



**Lab 02:** Network Performance, Layers & Security  
**Course:** Networks System Design  
**Name:** Do Davin  
**Student ID:** P20230018  
**Instructor:** Mr. Kuy Movsun  
**Due Date:** October 28, 2025 (12:00 AM)

# Activity 1 — Network Performance: Delay, Loss & Throughput

## Part A — Manual Calculation (Math-Safe Markdown)

Given Information

Parameter	Symbol	Value
File size	F	5 MB = $5 \times 10^6$ bytes
Link rate	R	10 Mbps = $10 \times 10^6$ bits/s
Packet size	L	1500 bytes = $1500 \times 8 = 12,000$ bits
Round-trip time	RTT	50 ms (not directly used below)
Packet loss	p	2% = 0.02

### 1) Transmission Delay for One 1500-Byte Packet

**Formula (text):**  $d_{trans} = L / R$   
**Substitute:**  $d_{trans} = 12,000 \text{ bits} / 10,000,000 \text{ bits/s}$   
**Result:**  $d_{trans} = 0.0012 \text{ s} = 1.2 \text{ ms}$

### 2) Total Transfer Time for 5 MB File (Ideal)

**Formula (text):**  $\text{Transfer time} = \text{File size (bits)} / R$   
**Convert size:**  $5 \times 10^6 \text{ bytes} \times 8 = 40 \times 10^6 \text{ bits}$   
**Substitute:**  $\text{Transfer time} = 40 \times 10^6 / 10 \times 10^6 = 4 \text{ s}$   
**Result:** 4 seconds

### 3) Effective Throughput with 2% Loss

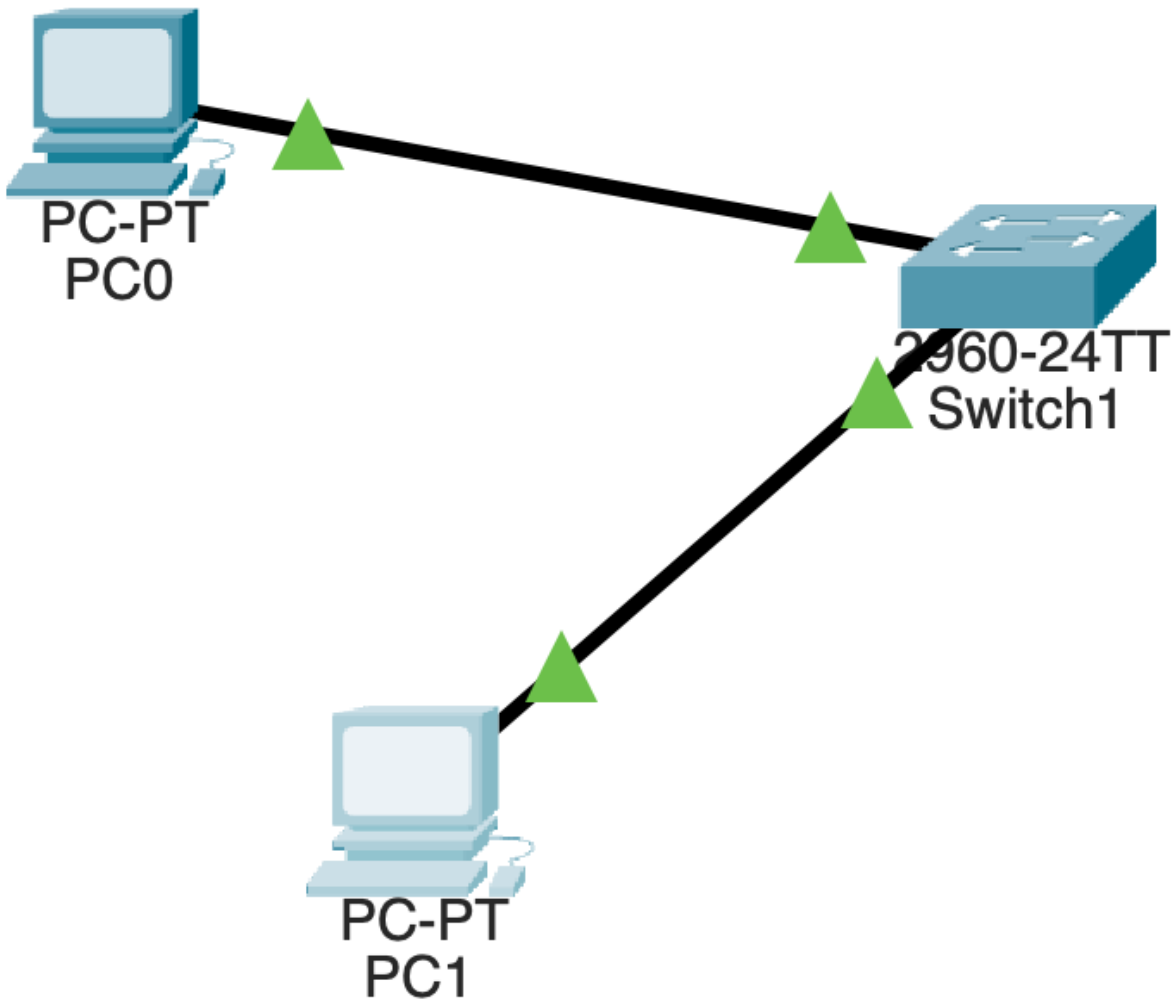
**Formula (text):**  $\text{Effective Throughput} = R \times (1 - \text{loss rate})$   
**Substitute:**  $10 \text{ Mbps} \times (1 - 0.02) = 9.8 \text{ Mbps}$   
**Result:** 9.8 Mbps

Final Answers

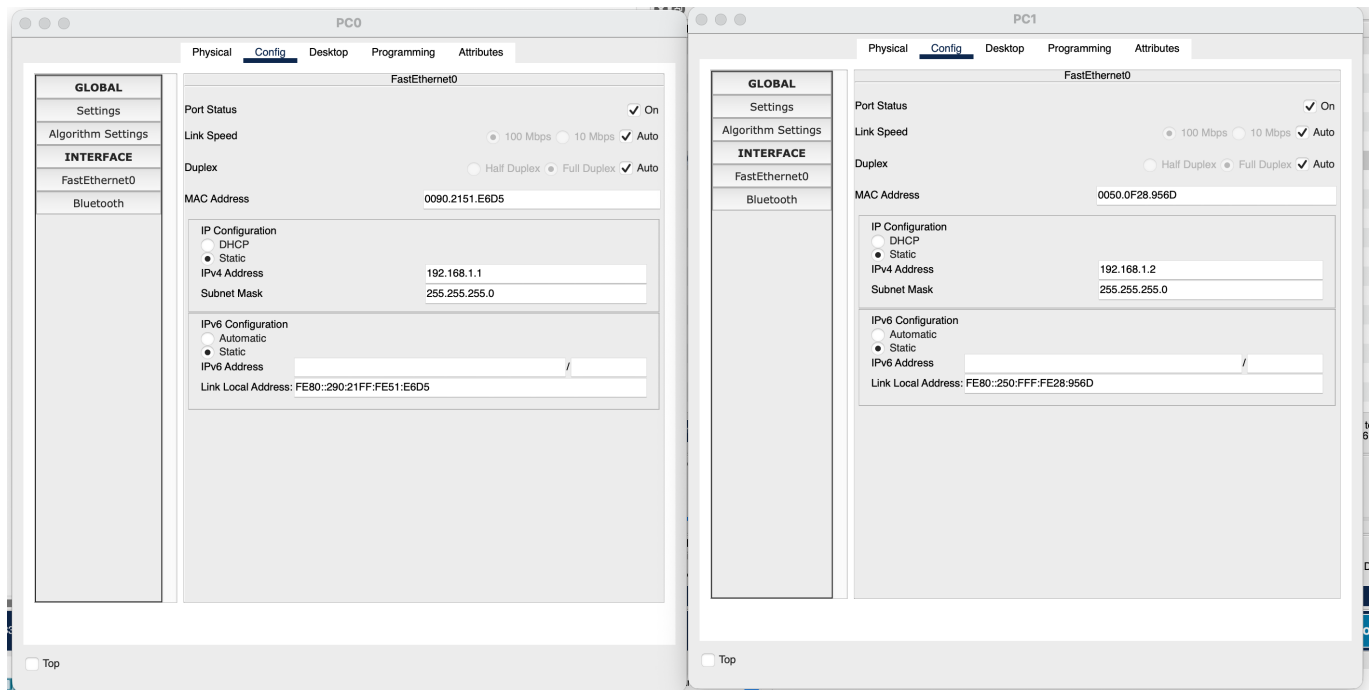
Calculation	Formula	Result
Transmission delay (one packet)	$d_{trans} = L / R$	1.2 ms
Ideal transfer time (5 MB)	File size / R	4 s
Effective throughput (2% loss)	$R \times (1 - \text{loss rate})$	9.8 Mbps

Part B. Packet Tracer Measurement

1. Create a simple topology: PC0 — Switch — PC1 with straight-through cables.



2. Assign IPs: PC0 192.168.1.1/24, PC1 192.168.1.2/24.



### 3. Test connectivity from PC0 using ping 192.168.1.2.

```
C:\>ping 192.168.1.2

Pinging 192.168.1.2 with 32 bytes of data:

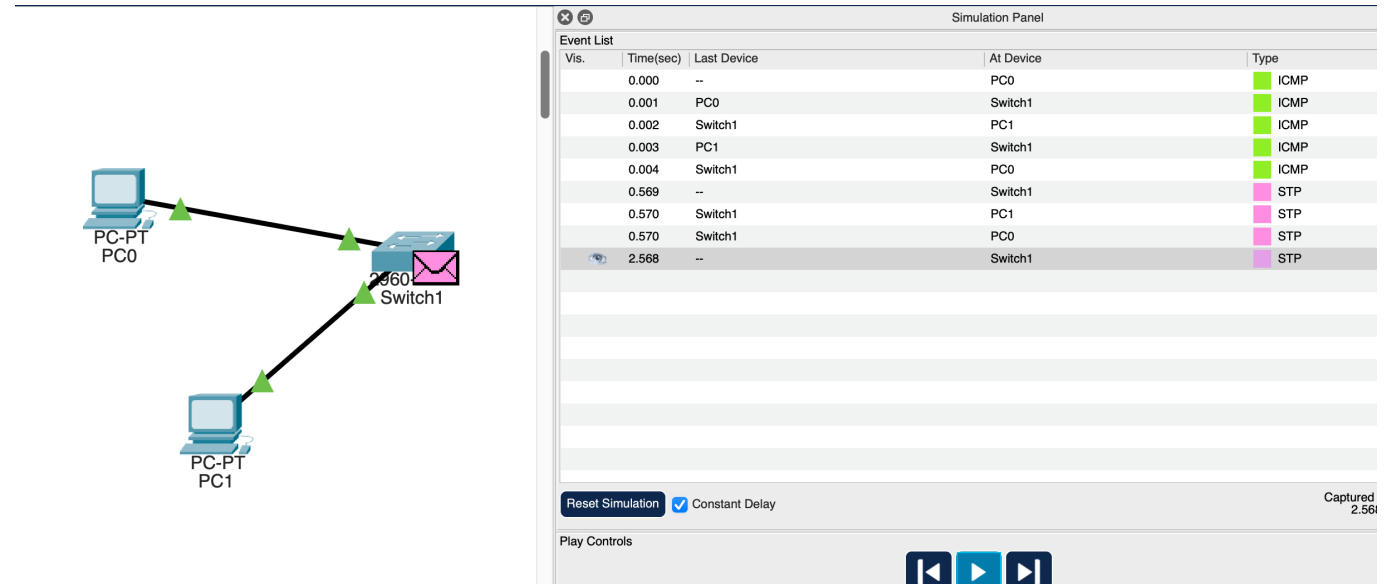
Reply from 192.168.1.2: bytes=32 time=4ms TTL=128
Reply from 192.168.1.2: bytes=32 time<1ms TTL=128
Reply from 192.168.1.2: bytes=32 time<1ms TTL=128
Reply from 192.168.1.2: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.1.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 4ms, Average = 1ms
```

### 4. Switch to Simulation Mode → use the Simple PDU (envelope icon) from PC0 → PC1.

### 5. Open the Event List and record the time when the Echo Request leaves PC0 and when the Echo Reply returns to PC0 (this is your RTT).

### 6. Assume each Simple PDU ≈ 600 bits (typical ICMP ping).



7. Calculate throughput using  $\text{Throughput (bps)} = \text{Packet Size (bits)} / \text{RTT (seconds)}$ .

Results

Test	Packet Size (bits)	RTT (ms)	Throughput (bps)	Loss (%)
1	600	4	150 000	0
2	600	4	150 000	0

Discussion Prompts

1. Delays observed in Simulation Mode

- **Transmission Delay:** Time to push bits onto the link.
- **Propagation Delay:** Time for the signal to travel through the medium (e.g., copper cable).
- **Processing Delay:** Time taken by devices (PCs, switches) to process or forward packets.
- **Queueing Delay:** Appears when many PDUs are sent quickly, causing temporary waiting in the switch buffer.

2. Effect of Sending Multiple PDUs Quickly

- Increases **queueing delay** as packets wait in line for transmission.
- May cause **packet loss** if the switch buffer becomes full.
- Results in **fluctuating (jittery) RTT values**, as some packets take longer to process.

3. Impact of Link Speed on Throughput

- **Higher link rate** → **lower transmission delay** and higher achievable throughput.
- With the same RTT, **throughput scales directly with link speed**.
- **Losses or congestion** reduce the *effective* throughput by:  
**Effective Throughput =  $R \times (1 - \text{loss rate})$**

# Activity 2 — Protocol Layers & Encapsulation

---

## Option A: Cisco Packet Tracer

### 1. Use the same PC0 – Switch – PC1 topology in Simulation Mode.

---

Device	IP Address	Subnet Mask
PC0	192.168.1.1	255.255.255.0
PC1	192.168.1.2	255.255.255.0

Connection Type: Copper Straight-Through Cable

- PC0 -> Switch1 (FastEthernet0/1)
- PC1 -> Switch1 (FastEthernet0/2)

### 2. Send a Simple PDU from PC0 → PC1.

#### Outbound PDU Details

PDU Information at Device: PC0

OSI Model

Outbound PDU Details

At Device: PC0  
Source: PC0  
Destination: PC1

In Layers

Layer7

Layer6

Layer5

Layer4

Layer3

Layer2

Layer1

Out Layers

Layer7

Layer6

Layer5

Layer4

Layer 3: IP Header Src. IP: 192.168.1.1, Dest. IP: 192.168.1.2  
ICMP Message Type: 8

Layer 2: Ethernet II Header  
0090.2151.E6D5 >>  
0050.0F28.956D

Layer 1: Port(s): FastEthernet0

1. The Ping process starts the next ping request.  
2. The Ping process creates an ICMP Echo Request message and sends it to the lower process.  
3. The source IP address is not specified. The device sets it to the port's IP address.  
4. The device sets TTL in the packet header.  
5. The destination IP address is in the same subnet. The device sets the next-hop to destination.

Challenge Me

<< Previous Layer

Next Layer >>

## Inbound PDU Details

6 / 11

PDU Information at Device: Switch1

OSI Model

Inbound PDU Details

Outbound PDU Details

At Device: Switch1  
Source: PC0  
Destination: PC1

In Layers

Layer7

Layer6

Layer5

Layer4

Layer3

Layer 2: Ethernet II Header  
0050.0F28.956D >>  
0090.2151.E6D5

Layer 1: Port FastEthernet0/2

Out Layers

Layer7

Layer6

Layer5

Layer4

Layer3

Layer 2: Ethernet II Header  
0050.0F28.956D >>  
0090.2151.E6D5

Layer 1: Port(s): FastEthernet0/1

1. FastEthernet0/2 receives the frame.

Challenge Me

<< Previous Layer

Next Layer >>

3. In the Event List, click the packet → PDU Information.

Layer	Protocol	Header Information	Description
Layer 3 (Network)	IP	Src: 192.168.1.1 Dest: 192.168.1.2 ICMP Type: 8	Adds IP header for logical addressing

Layer	Protocol	Header Information	Description
Layer 2 (Data Link)	Ethernet II	Src MAC: 0090.2151.E6D5 Dest MAC: 0050.0F28.956D	Adds Ethernet frame for local delivery
Layer 1 (Physical)	FastEthernet0	Port: FastEthernet0	Converts data to bits for transmission

## Encapsulation Order:

ICMP -> IP -> Ethernet

The Ping (ICMP Echo Request) message is wrapped in an IP packet, then inside an Ethernet frame before being sent out through the physical port.

### 4. Review Outbound PDU Details (headers added) and Inbound PDU Details (headers removed).

Direction	Layer	Protocol	Description
Inbound	L2	Ethernet II Header 0050.0F28.956D >> 0090.2151.E6D5	Switch receives the frame on <b>FastEthernet0/2</b>
Outbound	L2	Ethernet II Header (same)	Switch forwards frame via <b>FastEthernet0/1</b> to PC1

The switch operates at Layer 2, so it doesn't modify IP or ICMP headers. It checks the destination MAC, looks it up in its MAC address table, and forwards the frame to the correct port (toward PC1).

### 5. Identify visible layers and headers (Ethernet/MAC, IP, ICMP).

Layer	Protocol	Description
Layer 2 (Ethernet)	Frame header removed	PC1 removes the Ethernet frame
Layer 3 (IP)	IP header removed	PC1 processes the IP packet
Layer 4 (ICMP)	ICMP message processed	PC1 receives and replies to Ping request

## Decapsulation Order:

Ethernet -> IP -> ICMP

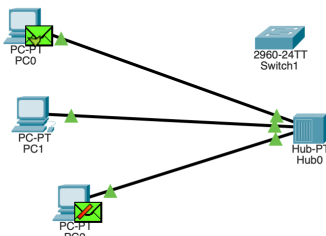
PC1 removes headers layer by layer as data moves up the OSI model, finally reading the ICMP Echo Request and sending an Echo Reply back to PC0.



# Activity 3 — Security Visibility: Hub vs Switch

## Part A. Hub Network

### 1. Build 3 PCs (PC0, PC1, PC2) + 1 Hub.



Simulation Panel

Vis.	Time(sec)	Last Device	At Device	Type
	0.000	--	PC0	ICMP
	0.001	PC0	Hub0	ICMP
	0.002	Hub0	PC1	ICMP
	0.002	Hub0	PC2	ICMP
	0.003	PC1	Hub0	ICMP
Visible	0.004	Hub0	PC0	ICMP
Visible	0.004	Hub0	PC2	ICMP

Reset Simulation

☒ Constant Delay

Captured to: 299.545 s

Play Controls

### 2. Assign IPs: PC0 192.168.2.1, PC1 192.168.2.2, PC2 192.168.2.3.

PC0 and PC1:

PC0

Physical

Config

Desktop

Programming

Attributes

GLOBAL

Settings

Algorithm Settings

INTERFACE

FastEthernet0

Bluetooth

Port Status

Link Speed

Duplex

MAC Address

IP Configuration

IPv4 Address

Subnet Mask

IPv6 Configuration

IPv6 Address

Link Local Address

On

100 Mbps

Full Duplex

0090.2151.E6D5

Static

192.168.1.1

255.255.255.0

Automatic

/

FE80::290:21FF:FE51:E6D5

PC1

Physical

Config

Desktop

Programming

Attributes

GLOBAL

Settings

Algorithm Settings

INTERFACE

FastEthernet0

Bluetooth

Port Status

Link Speed

Duplex

MAC Address

IP Configuration

IPv4 Address

Subnet Mask

IPv6 Configuration

IPv6 Address

Link Local Address

On

100 Mbps

Full Duplex

0050.0F28.956D

Static

192.168.1.2

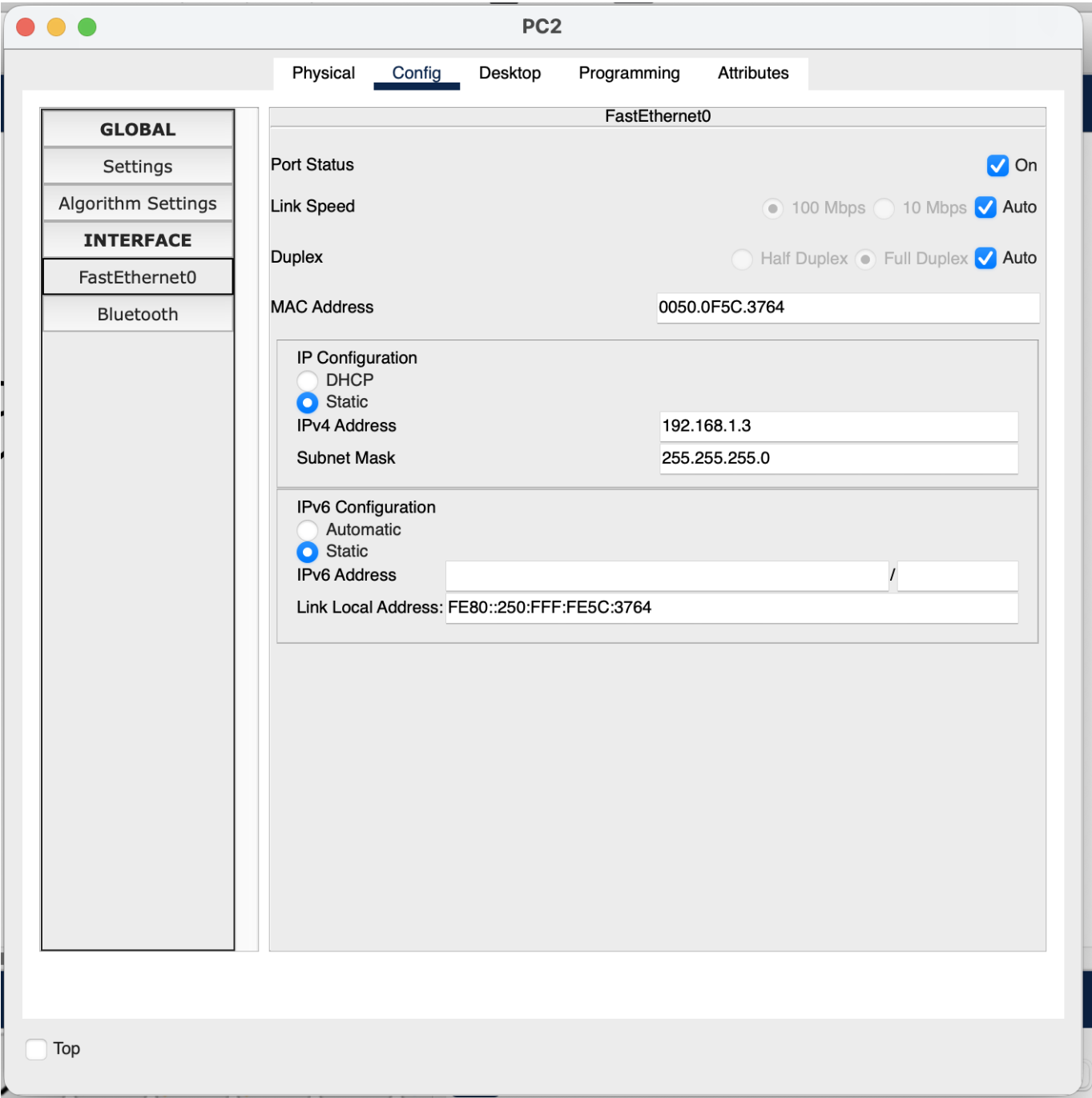
255.255.255.0

Automatic

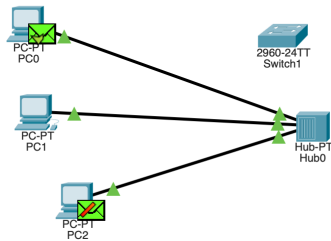
/

FE80::250:FFF:FE28:956D

PC2:



3. Switch to Simulation Mode → send Simple PDU from PC0 → PC1.
4. Observe whether PC2 receives a copy of the packet.



Simulation Panel

Event List				
Vis.	Time(sec)	Last Device	At Device	Type
	0.000	--	PC0	ICMP
	0.001	PC0	Hub0	ICMP
	0.002	Hub0	PC1	ICMP
	0.002	Hub0	PC2	ICMP
	0.003	PC1	Hub0	ICMP
Visible	0.004	Hub0	PC0	ICMP
Visible	0.004	Hub0	PC2	ICMP

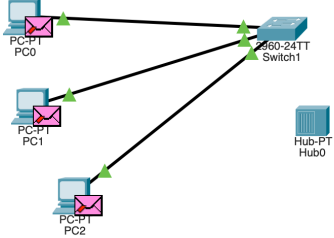
Reset Simulation ☒ Constant Delay

Play Controls

Captured to: 299.545 s

Part B. Switch Network

1. Replace the hub with a Switch and reconnect PCs (same IPs).



PC-PT PC0  
PC-PT PC1  
PC-PT PC2  
260-24TT Switch1  
Hub-PT Hub0

Simulation Panel

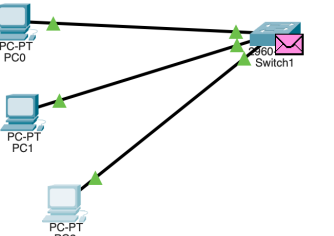
Event List				
Vis.	Time(sec)	Last Device	At Device	Type
	0.000	--	PC0	ICMP
	0.001	PC0	Switch1	ICMP
	0.002	Switch1	PC1	ICMP
	0.003	PC1	Switch1	ICMP
	0.004	Switch1	PC0	ICMP
	1.998	--	Switch1	STP
Visible	1.999	Switch1	PC0	STP
Visible	1.999	Switch1	PC1	STP
Visible	1.999	Switch1	PC2	STP

Reset Simulation ☒ Constant Delay

Captured to 1.999

Play Controls

2. Repeat the Simple PDU test PC0 → PC1 and observe packet visibility at PC2.



PC-PT PC0  
PC-PT PC1  
PC-PT PC2  
260-24TT Switch1  
Hub-PT Hub0

Simulation Panel

Event List				
Vis.	Time(sec)	Last Device	At Device	Type
	0.000	--	PC0	ICMP
	0.001	PC0	Switch1	ICMP
	0.002	Switch1	PC1	ICMP
	0.003	PC1	Switch1	ICMP
	0.004	Switch1	PC0	ICMP
Visible	0.952	--	Switch1	STP

Reset Simulation ☒ Constant Delay

Captured to 0.952

Play Controls

END