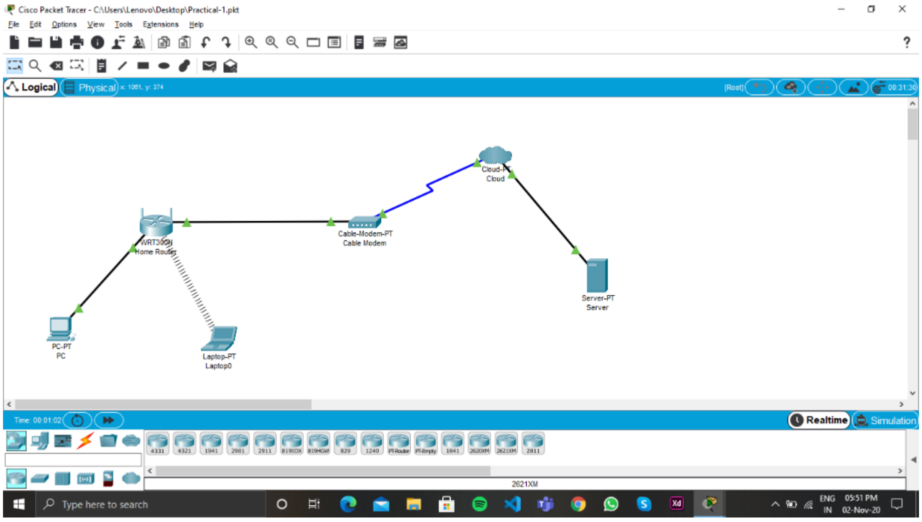
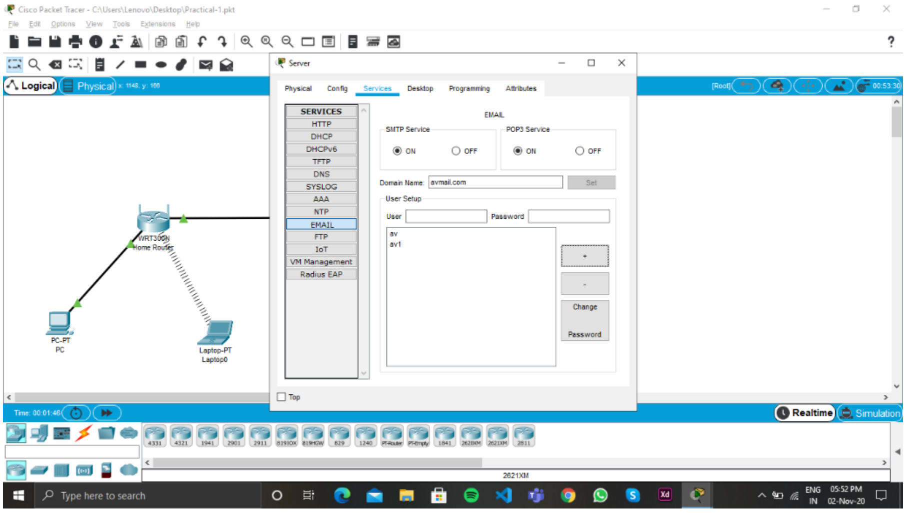
Computer Network Lab session 14

Q-1) Use the network topology used in DNS configuration (add one more end device) and perform SMTP configuration on the server. After configuration, both the device should be able to send and receive email to each other.

- **Topology**



- **Configure SMTP service on server.**



EMAIL

SMTP Service: ● ON ○ OFF

POP3 Service: ● ON ○ OFF

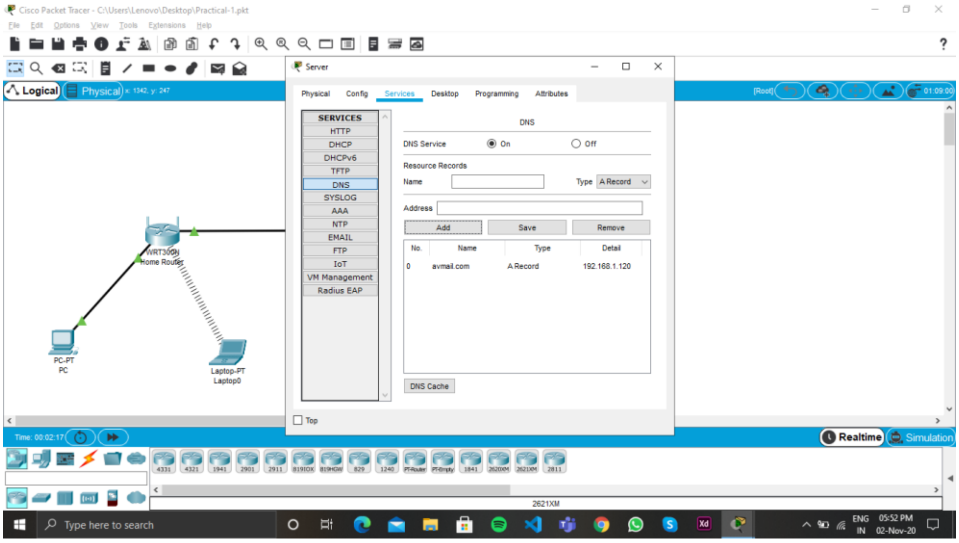
Domain Name: avmail.com

User Setup

User Password

av
av1

➢ Add “avmail.com” in DNS record.



DNS

DNS Service ● On ○ Off

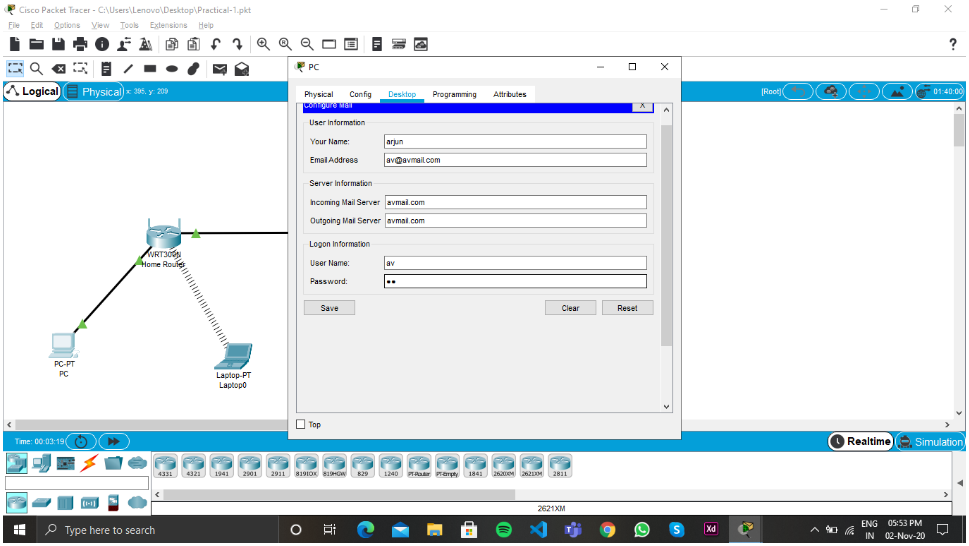
Resource Records

Name Type A Record

Address

No.	Name	Type	Detail
0	avmail.com	A Record	192.168.1.120

➢ Create email and login



User Information

Your Name: arjun

Email Address: av@avmail.com

Server Information

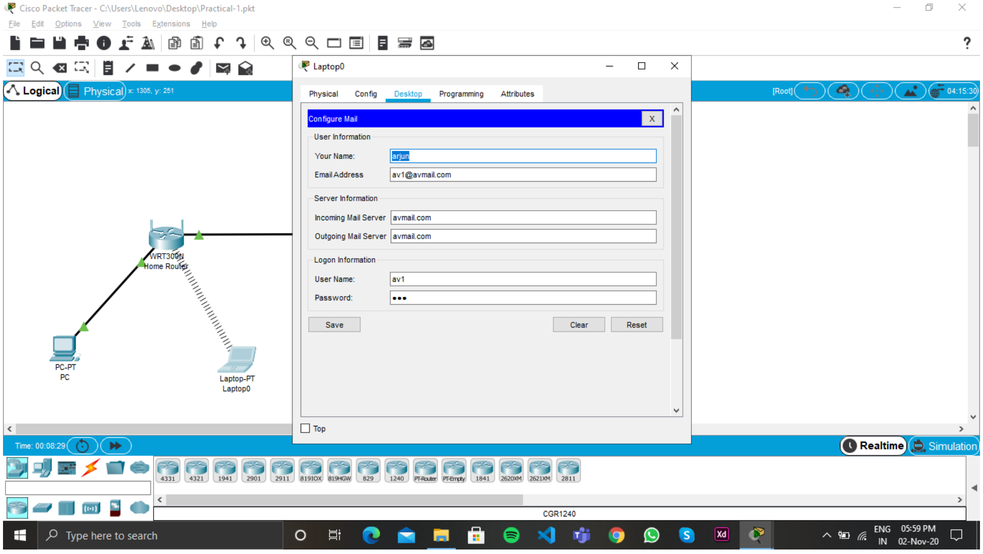
Incoming Mail Server: avmail.com

Outgoing Mail Server: avmail.com

Logon Information

User Name: av

Password: ••



User Information

Your Name: arjun

Email Address: av1@avmail.com

Server Information

Incoming Mail Server: avmail.com

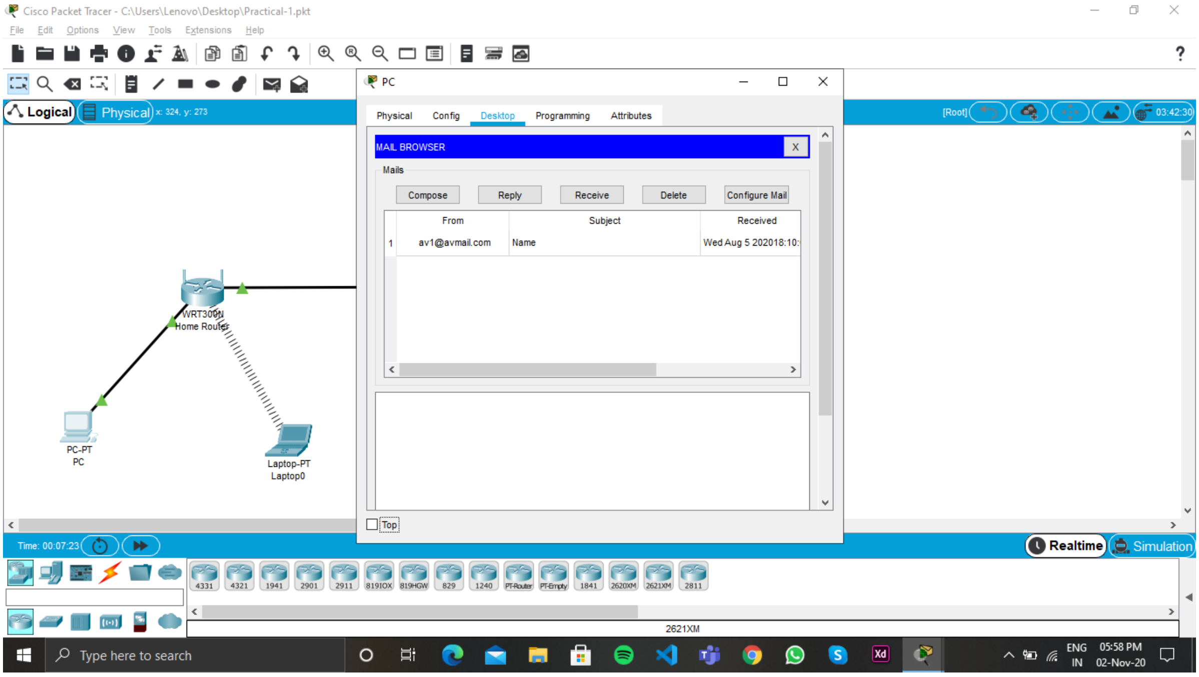
Outgoing Mail Server: avmail.com

Logon Information

User Name: av1

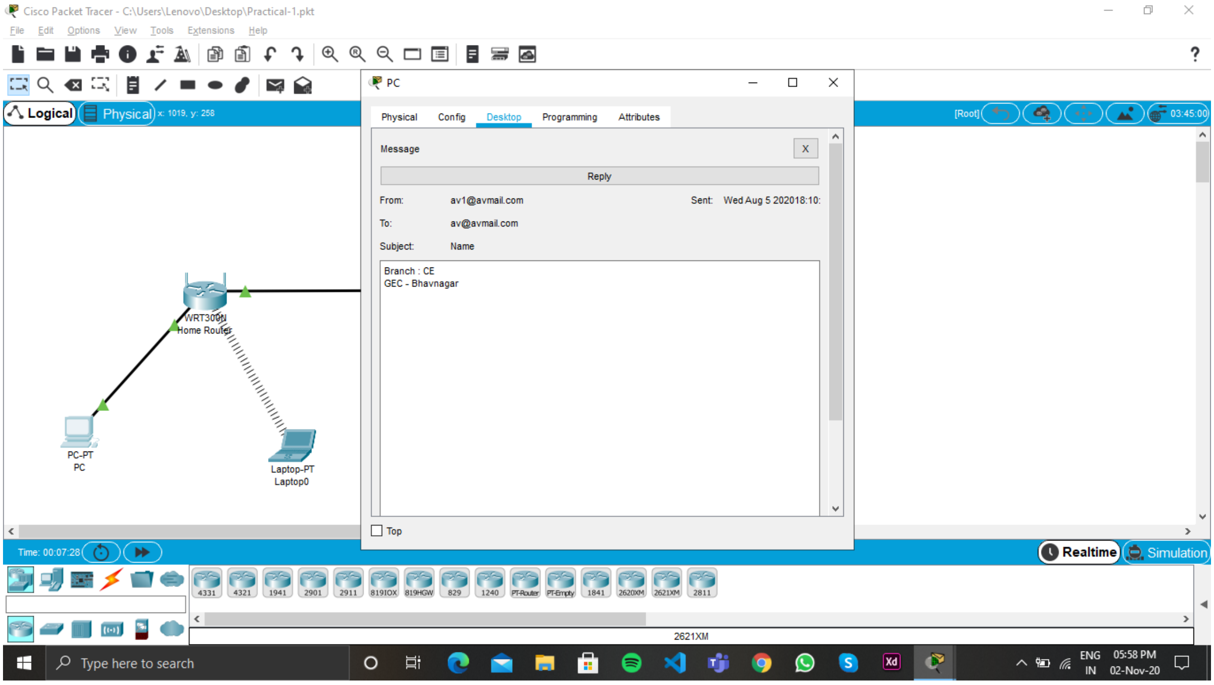
Password: •••

➢ Send emails:



Mails

	From	Subject	Received
1	av1@avmail.com	Name	Wed Aug 5 202018:10:



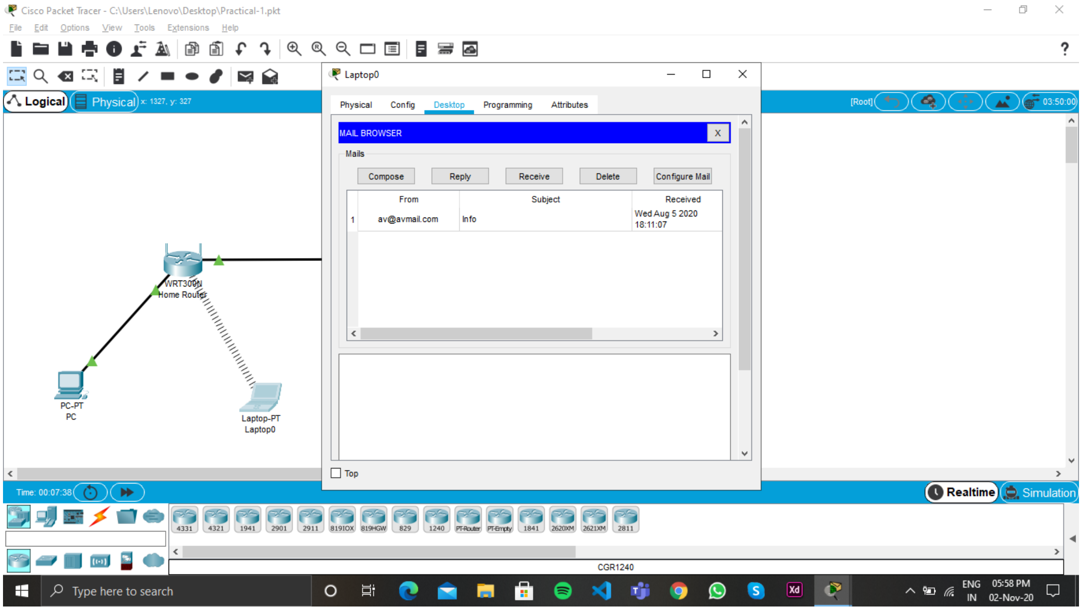
From: av1@avmail.com Sent: Wed Aug 5 202018:10:

To: av@avmail.com

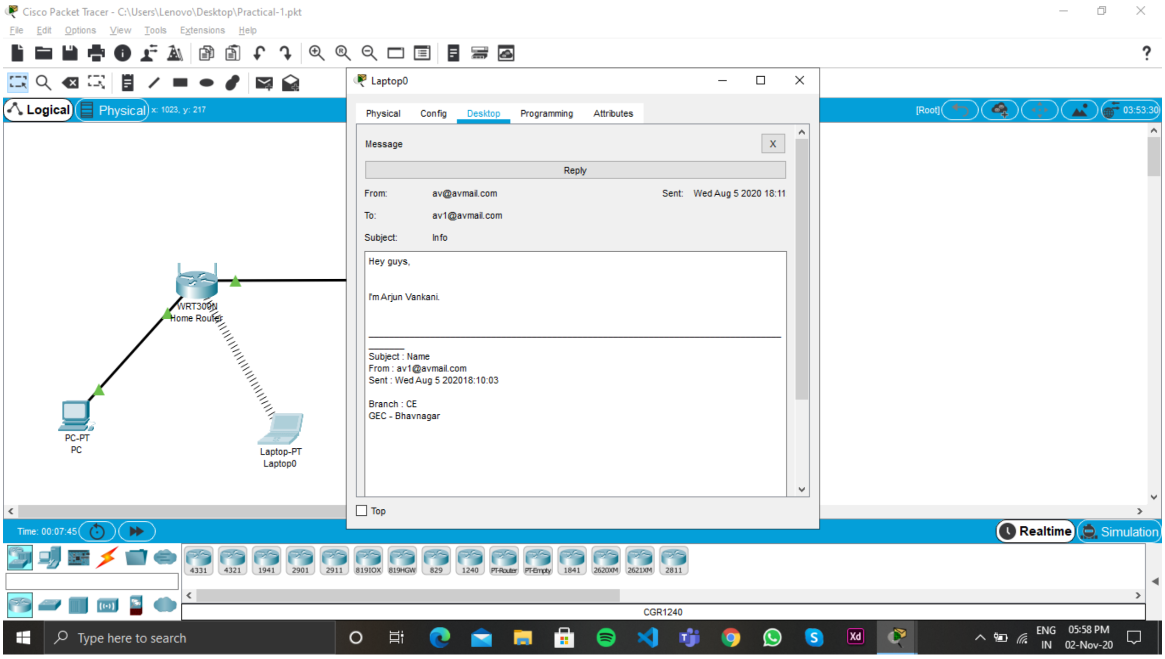
Subject: Name

Branch : CE
GEC - Bhavnagar

➢ Receiving mail



	From	Subject	Received
1	av@avmail.com	Info	Wed Aug 5 2020 18:11:07



From: av@avmail.com
Sent: Wed Aug 5 2020 18:11
To: av1@avmail.com
Subject: Info

Hey guys,

I'm Arjun Vankani.

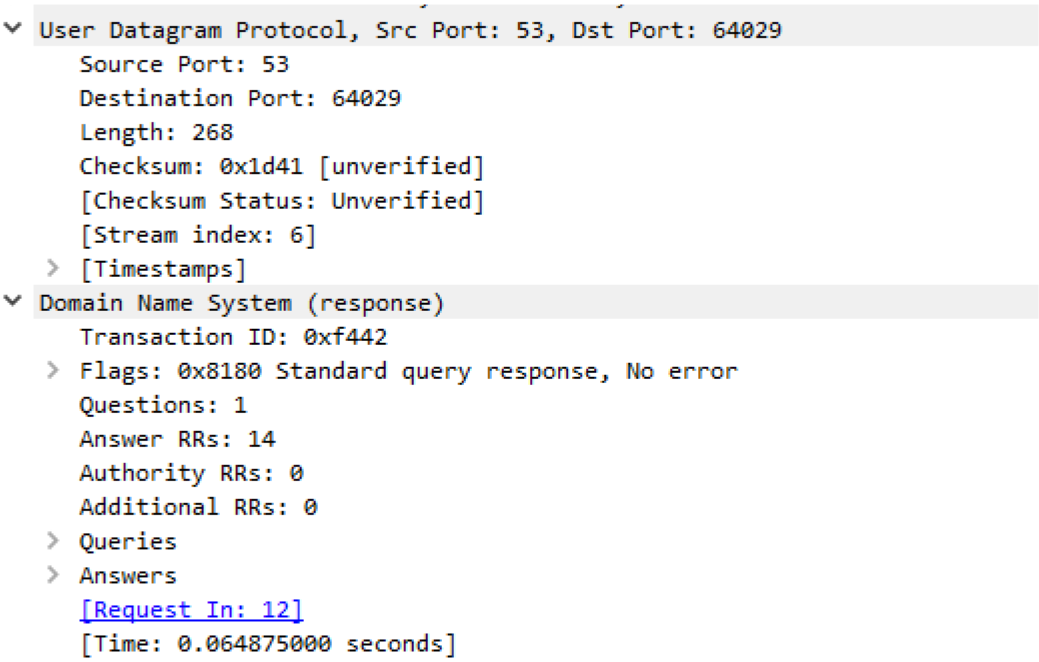
Subject : Name
From : av1@avmail.com
Sent : Wed Aug 5 202018:10:03

Branch : CE
GEC - Bhavnagar

Q-2) **Use Wireshark to analyze the network traffic of TCP and UDP Protocol.**

❖ UDP:

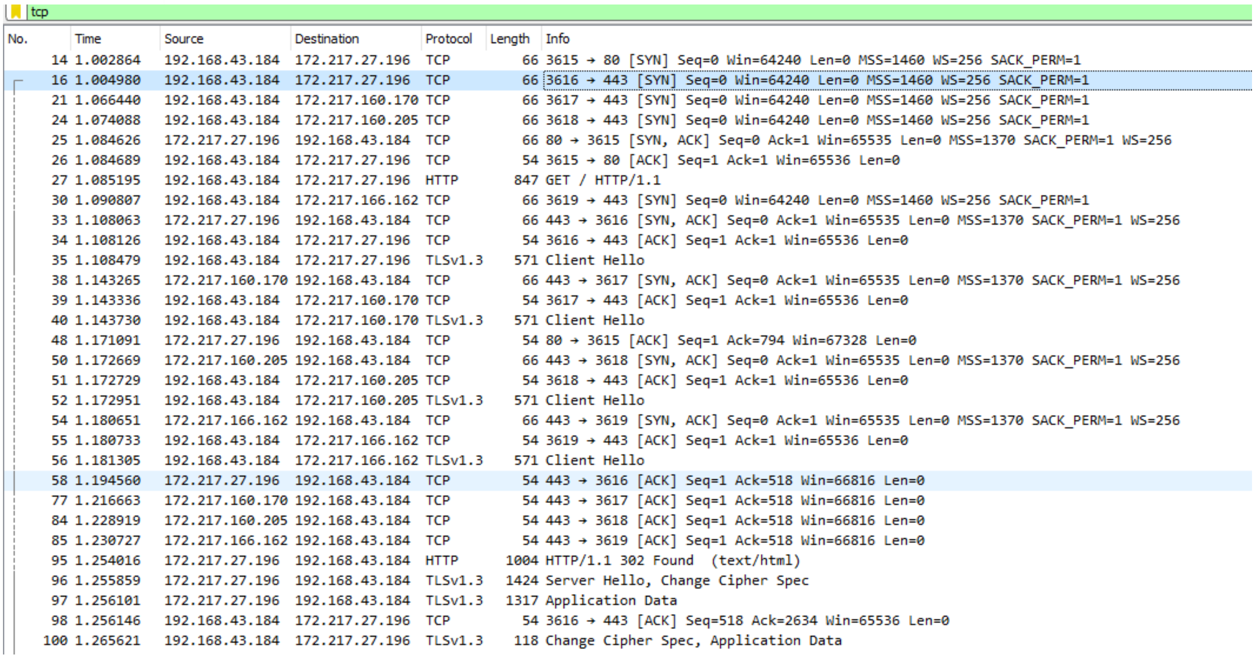
- We first analyze on UDP protocol in Transport layer, we know that the DNS uses the UDP protocol then we capture UDP from a DNS request of www.google.com.
- Here this DNS request is sent from google server port 53 to our system port 64029, that specified in UDP segment.



```
˅ User Datagram Protocol, Src Port: 53, Dst Port: 64029
     Source Port: 53
     Destination Port: 64029
     Length: 268
     Checksum: 0x1d41 [unverified]
     [Checksum Status: Unverified]
     [Stream index: 6]
   > [Timestamps]
˅ Domain Name System (response)
     Transaction ID: 0xf442
   > Flags: 0x8180 Standard query response, No error
     Questions: 1
     Answer RRs: 14
     Authority RRs: 0
     Additional RRs: 0
   > Queries
   > Answers
     [Request In: 12]
     [Time: 0.064875000 seconds]
```

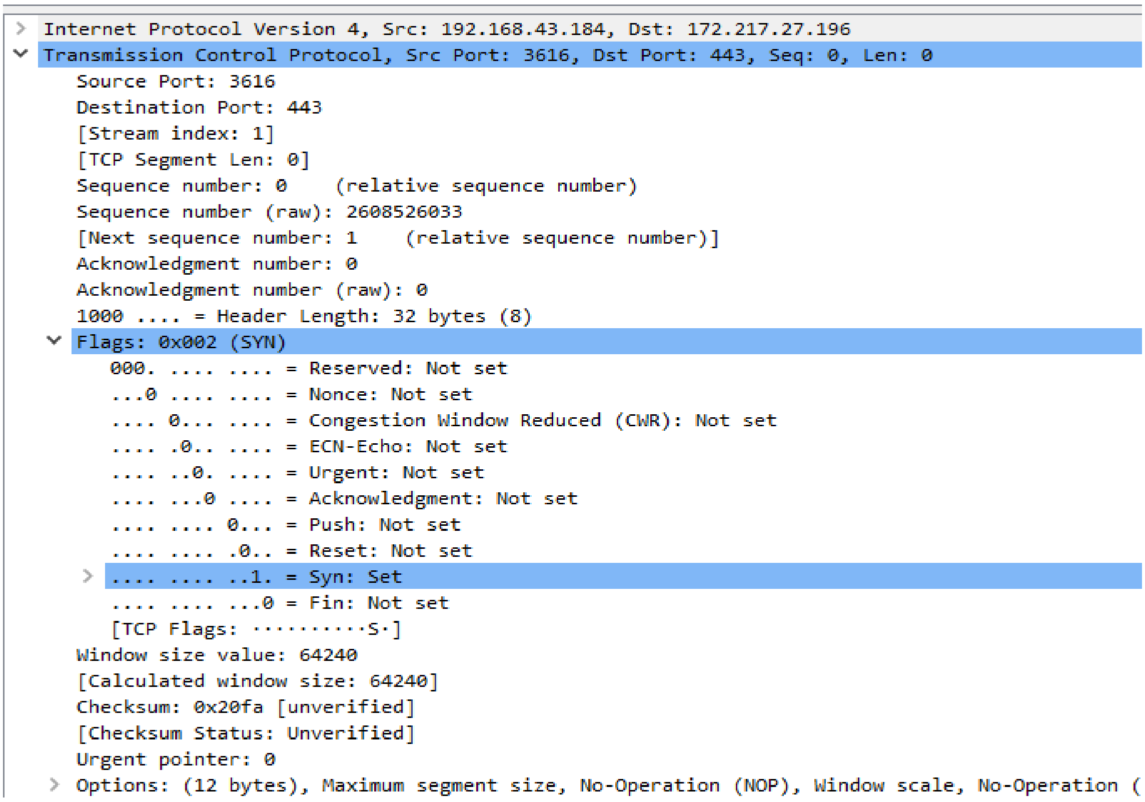
❖ TCP:

Now on going to analyse TCP protocol where we detect synchronize flag



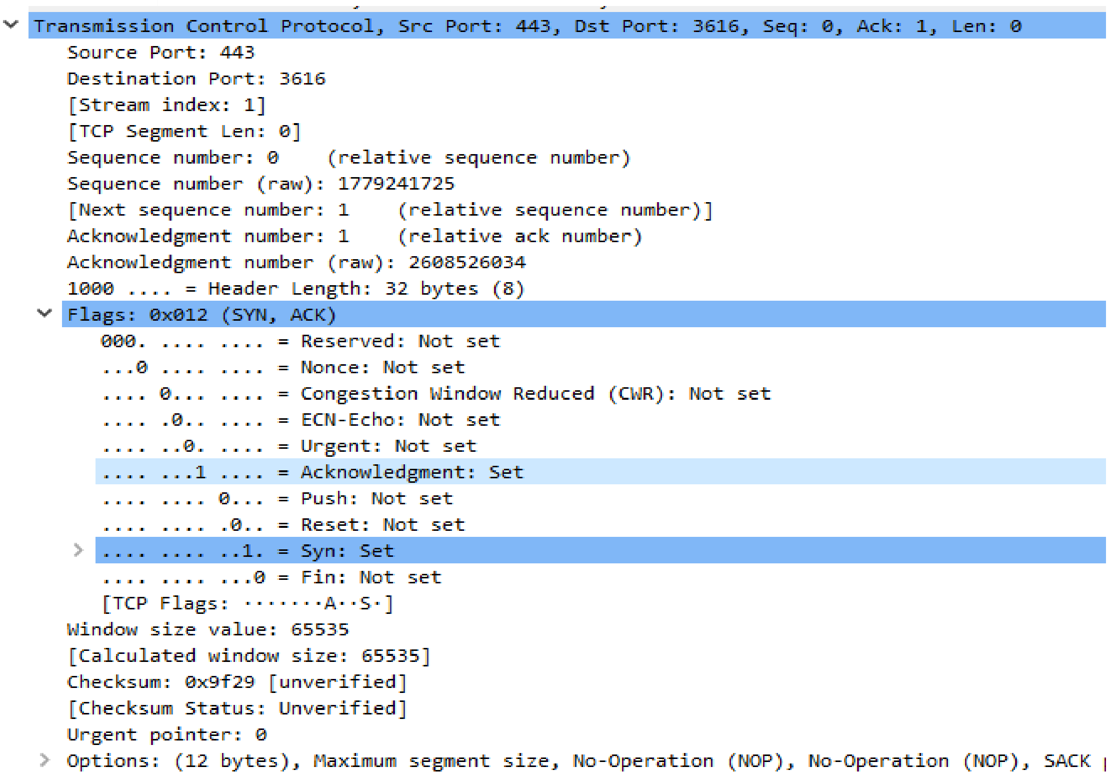
```
tcp
No.     Time       Source          Destination     Protocol Length Info
     14 1.002864   192.168.43.184  172.217.27.196  TCP        66 3615 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
     16 1.004980   192.168.43.184  172.217.27.196  TCP        66 3616 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
     21 1.066440   192.168.43.184  172.217.160.170 TCP        66 3617 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
     24 1.074088   192.168.43.184  172.217.160.205 TCP        66 3618 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
     25 1.084626   172.217.27.196  192.168.43.184  TCP        66 80 → 3615 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1370 SACK_PERM=1 WS=256
     26 1.084689   192.168.43.184  172.217.27.196  TCP        54 3615 → 80 [ACK] Seq=1 Ack=1 Win=65536 Len=0
     27 1.085195   192.168.43.184  172.217.27.196  HTTP      847 GET / HTTP/1.1
     30 1.090807   192.168.43.184  172.217.166.162 TCP        66 3619 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
     33 1.108063   172.217.27.196  192.168.43.184  TCP        66 443 → 3616 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1370 SACK_PERM=1 WS=256
     34 1.108126   192.168.43.184  172.217.27.196  TCP        54 3616 → 443 [ACK] Seq=1 Ack=1 Win=65536 Len=0
     35 1.108479   192.168.43.184  172.217.27.196  TLSv1.3   571 Client Hello
     38 1.143265   172.217.160.170 192.168.43.184  TCP        66 443 → 3617 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1370 SACK_PERM=1 WS=256
     39 1.143336   192.168.43.184  172.217.160.170 TCP        54 3617 → 443 [ACK] Seq=1 Ack=1 Win=65536 Len=0
     40 1.143730   192.168.43.184  172.217.160.170 TLSv1.3   571 Client Hello
     48 1.171091   172.217.27.196  192.168.43.184  TCP        54 80 → 3615 [ACK] Seq=1 Ack=794 Win=67328 Len=0
     50 1.172669   172.217.160.205 192.168.43.184  TCP        66 443 → 3618 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1370 SACK_PERM=1 WS=256
     51 1.172729   192.168.43.184  172.217.160.205 TCP        54 3618 → 443 [ACK] Seq=1 Ack=1 Win=65536 Len=0
     52 1.172951   192.168.43.184  172.217.160.205 TLSv1.3   571 Client Hello
     54 1.180651   172.217.166.162 192.168.43.184  TCP        66 443 → 3619 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1370 SACK_PERM=1 WS=256
     55 1.180733   192.168.43.184  172.217.166.162 TCP        54 3619 → 443 [ACK] Seq=1 Ack=1 Win=65536 Len=0
     56 1.181305   192.168.43.184  172.217.166.162 TLSv1.3   571 Client Hello
     58 1.194560   172.217.27.196  192.168.43.184  TCP        54 443 → 3616 [ACK] Seq=1 Ack=518 Win=66816 Len=0
     77 1.216663   172.217.160.170 192.168.43.184  TCP        54 443 → 3617 [ACK] Seq=1 Ack=518 Win=66816 Len=0
     84 1.228919   172.217.160.205 192.168.43.184  TCP        54 443 → 3618 [ACK] Seq=1 Ack=518 Win=66816 Len=0
     85 1.230727   172.217.166.162 192.168.43.184  TCP        54 443 → 3619 [ACK] Seq=1 Ack=518 Win=66816 Len=0
     95 1.254016   172.217.27.196  192.168.43.184  HTTP     1004 HTTP/1.1 302 Found  (text/html)
     96 1.255859   172.217.27.196  192.168.43.184  TLSv1.3  1424 Server Hello, Change Cipher Spec
     97 1.256101   172.217.27.196  192.168.43.184  TLSv1.3  1317 Application Data
     98 1.256146   192.168.43.184  172.217.27.196  TCP        54 3616 → 443 [ACK] Seq=518 Ack=2634 Win=65536 Len=0
    100 1.265621   192.168.43.184  172.217.27.196  TLSv1.3   118 Change Cipher Spec, Application Data
```

A) Frist sent a segment with SYN to google.com that is located in 16th line in figure.



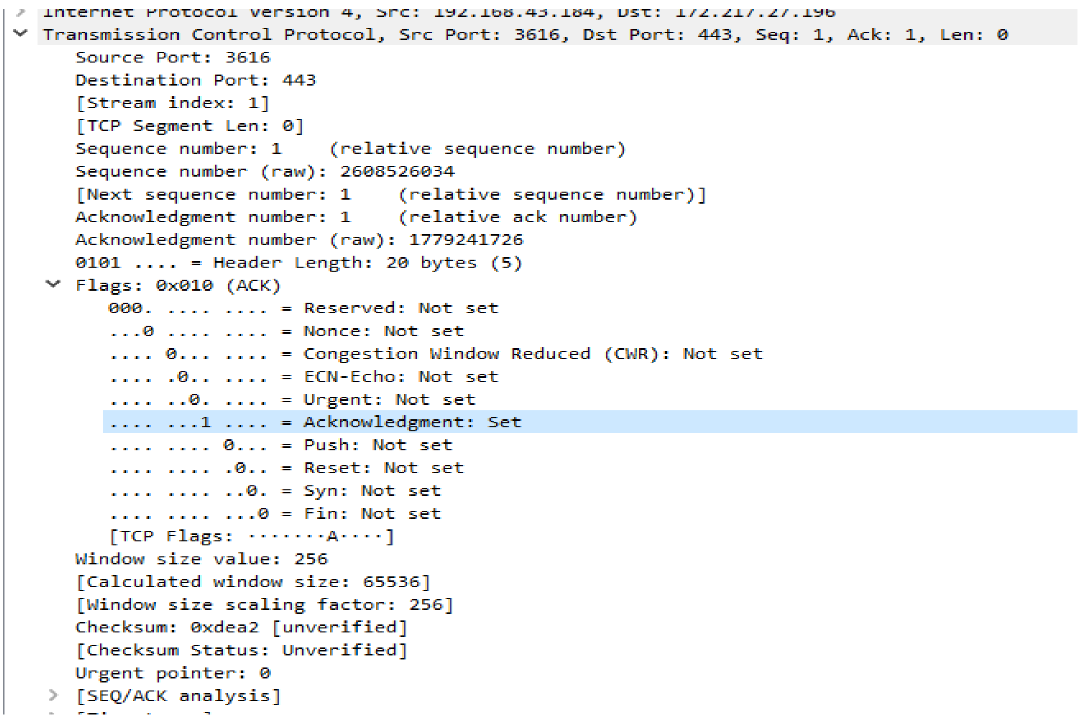
```
> Internet Protocol Version 4, Src: 192.168.43.184, Dst: 172.217.27.196
v Transmission Control Protocol, Src Port: 3616, Dst Port: 443, Seq: 0, Len: 0
    Source Port: 3616
    Destination Port: 443
    [Stream index: 1]
    [TCP Segment Len: 0]
    Sequence number: 0    (relative sequence number)
    Sequence number (raw): 2608526033
    [Next sequence number: 1    (relative sequence number)]
    Acknowledgment number: 0
    Acknowledgment number (raw): 0
    1000 .... = Header Length: 32 bytes (8)
  v Flags: 0x002 (SYN)
      000. .... .... = Reserved: Not set
      ...0 .... .... = Nonce: Not set
      .... 0... .... = Congestion Window Reduced (CWR): Not set
      .... .0.. .... = ECN-Echo: Not set
      .... ..0. .... = Urgent: Not set
      .... ...0 .... = Acknowledgment: Not set
      .... .... 0... = Push: Not set
      .... .... .0.. = Reset: Not set
    > .... .... ..1. = Syn: Set
      .... .... ...0 = Fin: Not set
      [TCP Flags: ··········S·]
    Window size value: 64240
    [Calculated window size: 64240]
    Checksum: 0x20fa [unverified]
    [Checksum Status: Unverified]
    Urgent pointer: 0
  > Options: (12 bytes), Maximum segment size, No-Operation (NOP), Window scale, No-Operation (
```

B) 2nd server response with SYN and ACK that located on 33rd line in figure.



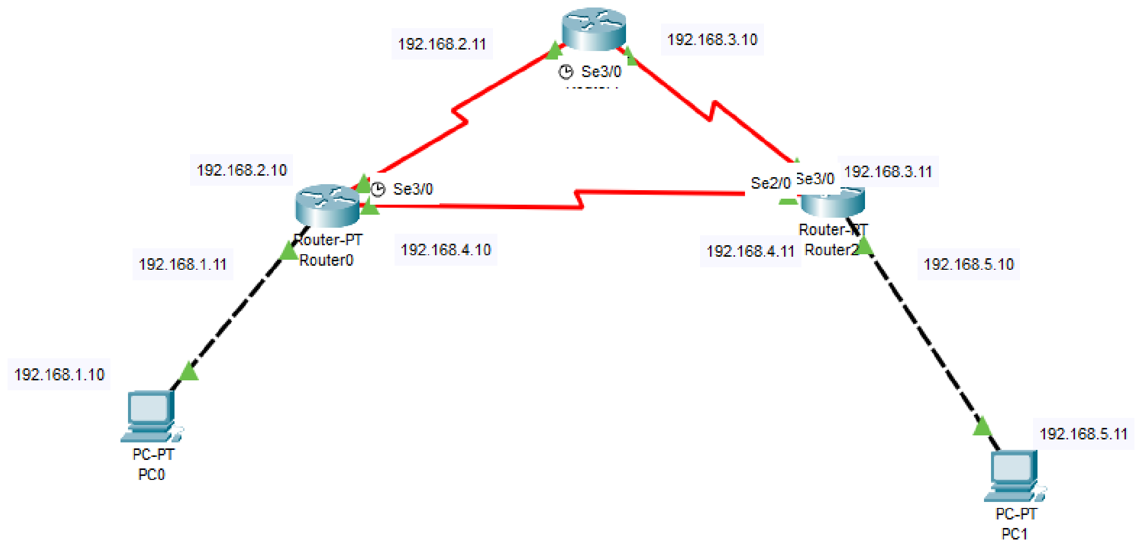
```
Transmission Control Protocol, Src Port: 443, Dst Port: 3616, Seq: 0, Ack: 1, Len: 0
    Source Port: 443
    Destination Port: 3616
    [Stream index: 1]
    [TCP Segment Len: 0]
    Sequence number: 0    (relative sequence number)
    Sequence number (raw): 1779241725
    [Next sequence number: 1    (relative sequence number)]
    Acknowledgment number: 1    (relative ack number)
    Acknowledgment number (raw): 2608526034
    1000 .... = Header Length: 32 bytes (8)
    Flags: 0x012 (SYN, ACK)
        000. .... .... = Reserved: Not set
        ...0 .... .... = Nonce: Not set
        .... 0... .... = Congestion Window Reduced (CWR): Not set
        .... .0.. .... = ECN-Echo: Not set
        .... ..0. .... = Urgent: Not set
        .... ...1 .... = Acknowledgment: Set
        .... .... 0... = Push: Not set
        .... .... .0.. = Reset: Not set
        .... .... ..1. = Syn: Set
        .... .... ...0 = Fin: Not set
        [TCP Flags: ·······A··S·]
    Window size value: 65535
    [Calculated window size: 65535]
    Checksum: 0x9f29 [unverified]
    [Checksum Status: Unverified]
    Urgent pointer: 0
    Options: (12 bytes), Maximum segment size, No-Operation (NOP), No-Operation (NOP), SACK
```

C) 3rd last ACK can send by user that located on 58th line in figure.



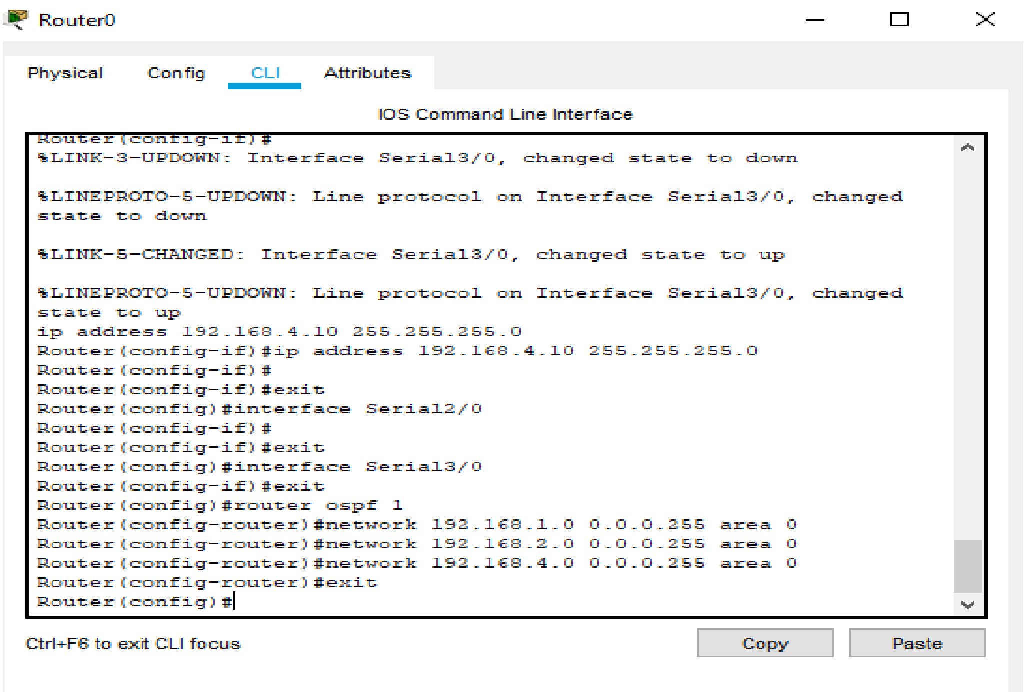
```
Internet Protocol Version 4, Src: 192.168.43.184, Dst: 172.217.27.196
Transmission Control Protocol, Src Port: 3616, Dst Port: 443, Seq: 1, Ack: 1, Len: 0
    Source Port: 3616
    Destination Port: 443
    [Stream index: 1]
    [TCP Segment Len: 0]
    Sequence number: 1    (relative sequence number)
    Sequence number (raw): 2608526034
    [Next sequence number: 1    (relative sequence number)]
    Acknowledgment number: 1    (relative ack number)
    Acknowledgment number (raw): 1779241726
    0101 .... = Header Length: 20 bytes (5)
    Flags: 0x010 (ACK)
        000. .... .... = Reserved: Not set
        ...0 .... .... = Nonce: Not set
        .... 0... .... = Congestion Window Reduced (CWR): Not set
        .... .0.. .... = ECN-Echo: Not set
        .... ..0. .... = Urgent: Not set
        .... ...1 .... = Acknowledgment: Set
        .... .... 0... = Push: Not set
        .... .... .0.. = Reset: Not set
        .... .... ..0. = Syn: Not set
        .... .... ...0 = Fin: Not set
        [TCP Flags: ·······A····]
    Window size value: 256
    [Calculated window size: 65536]
    [Window size scaling factor: 256]
    Checksum: 0xdea2 [unverified]
    [Checksum Status: Unverified]
    Urgent pointer: 0
    [SEQ/ACK analysis]
```

Q-3) Create a simple network topology and configure OSPF routing for data communication. Record necessary steps and screenshots.



OSPF as Open shortest path first

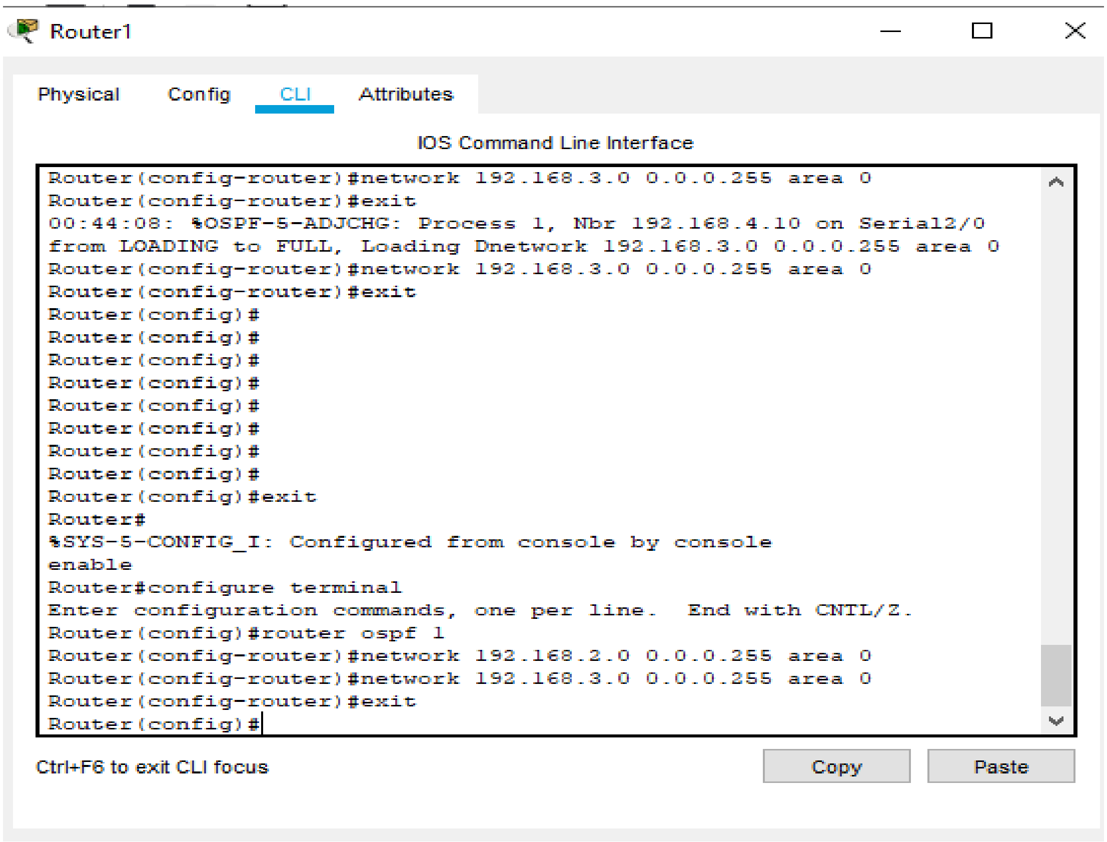
- Frist, configure all routers and pc with IP address and after setup whole network we can configure for OSPF routing on both router.
- OSPF configuration on Router 0 by using below commands.



```
Router(config-if)#
%LINK-3-UPDOWN: Interface Serial3/0, changed state to down

%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial3/0, changed
state to down

%LINK-5-CHANGED: Interface Serial3/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial3/0, changed
state to up
ip address 192.168.4.10 255.255.255.0
Router(config-if)#ip address 192.168.4.10 255.255.255.0
Router(config-if)#
Router(config-if)#exit
Router(config)#interface Serial2/0
Router(config-if)#
Router(config-if)#exit
Router(config)#interface Serial3/0
Router(config-if)#exit
Router(config)#router ospf 1
Router(config-router)#network 192.168.1.0 0.0.0.255 area 0
Router(config-router)#network 192.168.2.0 0.0.0.255 area 0
Router(config-router)#network 192.168.4.0 0.0.0.255 area 0
Router(config-router)#exit
Router(config)#
```

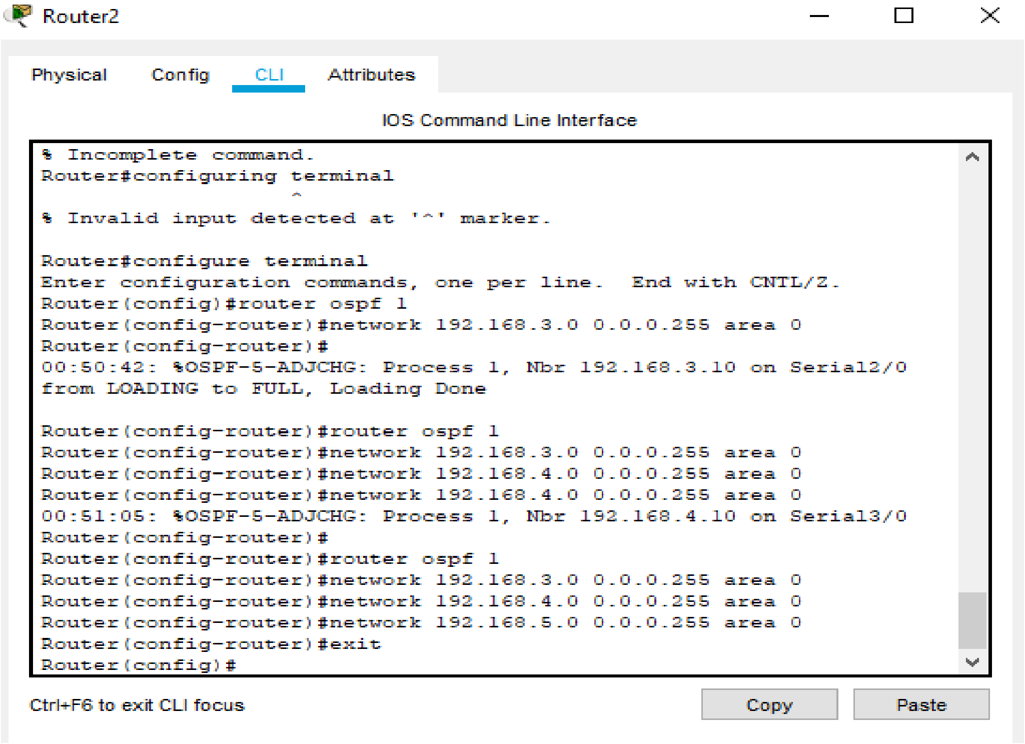
- Here, Router 0 can connect with network 192.168.1.0, 192.168.2.0 and 192.168.4.0. So, we can define that in OSPF configuration.

- OSPF configuration for Router 1 by using below commands.



```
Router(config-router)#network 192.168.3.0 0.0.0.255 area 0
Router(config-router)#exit
00:44:08: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.4.10 on Serial2/0
from LOADING to FULL, Loading Dnetwork 192.168.3.0 0.0.0.255 area 0
Router(config-router)#network 192.168.3.0 0.0.0.255 area 0
Router(config-router)#exit
Router(config)#
Router(config)#
Router(config)#
Router(config)#
Router(config)#
Router(config)#
Router(config)#
Router(config)#
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
enable
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#router ospf 1
Router(config-router)#network 192.168.2.0 0.0.0.255 area 0
Router(config-router)#network 192.168.3.0 0.0.0.255 area 0
Router(config-router)#exit
Router(config)#
```

- Here, Router 1 can connect with network 192.168.2.0 and 192.168.3.0. So, we can define that in OSPF configuration.

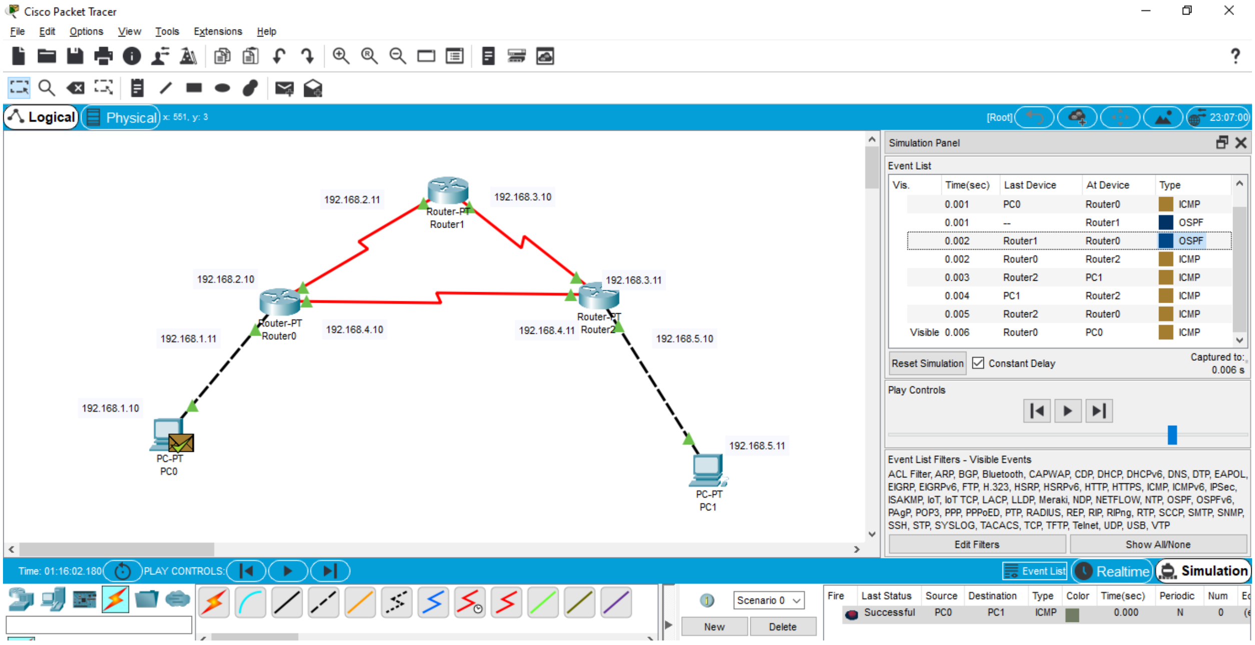


```
% Incomplete command.
Router#configuring terminal
                 ^
% Invalid input detected at '^' marker.

Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#router ospf 1
Router(config-router)#network 192.168.3.0 0.0.0.255 area 0
Router(config-router)#
00:50:42: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.3.10 on Serial2/0
from LOADING to FULL, Loading Done

Router(config-router)#router ospf 1
Router(config-router)#network 192.168.3.0 0.0.0.255 area 0
Router(config-router)#network 192.168.4.0 0.0.0.255 area 0
Router(config-router)#network 192.168.4.0 0.0.0.255 area 0
00:51:05: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.4.10 on Serial3/0
Router(config-router)#
Router(config-router)#router ospf 1
Router(config-router)#network 192.168.3.0 0.0.0.255 area 0
Router(config-router)#network 192.168.4.0 0.0.0.255 area 0
Router(config-router)#network 192.168.5.0 0.0.0.255 area 0
Router(config-router)#exit
Router(config)#
```

- OSPF configuration for Router 2 by using below common

- Here, Router 0 can connect with network 192.168.3.0, 192.168.4.0 and 192.168.5.0. So, we can define that in OSPF configuration.

- In command “Router OSPF 1” means here 1 is process_id you can assign it to anything.



Simulation Panel

Event List

Vis.	Time(sec)	Last Device	At Device	Type
	0.001	PC0	Router0	ICMP
	0.001	--	Router1	OSPF
	0.002	Router1	Router0	OSPF
	0.002	Router0	Router2	ICMP
	0.003	Router2	PC1	ICMP
	0.004	PC1	Router2	ICMP
	0.005	Router2	Router0	ICMP
Visible	0.006	Router0	PC0	ICMP

Reset Simulation ☑ Constant Delay Captured to: 0.006 s

Play Controls

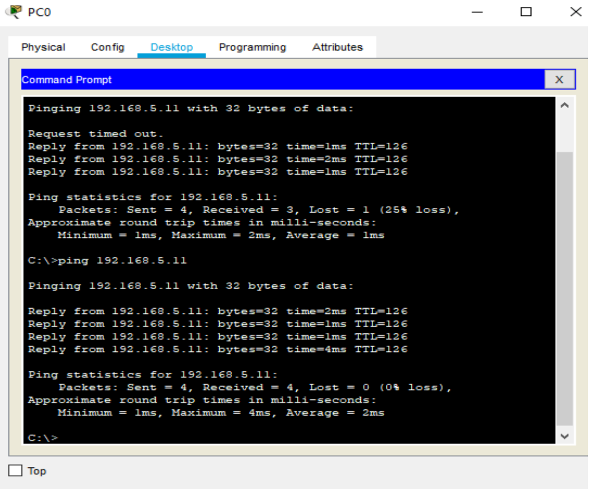
Event List Filters - Visible Events

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, Meraki, 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

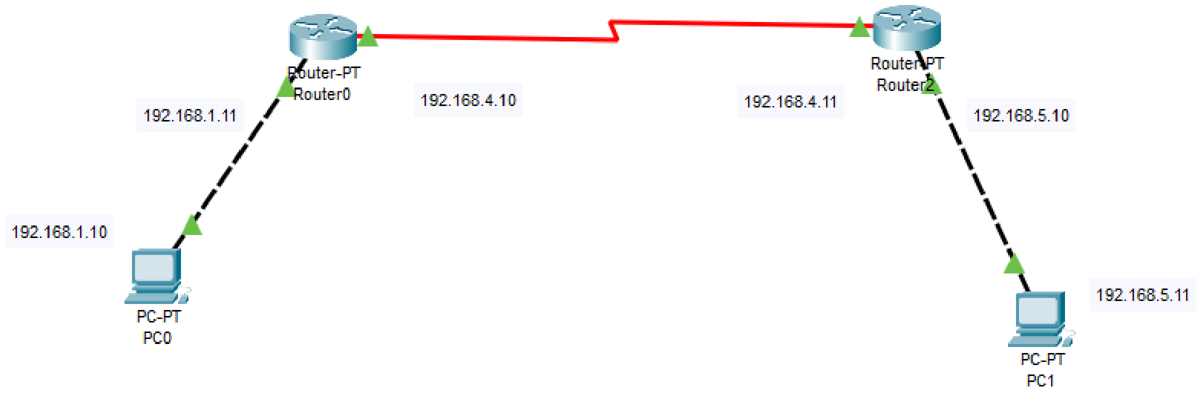
Fire	Last Status	Source	Destination	Type	Color	Time(sec)	Periodic	Num	Ed
●	Successful	PC0	PC1	ICMP	■	0.000	N	0	(e

- Now let’s check if it working or not, for checking we can ping PC0 from PC1.



```
Pinging 192.168.5.11 with 32 bytes of data:

Request timed out.
Reply from 192.168.5.11: bytes=32 time=1ms TTL=126
Reply from 192.168.5.11: bytes=32 time=2ms TTL=126
Reply from 192.168.5.11: bytes=32 time=1ms TTL=126

Ping statistics for 192.168.5.11:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 2ms, Average = 1ms

C:\>ping 192.168.5.11

Pinging 192.168.5.11 with 32 bytes of data:

Reply from 192.168.5.11: bytes=32 time=2ms TTL=126
Reply from 192.168.5.11: bytes=32 time=1ms TTL=126
Reply from 192.168.5.11: bytes=32 time=1ms TTL=126
Reply from 192.168.5.11: bytes=32 time=4ms TTL=126

Ping statistics for 192.168.5.11:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 4ms, Average = 2ms

C:\>
```

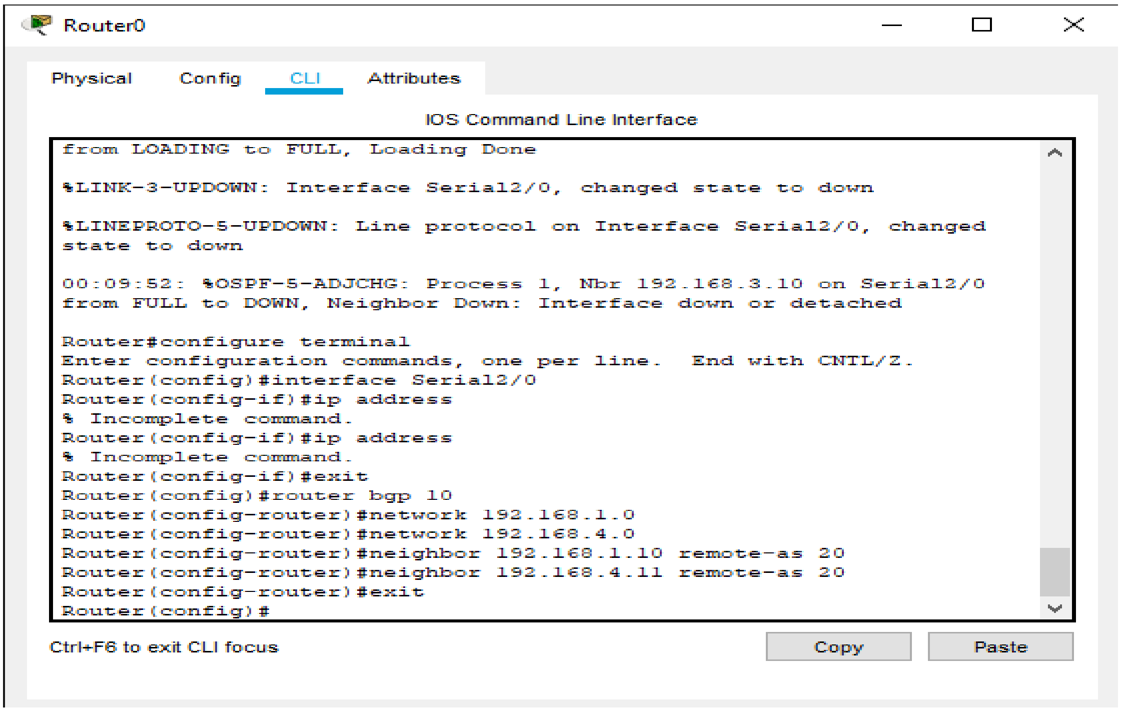
Q-4) Create a simple network topology and configure BGP routing for data communication. Record necessary steps and screenshots.



BGP as Border gateway protocol

- 1st configure all routers and pc with IP address and after setup whole network we can configure for BGP routing on both router.

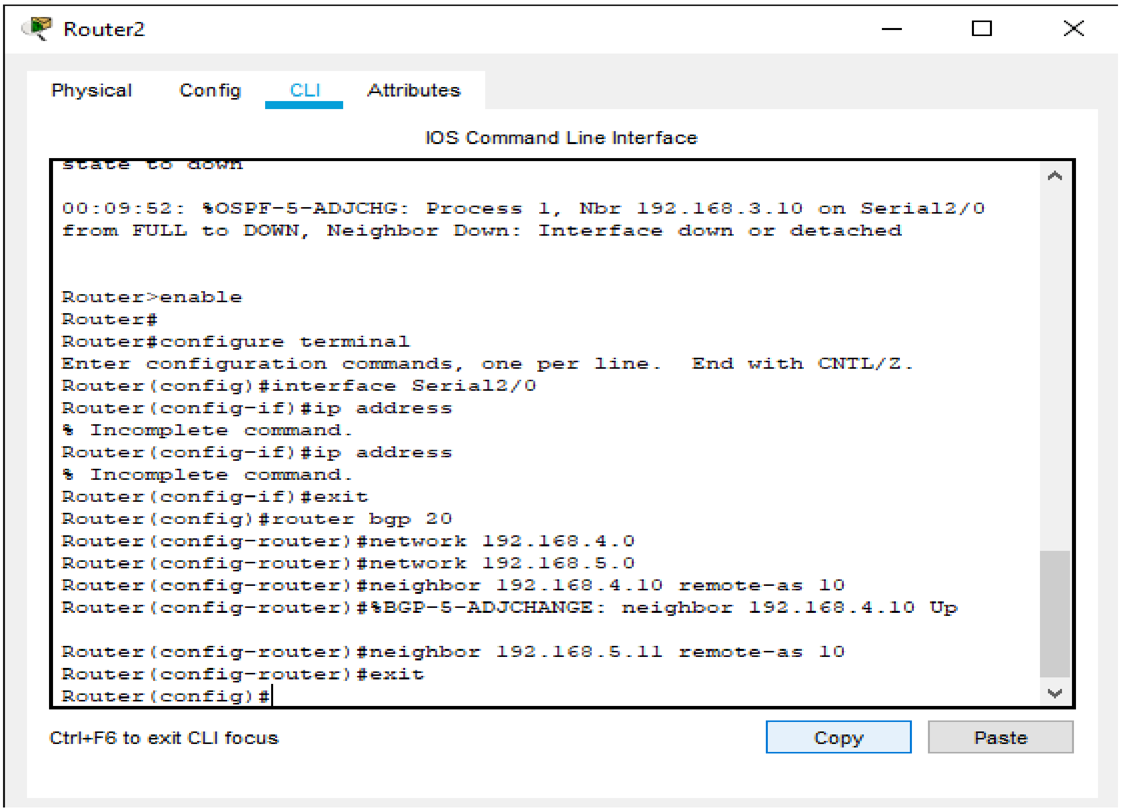
- BGP configuration on Router 0 by using below commands.



```
from LOADING to FULL, Loading Done

%LINK-3-UPDOWN: Interface Serial2/0, changed state to down

%LINEPROTO-5-UPDOWN: Line protocol on Interface Serial2/0, changed
state to down

00:09:52: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.3.10 on Serial2/0
from FULL to DOWN, Neighbor Down: Interface down or detached

Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface Serial2/0
Router(config-if)#ip address
% Incomplete command.
Router(config-if)#ip address
% Incomplete command.
Router(config-if)#exit
Router(config)#router bgp 10
Router(config-router)#network 192.168.1.0
Router(config-router)#network 192.168.4.0
Router(config-router)#neighbor 192.168.1.10 remote-as 20
Router(config-router)#neighbor 192.168.4.11 remote-as 20
Router(config-router)#exit
Router(config)#
```

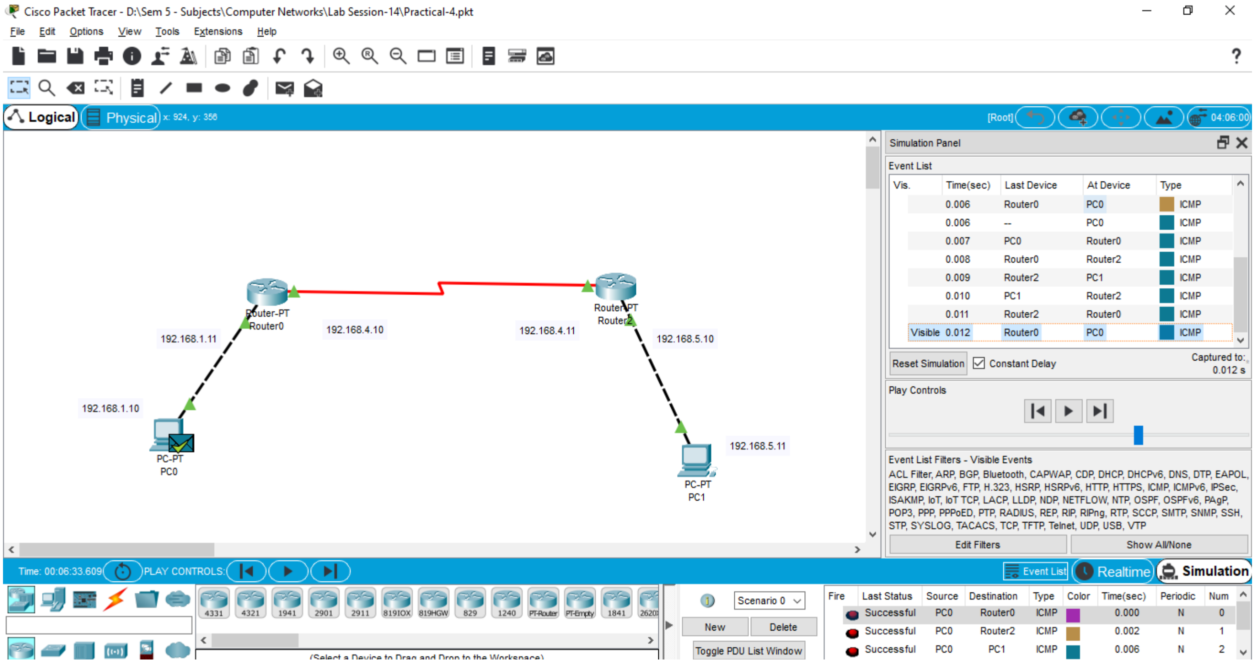
(Here, after BGP 10 is the id of that BGP)

- BGP configuration for Router 2 by using below commands.



```
state to down

00:09:52: %OSPF-5-ADJCHG: Process 1, Nbr 192.168.3.10 on Serial2/0
from FULL to DOWN, Neighbor Down: Interface down or detached


Router>enable
Router#
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface Serial2/0
Router(config-if)#ip address
% Incomplete command.
Router(config-if)#ip address
% Incomplete command.
Router(config-if)#exit
Router(config)#router bgp 20
Router(config-router)#network 192.168.4.0
Router(config-router)#network 192.168.5.0
Router(config-router)#neighbor 192.168.4.10 remote-as 10
Router(config-router)#%BGP-5-ADJCHANGE: neighbor 192.168.4.10 Up

Router(config-router)#neighbor 192.168.5.11 remote-as 10
Router(config-router)#exit
Router(config)#
```

- Now let’s check if it working or not.

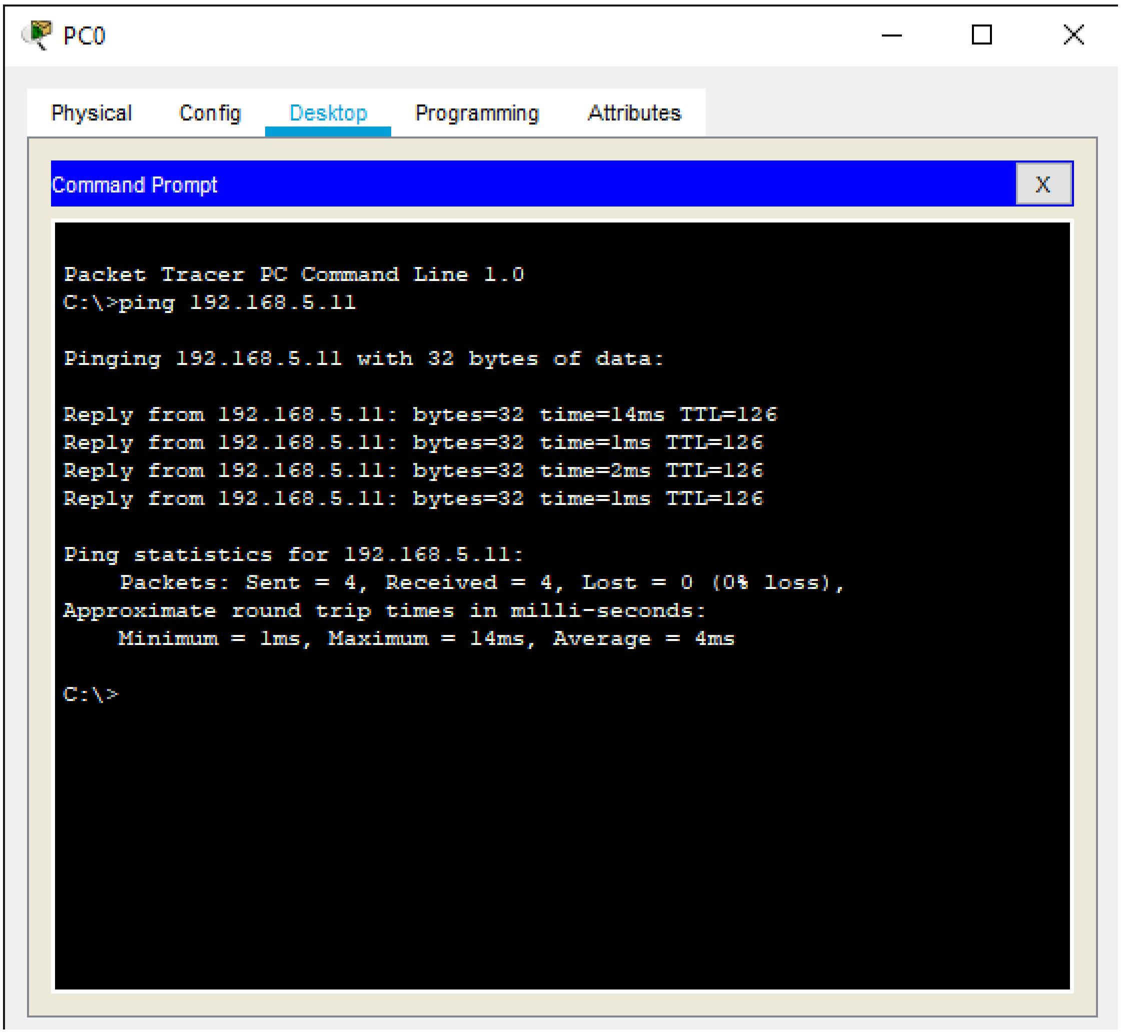


Vis.	Time(sec)	Last Device	At Device	Type
	0.006	Router0	PC0	ICMP
	0.006	--	PC0	ICMP
	0.007	PC0	Router0	ICMP
	0.008	Router0	Router2	ICMP
	0.009	Router2	PC1	ICMP
	0.010	PC1	Router2	ICMP
	0.011	Router2	Router0	ICMP
Visible	0.012	Router0	PC0	ICMP

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

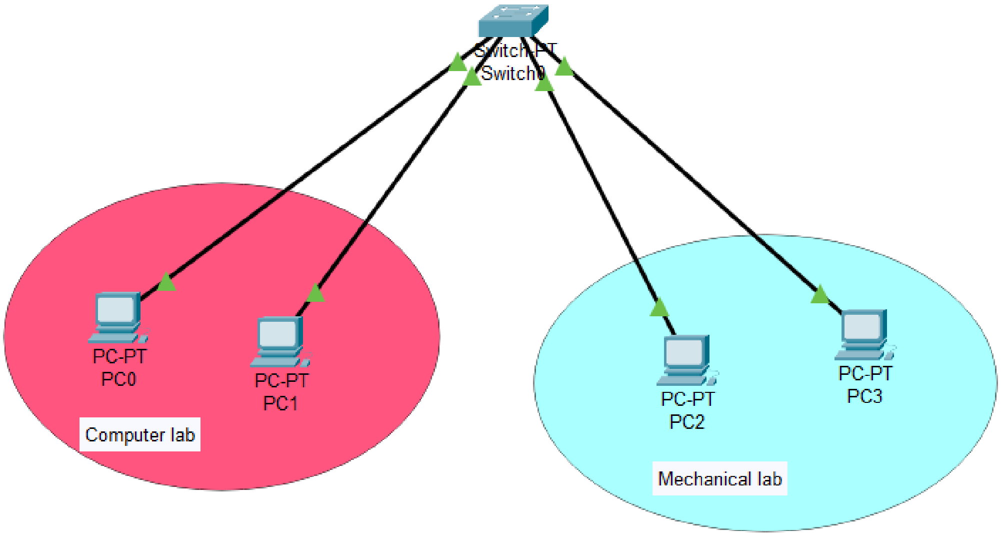
Fire	Last Status	Source	Destination	Type	Color	Time(sec)	Periodic	Num
●	Successful	PC0	Router0	ICMP		0.000	N	0
●	Successful	PC0	Router2	ICMP		0.002	N	1
●	Successful	PC0	PC1	ICMP		0.006	N	2

- Check using to send ping request from pc0 to pc1.

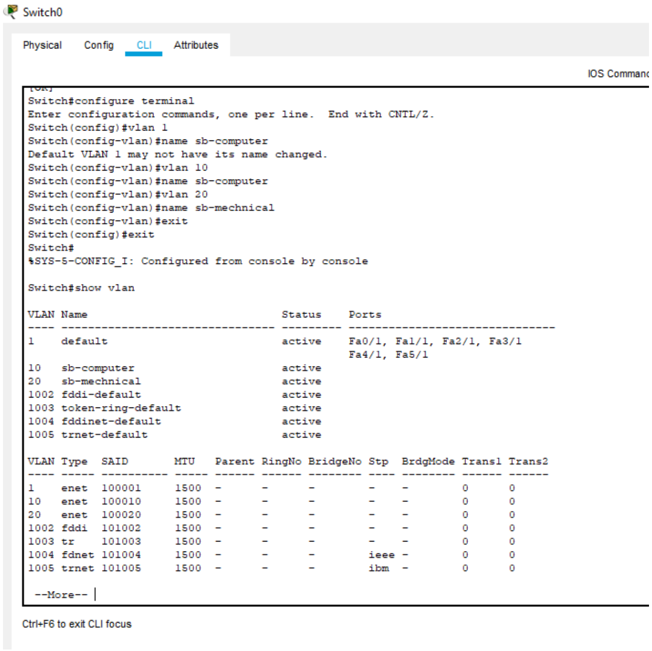


```
Packet Tracer PC Command Line 1.0
C:\>ping 192.168.5.11

Pinging 192.168.5.11 with 32 bytes of data:

Reply from 192.168.5.11: bytes=32 time=14ms TTL=126
Reply from 192.168.5.11: bytes=32 time=1ms TTL=126
Reply from 192.168.5.11: bytes=32 time=2ms TTL=126
Reply from 192.168.5.11: bytes=32 time=1ms TTL=126

Ping statistics for 192.168.5.11:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 14ms, Average = 4ms

C:\>
```

Q-5) Implement the concept of VLAN using Network Simulator. Create a small network topology and implement at least 2 different VLAN. The data communication is possible only between the machines of same VLAN.



- Creating the VLAN:

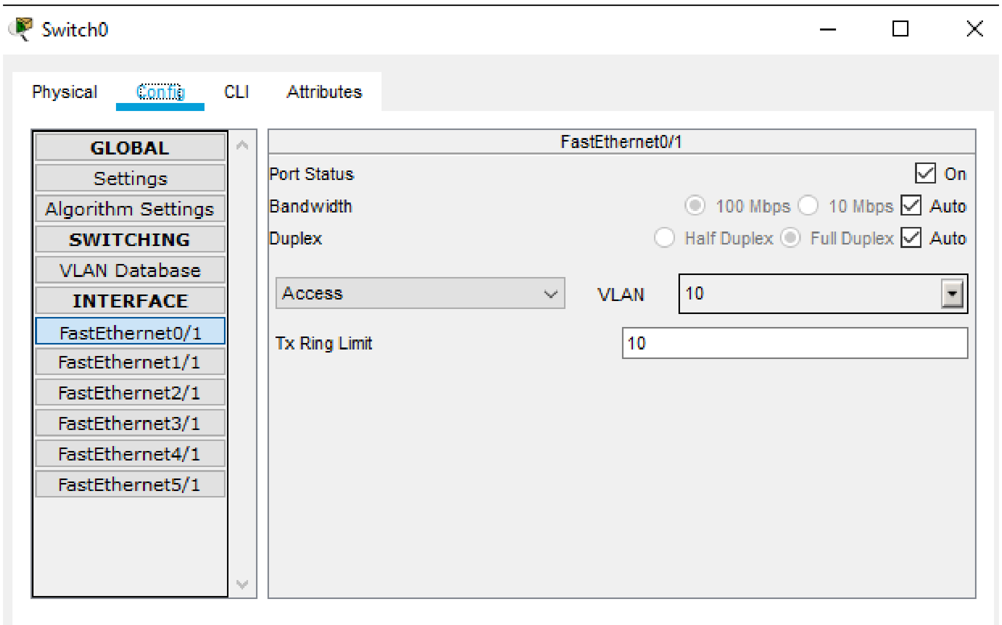


```
Switch#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#vlan 1
Switch(config-vlan)#name sb-computer
Default VLAN 1 may not have its name changed.
Switch(config-vlan)#vlan 10
Switch(config-vlan)#name sb-computer
Switch(config-vlan)#vlan 20
Switch(config-vlan)#name sb-mechnical
Switch(config-vlan)#exit
Switch(config)#exit
Switch#
%SYS-5-CONFIG_I: Configured from console by console

Switch#show vlan

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Fa0/1, Fa1/1, Fa2/1, Fa3/1
                                                Fa4/1, Fa5/1
10   sb-computer                      active
20   sb-mechnical                     active
1002 fddi-default                     active
1003 token-ring-default               active
1004 fddinet-default                  active
1005 trnet-default                    active

VLAN Type  SAID       MTU   Parent RingNo BridgeNo Stp  BrdgMode Trans1 Trans2
---- ----- ---------- ----- ------ ------ -------- ---- -------- ------ ------
1    enet  100001     1500  -      -      -        -    -        0      0
10   enet  100010     1500  -      -      -        -    -        0      0
20   enet  100020     1500  -      -      -        -    -        0      0
1002 fddi  101002     1500  -      -      -        -    -        0      0
1003 tr    101003     1500  -      -      -        -    -        0      0
1004 fdnet 101004     1500  -      -      -        ieee -        0      0
1005 trnet 101005     1500  -      -      -        ibm  -        0      0

 --More--
```

Ctrl+F6 to exit CLI focus

(“vlan 10” create vlan 10 and for give name you can use “name <any_name>”. For see how many vlan is created in switch you can write “show vlan” command.) After creating VLAN set interfaces to particular VLAN.

- Here, fa0/1 is in VLAN 10
  fa1/1 is also in VLAN 10
  fa2/1 is in VLAN 20
  fa3/1 is also in VLAN 20



FastEthernet0/1

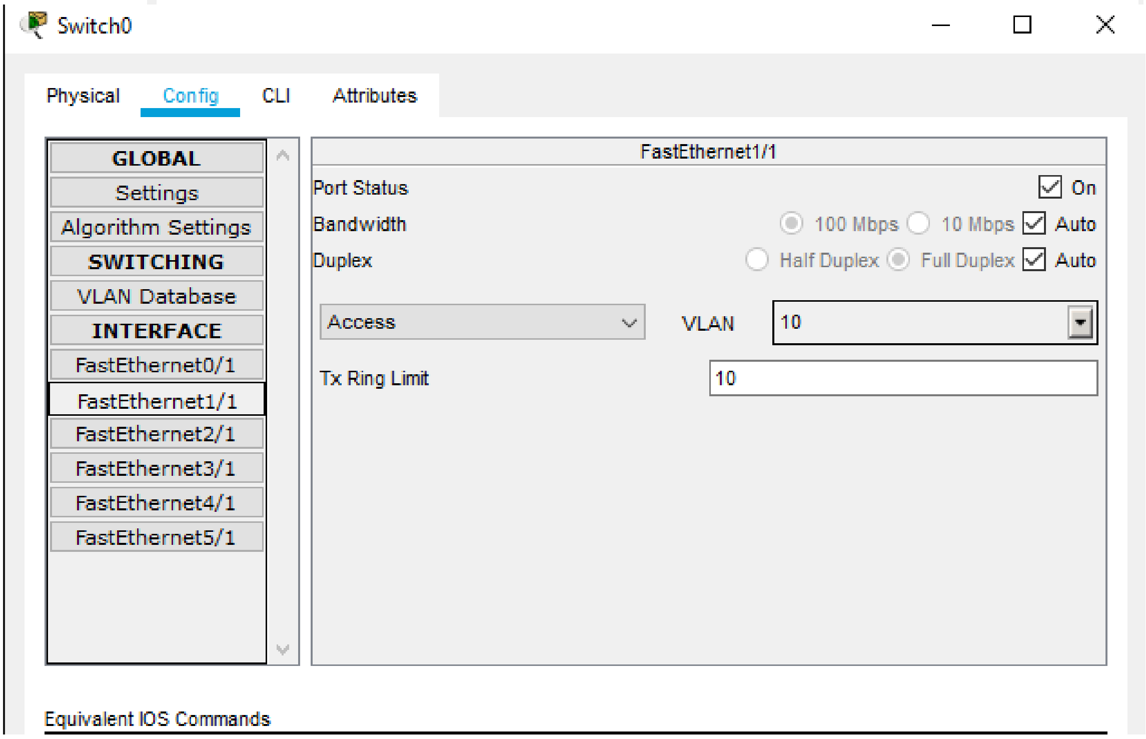
Port Status ☑ On

Bandwidth ● 100 Mbps ○ 10 Mbps ☑ Auto

Duplex ○ Half Duplex ● Full Duplex ☑ Auto

Access VLAN 10

Tx Ring Limit 10



FastEthernet1/1

Port Status ☑ On

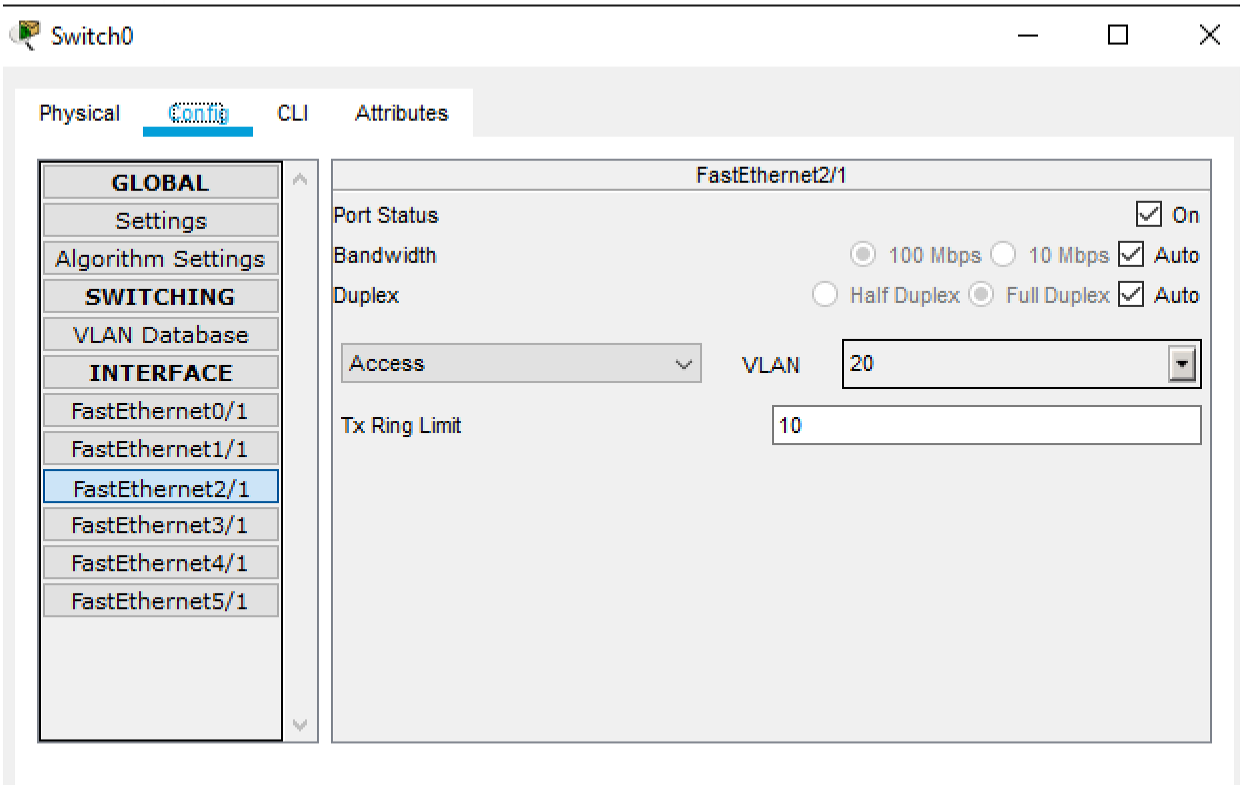
Bandwidth ● 100 Mbps ○ 10 Mbps ☑ Auto

Duplex ○ Half Duplex ● Full Duplex ☑ Auto

Access VLAN 10

Tx Ring Limit 10

Equivalent IOS Commands



FastEthernet2/1

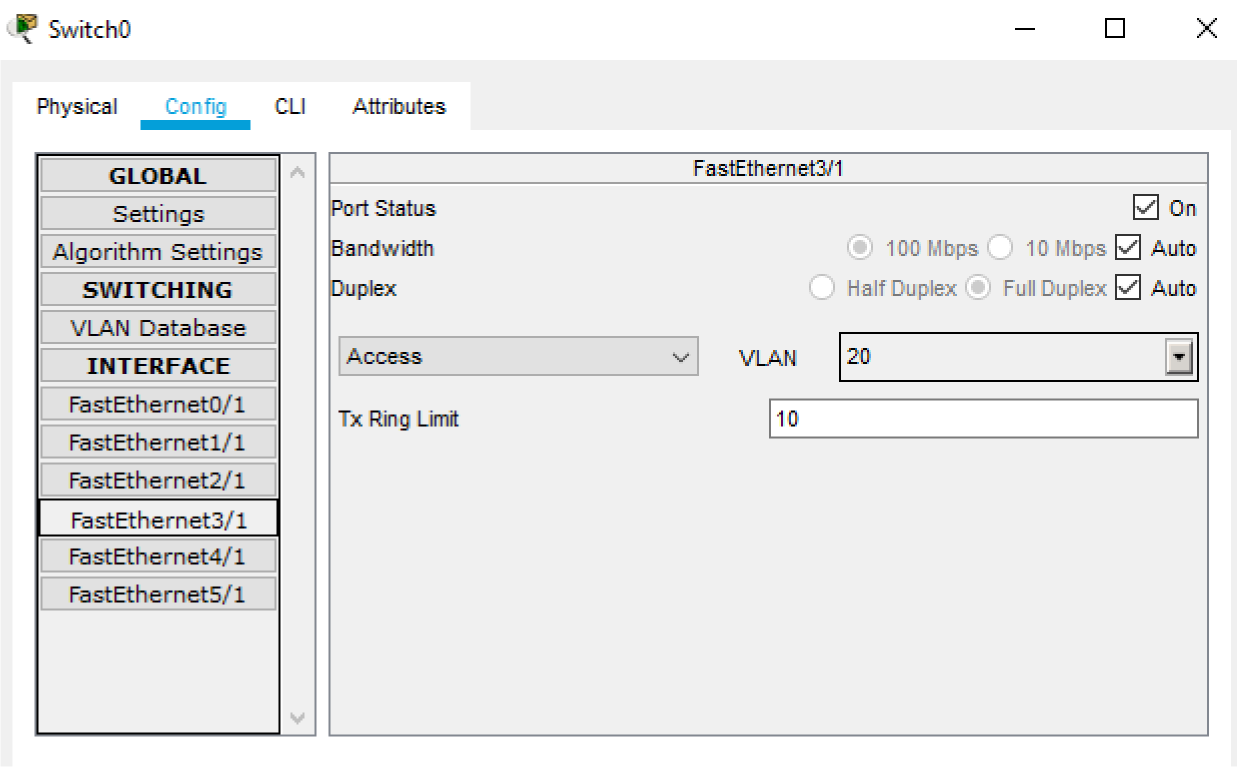
Port Status ☑ On

Bandwidth ● 100 Mbps ○ 10 Mbps ☑ Auto

Duplex ○ Half Duplex ● Full Duplex ☑ Auto

Access VLAN 20

Tx Ring Limit 10



FastEthernet3/1

Port Status ☑ On

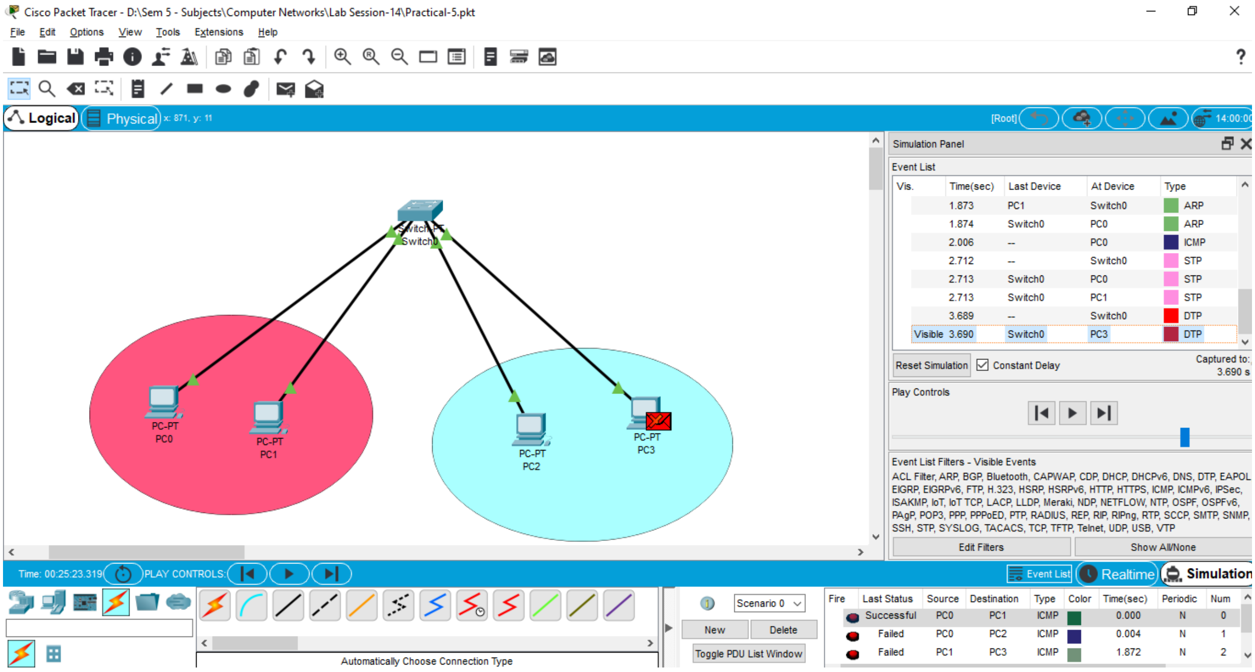
Bandwidth ● 100 Mbps ○ 10 Mbps ☑ Auto

Duplex ○ Half Duplex ● Full Duplex ☑ Auto

Access VLAN 20

Tx Ring Limit 10

- After you can see below packet sending results....



Cisco Packet Tracer - D:\Sem 5 - Subjects\Computer Networks\Lab Session-14\Practical-5.pkt

Simulation Panel

Event List

Vis.	Time(sec)	Last Device	At Device	Type
	1.873	PC1	Switch0	🟩 ARP
	1.874	Switch0	PC0	🟩 ARP
	2.006	--	PC0	🟦 ICMP
	2.712	--	Switch0	🟪 STP
	2.713	Switch0	PC0	🟪 STP
	2.713	Switch0	PC1	🟪 STP
	3.689	--	Switch0	🟥 DTP
Visible	3.690	Switch0	PC3	🟥 DTP

Reset Simulation ☑ Constant Delay Captured to: 3.690 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, Meraki, 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

Time: 00:25:23.319

Scenario 0

Fire	Last Status	Source	Destination	Type	Color	Time(sec)	Periodic	Num
●	Successful	PC0	PC1	ICMP	🟩	0.000	N	0
●	Failed	PC0	PC2	ICMP	🟦	0.004	N	1
●	Failed	PC1	PC3	ICMP	🟩	1.872	N	2

Automatically Choose Connection Type

(Here, you can see PC0 to PC1 packet is successfully transfer because both pc are in same VLAN and PC0 to PC2 is failed because those both pc are in different VLAN, and same for PC1 to PC3)