

10/11/22

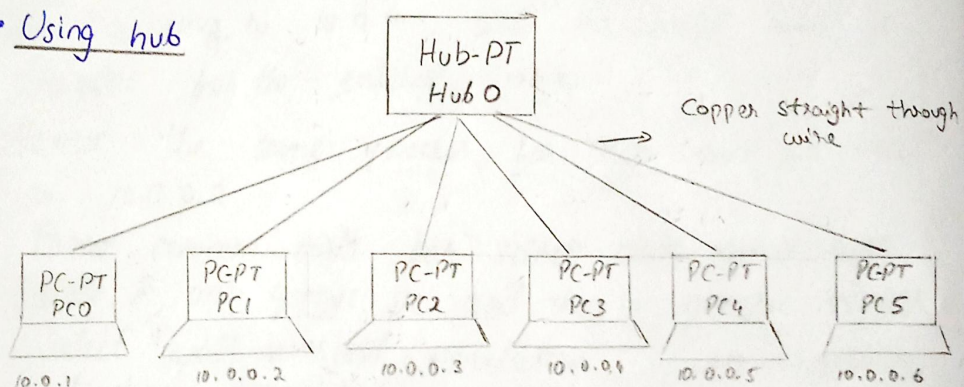
Lab 1

Hubs and Switches

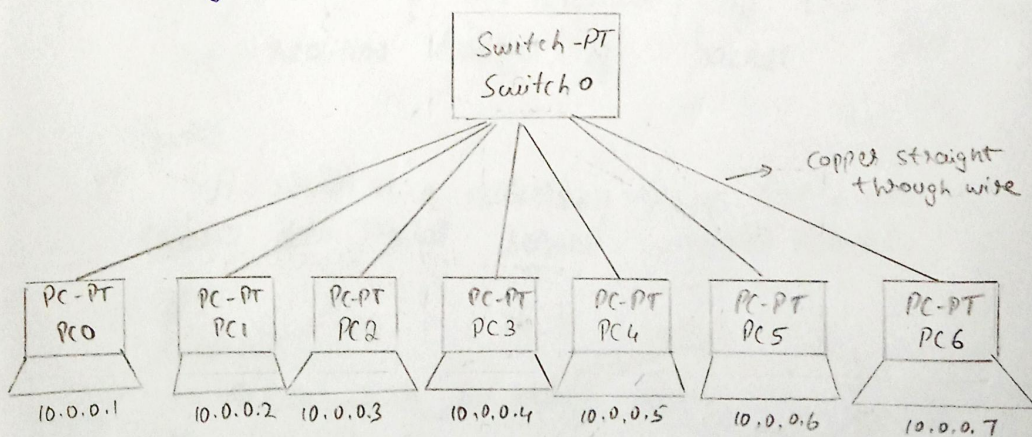
Aim: Creating a topology and simulate sending a simple PDU from source to destination using simple hub and switch as connecting devices.

Topology:

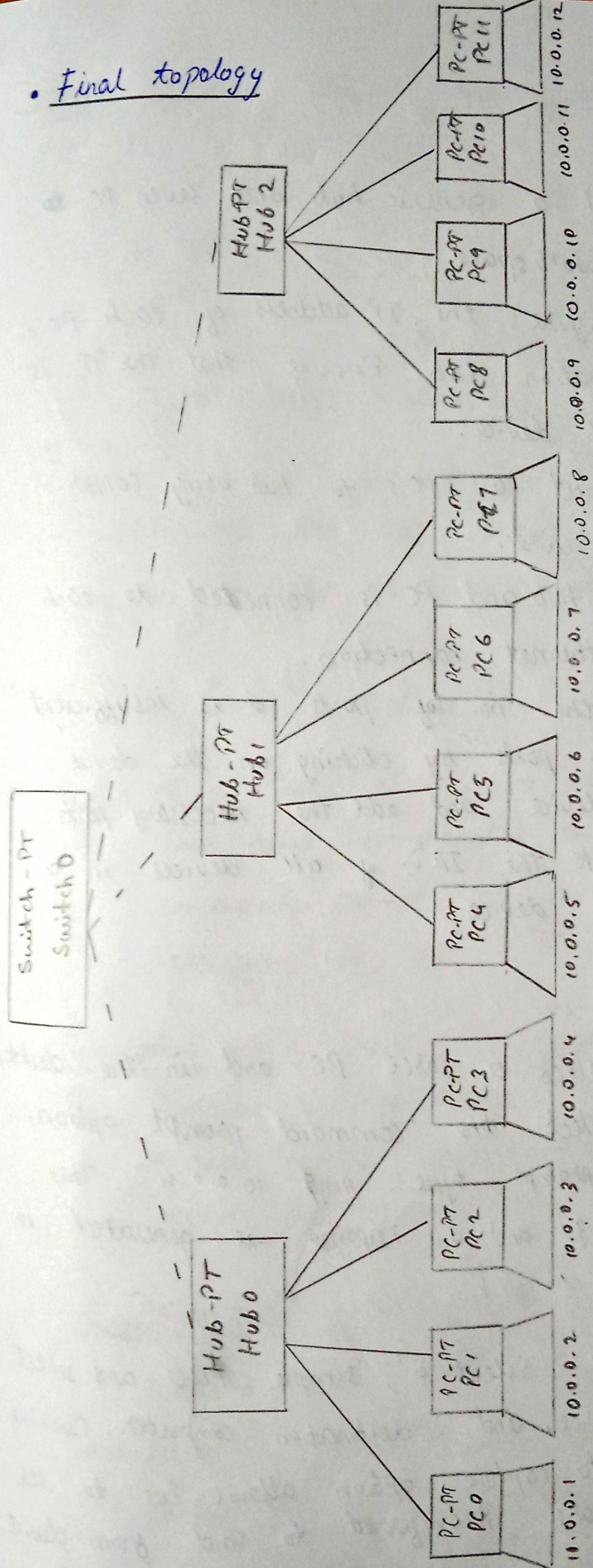
• Using hub



• Using switch



• Final topology



Procedure:

- Using hub:
- Add a generic hub and seven PC to the workspace.
 - Configure the IP address of each PC's in the configuration tab. Ensure that the IP is different for each device.
 - Connect all PC's to hub using copper straight through wire.
 - The hub and PC is connected to each other's fast ethernet connection.
 - If the no. of ports is insufficient then add extra ports by clicking on the device. Turn off the device and add the necessary ports.
 - Write the IP's of all devices in a note below the device

~~Using router~~

Real time: Select a source PC and in the desktop tab select the command prompt option. In command prompt type "ping 10.0.0.4". This pings the PC3 and a response is generated in PC0.

Simulation time: Select a simple PDU and select a source and destination computer. Clicking on auto capture option allows us to see how packets are transferred to and from device.

• Configure the IP of each of the PC in configure and add a note below each PC containing IP add.

Real time mode: Select the PC you want to send the packet from and open its command prompt, specify the destination PC by specifying its IP address. A response is sent by the destination PC to source PC.

Simulation mode: Add a simple PDC by selecting the pair of PC and click on autocapture from right panel.

Observation

HUB:

Learning outcome:

When a source sends a packet in the network, the hub receives the packet and sends broadcast over the network, i.e. it sends data to all the end devices in the network and the node whose ip matches with the specified address accepts the packet and acknowledges it, remaining nodes discards / ignores the message.

The communication between hub and end devices is established through copper straight through wire as they belong to different layers.

Result

PC > ping 10.0.0.3

pinging 10.0.0.3 with 32 bytes of data:

Reply from 10.0.0.3: byte = 32 time = 2ms

Reply from 10.0.0.3: byte = 32 time = 0ms

Reply from 10.0.0.3: byte = 32 time = 0ms

Reply from 10.0.0.3: byte = 32 time = 0ms

Ping statistic for 10.0.0.3

Packets sent = 4, received = 4, loss = 0

Switch

Learning Outcome

When a source device sends a message sends a message to the switch once the connection is established, which takes some time called as learning time, the switch receives the packet. It initially broadcasts the packet to all connected devices to locate the destination. Once the destination is located the message is sent only to that device.

The connection between the switch and end device is established using copper straight through as they belong to different network layer.

Result:

PC > 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

Reply from 10.0.0.2: bytes=32 time=3ms

Reply from 10.0.0.2: bytes=32 time=0ms

Reply from 10.0.0.2: bytes=32 time=0ms

Ping statistics for 10.0.0.2

packets sent=4, Received=4, lost=0 (0% loss)

Final structure (Hybrid model)

Learning outcome:

The switch and hub are connected through copper crossover as they belong to the same network layer. But PC and hubs are connected through copper straight through as they belong to different network layers.

The message from the source PC to destination is sent through the hub which then sends to all its connected PCs and the switch. The switch then sends the message to the respective hub and the hub sends the message to all its connected PCs. The destination PC acknowledges that it has received the message by sending an acknowledgement back to the source PC.

Results :

PC > ~~PC~~ ping 10.0.0.5

pinging 10.0.0.5 with 32 bytes of data

Reply from 10.0.0.5: ~~60~~ bytes = 32 time = 1ms TTL = 128

Reply from 10.0.0.5: bytes = 32 time = 1ms TTL = 128

Reply from 10.0.0.5: bytes = 32 time = 1ms TTL = 128

Reply from 10.0.0.5: bytes = 32 time = 1ms TTL = 128

Ping statistics for 10.0.0.7:

Packet : sent = 4 received = 4, lost = 0 (0% loss)

Minimum = 0ms, Maximum = 1ms, Average = 0ms

HUBS----SIMULATION

Manoj Howale S
1BM20CS082
10 Nov 2022

Simulation Panel

Vis.	Time(sec)	Last Device	At Device	Type	Info
	0.002	Hub0	PC3	ICMP	
	0.002	Hub0	PC8	ICMP	
	0.003	PC3	Hub0	ICMP	
	0.004	Hub0	PC0	ICMP	
	0.004	Hub0	PC1	ICMP	
	0.004	Hub0	PC2	ICMP	
	0.004	Hub0	PC3	ICMP	
	0.004	Hub0	PC4	ICMP	
	0.004	Hub0	PC6	ICMP	

Reset Simulation ☒ Constant Delay Captured to: 438.582 s

Play Controls: Back Auto Capture / Play Capture / Forward

Event List Filters - Visible Events
ACL Filter, ARP, BGP, CDP, DHCP, DHCPv6, DNS, DTP, EIGRP, EIGRPv6, FTP, H.323, HSRP, HSRPv6, HTTPS, HTTP, ICMP, ICMPv6, IPsec, ISAKMP, LACP, NDI, NETFLOW, NTP, OSPF, OSPFv6, PAPI, POP3, RADIUS, RDP, RDPv6, RTR, SCOP, SMTP, SNMP, SSH, STP, SYSLOG, TACACS, TFTP, Telnet, UDP, VTP

Edit Filters Show All/None

Time: 00:15:34.125 Power Cycle Devices PLAY CONTROLS: Back Auto Capture / Play Capture / Forward

Connections: Automatically Choose Connection Type

Scenario 0
New Delete
Toggle PDU List Window

Fire	Last Status	Source	Destination	Type	Color	Time(sec)	Periodic	Num	Edit	Delete
	Successful	PC1	PC5	ICMP		0.000	N	0	(edit)	(delete)

Simulation

HUBS----REAL TIME

Manoj Howale S
1BM20CS082
10 Nov 22

Command Prompt

```

Packet Tracer PC Command Line 1.0
C:\>ping 10.0.0.5

Pinging 10.0.0.5 with 32 bytes of data:

Reply from 10.0.0.5: bytes=32 time=1ms TTL=128
Reply from 10.0.0.5: bytes=32 time=1ms TTL=128
Reply from 10.0.0.5: bytes=32 time=1ms TTL=128
Reply from 10.0.0.5: bytes=32 time=1ms TTL=128

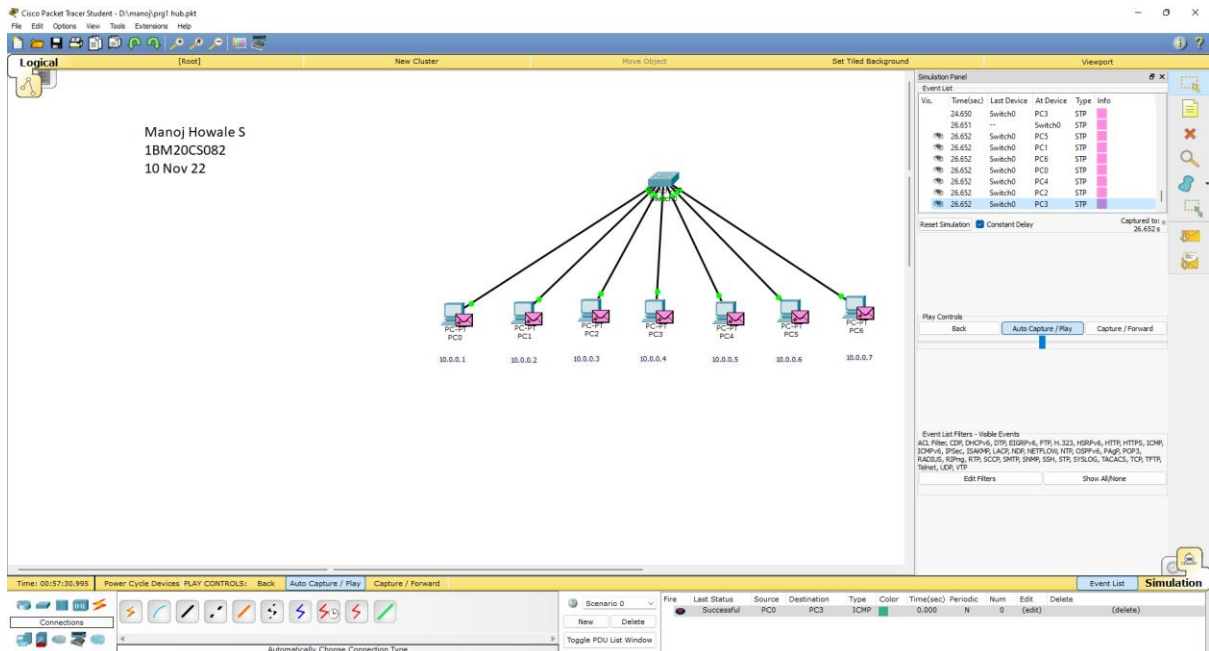
Ping statistics for 10.0.0.5:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milliseconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms
C:\>
    
```

Simulation Panel

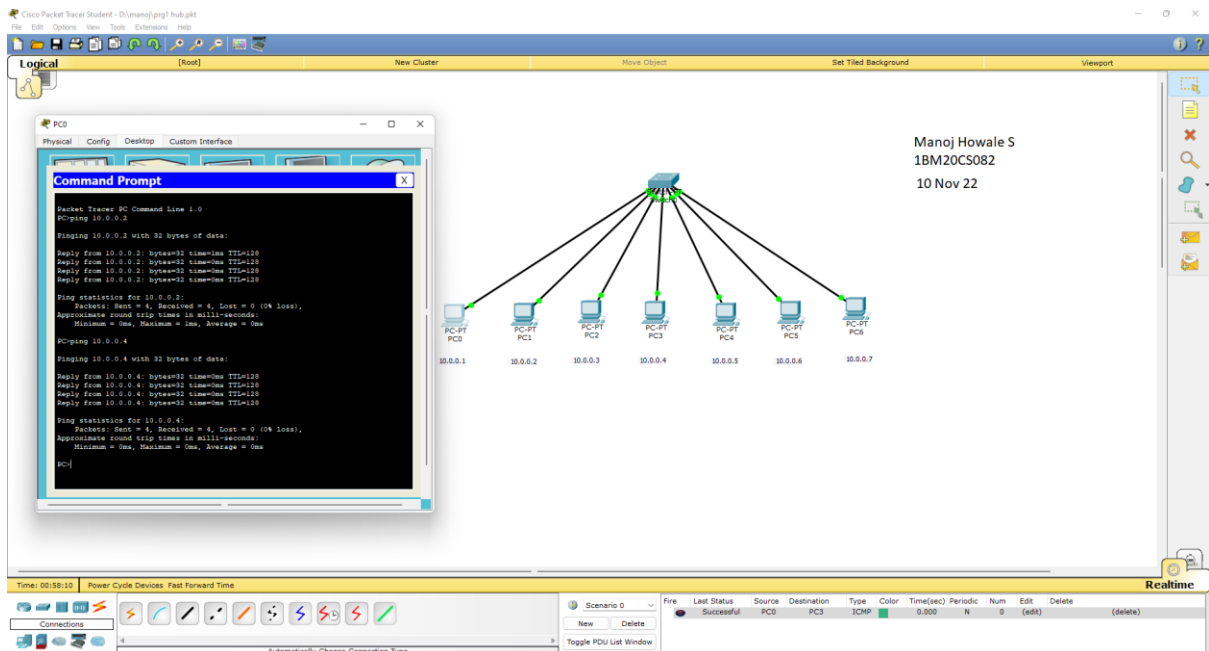
Fire	Last Status	Source	Destination	Type	Color	Time(sec)	Periodic	Num	Edit	Delete
	Successful	PC1	PC5	ICMP		0.000	N	0	(edit)	(delete)

Realtime

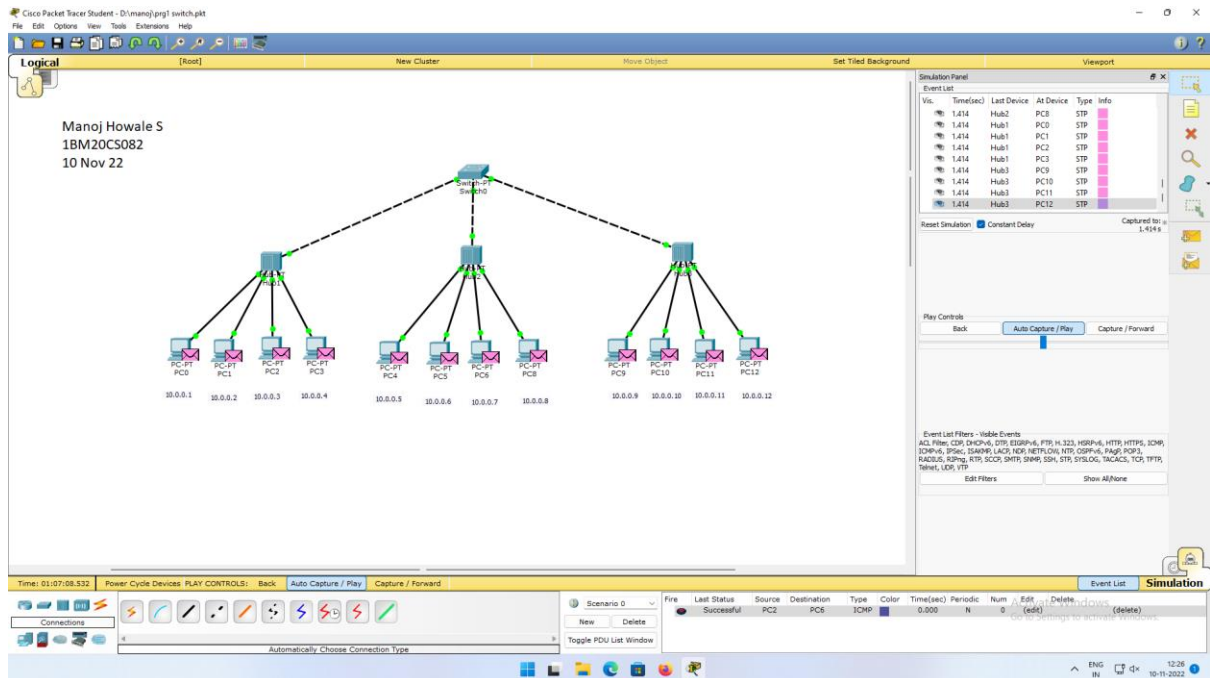
SWITCHES-----SIMULATION



SWITCHES-----REAL TIME



FINAL NETWORK-----SIMULATION



FINAL NETWORK-----REAL TIME

