ACTIVE CLASS WEEK 7

ACTIVITY 1:

Activity 1: Routing - Group Activity

1. Network Setup:

- **Configuration:**

  - **Router0** is connected to **Switch1** and **PC1** on LAN **10.1.1.0/24**.

  - **Router1** is connected to **Switch2** and **PC3** on LAN **198.168.10.0/24**.

  - **Router0** and **Router1** are directly connected to each other.

- **Roles and Responsibilities:**

  - **Nadia:** Manages LAN **10.1.1.0/24** (handles PC1).

  - **Archit:** Manages LAN **198.168.10.0/24** (handles PC3).

  - **Jasveena:** Operates as the data plane for Router0.

  - **Pranika:** Operates as the data plane for Router1.

  - **Gitanshi:** Manages the control plane for both routers.

2. Ping Test:

- **Nadia** (from PC1) will send two ping requests to **Archit** (PC3).

3. Step-by-Step Instructions:

- **Gitanshi** (control plane) will ensure that Router0 and Router1 are exchanging routing information properly.

- **Jasveena** (Router0's data plane) will update the routing table to recognize LAN **198.168.10.0/24** from Router1.

- **Pranika** (Router1’s data plane) will update the routing table to recognize LAN **10.1.1.0/24** from Router0.

- **Nadia** (PC1) initiates a ping to **Archit** (PC3). The ping packet is forwarded to **Jasveena** (Router0).

- **Jasveena** (Router0) examines the routing table and forwards the packet to **Pranika** (Router1).

- **Pranika** (Router1) checks the routing table and forwards the packet to **Archit** (PC3).

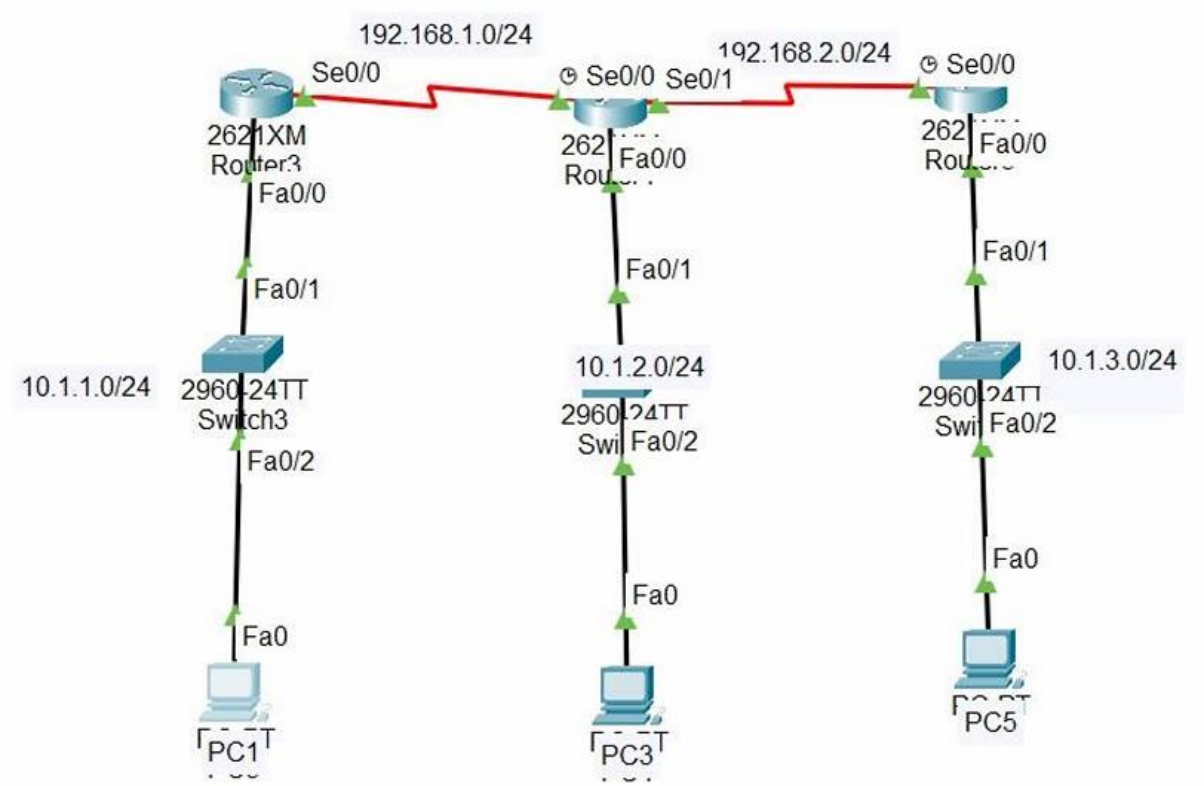
- **Archit** (PC3) receives the ping and sends a reply back to **Nadia** (PC1), following the same route in reverse.

4. Control Plane Responsibilities:

- **Gitanshi** (control plane) is responsible for maintaining up-to-date routing tables, exchanging routing information, and determining the optimal path for data packets. This includes ensuring that Router0 and Router1 are aware of each other’s network information.

ACTIVITY2:

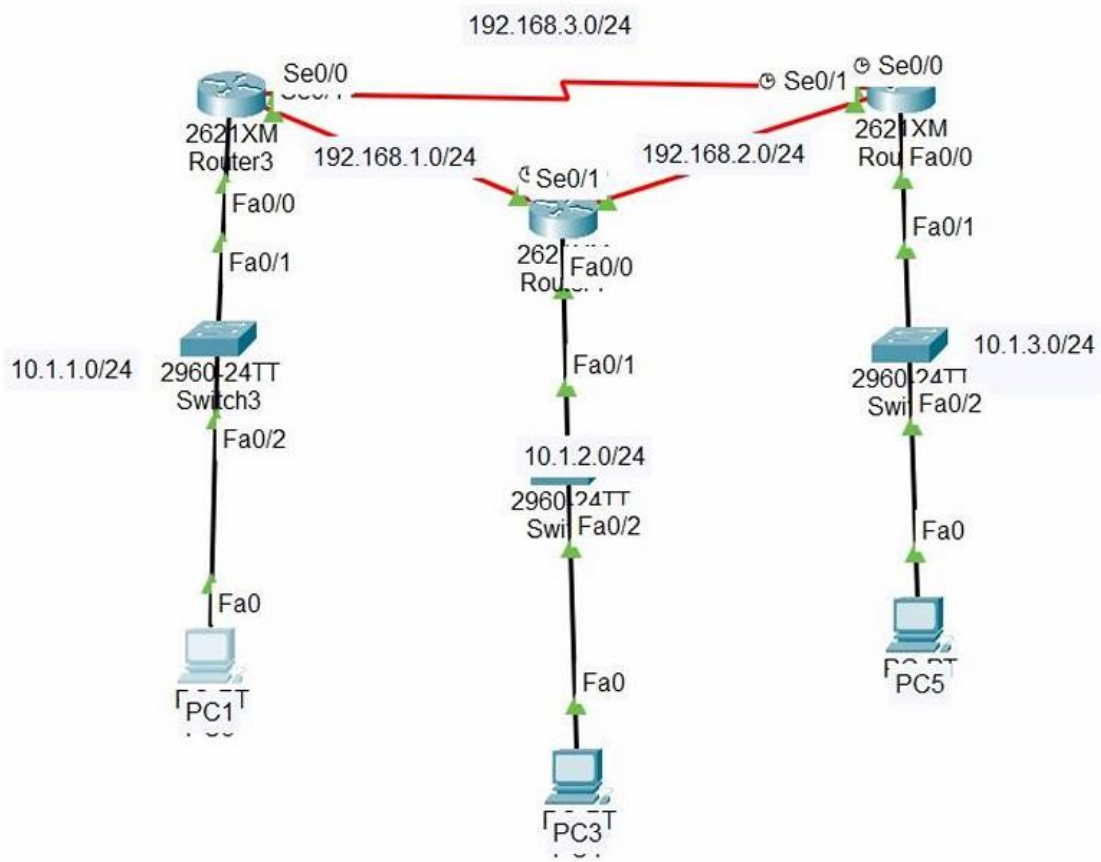
1. The cisco network setup



Now we need to verify that PC1 can communicate with both PC3 and PC5 using PING command and simulations.

```
C:\>ping 10.1.2.1

Pinging 10.1.2.1 with 32 bytes of data:

Reply from 10.1.2.1: bytes=32 time=17ms TTL=126
Reply from 10.1.2.1: bytes=32 time=11ms TTL=126
Reply from 10.1.2.1: bytes=32 time=1ms TTL=126
Reply from 10.1.2.1: bytes=32 time=9ms TTL=126

Ping statistics for 10.1.2.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 17ms, Average = 9ms

C:\>ping 10.1.3.1

Pinging 10.1.3.1 with 32 bytes of data:

Reply from 10.1.3.1: bytes=32 time=16ms TTL=125
Reply from 10.1.3.1: bytes=32 time=2ms TTL=125
Reply from 10.1.3.1: bytes=32 time=19ms TTL=125
Reply from 10.1.3.1: bytes=32 time=2ms TTL=125

Ping statistics for 10.1.3.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 2ms, Maximum = 19ms, Average = 9ms
```

2. Discuss what changes now you need to make to enable the communication between PC1 and PC5 (Tip: what static routes you need to add).
    - The modifications included adding extra WIC-1T cards to both Router 1 and Router 3 to create a direct serial connection between them. In terms of static routing, previously, the next hop for Router 1 and Router 3 pointed to the port numbers of Router 2, which connected to both routers. However, with the new setup, the port numbers for Router 1 and Router 3 are now directly linked to each other, eliminating the need for Router 2 in their communication.



Implement the changes and verify that PC1 and communicate with PC5 using PING command and simulations.

```
C:\>ping 10.1.3.1

Pinging 10.1.3.1 with 32 bytes of data:

Request timed out.
Reply from 10.1.3.1: bytes=32 time=9ms TTL=126
Reply from 10.1.3.1: bytes=32 time=1ms TTL=126
Reply from 10.1.3.1: bytes=32 time=11ms TTL=126

Ping statistics for 10.1.3.1:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 11ms, Average = 7ms

C:\>
```

Activity 3: Alternatives to Static Routing

Question: Is there an alternative to manually configuring the routing table each time connectivity changes?

Answer: To avoid the need for manual updates to the routing table whenever there are changes in network connectivity, dynamic routing protocols can be utilized. Protocols such as RIP (Routing Information Protocol), OSPF (Open Shortest Path First), and EIGRP (Enhanced Interior Gateway Routing Protocol) automatically adjust routing tables in response to network changes, streamlining network management and improving efficiency.

Question: What are the latest routing algorithms available?

Answer: Among the latest routing algorithms are:

- **OSPFv3:** This is an updated iteration of the OSPF protocol, designed specifically for IPv6 networks.
- **BGP-4 (Border Gateway Protocol):** This protocol is utilized for routing in large-scale Internet backbone networks, offering robust path selection and scalability.
- **EIGRP:** An enhanced version of IGRP, EIGRP provides improved convergence times and scalability, making it suitable for modern network environments.

Question: Are there routing algorithms available in Cisco Packet Tracer?

Answer: Cisco Packet Tracer supports several routing algorithms, including RIP v1/v2, OSPFv2, and EIGRP. These protocols can be configured through the graphical interface of the software by accessing the settings for each router.

Question: Use the simulation tool to send a packet from PC1 to PC5. Have you seen any differences compared to the route you have seen in Activity 2? Explain what happened.

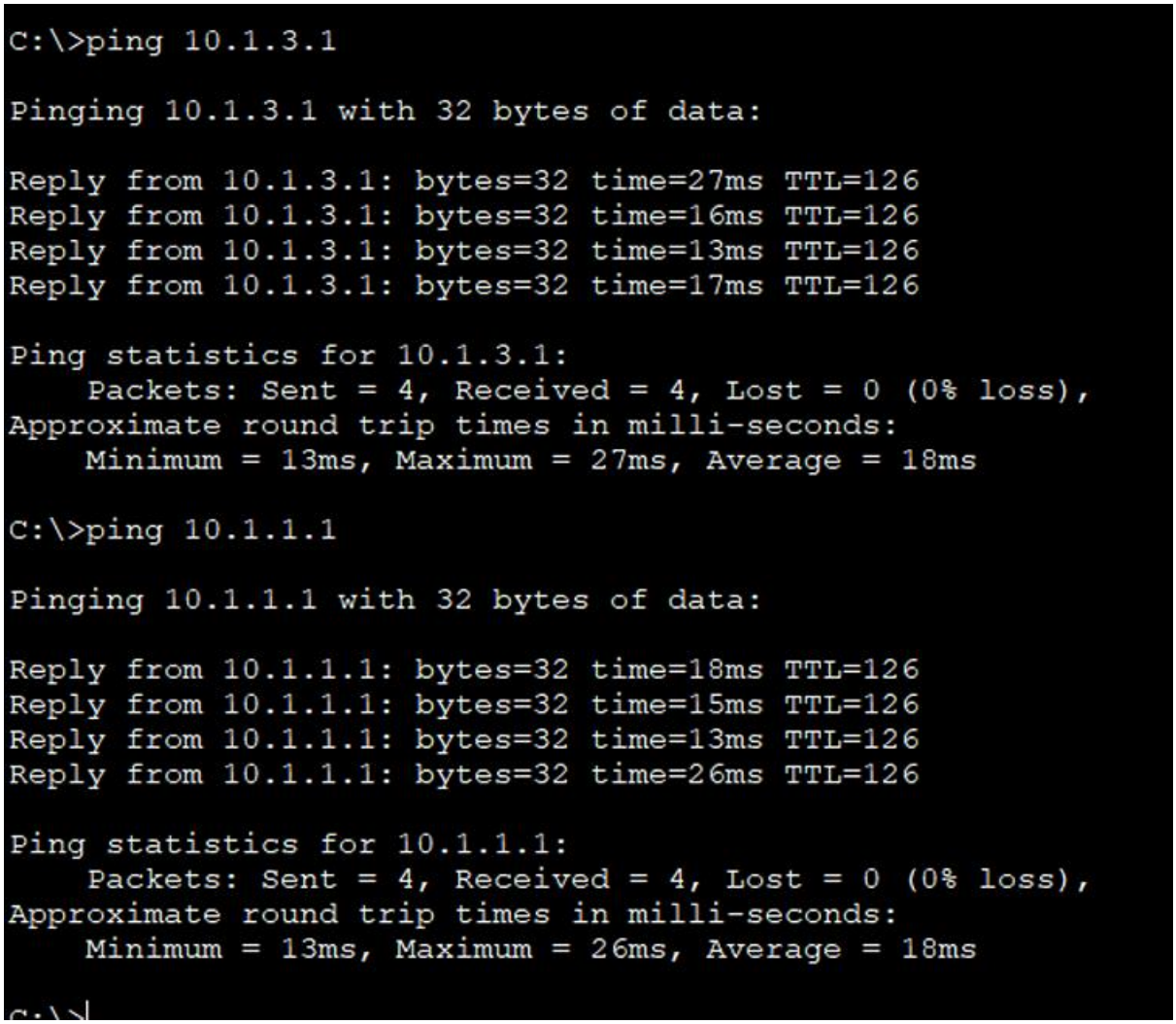
To send a packet form PC1 to PC5 the packet was sent to all the router and then from router 5 to PC5

Question: Verify the network connections using PING (in each PC).

Pinging from PC1

```
C:\>ping 10.1.2.1

Pinging 10.1.2.1 with 32 bytes of data:

Reply from 10.1.2.1: bytes=32 time=21ms TTL=126
Reply from 10.1.2.1: bytes=32 time=12ms TTL=126
Reply from 10.1.2.1: bytes=32 time=10ms TTL=126
Reply from 10.1.2.1: bytes=32 time=3ms TTL=126

Ping statistics for 10.1.2.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 3ms, Maximum = 21ms, Average = 11ms

C:\>ping 10.1.3.1

Pinging 10.1.3.1 with 32 bytes of data:

Reply from 10.1.3.1: bytes=32 time=16ms TTL=126
Reply from 10.1.3.1: bytes=32 time=1ms TTL=126
Reply from 10.1.3.1: bytes=32 time=13ms TTL=126
Reply from 10.1.3.1: bytes=32 time=1ms TTL=126

Ping statistics for 10.1.3.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 16ms, Average = 7ms

C:\>
```

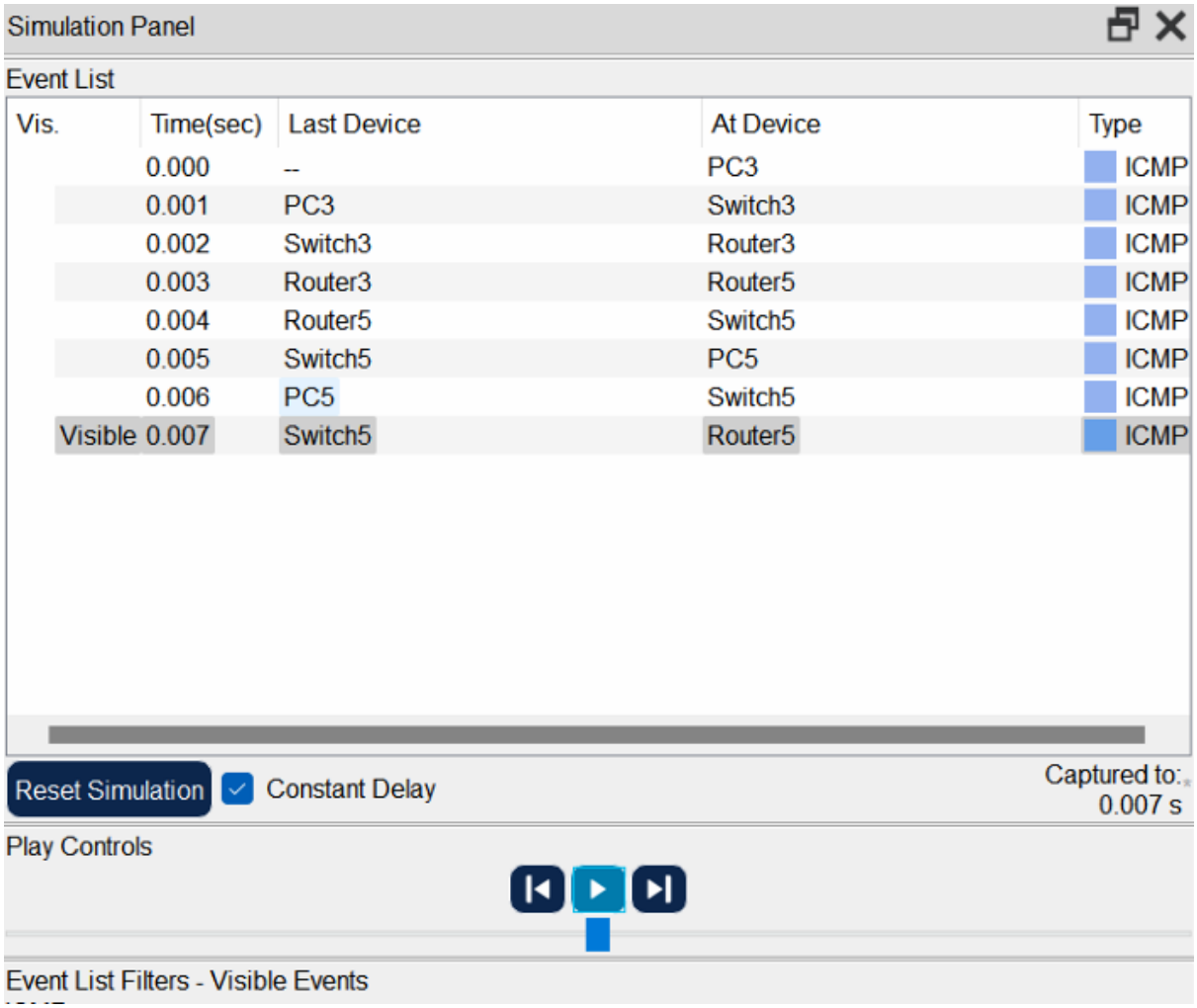
Pinging from PC3



Pinging from PC5

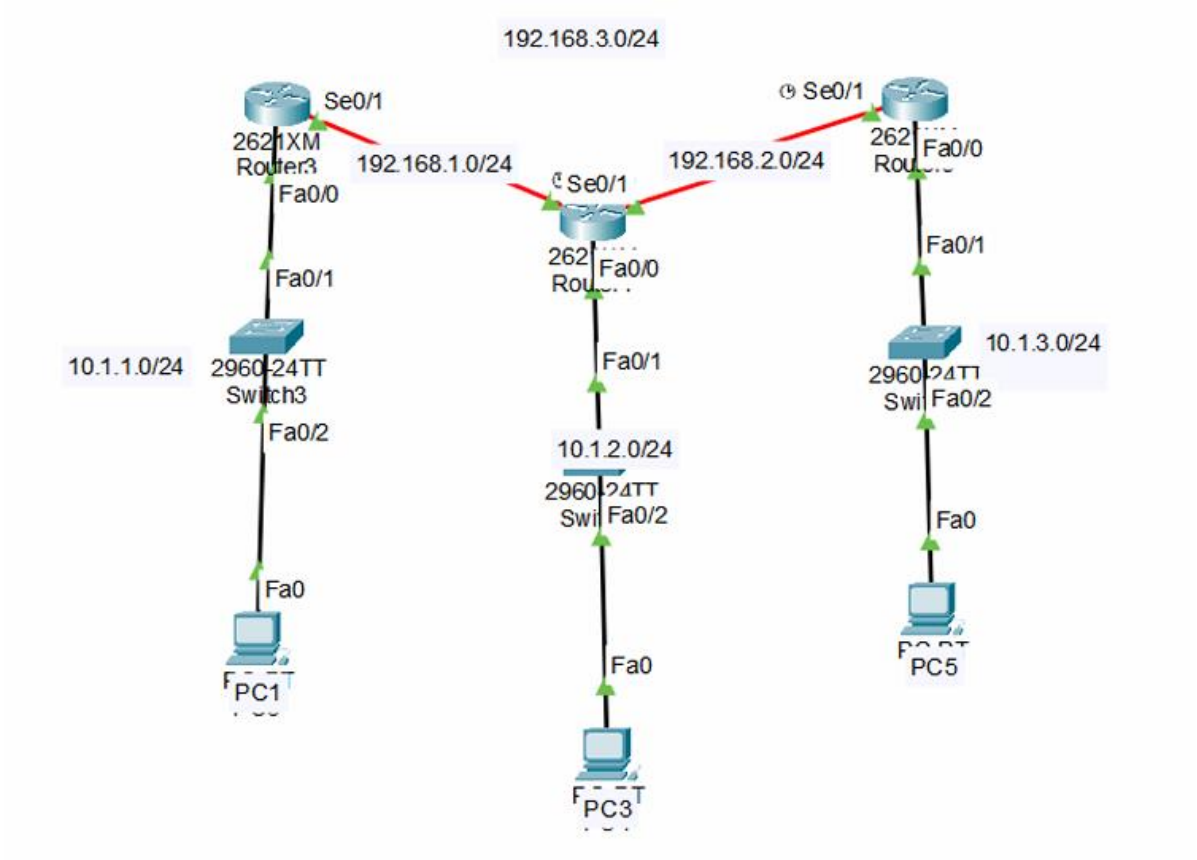
```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 10.1.1.1

Pinging 10.1.1.1 with 32 bytes of data:

Reply from 10.1.1.1: bytes=32 time=12ms TTL=126
Reply from 10.1.1.1: bytes=32 time=15ms TTL=126
Reply from 10.1.1.1: bytes=32 time=1ms TTL=126
Reply from 10.1.1.1: bytes=32 time=10ms TTL=126

Ping statistics for 10.1.1.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 15ms, Average = 9ms

C:\>ping 10.1.2.1

Pinging 10.1.2.1 with 32 bytes of data:

Reply from 10.1.2.1: bytes=32 time=17ms TTL=126
Reply from 10.1.2.1: bytes=32 time=18ms TTL=125
Reply from 10.1.2.1: bytes=32 time=15ms TTL=126
Reply from 10.1.2.1: bytes=32 time=8ms TTL=125

Ping statistics for 10.1.2.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 8ms, Maximum = 18ms, Average = 14ms

C:\>
```

Question : Use the simulation tool to send a packet from PC1 to PC5. Have you seen any differences compared to the route you have seen in Activity 2? Explain what happened.

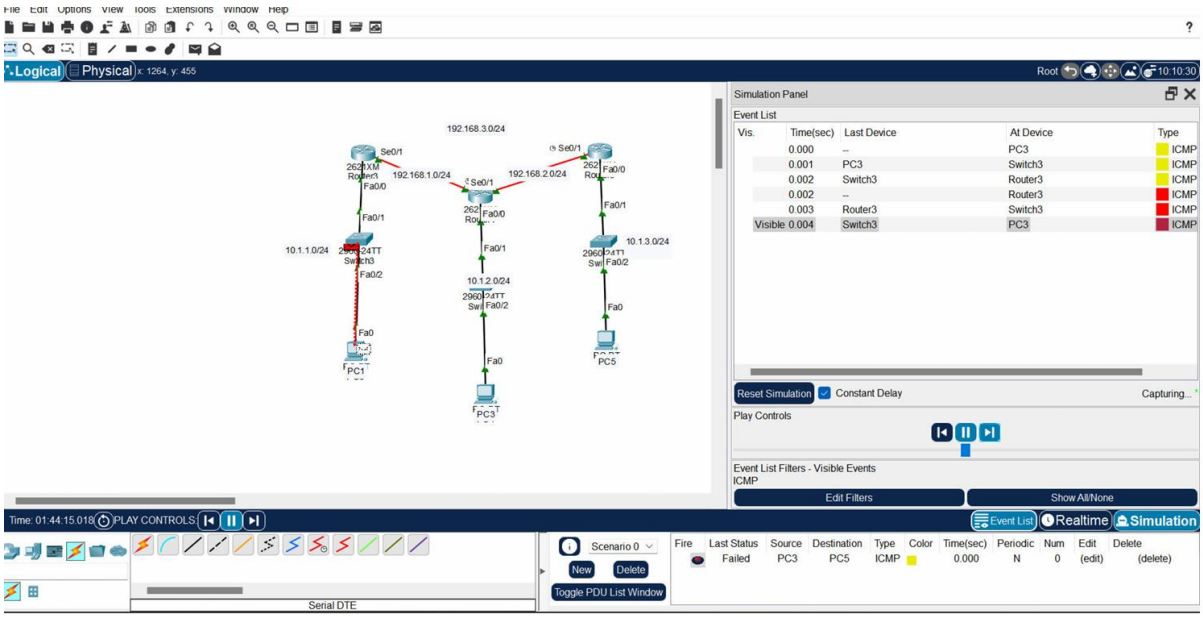


This packet chose the shortest and the easiest route which was going to router 3 and then to router 5, rather than going from router 4 to router 4.

Question : Now remove the connection between router 1 and 3 and use the simulation tool to send a packet from PC1 to PC5 without changing any configuration.



Screenshot of sending the packet :



Here , the packet transfer failed as we can see , this happened when the connection between the relevant server

Observation : we need to manually re-configure if we are using static routing in case when we physically remove a connection. If not re-configured manually, the packet transfer will fail.

ACTIVE CLASS WEEK 8

ACTIVITY 1

1.Role play:

Jasveena: Welcome everyone! Today, we need to discuss how we can manage the increasing number of IoT devices in our network. Let's start with the first question. Do we have enough IP addresses to assign for each device that connects to the network?

Nadia: With the traditional IPv4 addressing scheme, we have approximately 4.3 billion unique IP addresses. However, with the rapid growth of IoT devices, this number is insufficient to provide a unique IP address for every device.

Pranika: That's correct. We need to find a solution to address the shortage of IPv4 addresses.

Archit: One solution is to transition to IPv6, which provides a vastly larger address space. However, this transition is still in progress and not all devices support IPv6 yet.

Gitanshi: Another solution is to use Network Address Translation (NAT). NAT allows multiple devices on a local network to share a single public IP address, conserving the number of public IP addresses needed.

Jasveena: That's a great point. We can also use the Dynamic Host Configuration Protocol (DHCP) to dynamically assign IP addresses to devices on the network. This helps manage IP address allocation efficiently. Archit: And don't forget about the Internet Control Message Protocol (ICMP). ICMP is used for diagnostic and error-reporting purposes in network communication. It helps us troubleshoot and diagnose network problems.

Gitanshi: Exactly. By using these protocols together, we can effectively manage the IP address space and ensure smooth network operations.

• Discussion:

Do we have enough IP addresses to assign for each device that connect to the network?

Answer: With the traditional IPv4 addressing scheme, we have approximately 4.3 billion unique IP addresses. However, with the rapid growth of IoT devices, this number is insufficient to provide a unique IP address for every device.

Do we have any solution?

Yes, there are several solutions to address the shortage of IPv4 addresses:

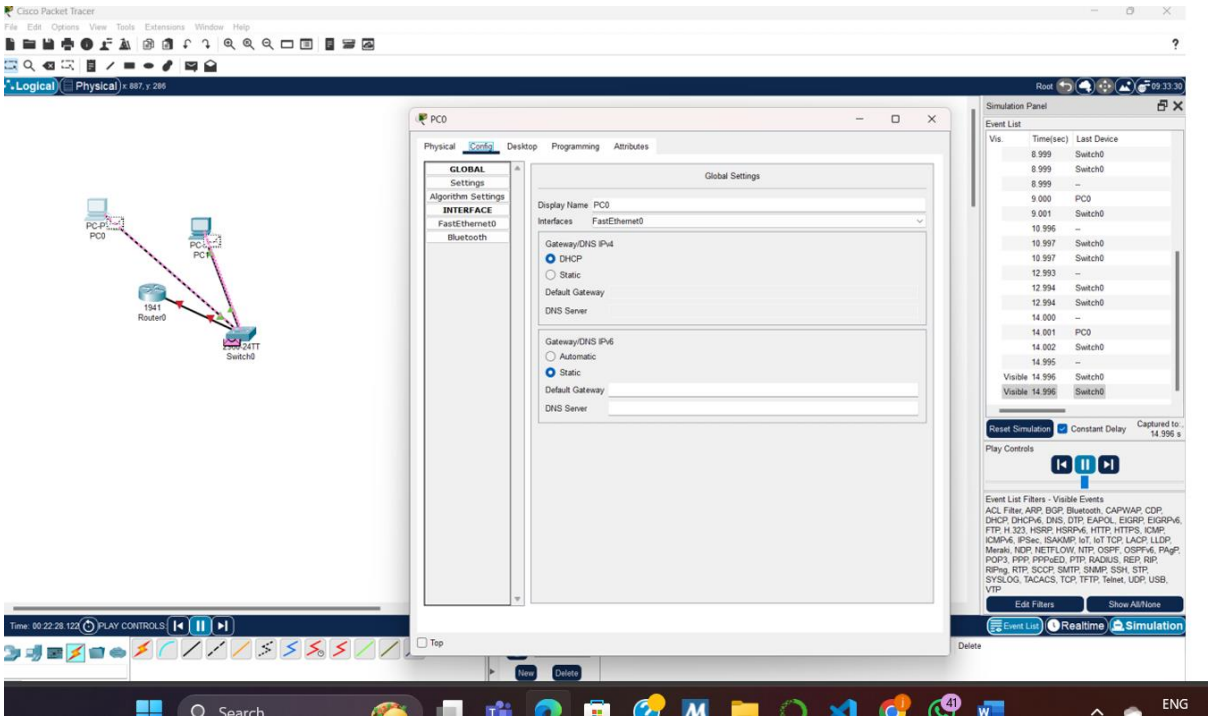
• IPv6: The most comprehensive solution is transitioning to IPv6, which provides a vastly larger address space (approximately 340 undecillion addresses).

• Network Address Translation (NAT): NAT allows multiple devices on a local network to share a single public IP address, conserving the number of public IP addresses needed.

• Dynamic Host Configuration Protocol (DHCP): DHCP dynamically assigns IP addresses to devices on a network, reusing addresses when devices are not actively connected

Explain a protocol that we can use along with IPv4 to conserve the global IP address space.

Answer: Network Address Translation (NAT): NAT is a protocol used to map multiple private IP addresses to a single public IP address (or a few public IP addresses). This allows multiple devices on a local network to access the internet using a single public IP address, effectively conserving the global IP address space. NAT is commonly used in home and office networks.



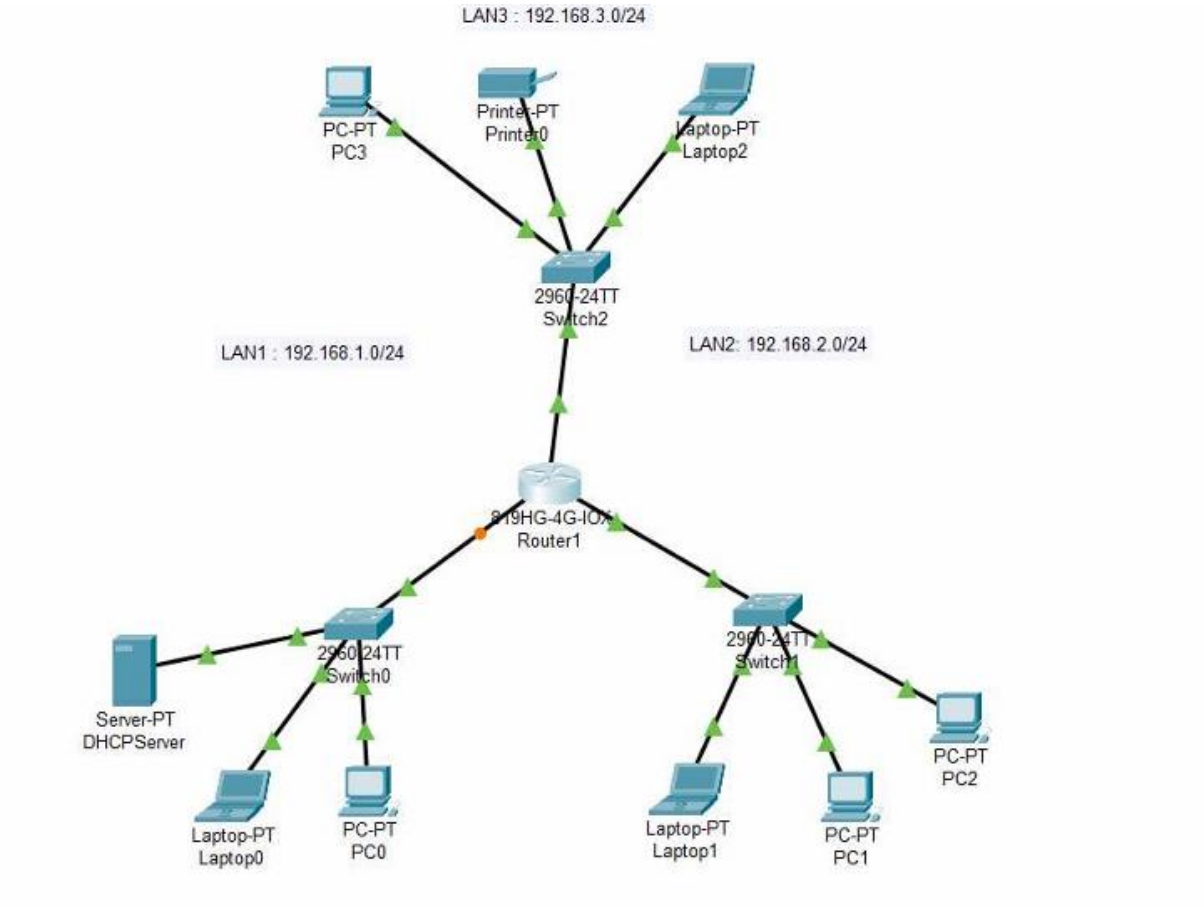
ACTIVITY 2

Role Play Scenario

1. **Gitanshi (Router):** “Broadcasting DHCPDISCOVER to find DHCP clients.”
2. **Nadia (Switch):** “Receiving and forwarding DHCPDISCOVER to all devices on the network.”
3. **Pranika (PC):** “Receiving DHCPDISCOVER and responding with a DHCPOFFER message.”
4. **Nadia (Switch):** “Forwarding DHCPOFFER from Pranika to the DHCP server (Gitanshi).”
5. **Gitanshi (Router):** “Receiving DHCPOFFER and sending a DHCPREQUEST message.”
6. **Jasveena (Printer):** “Receiving DHCPREQUEST and responding with a DHCPACK message.”
7. **Archit (Network Storage Device):** “Receiving IP address from DHCPACK, connectivity established.”

TIME	GITANSHI (ROUTER)	Nadia (Switch)	Pranika (PC)	Jasveena (Printer)	Archit (Network Storag	

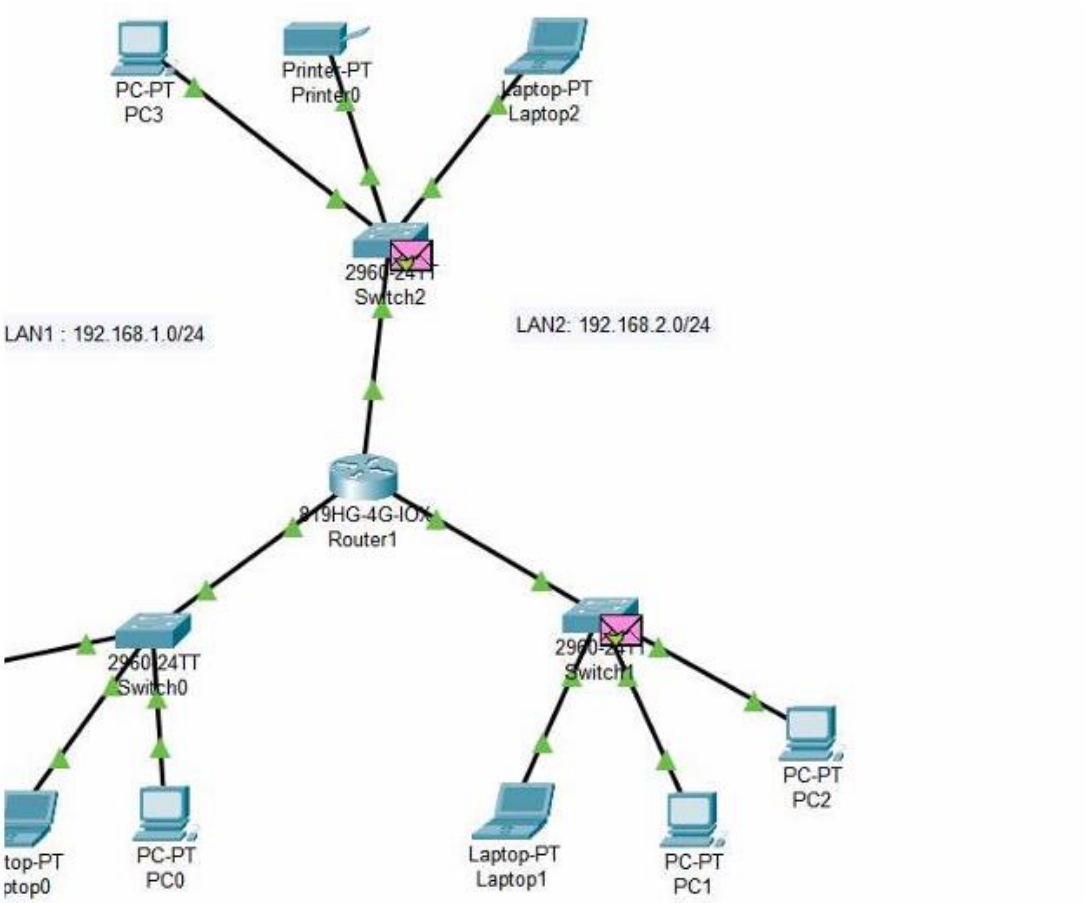
					e Device)	
T1	Broadcasting DHCPDISCOVER	Receiving and forwarding DHCPDISCOVER				
T2		Forwarding DHCPDISCOVER	Receiving and responding with DHCPOFFER			
T3		Forwarding DHCPOFFER				
T4	Receiving DHCPOFFER					
T5	Sending DHCPREQUEST			Receiving DHCPREQUEST		
T6				Responding with DHCPACK		
T7						Receiving IP address, connectivity established

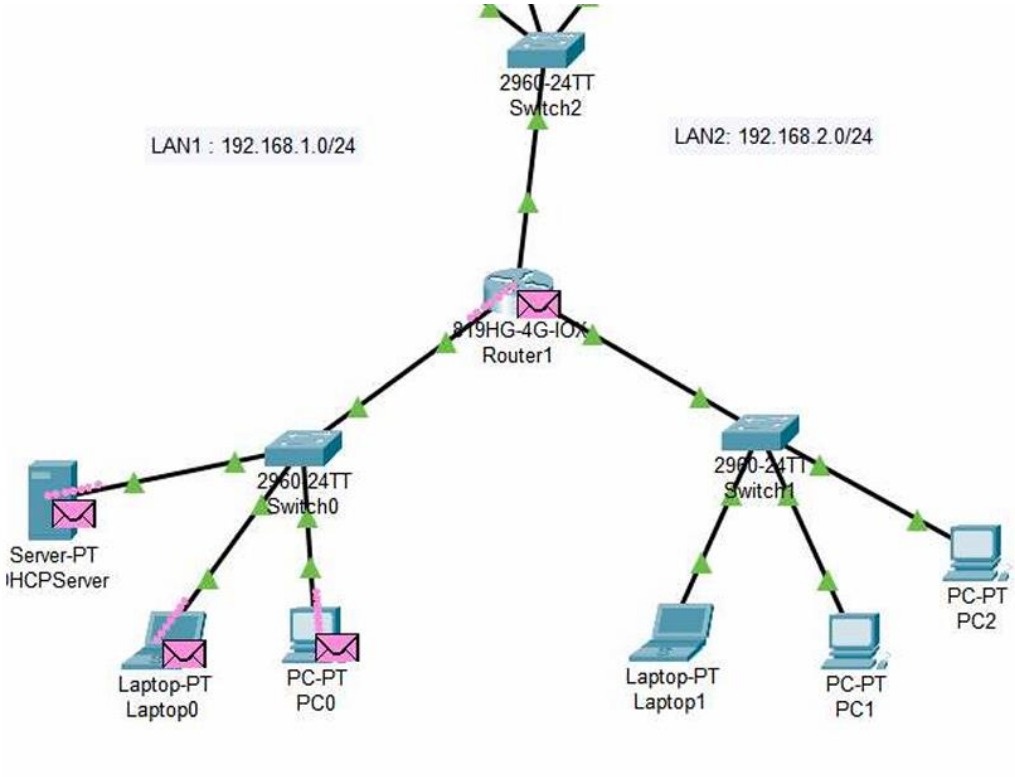

LAN3 : 192.168.3.0/24
PC-PT
PC3
Printer-PT
Printer0
Laptop-PT
Laptop2
2960-24TT
Switch2
LAN1 : 192.168.1.0/24
LAN2: 192.168.2.0/24
Router1
2960-24TT
Switch0
2960-24TT
Switch1
Server-PT
DHCPServer
Laptop-PT
Laptop0
PC-PT
PC0
Laptop-PT
Laptop1
PC-PT
PC1
PC-PT
PC2

Physical Config CLI Attributes

IOS Command Line Interface

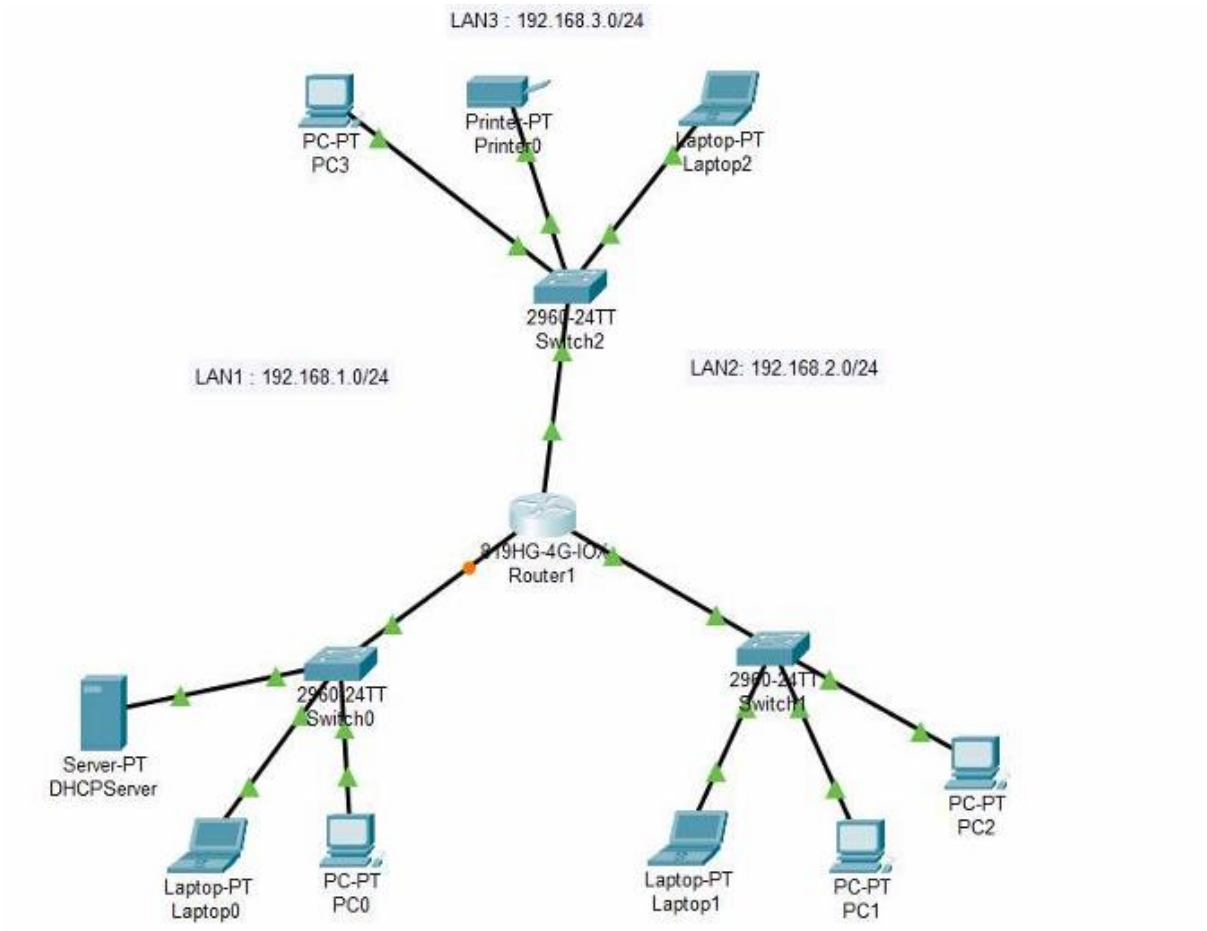
```
Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up
exit
Router(config)#interface gigabitethernet0/1
Router(config-if)#ip address 192.168.2.1 255.255.255.0
Router(config-if)#no shutdown

Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/1, changed state to up
exit
Router(config)#
Router(config)#interface gigabitethernet0/2
%Invalid interface type and number
Router(config)#ip address 192.168.3.1 255.255.255.0
                  ^
% Invalid input detected at '^' marker.

Router(config)#
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed state to up

Router(config)#configure terminal
                  ^
% Invalid input detected at '^' marker.

Router(config)#ip dhcp pool LAN1
Router(dhcp-config)#network 192.168.1.0 255.255.255.0
Router(dhcp-config)#default-router 192.168.1.1
Router(dhcp-config)#dns-server 8.8.8.8
Router(dhcp-config)#exit
Router(config)#ip dhcp pool LAN2
Router(dhcp-config)#network 192.168.2.0 255.255.255.0
Router(dhcp-config)#default-router 192.168.2.1
Router(dhcp-config)#dns-server 8.8.8.8
Router(dhcp-config)#exit
Router(config)#ip dhcp pool LAN3
Router(dhcp-config)#network 192.168.3.0 255.255.255.0
Router(dhcp-config)#default-router 192.168.3.1
Router(dhcp-config)#dns-server 8.8.8.8
Router(dhcp-config)#exit
Router(config)#
```

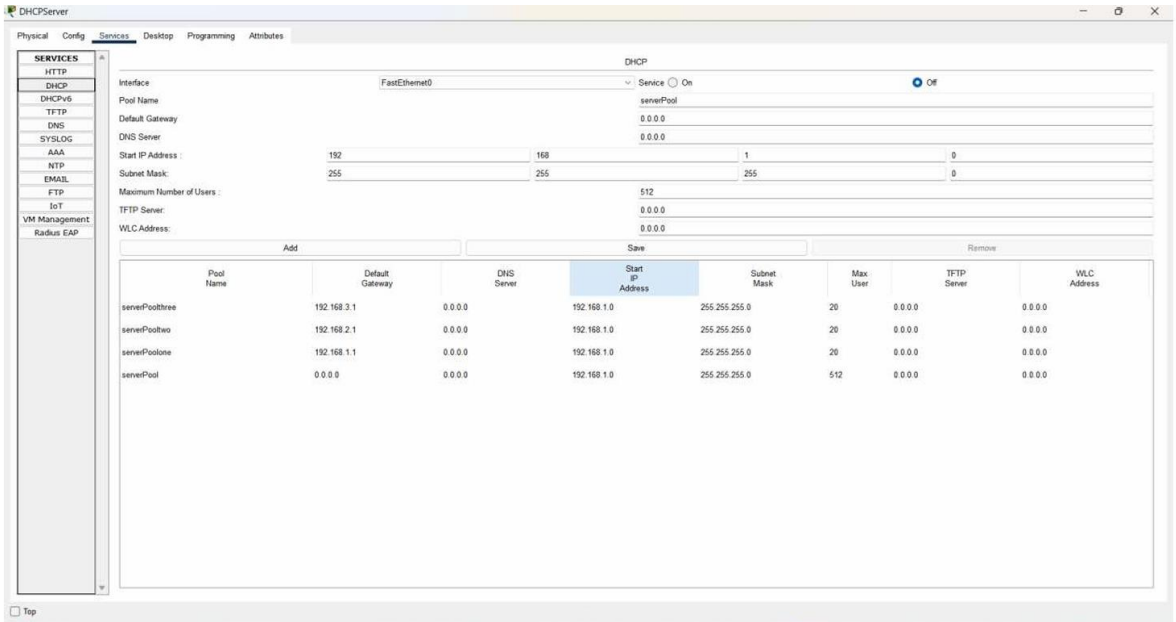

PC-PT
PC3
Printer-PT
Printer0
Laptop-PT
Laptop2
2960-24TT
Switch2
LAN1 : 192.168.1.0/24
LAN2: 192.168.2.0/24
819HG-4G-IOX
Router1
2960-24TT
Switch0
top-PT
ptop0
PC-PT
PC0
Laptop-PT
Laptop1
PC-PT
PC1
PC-PT
PC2

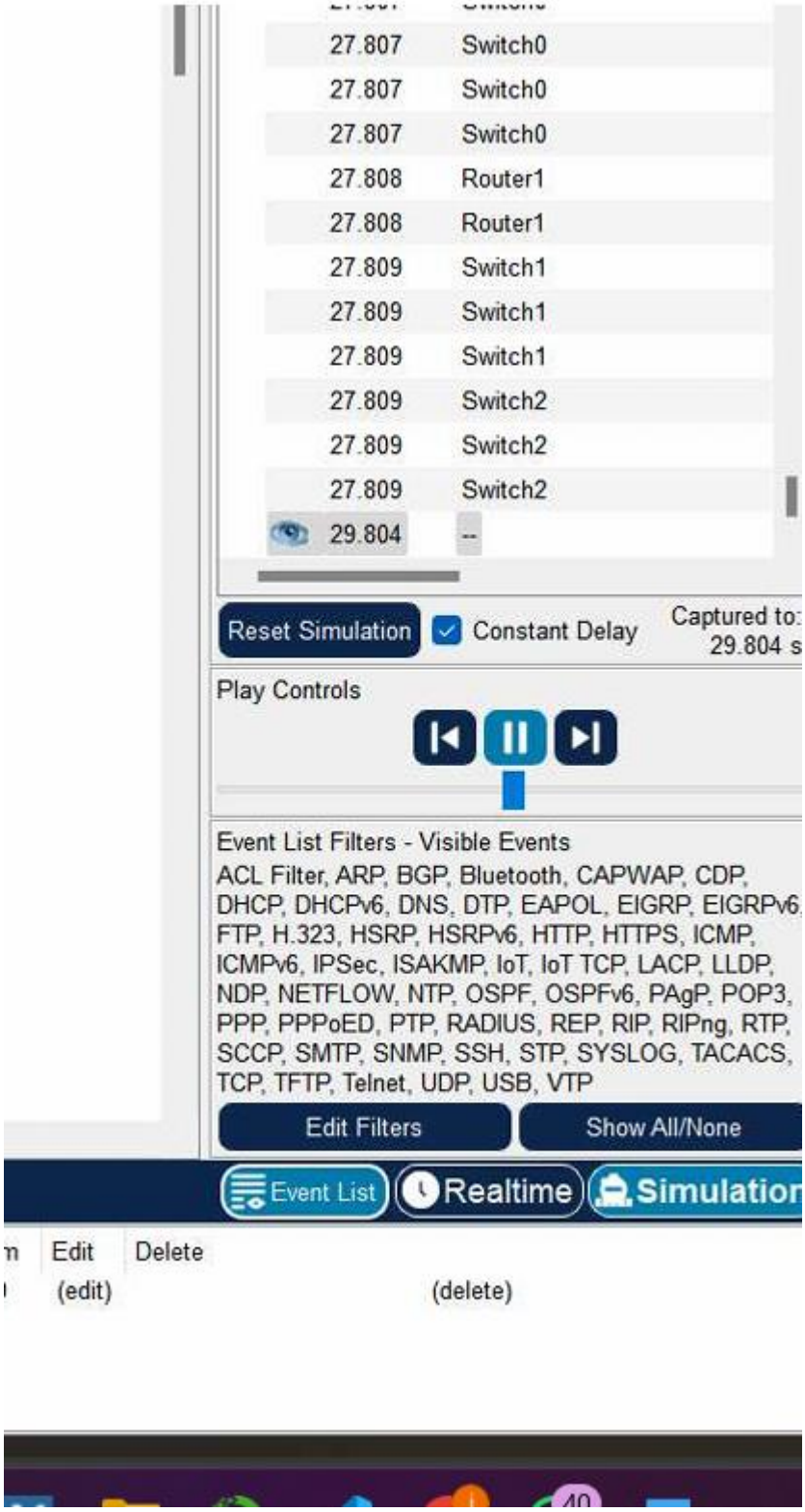

2960-24TT
Switch2
LAN1 : 192.168.1.0/24
LAN2: 192.168.2.0/24
819HG-4G-IOX
Router1
2960-24TT
Switch0
Server-PT
HCPServer
Laptop-PT
Laptop0
PC-PT
PC0
Laptop-PT
Laptop1
PC-PT
PC1
PC-PT
PC2

ACTIVITY 3

1.


LAN3 : 192.168.3.0/24
PC-PT
PC3
Printer-PT
Printer0
Laptop-PT
Laptop2
2960-24TT
Switch2
LAN1 : 192.168.1.0/24
LAN2: 192.168.2.0/24
Router1
2960-24TT
Switch0
2960-24TT
Switch1
Server-PT
DHCPServer
PC-PT
PC2
Laptop-PT
Laptop0
PC-PT
PC0
Laptop-PT
Laptop1
PC-PT
PC1


DHCPServer
Physical
Config
Services
Desktop
Programming
Attributes
SERVICES
HTTP
DHCP
DHCPv6
TFTP
DNS
SYSLOG
AAA
NTP
EMAIL
FTP
IoT
VM Management
Radius EAP
DHCP
Interface
FastEthernet0
Service
On
Off
Pool Name
serverPool
Default Gateway
0.0.0.0
DNS Server
0.0.0.0
Start IP Address :
192
168
1
0
Subnet Mask:
255
255
255
0
Maximum Number of Users :
512
TFTP Server:
0.0.0.0
WLC Address:
0.0.0.0
Add
Save
Remove
Pool Name
Default Gateway
DNS Server
Start IP Address
Subnet Mask
Max User
TFTP Server
WLC Address
serverPoolthree
192.168.3.1
0.0.0.0
192.168.1.0
255.255.255.0
20
0.0.0.0
0.0.0.0
serverPooltwo
192.168.2.1
0.0.0.0
192.168.1.0
255.255.255.0
20
0.0.0.0
0.0.0.0
serverPoolone
192.168.1.1
0.0.0.0
192.168.1.0
255.255.255.0
20
0.0.0.0
0.0.0.0
serverPool
0.0.0.0
0.0.0.0
192.168.1.0
255.255.255.0
512
0.0.0.0
0.0.0.0
Top


27.807 Switch0
27.807 Switch0
27.807 Switch0
27.808 Router1
27.808 Router1
27.809 Switch1
27.809 Switch1
27.809 Switch1
27.809 Switch2
27.809 Switch2
27.809 Switch2
29.804 --
Reset Simulation
Constant Delay
Captured to:
29.804 s
Play Controls
Event List Filters - Visible Events
ACL Filter, ARP, BGP, Bluetooth, CAPWAP, CDP, DHCP, DHCPv6, DNS, DTP, EAPOL, EIGRP, EIGRPv6, FTP, H.323, HSRP, HSRPv6, HTTP, HTTPS, ICMP, ICMPv6, IPSec, ISAKMP, IoT, IoT TCP, LACP, LLDP, NDP, NETFLOW, NTP, OSPF, OSPFv6, PAgP, POP3, PPP, PPPoED, PTP, RADIUS, REP, RIP, RIPng, RTP, SCCP, SMTP, SNMP, SSH, STP, SYSLOG, TACACS, TCP, TFTP, Telnet, UDP, USB, VTP
Edit Filters
Show All/None
Event List
Realtime
Simulation
Edit
Delete
(edit)
(delete)

PDU Information at Device: PC2

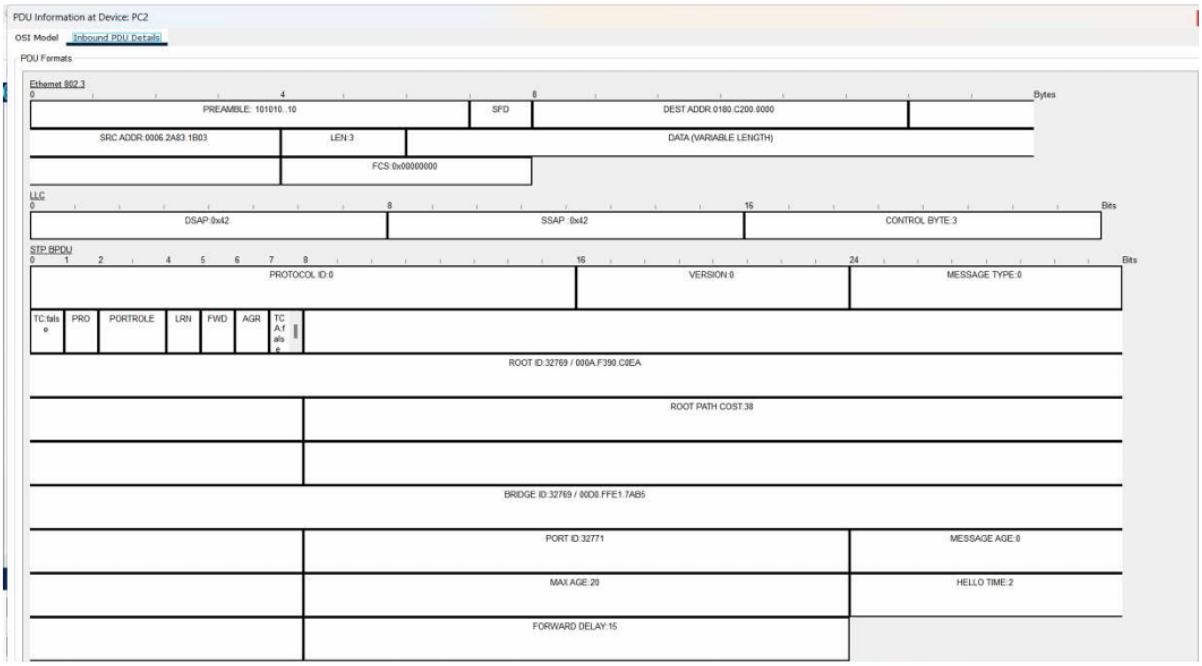
OSI Model | Inbound PDU Details

At Device: PC2
Source: Switch0
Destination: STP Multicast Address

In Layers	Out Layers
Layer7	Layer7
Layer6	Layer6
Layer5	Layer5
Layer4	Layer4
Layer3	Layer3
Layer 2: IEEE 802.3 Header 0006.2A83.1B03 >> 0180.C200.0000 LLC STP BPDU	Layer2
Layer 1: Port FastEthernet0	Layer1

1. FastEthernet0 receives the frame.

Challenge Me | << Previous Layer | Next Layer >>



QUES 5 AND 6

ICMP Message Format and Details

ICMP Message Format:

The Internet Control Message Protocol (ICMP) message format consists of the following fields:

1. **Type:** Indicates the type of the ICMP message (e.g., Echo Request, Echo Reply).

2. **Code:** Provides further information about the type (e.g., for Type 3 (Destination Unreachable), Code 0 means "Network Unreachable").

3. **Checksum:** Error-checking data to ensure the message has not been corrupted.

4. **Identifier:** Used to match requests and replies (mainly for Echo Requests and Replies).

5. **Sequence Number:** Used to identify multiple messages or packets of the same type.

6. **Data:** Contains additional information relevant to the ICMP message type, such as the payload of an Echo Request or Reply.

ICMP Message Types:

- **Type 0:** Echo Reply

- **Type 3:** Destination Unreachable

- **Type 8:** Echo Request

- **Type 11:** Time Exceeded (used by traceroute)

Changes in ICMP Messages between Ping and Traceroute

Ping:

- **Type:** 8 (Echo Request) for sending, 0 (Echo Reply) for responses.

- **Code:** Generally 0 for both Echo Requests and Replies.

- **Purpose:** Used to check if a host is reachable and to measure round-trip time.

Traceroute:

- **Type:** 11 (Time Exceeded) for reporting hops along the path to the destination.

- **Code:** Typically 0 for TTL Expired in Transit.

- **Purpose:** Determines the path packets take to reach the destination by sending packets with incrementally increasing TTL (Time-to-Live) values and recording the ICMP Time Exceeded messages returned by intermediate routers.

Behavior in Traceroute vs. Ping

- **Ping:** Sends Echo Request packets to the destination and waits for Echo Reply packets. It measures the round-trip time and checks reachability.

- **Traceroute:** Sends packets with increasing TTL values. Each router along the path decrements the TTL and, when it reaches zero, sends back an ICMP Time Exceeded message. This process helps in mapping out the route by identifying each hop between the source and destination.