush PTR _nsdswc._tcp.local PTR SwcPageMapping._nsdswc._tcp.local...

## UDP

**Result:** Using Wireshark, TCP packets were observed with handshake and acknowledgment fields, while UDP packets showed simple headers without connection setup. This confirms TCP is reliable and connection-oriented, whereas UDP is fast and connectionless.

## Practical - 3

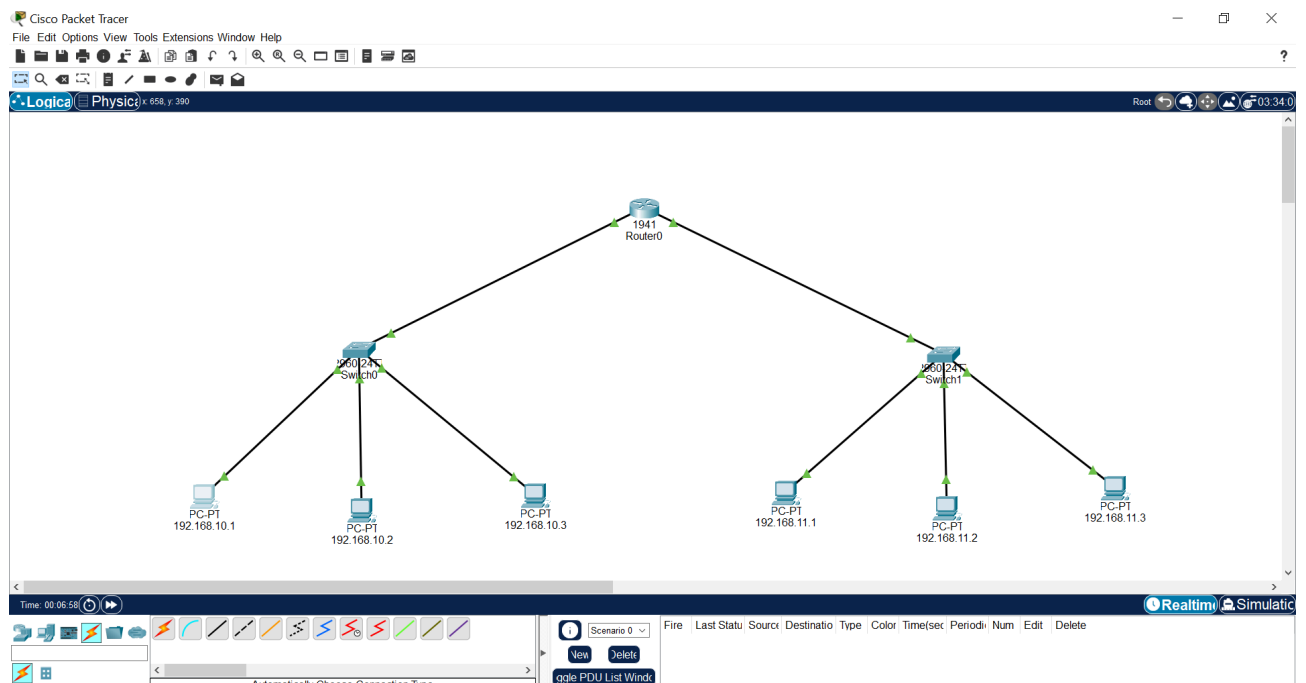
**Aim:** To examine and verify a device's default gateway in a network.

### Network Topology Devices

- **1 Router** (gateway)
- **2 Switches** (Switch0, Switch1)
- **6 PCs** (PC0–PC5)
- Copper Straight-Through cables everywhere

### Network Setup:

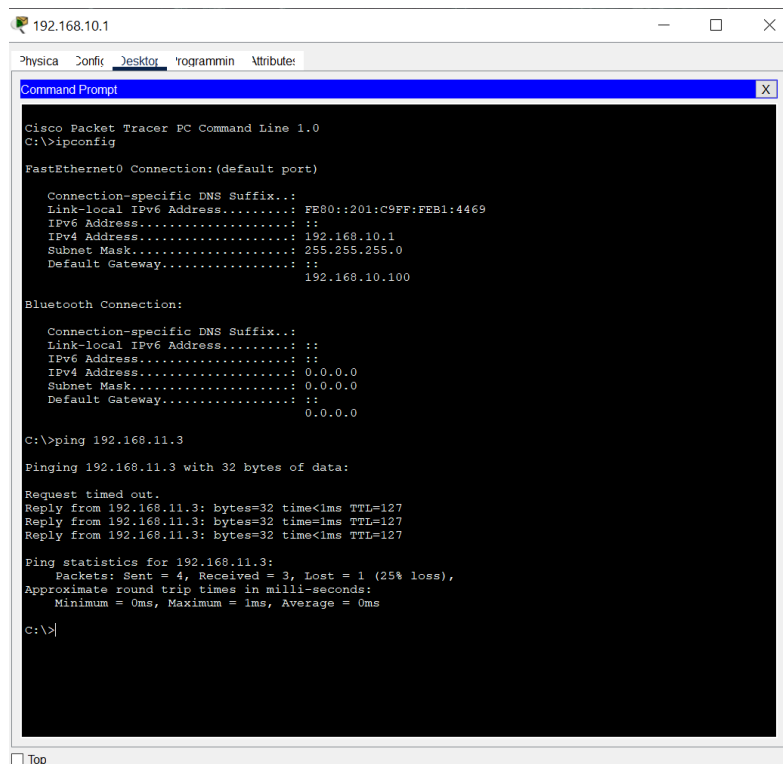
Device	IP Address	Subnet Mask	Default Gateway
PC0	192.168.10.1	255.255.255.0	192.168.10.100
PC1	192.168.10.2	255.255.255.0	192.168.10.100
PC2	192.168.10.3	255.255.255.0	192.168.10.100
PC3	192.168.11.1	255.255.255.0	192.168.11.100
PC4	192.168.11.2	255.255.255.0	192.168.11.100
PC5	192.168.11.3	255.255.255.0	192.168.11.100



## Procedure:

1. Open Cisco Packet Tracer and create the topology as shown in the network setup diagram:
  - 1 Router connected to Switch0 and Switch1
  - 3 PCs (PC0–PC2) connected to Switch0
  - 3 PCs (PC3–PC5) connected to Switch1
2. Use Copper Straight-Through cables for all connections.
3. **Configure Router (Gateway):**
  - On Router
    - FastEthernet0/0 → Assign IP: 192.168.10.100, Subnet: 255.255.255.0
    - FastEthernet0/1 → Assign IP: 192.168.11.100, Subnet: 255.255.255.0
  - Turn **Port Status: On**.
4. Assign IP & Gateway to each PC according to the network setup.
5. **Verify the Gateway:**
  - On any PC (connected to Switch0) → Desktop → Command Prompt: Enter: ipconfig → ping 192.168.11.1

## Output:



```
C:\192.168.10.1
Cisco Packet Tracer PC Command Line 1.0
C:\>ipconfig

FastEthernet0 Connection: (default port)

    Connection-specific DNS Suffix...: FE80::201:C9FF:FEB1:4469
    Link-local IPv6 Address...: ::
    IPv6 Address...: 192.168.10.1
    Subnet Mask...: 255.255.255.0
    Default Gateway...: 192.168.10.100

Bluetooth Connection:

    Connection-specific DNS Suffix...: 
    Link-local IPv6 Address...: ::
    IPv6 Address...: 0.0.0.0
    Subnet Mask...: 0.0.0.0
    Default Gateway...: 0.0.0.0

C:\>ping 192.168.11.3

Pinging 192.168.11.3 with 32 bytes of data:

Request timed out.
Reply from 192.168.11.3: bytes=32 time<1ms TTL=127
Reply from 192.168.11.3: bytes=32 time<1ms TTL=127
Reply from 192.168.11.3: bytes=32 time<1ms TTL=127

Ping statistics for 192.168.11.3:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms

C:\>|
```

**Result:** All PCs successfully pinged through the router, confirming the gateway worked.



## Practical - 4

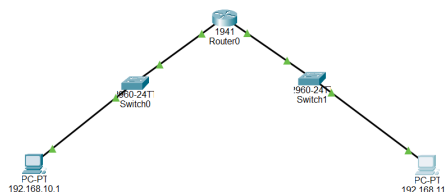
**Aim:** To study and verify network connectivity using **Ping** and **Traceroute** using *ping* and *tracert* commands, and to display routing information using the *show ip route* command in Cisco Packet Tracer.

### Network Topology Devices

- **1 Router** (gateway)
- **2 Switches** (Switch0, Switch1)
- **2 PCs** (PC0, PC1)
- Copper Straight-Through cables everywhere

### Network Setup:

Device	IP Address	Subnet Mask	Default Gateway
PC0	192.168.10.1	255.255.255.0	192.168.10.100
PC1	192.168.11.1	255.255.255.0	192.168.11.100



### Procedure:

1. Open Cisco Packet Tracer and create a new project.
2. Place the required devices from the device list:

- 2 PCs
- 2 Switches
- 1 Router
- Copper Straight-through cables

3. Connect the devices using cables:

- Connect **PC0** → **Switch0** using a Straight-through cable
- Connect **Switch0** → **Router0 (G0/0)** using a Straight-through cable
- Connect **Router0 (G0/1)** → **Switch1** using a Straight-through cable
- Connect **Switch1** → **PC1** using a Straight-through cable

#### 4. Assign IP addresses to PCs:

- Open **PC0** → **Desktop** → **IP Configuration**
  - IP: 192.168.10.1
  - Subnet Mask: 255.255.255.0
  - Default Gateway: 192.168.10.100
- Open **PC1** → **Desktop** → **IP Configuration**
  - IP: 192.168.11.1
  - Subnet Mask: 255.255.255.0
  - Default Gateway: 192.168.11.100

#### 5. Configure Router Interfaces:

Select router → CLI tab

Enter the following commands:

```
enable
configure terminal
interface g0/0
ip address 192.168.1.1 255.255.255.0
no shutdown
exit
interface g0/1
ip address 192.168.2.1 255.255.255.0
no shutdown
exit
end
```

#### 6. Check Router Routing Table:

```
show ip route
```

#### 7. Test connectivity using Ping:

On **PC1**, open Command Prompt and type:

```
ping 192.168.10.1
```

#### 8. Use Traceroute to view network path:

On **PC1**, type:

```
tracert 192.168.10.1
```

9. Observe results—successful ping replies and traceroute hop display confirm correct routing.

## Output:

```
Router0
IOS Command Line Interface

Router(config-if)#ip address 192.168.10.100 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up
Router(config-if)#
Router(config-if)#exit
Router(config)#interface GigabitEthernet0/1
Router(config-if)#ip address 192.168.11.100 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/1, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed state to up
Router(config-if)#show ip route
% Invalid input detected at '' marker.
Router(config-if)#
Router#
%SYS-5-CONFIG-I: Configured from console by console
Router#show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       O - OSPF, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

192.168.10.0/24 is variably subnetted, 2 subnets, 2 masks
C       192.168.10.0/24 is directly connected, GigabitEthernet0/0
L       192.168.10.100/32 is directly connected, GigabitEthernet0/0
C       192.168.11.0/24 is variably subnetted, 2 subnets, 2 masks
C       192.168.11.0/24 is directly connected, GigabitEthernet0/1
L       192.168.11.100/32 is directly connected, GigabitEthernet0/1

Router#
```

```
192.168.11.1
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.10.1

Pinging 192.168.10.1 with 32 bytes of data:

Reply from 192.168.10.1: bytes=32 time=13ms TTL=127
Reply from 192.168.10.1: bytes=32 time<1ms TTL=127
Reply from 192.168.10.1: bytes=32 time<1ms TTL=127
Reply from 192.168.10.1: bytes=32 time=1ms TTL=127

Ping statistics for 192.168.10.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 13ms, Average = 3ms

C:\>tracert 192.168.10.1

Tracing route to 192.168.10.1 over a maximum of 30 hops:
  0  0 ms    0 ms    0 ms    192.168.11.100
  1  0 ms    0 ms    0 ms    192.168.10.1
Trace complete.

C:\>
```

## Result:

Ping and traceroute were successfully executed between PC1 and PC2 through the router and switch.

The show ip route command displayed the connected networks, proving that routing and connectivity worked correctly.

## Practical - 5

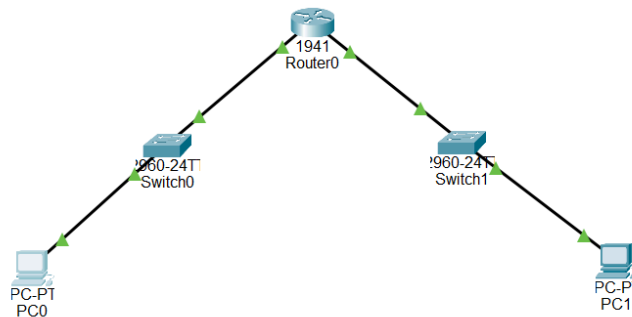
**Aim:** To examine **ICMP (Internet Control Message Protocol) packets** by generating ping traffic and analyzing packet flow in Cisco Packet Tracer using the simulation mode.

### Network Topology Devices

Components	Quantity
PCs	2
Switch	1
Router	1
Copper straight-through cables	As required

### Network Setup

Device	Interface	IP Address	Subnet Mask	Gateway
PC0	NIC	192.168.1.10	255.255.255.0	192.168.1.1
PC1	NIC	192.168.2.10	255.255.255.0	192.168.2.1
Router	G0/0	192.168.1.1	255.255.255.0	–
Router	G0/1	192.168.2.1	255.255.255.0	–



## Procedure

### Step 1: Create Network Topology

1. Open **Cisco Packet Tracer**
2. Drag 2 PCs, 1 Router, and 1 Switch.
3. Connect:
  - PC0 → Switch (straight-through)
  - PC1 → Switch (straight-through)
  - Switch → Router interface G0/0 and G0/1

### Step 2: Configure IP Addresses

#### On PC0

Desktop → IP Configuration:

IP: 192.168.1.10

Mask: 255.255.255.0

Gateway: 192.168.1.1

#### On PC1

IP: 192.168.2.10

Mask: 255.255.255.0

Gateway: 192.168.2.1

### Step 3: Configure Router

Open CLI:

```
Router> enable
```

```
Router# configure terminal
```

```
Router(config)# interface g0/0
```

```
Router(config-if)# ip address 192.168.1.1 255.255.255.0
```

```
Router(config-if)# no shutdown
```

```
Router(config-if)# exit
```

```
Router(config)# interface g0/1
```

```
Router(config-if)# ip address 192.168.2.1 255.255.255.0
```

```
Router(config-if)# no shutdown
```

```
Router(config-if)# exit
```

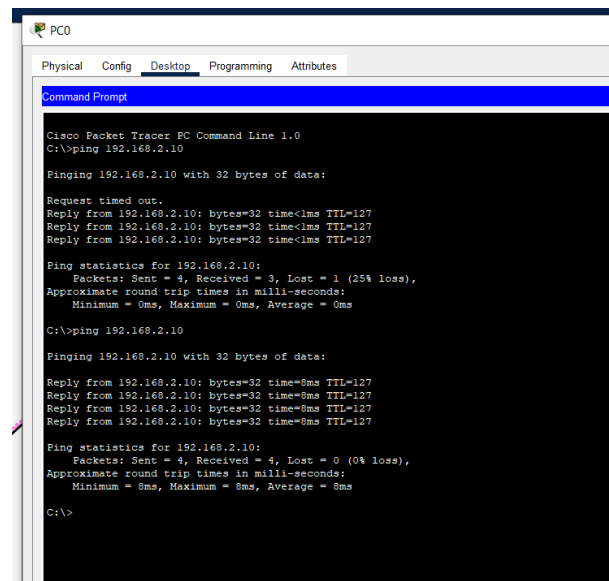
## Step 4: Use Simulation Mode to Observe ICMP Packets

1. Click **Simulation Mode** (bottom right)
2. Select **ICMP** from the Event Filters

On **PC1**, open Command Prompt: ping 192.168.2.10

3. Observe the packet movement frame-by-frame.
4. Click **Play** or **Capture/Forward** to view each stage.

### Output:



```
PC0
Physical Config Desktop Programming Attributes
Command Prompt
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.2.10

Pinging 192.168.2.10 with 32 bytes of data:

Request timed out.
Reply from 192.168.2.10: bytes=32 time=1ms TTL=127
Reply from 192.168.2.10: bytes=32 time=1ms TTL=127
Reply from 192.168.2.10: bytes=32 time=1ms TTL=127

Ping statistics for 192.168.2.10:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\>ping 192.168.2.10

Pinging 192.168.2.10 with 32 bytes of data:

Reply from 192.168.2.10: bytes=32 time=8ms TTL=127
Reply from 192.168.2.10: bytes=32 time=8ms TTL=127
Reply from 192.168.2.10: bytes=32 time=8ms TTL=127
Reply from 192.168.2.10: bytes=32 time=8ms TTL=127

Ping statistics for 192.168.2.10:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 8ms, Maximum = 8ms, Average = 8ms
C:\>
```

### On Simulation Screen

- ICMP Echo Request packet travels **PC0** → **Switch** → **Router** → **Switch** → **PC1**
- ICMP Echo Reply packet returns **PC1** → **Switch** → **Router** → **Switch** → **PC0**
- Packet type is shown as ICMP with colored arrows

**Result:** ICMP packet behavior was successfully examined. The Echo Request and Echo Reply packets were visualized during transmission across the router using simulation mode in Cisco Packet Tracer.

## **Practical-6**

**Aim:** To understand and perform **IPv4 Address Subnetting** by dividing a large network into smaller subnetworks for efficient utilization of IP addresses.

**Theory:** Subnetting is the process of dividing a single large IP network into multiple smaller logical networks. It helps in:

- Better network management
- Improved security
- Reduced broadcast traffic
- Efficient usage of IP addresses

IPv4 addresses include **Network Portion + Host Portion**, determined by the Subnet Mask.

### **Given Problem**

Network Address: **192.168.1.0 /24**

Requirement: **4 Subnets**

### **Steps to Calculate Subnets**

#### **Step 1: Determine number of bits to borrow**

To create 4 subnets:

$$2^n \geq 4$$

$$n = 2 \text{ bits}$$

So borrow **2 bits** from the host portion.

#### **Step 2: New Subnet Mask**

Original mask: /24

Borrowed 2 bits: /24 + 2 = **/26**

Subnet mask for /26 = **255.255.255.192**

#### **Step 3: Calculate Subnet Increment**

$$\text{Formula: } 256 - 192 = 64$$

Therefore, each subnet increases by 64.

#### Step 4: Create Subnet Table

Subnet No.	Network ID	First Usable Host	Last Usable Host	Broadcast Address
1	192.168.1.0	192.168.1.1	192.168.1.62	192.168.1.63
2	192.168.1.64	192.168.1.65	192.168.1.126	192.168.1.127
3	192.168.1.128	192.168.1.129	192.168.1.190	192.168.1.191
4	192.168.1.192	192.168.1.193	192.168.1.254	192.168.1.255

#### Example Host Assignment

Device	IP Address	Subnet Mask	Gateway
PC1	192.168.1.10	255.255.255.192	192.168.1.1
PC2	192.168.1.50	255.255.255.192	192.168.1.1
PC3	192.168.1.70	255.255.255.192	192.168.1.65
PC4	192.168.1.130	255.255.255.192	192.168.1.129

#### Result

Subnetting for the network **192.168.1.0/24** was successfully performed. Four subnets were created with a subnet mask of **255.255.255.192 (/26)**, and the Network ID, Broadcast address, and valid host ranges were calculated.

#### Conclusion

Subnetting divides a large network into smaller networks, improving IP utilization, security, and performance.



## Practical-7

**Aim:** To perform IPv4 subnetting and configure a router to enable communication between different subnets.

### Network Topology Devices

Components	Quantity
Router	1
Switches	2
PCs	4
Copper straight-through cables	As required

### Network Address Information

Given Network: **192.168.10.0 /24**

Required: **4 Subnets**

### Subnetting Calculation

Borrow bits:  $2^n \geq 4 \rightarrow n = 2$  bits

New Prefix:

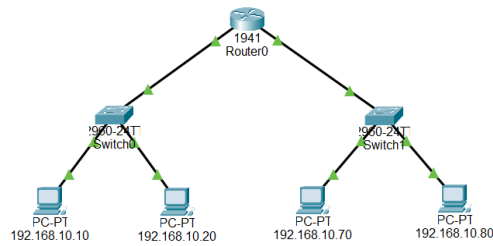
$/24 + 2 = /26$

Subnet Mask = 255.255.255.192

Increment = 64

### Subnet Table & Network Setup

Subnet	Network ID	First Host	Last Host	Broadcast Address
1	192.168.10.0	192.168.10.1	192.168.10.62	192.168.10.63
2	192.168.10.64	192.168.10.65	192.168.10.126	192.168.10.127
3	192.168.10.128	192.168.10.129	192.168.10.190	192.168.10.191
4	192.168.10.192	192.168.10.193	192.168.10.254	192.168.10.255



## IP Address Assignment

Device	IP Address	Subnet Mask	Default Gateway
PC1	192.168.10.10	255.255.255.192	192.168.10.1
PC2	192.168.10.20	255.255.255.192	192.168.10.1
PC3	192.168.10.70	255.255.255.192	192.168.10.65
PC4	192.168.10.80	255.255.255.192	192.168.10.65
Router G0/0	192.168.10.1	255.255.255.192	—
Router G0/1	192.168.10.65	255.255.255.192	—

## Procedure

1. Subnet the given network (192.168.10.0/24) into 4 subnets.
2. Create the network topology using: 1 Router, **2 Switches**, and 4 PCs.
3. Connect PC1 & PC2 to Switch 1, and PC3 & PC4 to Switch 2.
4. Connect both switches to the router using straight-through cables.
5. Assign IP addresses to PCs according to subnet table.
6. Open router CLI and configure router interfaces:

```
Router> enable
```

```
Router# configure terminal
```

```
Router(config)# interface g0/0
```

```
Router(config-if)# ip address 192.168.10.1
255.255.255.192
```

```
Router(config-if)# no shutdown
```

```
Router(config-if)# exit
```

```
Router(config)# interface g0/1
```

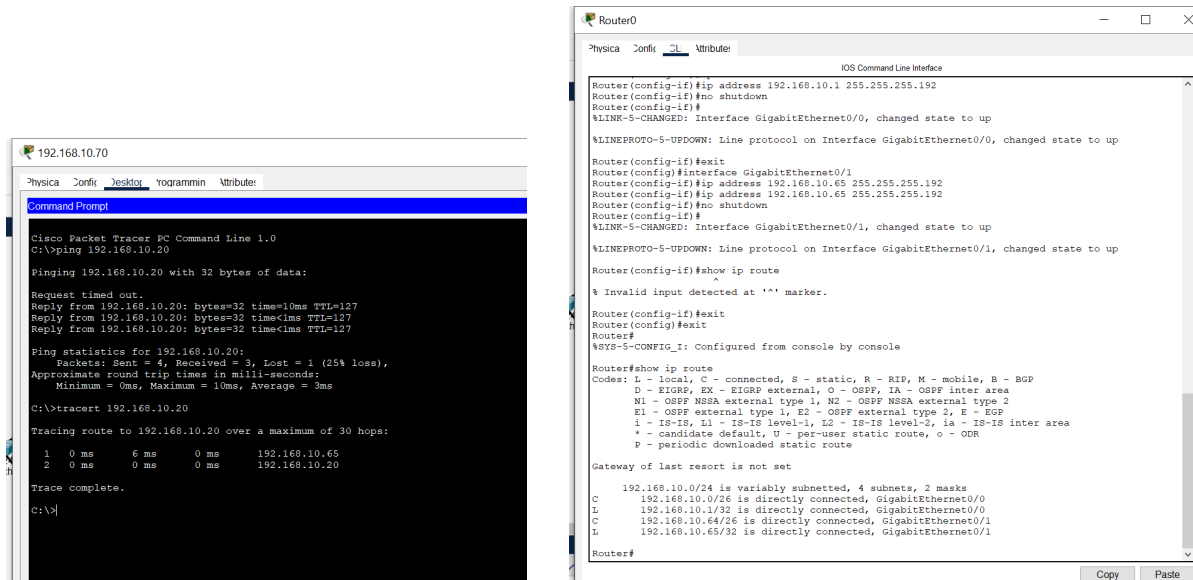
```
Router(config-if)# ip address 192.168.10.65  
255.255.255.192
```

```
Router(config-if)# no shutdown
```

7. Verify router routing table: Router# show ip route

8. Test communication: On PC3 using *ping* 192.168.10.20

## Output:



```
192.168.10.70
C:\>ping 192.168.10.20

Pinging 192.168.10.20 with 32 bytes of data:

Request timed out.
Reply from 192.168.10.20: bytes=32 time=10ms TTL=127
Reply from 192.168.10.20: bytes=32 time<1ms TTL=127
Reply from 192.168.10.20: bytes=32 time<1ms TTL=127

Ping statistics for 192.168.10.20:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 10ms, Average = 3ms

C:\>tracert 192.168.10.20

Tracing route to 192.168.10.20 over a maximum of 30 hops:
  0  0 ms    6 ms    0 ms   192.168.10.65
  1  0 ms    0 ms    0 ms   192.168.10.20
Trace complete.

C:\>

Router0
IOS Command Line Interface
Router(config-if)#ip address 192.168.10.1 255.255.255.192
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up
Router(config-if)#exit
Router(config)#interface GigabitEthernet0/1
Router(config-if)#ip address 192.168.10.65 255.255.255.192
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/1, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed state to up
Router(config-if)#show ip route
% Invalid input detected at '^' marker.
Router(config-if)#exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
Router#show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
        D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
        N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
        E1 - OSPF external type 1, E2 - OSPF external type 2, E - EGP
        I - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
        * - candidate default, U - per-user static route, o - ODR
        P - periodic downloaded static route

Gateway of last resort is not set

    192.168.10.0/24 is variably subnetted, 4 subnets, 2 masks
C       192.168.10.0/26 is directly connected, GigabitEthernet0/0
L       192.168.10.1/32 is directly connected, GigabitEthernet0/0
C       192.168.10.64/26 is directly connected, GigabitEthernet0/1
L       192.168.10.65/32 is directly connected, GigabitEthernet0/1

Router#
```

**Result:** Subnetting was successfully implemented and router interfaces were configured. Devices from different subnets communicated successfully through the router.