CONSOLIDATED LAB RECORD

COURSE: COMPUTER NETWORKS

COURSE CODE: BCSE308P

BACHELOR OF TECHNOLOGY

in

ELECTRONICS AND COMPUTER ENGINEERING

SUBMITTED BY: Thurlapati Sai Sree Praneetha

22BLC1350

Study Experiment: 1a **Date: 22/07/24**

Study of Basic Networking Commands

Name **:** Thurlapati Sai Sree Praneetha
Reg. No **:** 22BLC1350
Faculty Name **:** Jaya Vignesh T

Aim:

To study and understand the basic networking commands.

Theory:

A computer network consists of several computers connected together. The network can be as simple as a few computers connected in your home or office, or as complicated as a large university network or even the entire Internet. When your computer is part of a network, you have access to those systems either directly or through services like mail and the web.

There are a variety of networking programs that you can use. Some are handy for performing diagnostics to see if everything is working properly. Others (like mail readers and web browsers) are useful for getting your work done and staying in contact with other people.

COMMANDS

1) ipconfig:

Description of the Command: ipconfig (Internet Protocol Configuration) view and configure the network settings on a Windows computer. It's often used to troubleshoot network issues, like connectivity problems or incorrect IP addresses.

Example:

```
Microsoft Windows [Version 10.0.22621.1555]
(c) Microsoft Corporation. All rights reserved.

C:\Users\student.ACN-08>ipconfig

Windows IP Configuration


Ethernet adapter Ethernet:

   Connection-specific DNS Suffix  . :
   Link-local IPv6 Address . . . . . : fe80::4fac:cb7d:cff6:ca68%2
   IPv4 Address. . . . . . . . . . . : 172.16.80.28
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 172.16.80.1
```

2) ping:

Description of the Command: Ping is a utility command that uses the protocol called the ICMP (Internet Control Message Protocol). It is a command used to send some test data and check if we can communicate a request message to a target host and waiting for a response. It sends an Echo Request Packet and receives an Echo Reply Packet.

Example:

```
C:\Users\student.ACN-08>ping 172.16.80.27

Pinging 172.16.80.27 with 32 bytes of data:
Reply from 172.16.80.27: bytes=32 time<1ms TTL=128
Reply from 172.16.80.27: bytes=32 time<1ms TTL=128
Reply from 172.16.80.27: bytes=32 time=1ms TTL=128
Reply from 172.16.80.27: bytes=32 time=1ms TTL=128

Ping statistics for 172.16.80.27:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

```
C:\Users\student.ACN-08>ping 172.16.80.44

Pinging 172.16.80.44 with 32 bytes of data:
Reply from 172.16.80.28: Destination host unreachable.
Reply from 172.16.80.28: Destination host unreachable.
Reply from 172.16.80.28: Destination host unreachable.
Reply from 172.16.80.28: Destination host unreachable.

Ping statistics for 172.16.80.44:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
```

3) <u>tracert:</u>

Description of the Command: The command tracert stands for traceroute. It is a tool that tracks the path of information packets from a source to a destination across an IP network**.**

Example:

```
C:\Users\student.ACN-08>tracert lms.vit.ac.in

Tracing route to lms.vit.ac.in [172.16.0.71]
over a maximum of 30 hops:

  1    <1 ms    <1 ms     5 ms  172.16.80.1
  2    <1 ms    <1 ms    <1 ms  lms.vit.ac.in [172.16.0.71]

Trace complete.
```

```
C:\Users\student.ACN-08>tracert 172.16.80.27

Tracing route to DESKTOP-5B7J2DE [172.16.80.27]
over a maximum of 30 hops:

  1    <1 ms    <1 ms    <1 ms  DESKTOP-5B7J2DE [172.16.80.27]

Trace complete.
```

```
C:\Users\student.ACN-08>tracert vit.ac.in

Tracing route to vit.ac.in [122.184.65.22]
over a maximum of 30 hops:

  1    <1 ms    <1 ms    <1 ms  172.16.80.1
  2    <1 ms    <1 ms    <1 ms  172.16.0.2
  3     *        *        *     Request timed out.
  4     1 ms     1 ms     1 ms  136.232.208.189
  5     *        *        *     Request timed out.
  6     *        *        *     Request timed out.
  7     *        *        *     Request timed out.
  8     2 ms     2 ms     2 ms  49.44.220.189
  9     *        *        *     Request timed out.
 10     3 ms     4 ms     3 ms  116.119.72.73
 11     5 ms     5 ms     5 ms  182.79.14.90
 12     *        *        *     Request timed out.
 13     *        *        *     Request timed out.
 14     *        *        *     Request timed out.
 15     6 ms     7 ms     6 ms  www.vit.ac.in [122.184.65.22]

Trace complete.
```

4) <u>nslookup</u>

Description of the Command: nslookup (Name Server Lookup) is a tool that lets users query their DNS service to find out a domain name's IP address or DNS record. It can also perform a reverse DNS lookup to find the domain name for a given IP address.

Example:

```
C:\Users\student.ACN-08>nslookup lms.vit.ac.in
Server:  vitccdns
Address:  172.16.1.11

Name:    lms.vit.ac.in
Address:  172.16.0.71
```

```
C:\Users\student.ACN-08>nslookup youtube.com
Server:  vitccdns
Address:  172.16.1.11

Non-authoritative answer:
Name:    youtube.com
Addresses:  2404:6800:4007:819::200e
          142.250.182.14
```

5) <u>getmac:</u>

Description of the Command: It is a hexadecimal representation. It has a 48bit address. It is also called as physical address or hardware address. The getmac command displays MAC (Media Access Control) address network protocols for each network adapter on a computer.

Example:

```
C:\Users\student.ACN-08>getmac

Physical Address    Transport Name
=================== ==========================================================
F4-6B-8C-8B-37-F5   \Device\Tcpip_{04BB5C63-3F02-47F3-9C64-80F629FF1373}
```

Result:

Thus a few basic networking commands are studied and practiced successfully.

Experiment No: 1b **Date: 23/07/24**

Client-Server Network Topology using Cisco packet tracer and Also build network topologies

Name : Thurlapati Sai Sree Praneetha
Reg. No : 22BLC1350
Faculty Name : Jaya Vignesh T

Aim:

a. To set up Client-Server Network Topology using Cisco packet tracer
b. To Analyze the Network behavior in both Simulation Mode and Real Time Mode
c. To set up HTTP, DNS services and analyze the network behavior
d. Build star, mesh topology using hub and switches and analyze network behavior

Tool Required:

Cisco Packet Tracer

Theory:

CISCO PACKET TRACER:

INTRODUCTION:

- Cisco Packet Tracer is an advanced network simulation tool created by Cisco Systems, renowned for its comprehensive capabilities in teaching and learning networking concepts.
- It serves as a fundamental resource for students and professionals preparing for Cisco certifications. Packet Tracer offers a virtual environment where users can design, configure, and troubleshoot network setups, mirroring real-world networking scenarios.
- Cisco Packet Tracer supports a wide range of networking protocols, including routing protocols (OSPF, EIGRP, BGP), switching protocols (VLANs, STP), and application protocols (HTTP, FTP, DNS).
- Its Real-time and Simulation Modes allow users to experience network operations as they occur in reality and analyze data flow step-by-step, respectively.

FEATURES:

- Network Simulation: Simulate complex networks with a variety of devices.
- Visualization: Interactive network diagram and device configuration.
- Multi-user Collaboration: Allows multiple users to work on the same project.
- Learning and Assessment: Integrated with Cisco Networking Academy.

WORKSPACES:

- Logical Workspace: Design and configure network topologies.
- Physical Workspace: Visualize physical layout and device placements.

OPERATING MODES:

- Real-time Mode: Operate networks as they would function in the real world.
- Simulation Mode: Visualize data flow and packet transfer in a step-by-step manner.

PROTOCOLS:

- Routing Protocols: OSPF, EIGRP, BGP
- Switching Protocols: VLANs, STP
- Application Protocols: HTTP, FTP, DNS

PROCEDURE:

1. **Setting Up the Environment**

   Open the Cisco Packet Tracer application.

   Go to File > New.

2. **Adding Devices**

   Drag and drop a server from the End Devices section onto the workspace.

   Drag and drop one or more PCs from the End Devices section onto the workspace.

3. **Adding a Switch**

   Drag and drop a switch from the Switches section onto the workspace.

4. **Connecting Devices**

   Use Copper Straight-Through cables to connect each client PC to the switch, noting the IO ports.

   Use a Copper Cross-Over cable to connect the server to the switch.

5. **Configuring IP Addresses**

   Server IP Configuration:

   - Click on the server.
   - Navigate to the Desktop tab.
   - Select IP Configuration.
   - Assign an IP address.

Client IP Configuration:

- Click on each client PC.
- Navigate to the Desktop tab.
- Select IP Configuration.
- Assign IP addresses from 192.168.1.1 to 192.168.1.11.

6. **Verifying Connectivity**

Ping Test:

- Click on a client PC.
- Go to the Desktop tab.
- Open the Command Prompt.
- Use the ping command to test connectivity between the Server and PCs.

7. **Testing the Network in Simulation Mode:**

- Open the simulation mode to visualize data transfer. The simulation panel and event list will display.
- Click the closed envelope symbol to select the source and receiver.
- Toggle Show All/None to view the appropriate protocol, then select ICMP under Edit Filters.
- Watch the packet flow in the simulation panel on the right.

8. **Configuring HTTP & DNS Services on Server in Real-Time Mode:**

Enable HTTP Service:

- Switch to Real-Time Mode.
- Click on the server, go to the Services tab, and select HTTP.
- Ensure the HTTP service is enabled and the default index.html page is available.
- On a client device, open the Desktop tab, launch the Web Browser, and enter the server's IP address.

Set Up DNS Service:

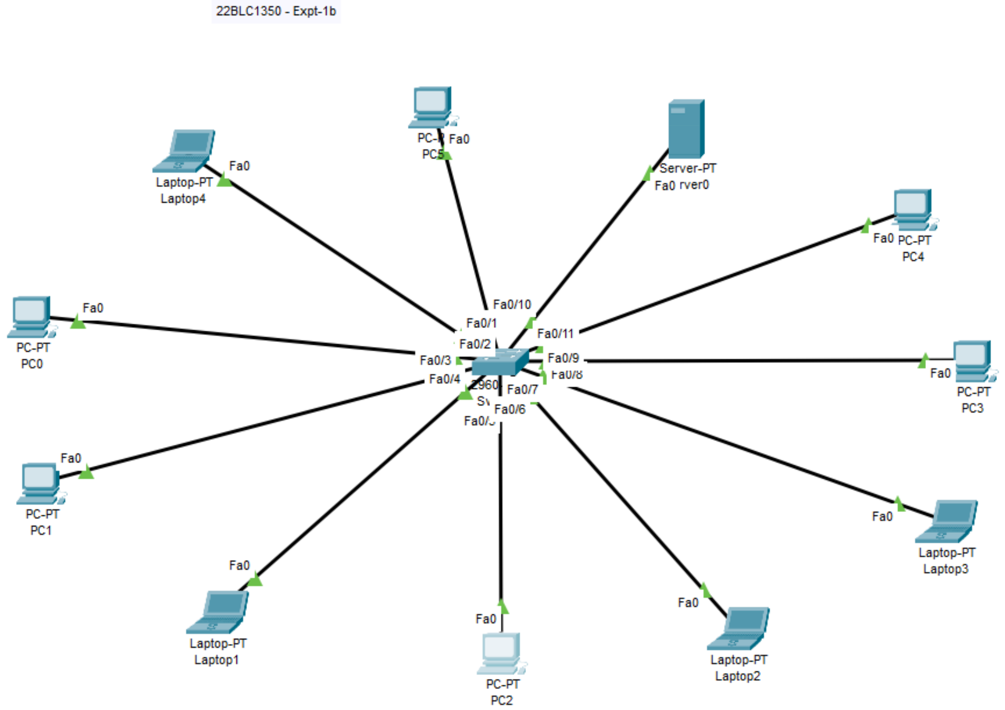
- Create a new server and rename it for DNS service.
- Assign the appropriate IP address to the DNS server.
- On the DNS server, go to the Services tab, select DNS, and enter the website name and IP address of the HTTP server.
- Click Add to save the DNS entry.

Configure DNS Settings on Devices:

Update each device's global settings to use the DNS server's IP address.

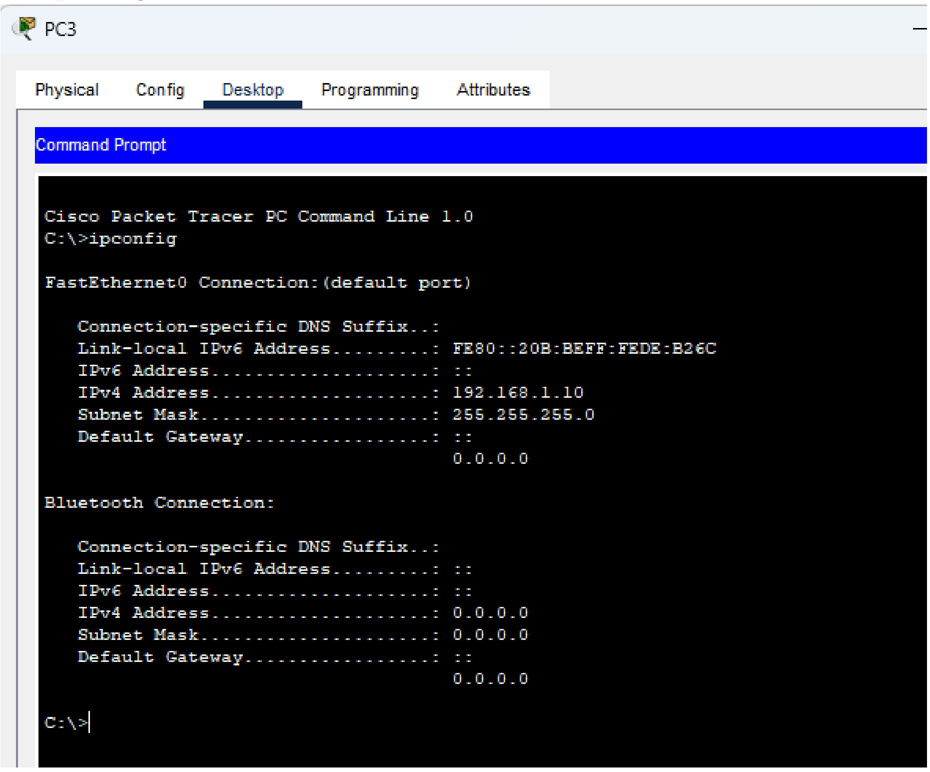
EXERCISE:

a) **Build a Local Area Network with at least 10 clients, 1 switch and a server using star topology**

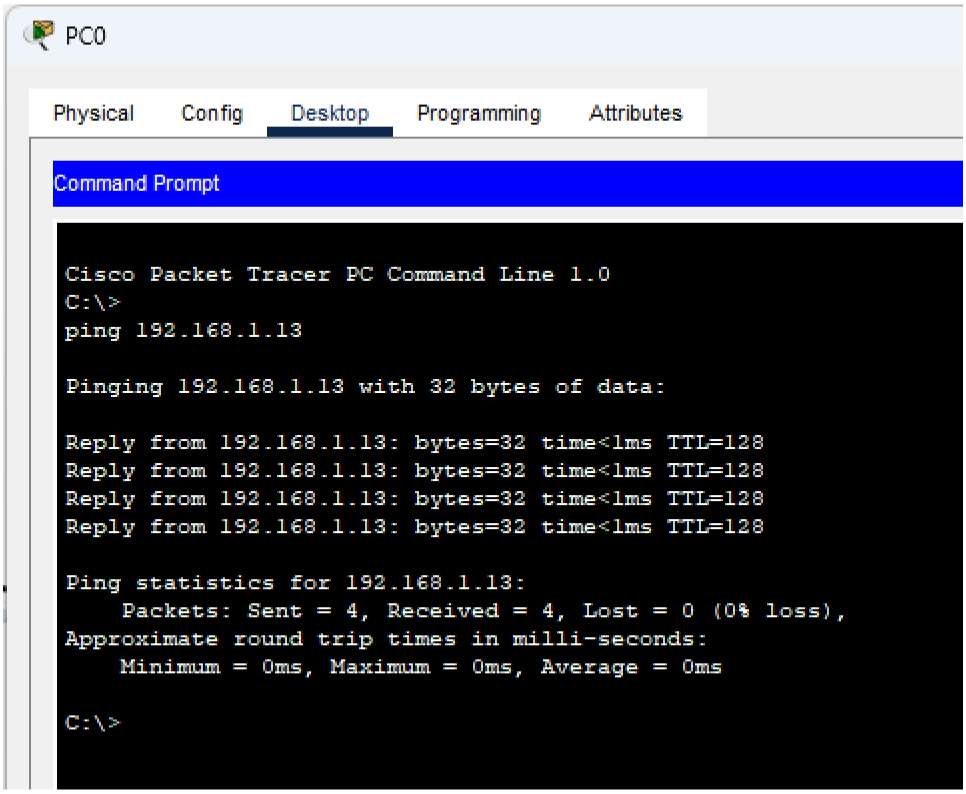


b) Configure the entire network and verify the connection between all clients and server

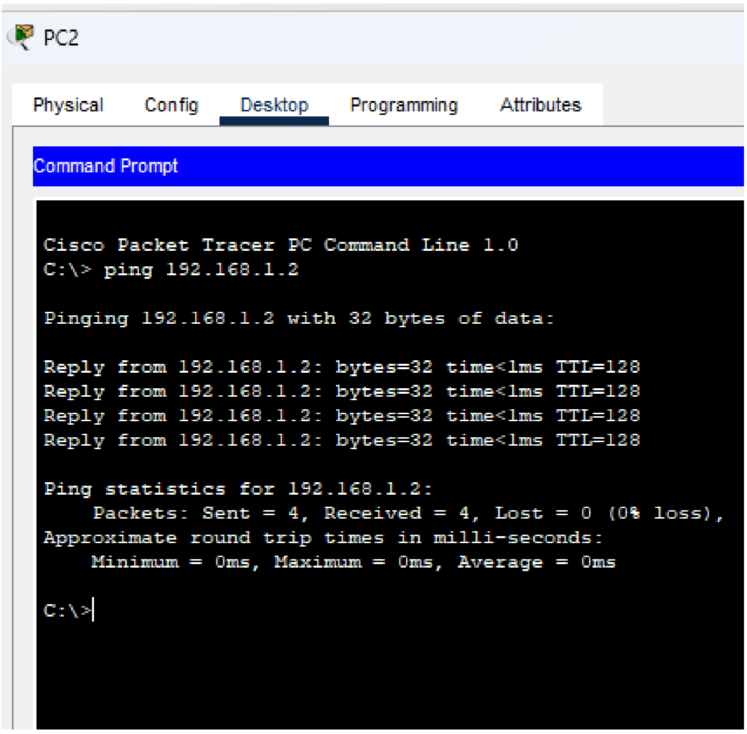
PC ipconfig:



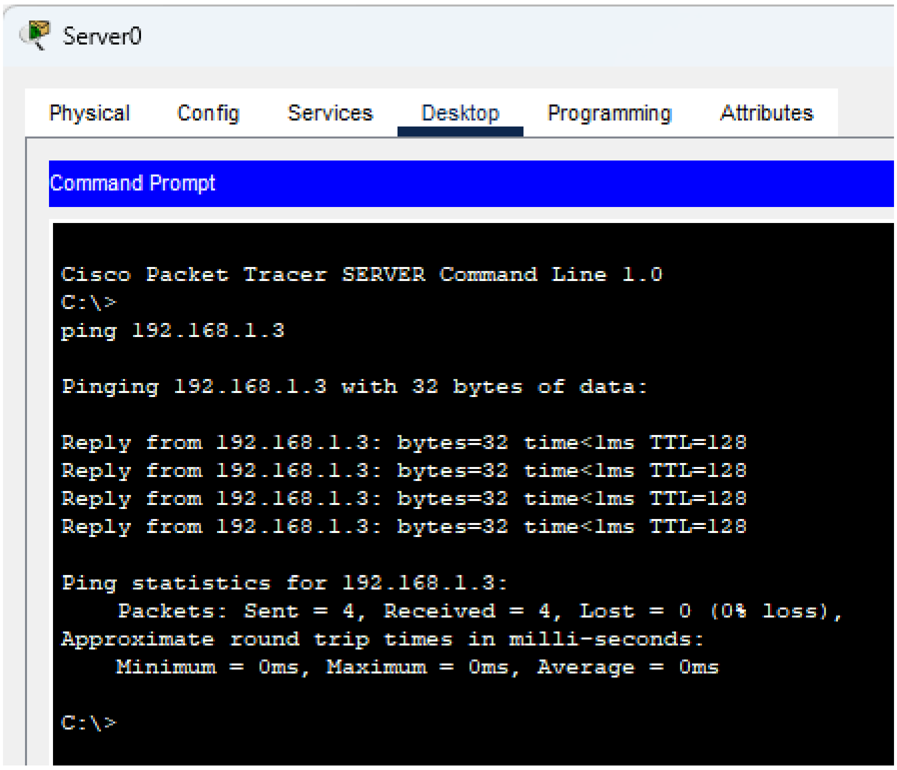
PINGING PC TO LAPTOP



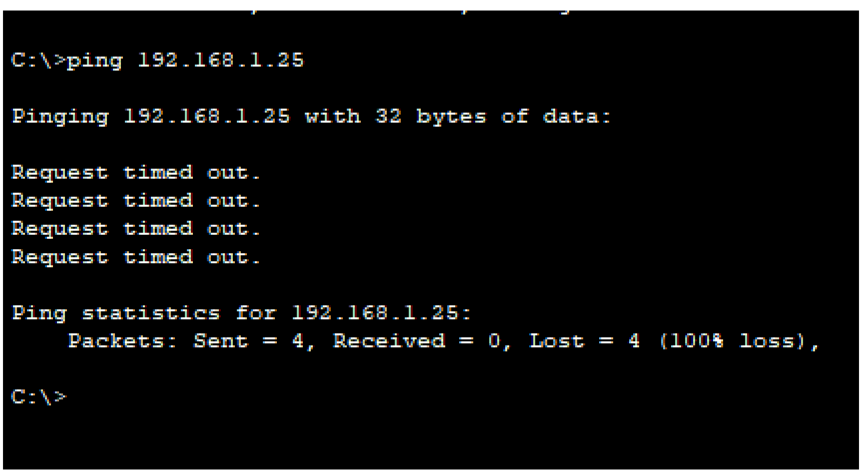
PINGING PC TO SERVER



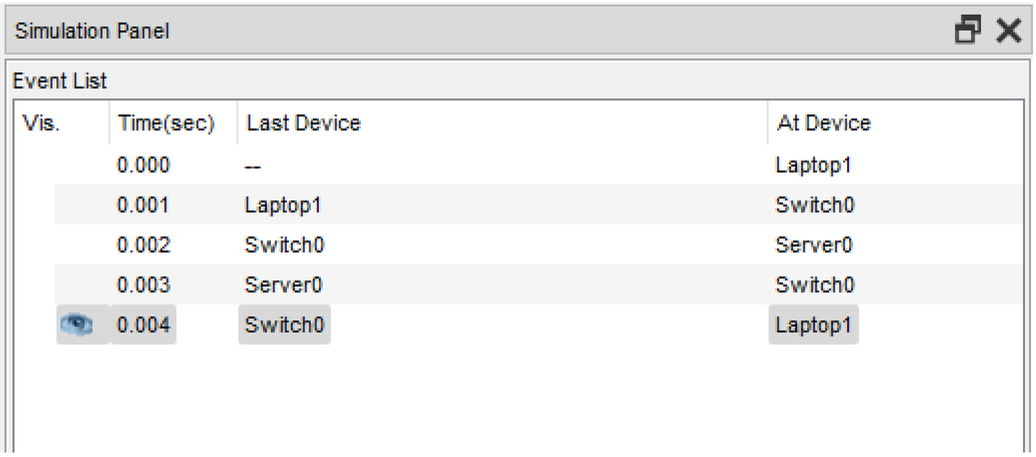
PINGING SERVER TO PC:



EXAMPLE OF AN ERROR:

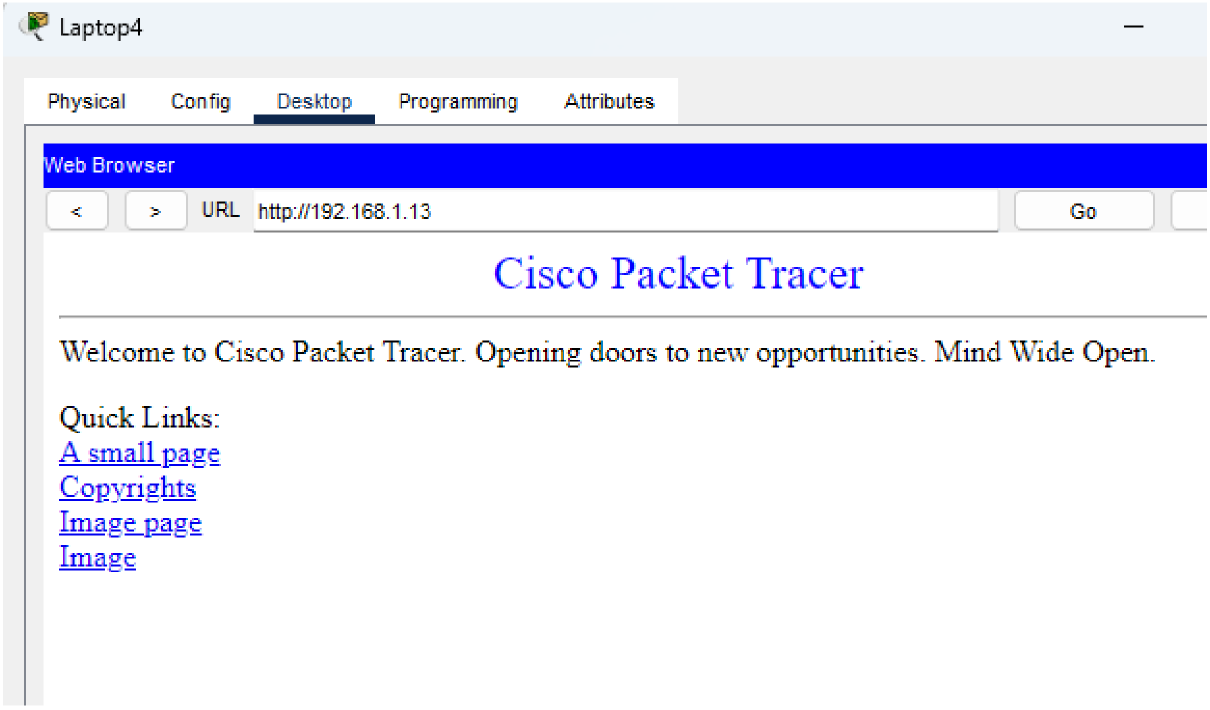


c) Analyze the network behaviour in simulation mode

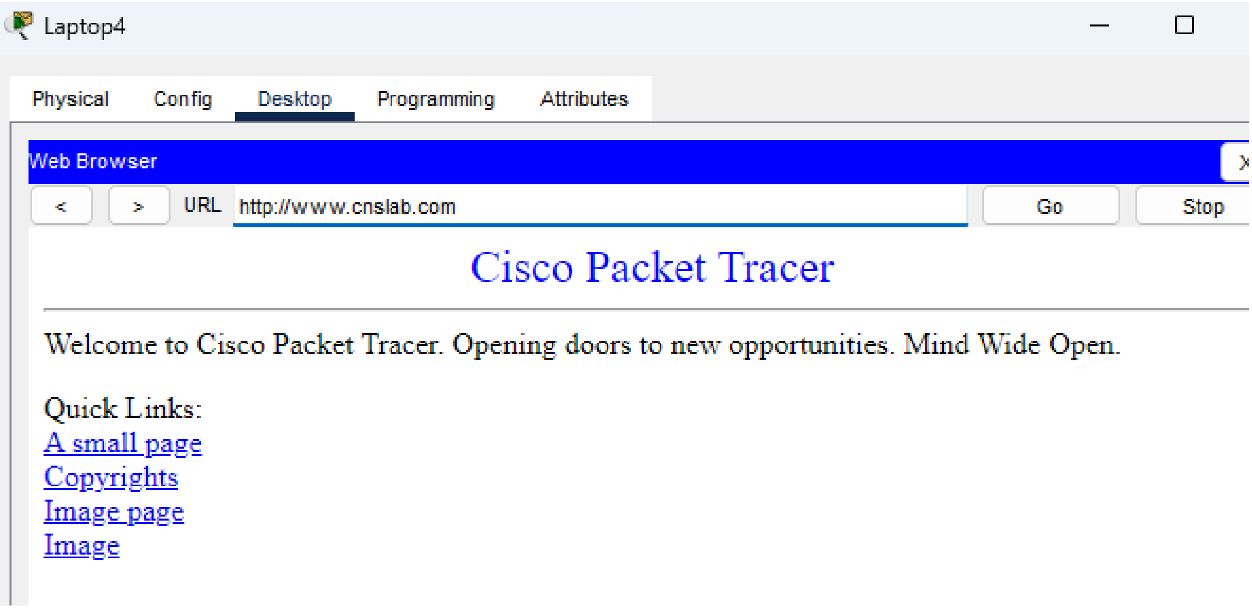


d) Configure the HTTP and DNS services in the server and analyse the network behaviour in real time mode

1. HTTP:



After adding the link of DNS Server:



INFERENCES & RESULT: I understood how to set up a client server network with PC's,Laptops and Servers. With the use of the network I have gained knowledge on how the real time mode and simulation mode works. I have also understood how the HTTP server handles web page requests, and the DNS server converts domain names into IP addresses.

Expt 2 : Study and Analysis of Network Performance using NETSIM

Name: Thurlapati Sai Sree Praneetha **Date**: 30-07-2024

Registration No: 22BLC1350

Aim: To study the performance of a Computer Network, by changing various factors of the network, using NetSim.

Different Factors are

- Increasing the traffic (Increasing the Packet Generation Rate for a fixed number of Applications)
- Increasing the number of users (Increasing the number of nodes / Applications with fixed packet generation rate)
- Increasing the bit error rate across links

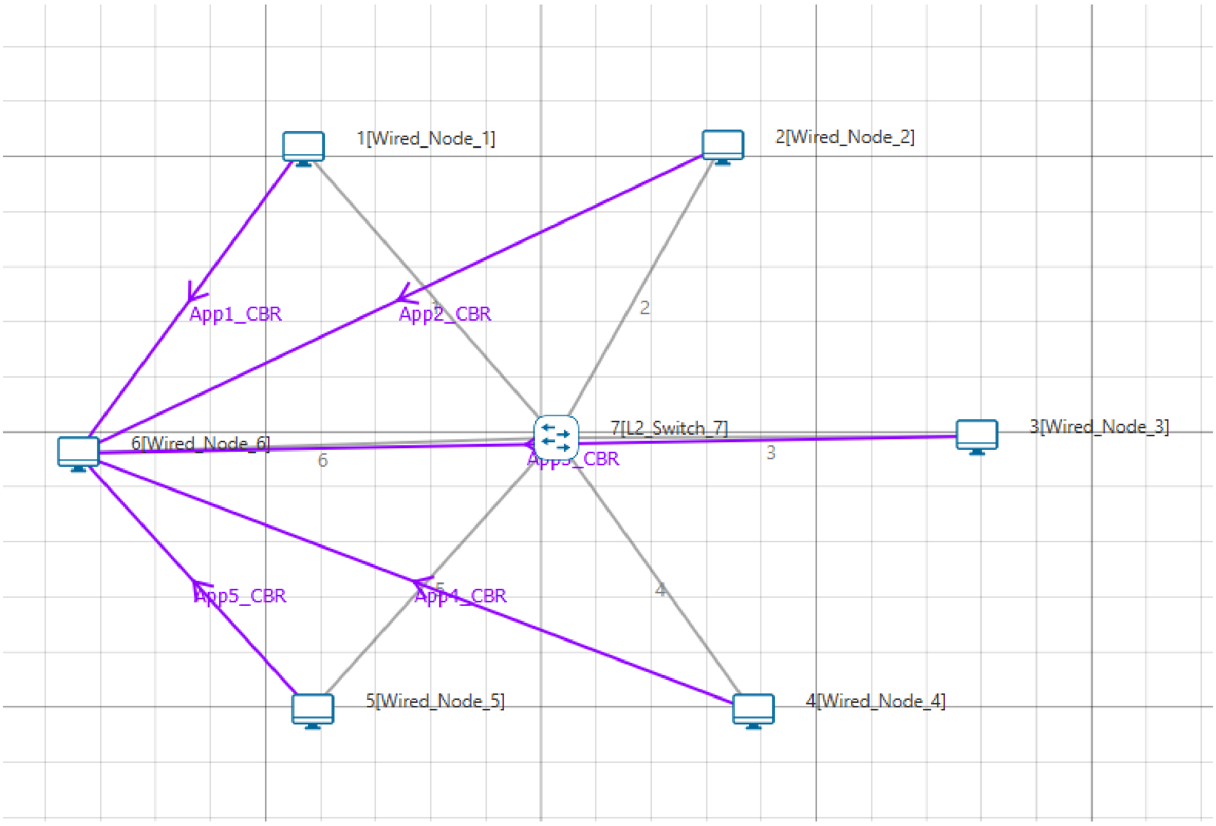
Procedure:

- Click on internetworks and click new simulation
- Place the nodes (pc/laptop) and switch. Click wired/wireless in the top bar and connect the nodes to the switch
- Right click on the link number (ie, the wire) and click on properties. Change the max uplink and max downlink to 5.

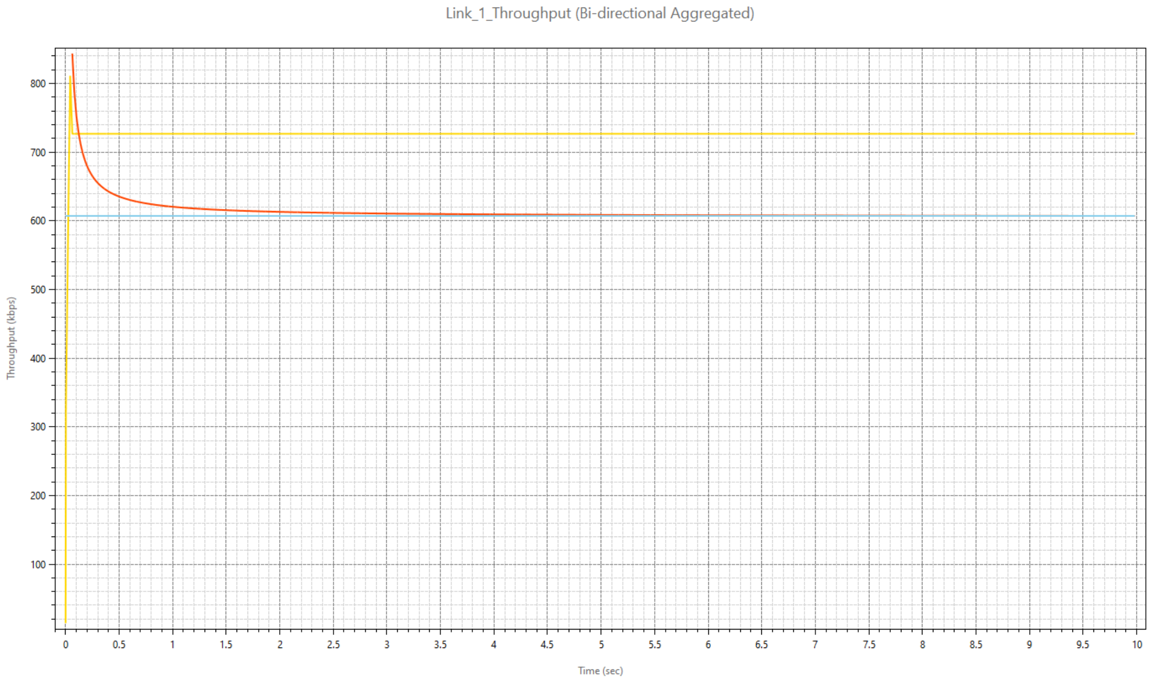
- Right click on the nodes and select properties. Under transport layer, we will see 2 protocols; UDP-faster delivery services where we can tolerate packet losses and TCP-clarity is needed and we cannot tolerate packet losses but is slower.
- Click on application in the top bar. Application method-unicast; application type-CBR (constant bit rate).

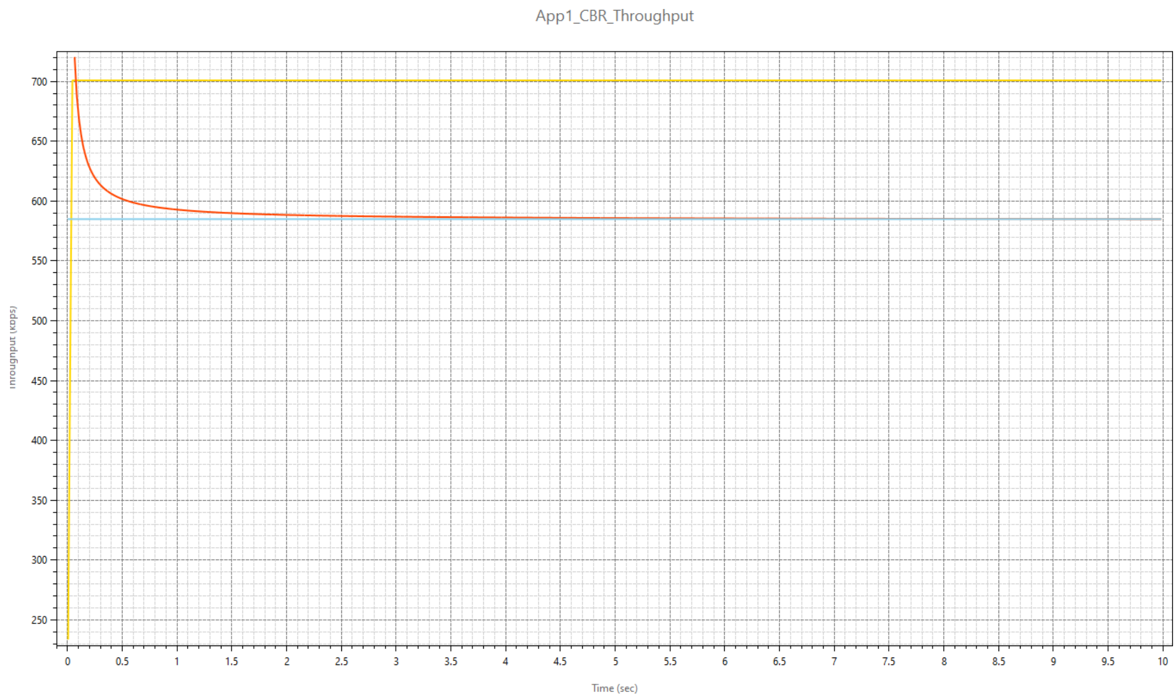
  20msec – 1 Packet
  For 1 msec = 1/20*10^-3 = 50 packets/sec
  Size of packet = 1500(ROUND OFF 1460)
  Total Bits/sec = 50*1500*8=0.6Mbits/sec
- Click on plot in the top bar and select ok. Click on packet trace in the top bar and click ok. Then click on run.
- Static ARP configuration - disable static ARP. Click accept. Go to the first tab (beside ARP configuration and change the time to 10s and click ok. A command prompt will open. Wait for a while. 3 more windows will open. Animation, Metrics and Plots
- Export the results to excel and take out the application metrics from it
- Find the average value of the throughput and delay from each of the cases and form a separate table.
- Now plot the graph of the values and understand how the throughput and delay changes with decrease in time period.
- Take the cases when time period is set to be 20msec(20000musec), 15msec(15000musec), 10msec(10000musec), 5msec(5000musec).

  NETWORK:



PLOTS WINDOW:





RESULTS:

Application_Metrics_Table

Application_Metrics

Application ID	Throughput Plot	Application Name	Packets Generated	Packets Received	Throughput (Mbps)	Delay (microsecond)
1	Application_Throughput_plot	App1_CBR	500	500	0.584000	4856.519200
2	Application_Throughput_plot	App2_CBR	500	499	0.582832	7279.882645
3	Application_Throughput_plot	App3_CBR	500	497	0.580496	9698.373602
4	Application_Throughput_plot	App4_CBR	500	498	0.581664	12112.007550
5	Application_Throughput_plot	App5_CBR	500	498	0.581664	14530.501365

TCP_Metrics_Table

TCP_Metrics

Source	Destination	Segment Sent	Segment Received	Ack Sent	Ack Received	Duplicate ack received
WIRED_NODE_1	ANY_DEVICE	0	0	0	0	0
WIRED_NODE_2	ANY_DEVICE	0	0	0	0	0
WIRED_NODE_3	ANY_DEVICE	0	0	0	0	0
WIRED_NODE_4	ANY_DEVICE	0	0	0	0	0
WIRED_NODE_5	ANY_DEVICE	0	0	0	0	0
WIRED_NODE_6	ANY_DEVICE	0	0	0	0	0

Link_Metrics_Table

Link_Metrics

Link ID	Link Throughput Plot	Packets Transmit...		Packets Errored		Packets Collided	
		Data	Control	Data	Control	Data	Control
All	NA	4996	40	8	0	0	0
1	Link_throughput	500	6	0	0	0	0
2	Link_throughput	500	6	1	0	0	0
3	Link_throughput	500	6	2	0	0	0
4	Link_throughput	500	6	1	0	0	0
5	Link_throughput	500	6	0	0	0	0
6	Link_throughput	2496	10	4	0	0	0

Queue_Metrics_Table

Queue_Metrics

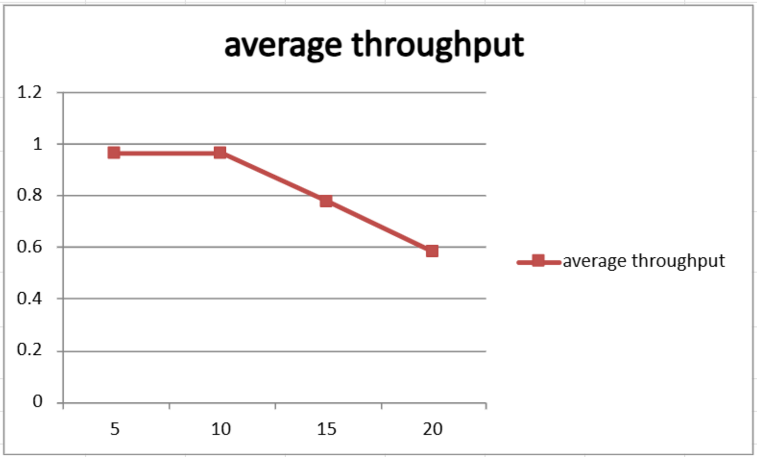
Device_id	Port_id	Queued_pa...	Dequeued_...	Dropped_p...

No content in table

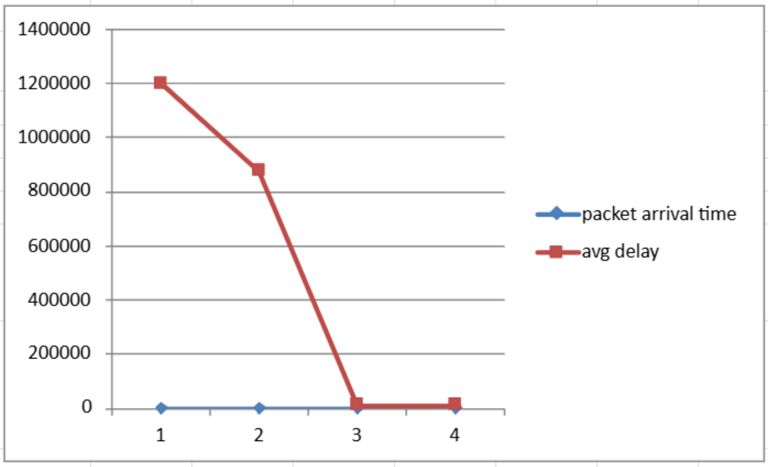
Network Simulation Specifications:

1. No of Nodes = 6; No of Destination Node = 1 (Node 6); No of Applications (CBR) = 3 (3 nodes Application Traffic); Vary the Packet Inter-arrival Time: 20ms, 15ms, 10ms, 5ms
   Link Speed = Switch to Destination = 5Mbps (Duplex);
   Only UDP sources; Dynamic ARP enabled; Simulation time: 10seconds
   - ✓ **Inference (Plots):** Calculate the Average Application Throughput & Average Delay vs Packet Inter-arrival Time

packet arrival time	average throughput
5	**0.9628992**
10	**0.9621984**
15	**0.7762528**
20	**0.5821312**

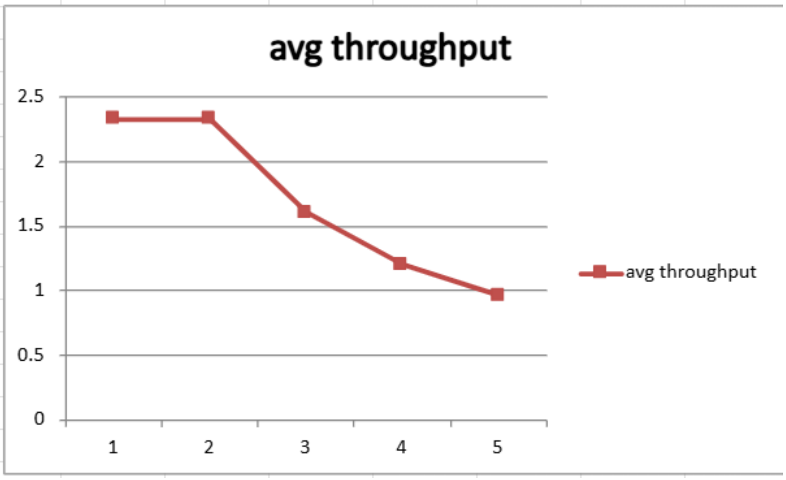


packet arrival time	avg delay
5	**1199587.108**
10	**875034.0366**
15	**9696.240762**
20	**9695.456872**

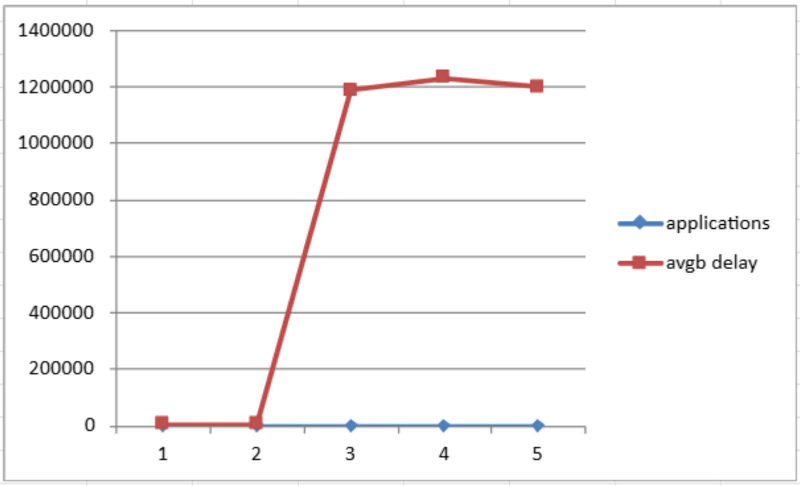


2. No of Nodes = 6 ; Packet Inter-arrival Time = 5ms ; Increase the number of Application Traffics (nodes in this case) = 1, 2, 3, 4, 5 ; No of Destination Node = 1
   Link Speed = Switch to Destination = 5Mbps (Duplex)
   - ✓ **Inference (Plots):** Calculate the Average Application Throughput & Average Delay vs No of Application Traffic/Nodes

applications	avg throughput
1	2.328992
2	2.328922
3	1.604053
4	1.203332
5	0.962899



applications	avgb delay
1	4855.093
2	6064.759
3	1187996
4	1231890
5	1199587



TYPES OF DELAY:

1. Transmission Delay: The time taken to transfer 1460 bits from packet to the wire.

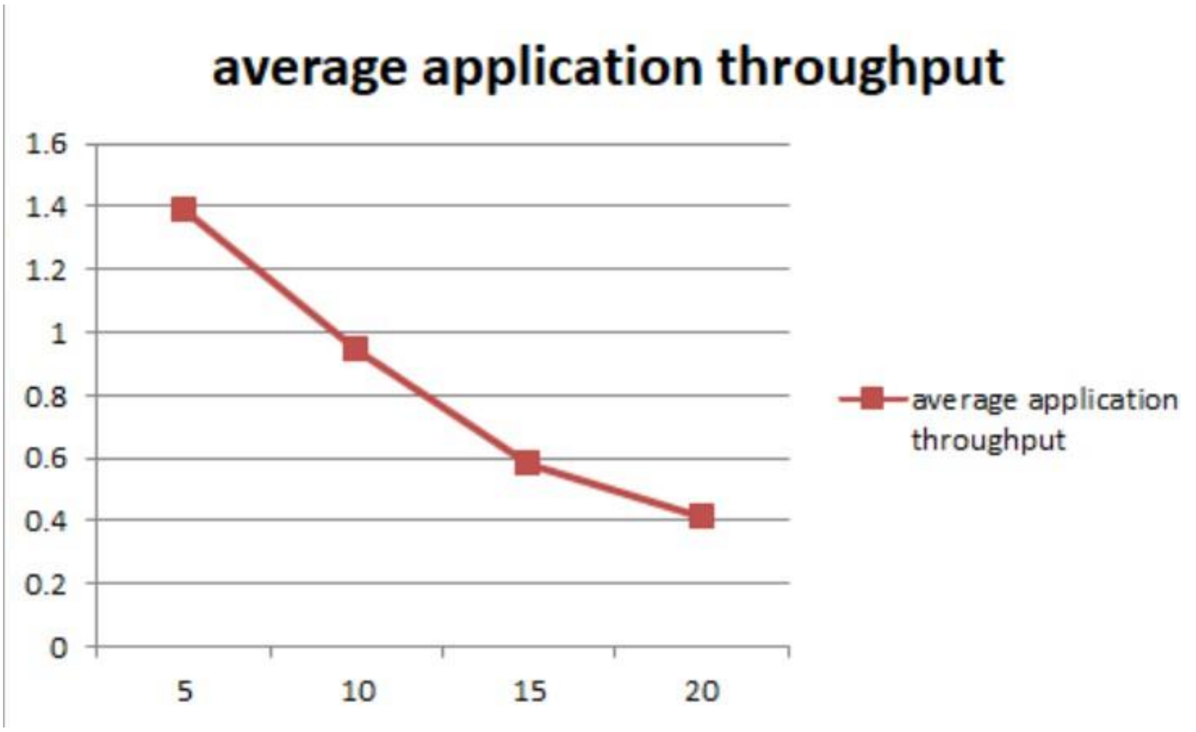
2. Propagation Delay: Delay time taken by the bits to travel through the wire and reach the destination.

3. Processing Delay: The time taken for a router or a switch to process a packet's header.

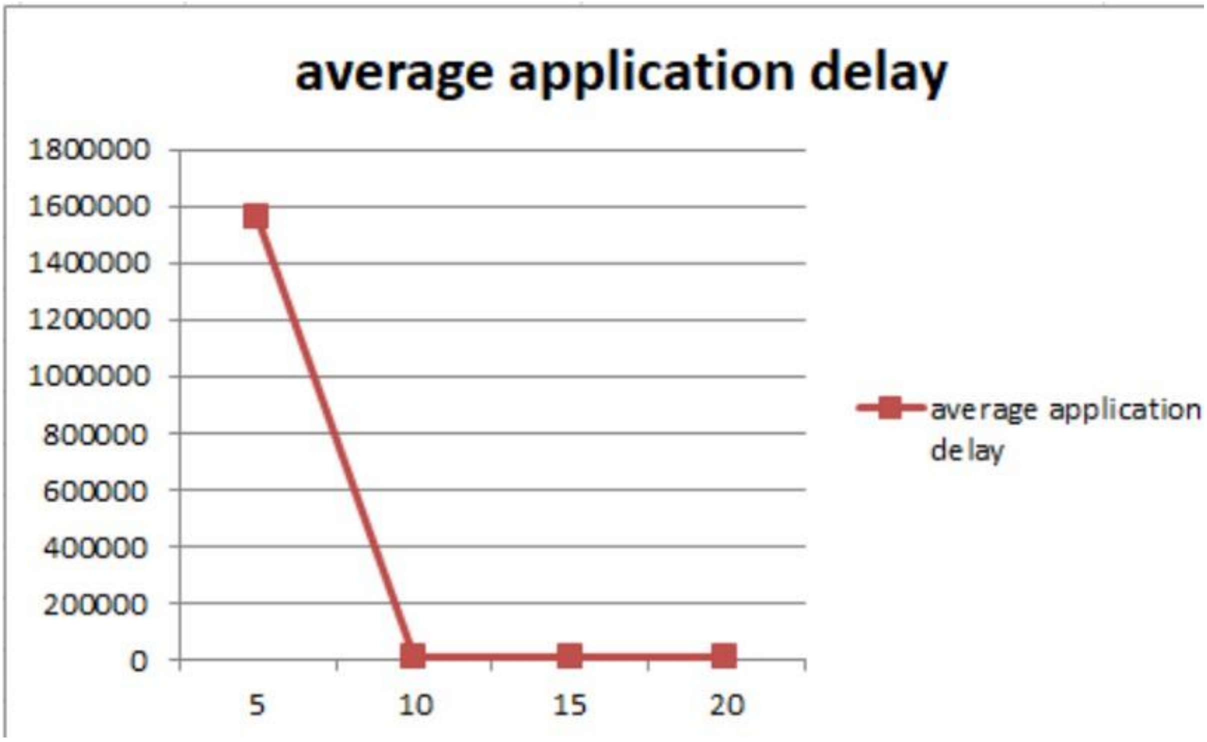
4. Queuing Delay: It occurs when a packet arrives at a router or switch and must wait in a queue before it can be transmitted to the next hop.

3. No of Nodes = 6; No of Destination Node = 1; No of Applications (CBR) = 3 (3 nodes Application Traffic); Packet Inter-arrival Time: 10ms; Vary the BER: 10^-8, 10^-6, 10^-5 across active links
   Link Speed = Switch to Destination = 5Mbps (Duplex)
   - ✓ **Inference (Plots):** Calculate the Average Application Throughput & Average Delay vs BER

packet interarrival time	average application throughput
5	1.39
10	0.94
15	0.58
20	0.41



packet interarrival time	average application delay
5	1558058.412
10	7483.731483
15	8921.1935
20	8921.194



CONCLUSION: Therefore, with the above we were able to create our own network, create traffic and find the average delays and plot the delays to come to a generalization of changes in delay in various cases.

Expt 2 : Study and Analysis of Network Performance using NETSIM

Name: Thurlapati Sai Sree Praneetha **Date**: 30-07-2024

Registration No: 22BLC1350

Aim: To study the performance of a Computer Network, by changing various factors of the network, using NetSim.

Different Factors are

- Increasing the traffic (Increasing the Packet Generation Rate for a fixed number of Applications)
- Increasing the number of users (Increasing the number of nodes / Applications with fixed packet generation rate)
- Increasing the bit error rate across links

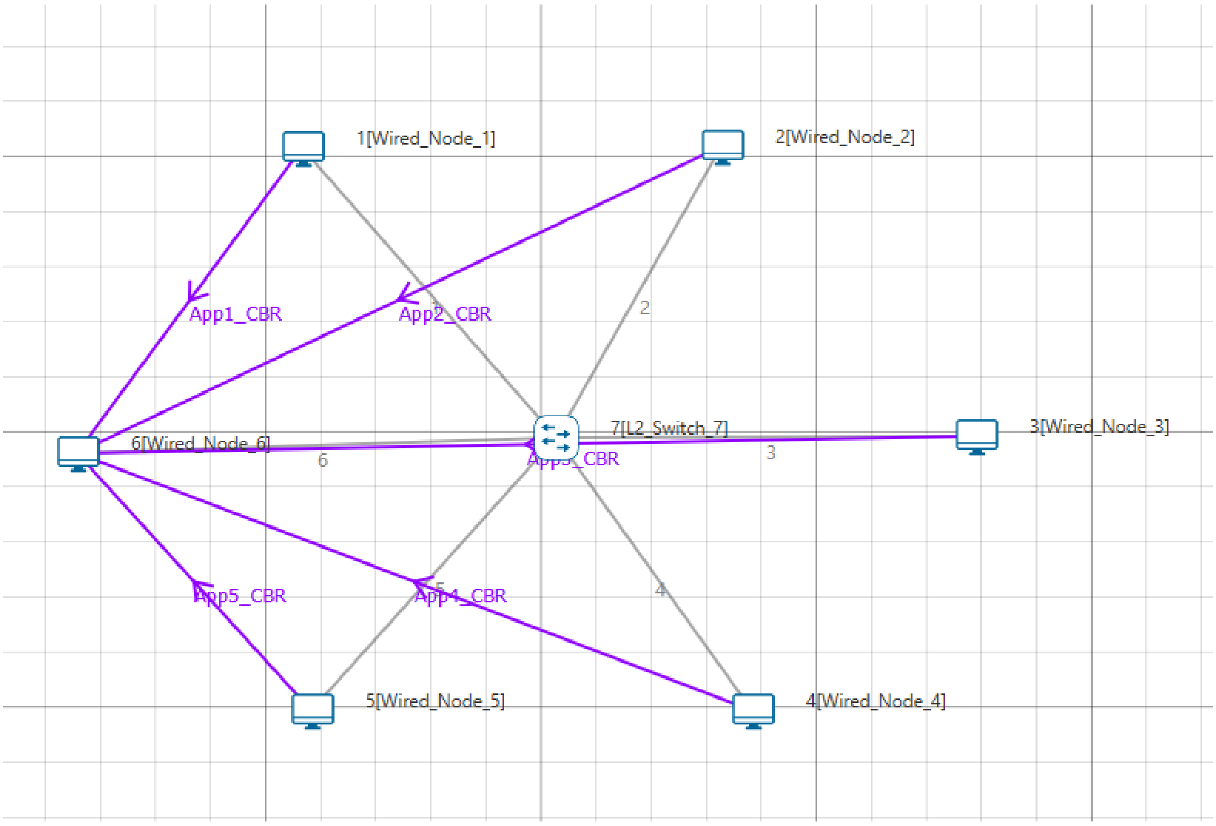
Procedure:

- Click on internetworks and click new simulation
- Place the nodes (pc/laptop) and switch. Click wired/wireless in the top bar and connect the nodes to the switch
- Right click on the link number (ie, the wire) and click on properties. Change the max uplink and max downlink to 5.

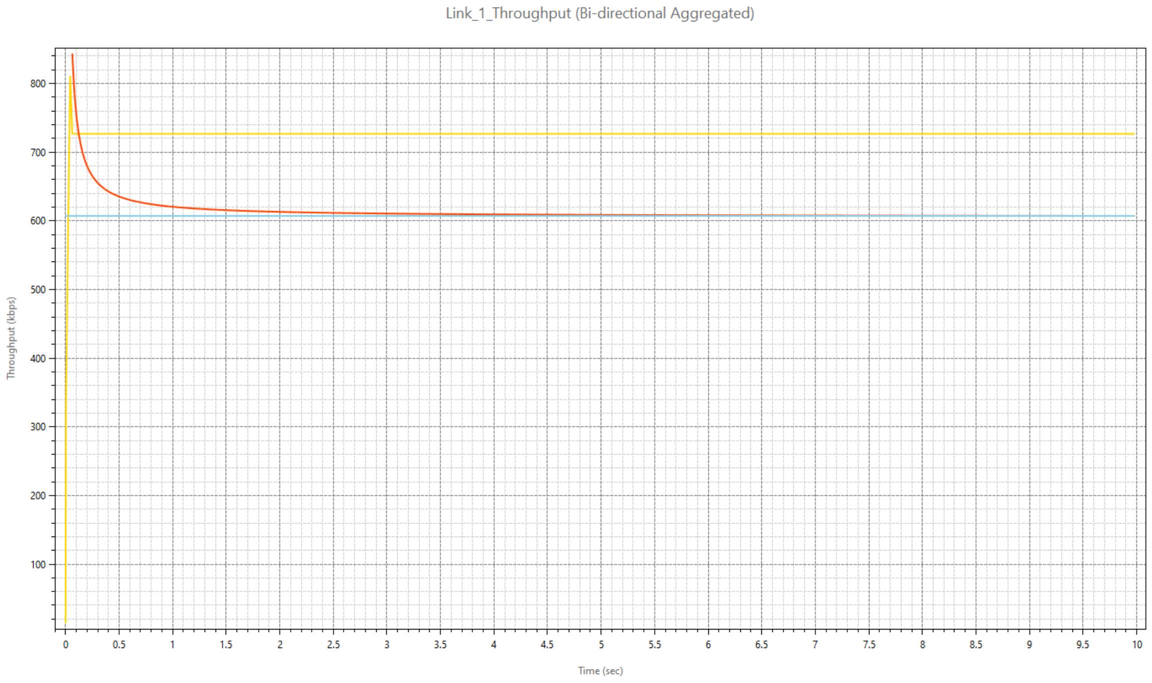
- Right click on the nodes and select properties. Under transport layer, we will see 2 protocols; UDP-faster delivery services where we can tolerate packet losses and TCP-clarity is needed and we cannot tolerate packet losses but is slower.
- Click on application in the top bar. Application method-unicast; application type-CBR (constant bit rate).

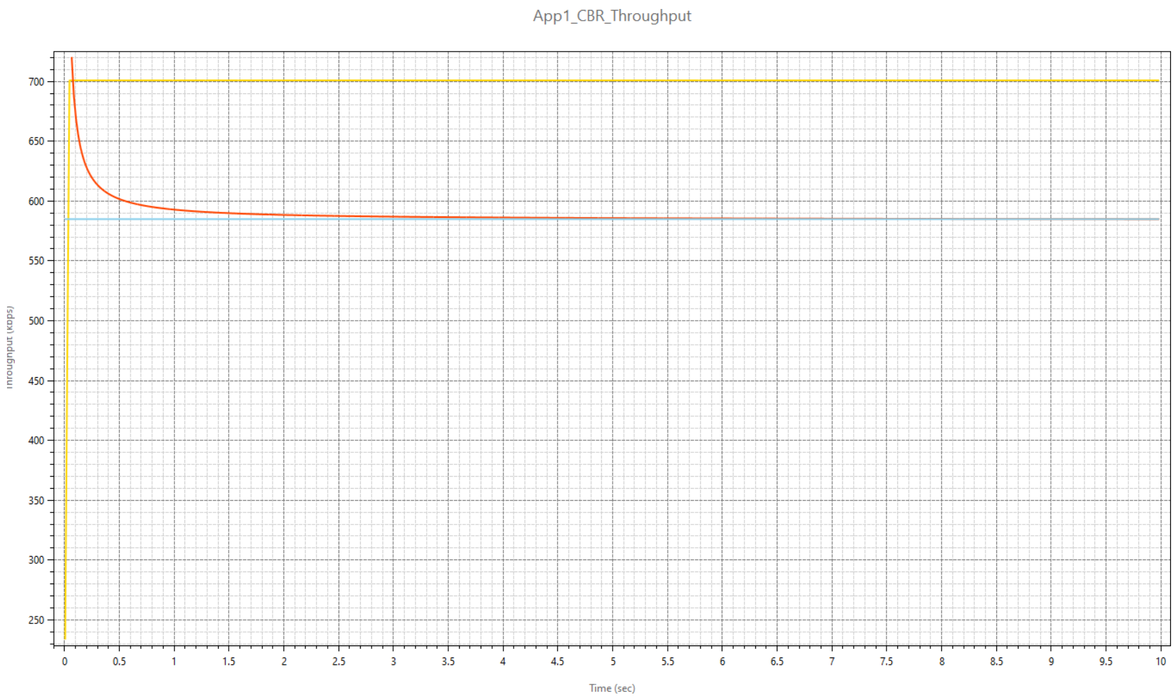
  20msec – 1 Packet
  For 1 msec = 1/20*10^-3 = 50 packets/sec
  Size of packet = 1500(ROUND OFF 1460)
  Total Bits/sec = 50*1500*8=0.6Mbits/sec
- Click on plot in the top bar and select ok. Click on packet trace in the top bar and click ok. Then click on run.
- Static ARP configuration - disable static ARP. Click accept. Go to the first tab (beside ARP configuration and change the time to 10s and click ok. A command prompt will open. Wait for a while. 3 more windows will open. Animation, Metrics and Plots
- Export the results to excel and take out the application metrics from it
- Find the average value of the throughput and delay from each of the cases and form a separate table.
- Now plot the graph of the values and understand how the throughput and delay changes with decrease in time period.
- Take the cases when time period is set to be 20msec(20000musec), 15msec(15000musec), 10msec(10000musec), 5msec(5000musec).

  NETWORK:

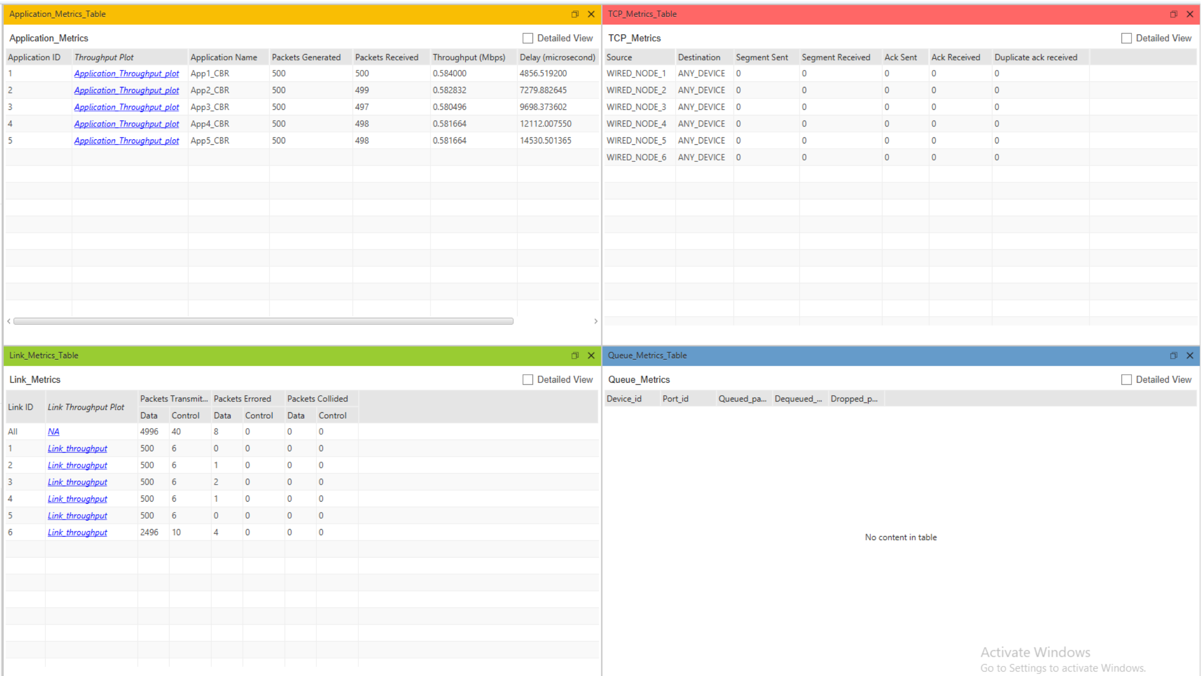


PLOTS WINDOW:





RESULTS:



Application_Metrics_Table

Application_Metrics

Application ID	Throughput Plot	Application Name	Packets Generated	Packets Received	Throughput (Mbps)	Delay (microsecond)
1	Application_Throughput_plot	App1_CBR	500	500	0.584000	4856.519200
2	Application_Throughput_plot	App2_CBR	500	499	0.582832	7279.882645
3	Application_Throughput_plot	App3_CBR	500	497	0.580496	9698.373602
4	Application_Throughput_plot	App4_CBR	500	498	0.581664	12112.007550
5	Application_Throughput_plot	App5_CBR	500	498	0.581664	14530.501365

TCP_Metrics_Table

TCP_Metrics

Source	Destination	Segment Sent	Segment Received	Ack Sent	Ack Received	Duplicate ack received
WIRED_NODE_1	ANY_DEVICE	0	0	0	0	0
WIRED_NODE_2	ANY_DEVICE	0	0	0	0	0
WIRED_NODE_3	ANY_DEVICE	0	0	0	0	0
WIRED_NODE_4	ANY_DEVICE	0	0	0	0	0
WIRED_NODE_5	ANY_DEVICE	0	0	0	0	0
WIRED_NODE_6	ANY_DEVICE	0	0	0	0	0

Link_Metrics_Table

Link_Metrics

Link ID	Link Throughput Plot	Packets Transmit...		Packets Errored		Packets Collided	
		Data	Control	Data	Control	Data	Control
All	NA	4996	40	8	0	0	0
1	Link_throughput	500	6	0	0	0	0
2	Link_throughput	500	6	1	0	0	0
3	Link_throughput	500	6	2	0	0	0
4	Link_throughput	500	6	1	0	0	0
5	Link_throughput	500	6	0	0	0	0
6	Link_throughput	2496	10	4	0	0	0

Queue_Metrics_Table

Queue_Metrics

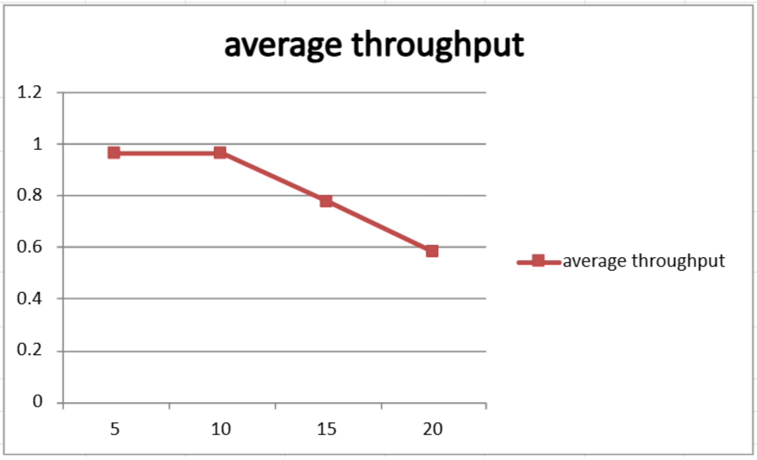
Device_id	Port_id	Queued_pa...	Dequeued_...	Dropped_p...

No content in table

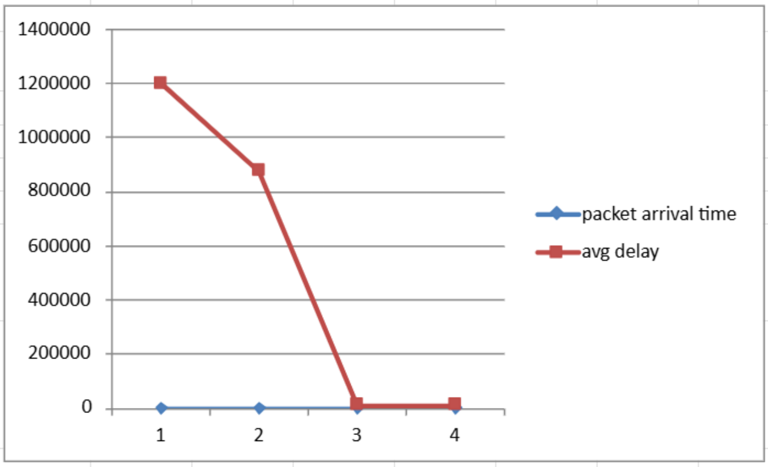
Network Simulation Specifications:

1. No of Nodes = 6; No of Destination Node = 1 (Node 6); No of Applications (CBR) = 3 (3 nodes Application Traffic); Vary the Packet Inter-arrival Time: 20ms, 15ms, 10ms, 5ms
   Link Speed = Switch to Destination = 5Mbps (Duplex);
   Only UDP sources; Dynamic ARP enabled; Simulation time: 10seconds
   - ✓ **Inference (Plots):** Calculate the Average Application Throughput & Average Delay vs Packet Inter-arrival Time

packet arrival time	average throughput
5	**0.9628992**
10	**0.9621984**
15	**0.7762528**
20	**0.5821312**

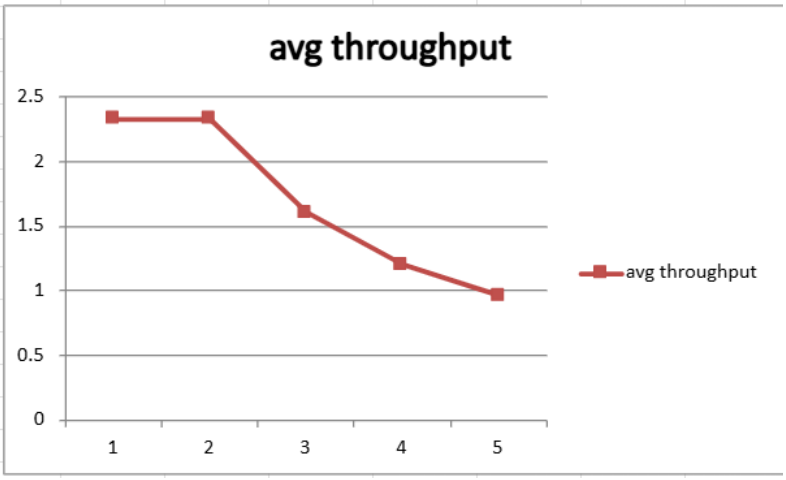


packet arrival time	avg delay
5	**1199587.108**
10	**875034.0366**
15	**9696.240762**
20	**9695.456872**

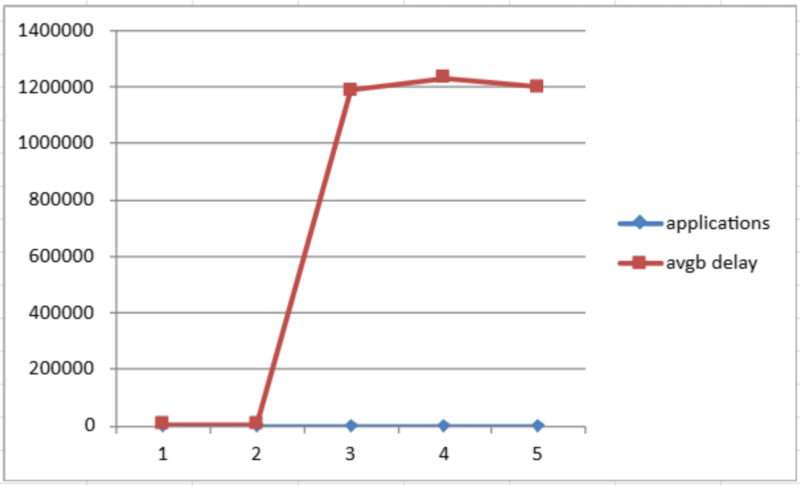


2. No of Nodes = 6 ; Packet Inter-arrival Time = 5ms ; Increase the number of Application Traffics (nodes in this case) = 1, 2, 3, 4, 5 ; No of Destination Node = 1
   Link Speed = Switch to Destination = 5Mbps (Duplex)
   - ✓ **Inference (Plots):** Calculate the Average Application Throughput & Average Delay vs No of Application Traffic/Nodes

applications	avg throughput
1	2.328992
2	2.328922
3	1.604053
4	1.203332
5	0.962899



applications	avgb delay
1	4855.093
2	6064.759
3	1187996
4	1231890
5	1199587



TYPES OF DELAY:

1. Transmission Delay: The time taken to transfer 1460 bits from packet to the wire.

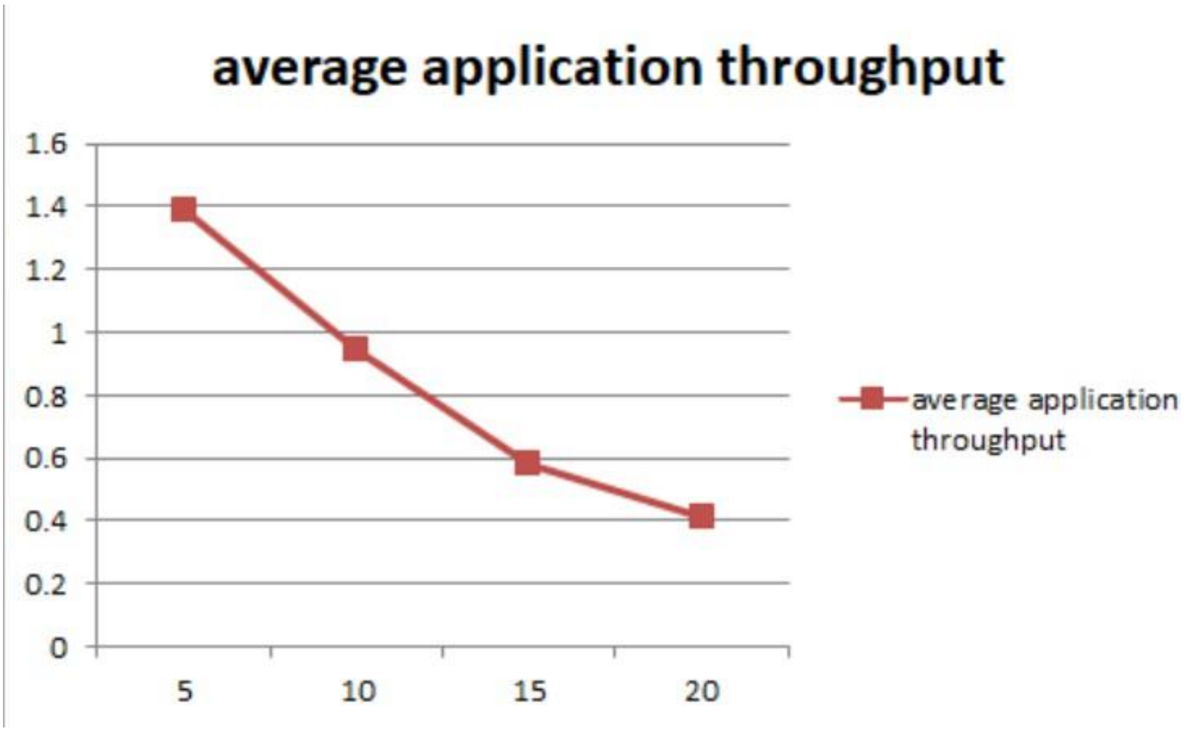
2. Propagation Delay: Delay time taken by the bits to travel through the wire and reach the destination.

3. Processing Delay: The time taken for a router or a switch to process a packet's header.

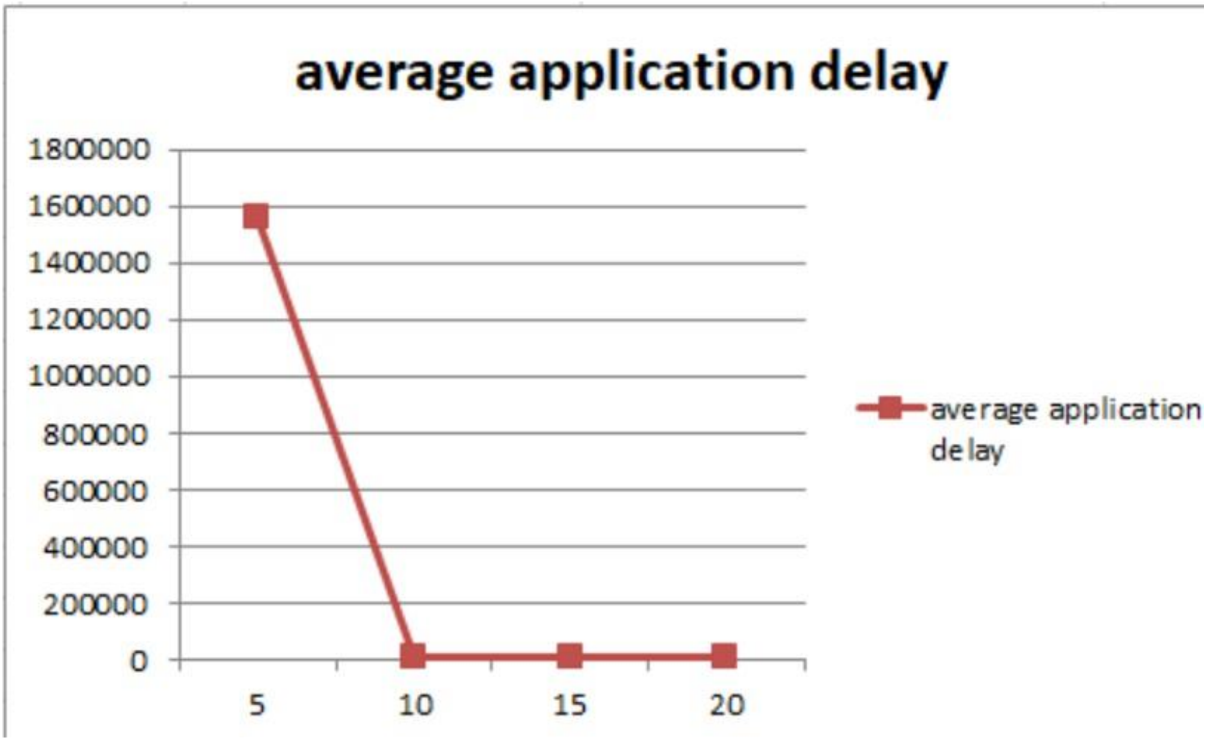
4. Queuing Delay: It occurs when a packet arrives at a router or switch and must wait in a queue before it can be transmitted to the next hop.

3. No of Nodes = 6; No of Destination Node = 1; No of Applications (CBR) = 3 (3 nodes Application Traffic); Packet Inter-arrival Time: 10ms; Vary the BER: 10^-8, 10^-6, 10^-5 across active links
   Link Speed = Switch to Destination = 5Mbps (Duplex)
   - ✓ **Inference (Plots):** Calculate the Average Application Throughput & Average Delay vs BER

packet interarrival time	average application throughput
5	1.39
10	0.94
15	0.58
20	0.41



packet interarrival time	average application delay
5	1558058.412
10	7483.731483
15	8921.1935
20	8921.194



CONCLUSION: Therefore, with the above we were able to create our own network, create traffic and find the average delays and plot the delays to come to a generalization of changes in delay in various cases.

Experiment 3 : Simulate and Analyze the impact of link failure and understand it's impact onnetwork performance

Name : Thurlapati Sai Sree Praneetha
Reg. No : 22BLC1350

Faculty Name : Jaya Vignesh T

AIM:

Task : Simulate and Analyze the impact of link failure and understand it's impact on networkperformance

SOFTWARE USED:

Netsim v13.1 Academic version or higher.

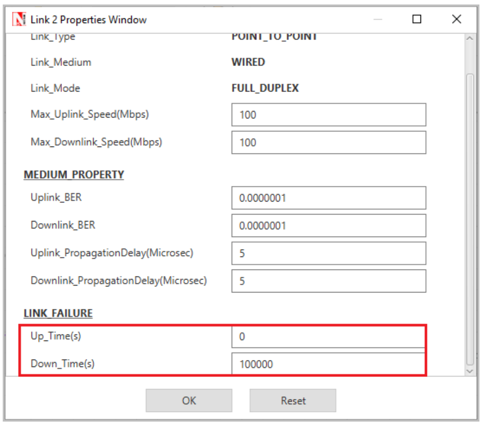
Objective

To model link failure, understand its impact on

network performance. Theory

A link failure can occur due to a) faults in the physical link and b) failure of the connected port. When a link fails, packets cannot be transported. This also means that established routes to destinations maybecome unavailable. In such cases, the routing protocol must recompute an alternate path around the failure.

In NetSim, only WAN links (connecting two routers) can be failed. Right click on a WAN link betweentwo routers and the Link Properties Window is as



shown below.

Figure 1: Wired Link Properties Window

Link Up Time refers to the time(s) at which the link is functional and Link Down Time refers to the time

(s) at which a link fails. Click on Up_Time or Down_Time to understand the configuration options.

NOTE: Link failure can be set only for "WAN Interfaces".

Network Setup

a) Link Failure Single WAN Interface

Design the configuration corresponding to this experiment as shown in below figure

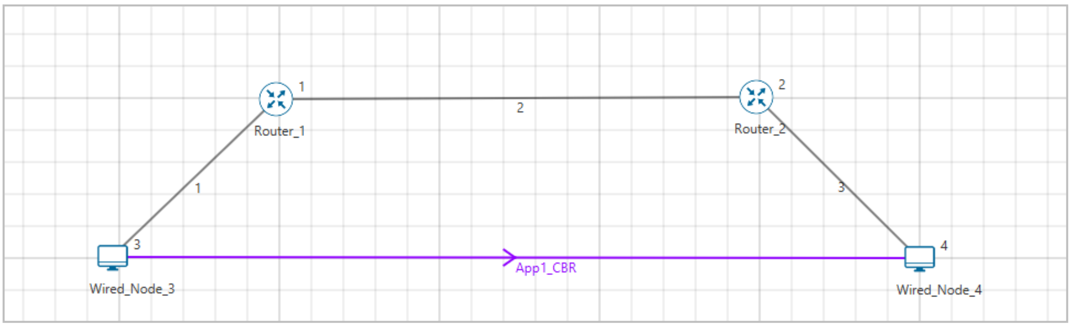


Figure **Error! No text of specified style in document.**: Network set up for studying the Link Failure SingleWAN Interface

Procedure

The following set of procedures were done to generate this sample:

Step 1: In the "Internetworks" library, and a network scenario is designed in NetSim comprising of 2Wired Nodes and 2 Routers.

Step 2: By default, Link Failure Up Time is set to 0,10,20 and Down Time is set to 5,15. This means thelink is up 0-5s, 10-15s and 20s onwards, and it is down 5-10s and 15-20s.

Step 3: Packet Trace is enabled in NetSim GUI. At the end of the simulation, a .csv file containing all thepacket information is available for performing packet level analysis.

Step 4: Right click on the Application Flow App1 CBR and select Properties or click on the Application icon present in the top ribbon/toolbar.

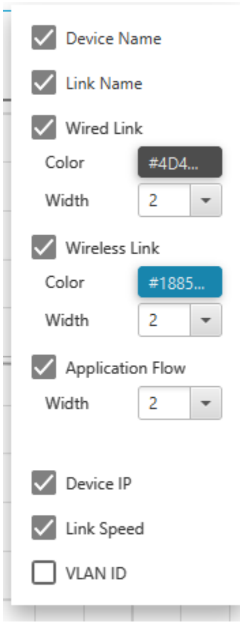
A CBR Application is generated from Wired Node 3 i.e., Source to Wired Node 4 i.e., Destination withPacket Size remaining 1460 Bytes and Inter Arrival Time remaining 20000µs.

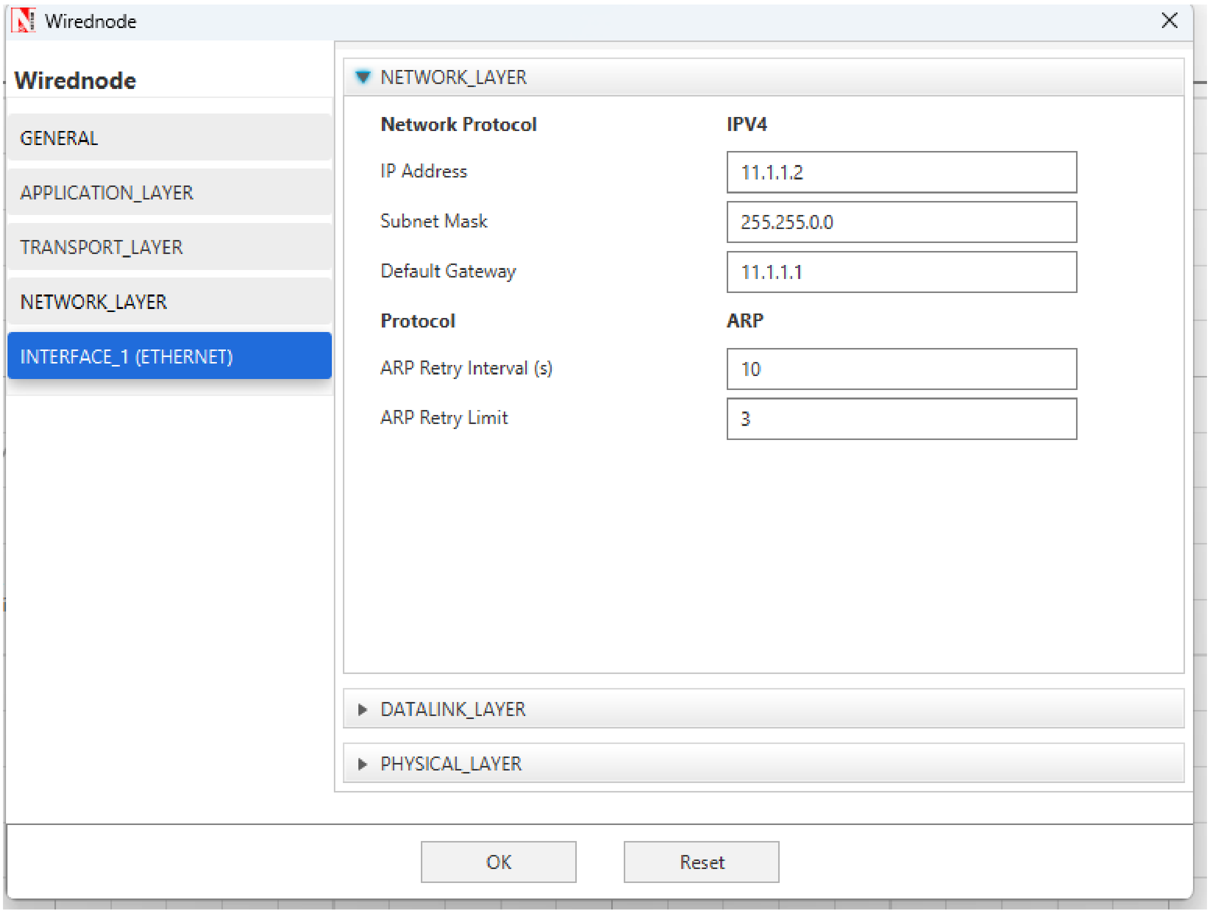
Step 5: Transport protocol set as UDP.

Step 6: Enable the plots and run the simulation

for 50 Seconds.Output

Go to NetSim Simulation Result Window and open the Application Throughput plot.



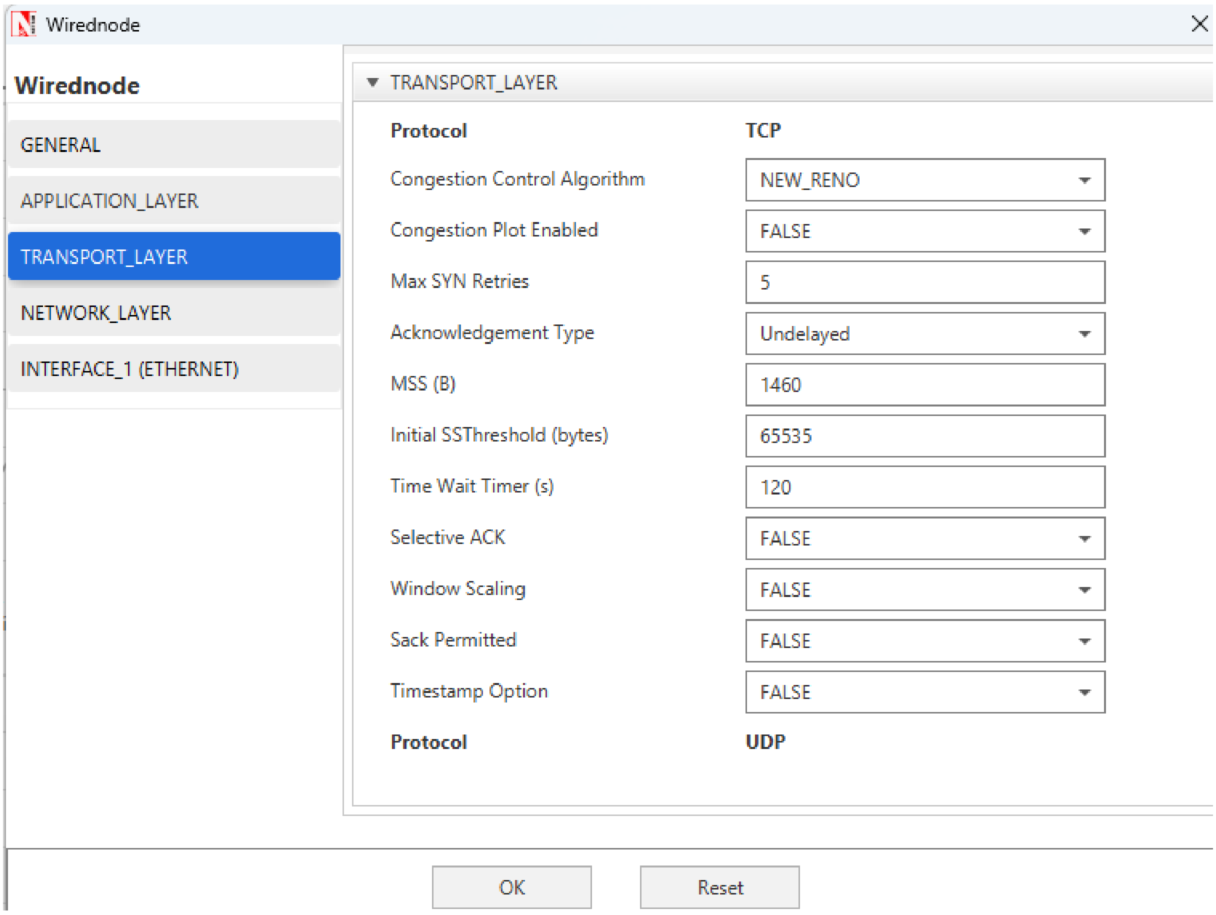


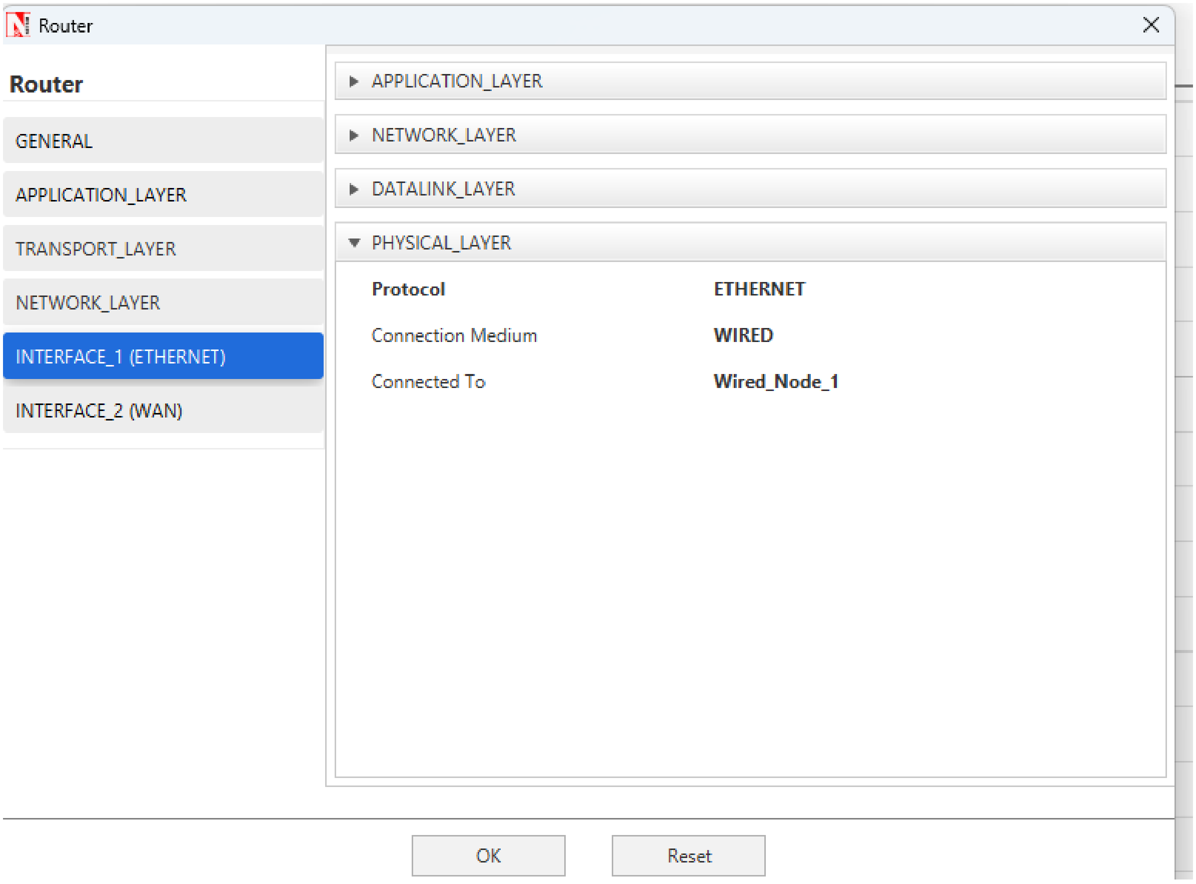
Arp request is a broadcast packet

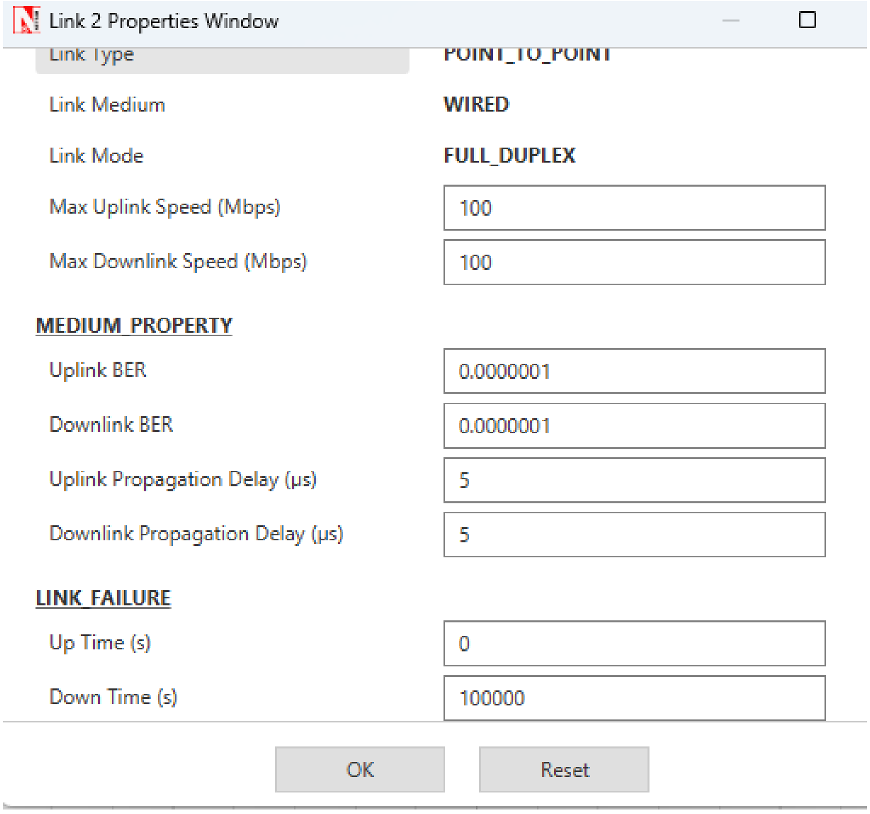
Arp retry limit – can retry for the packet

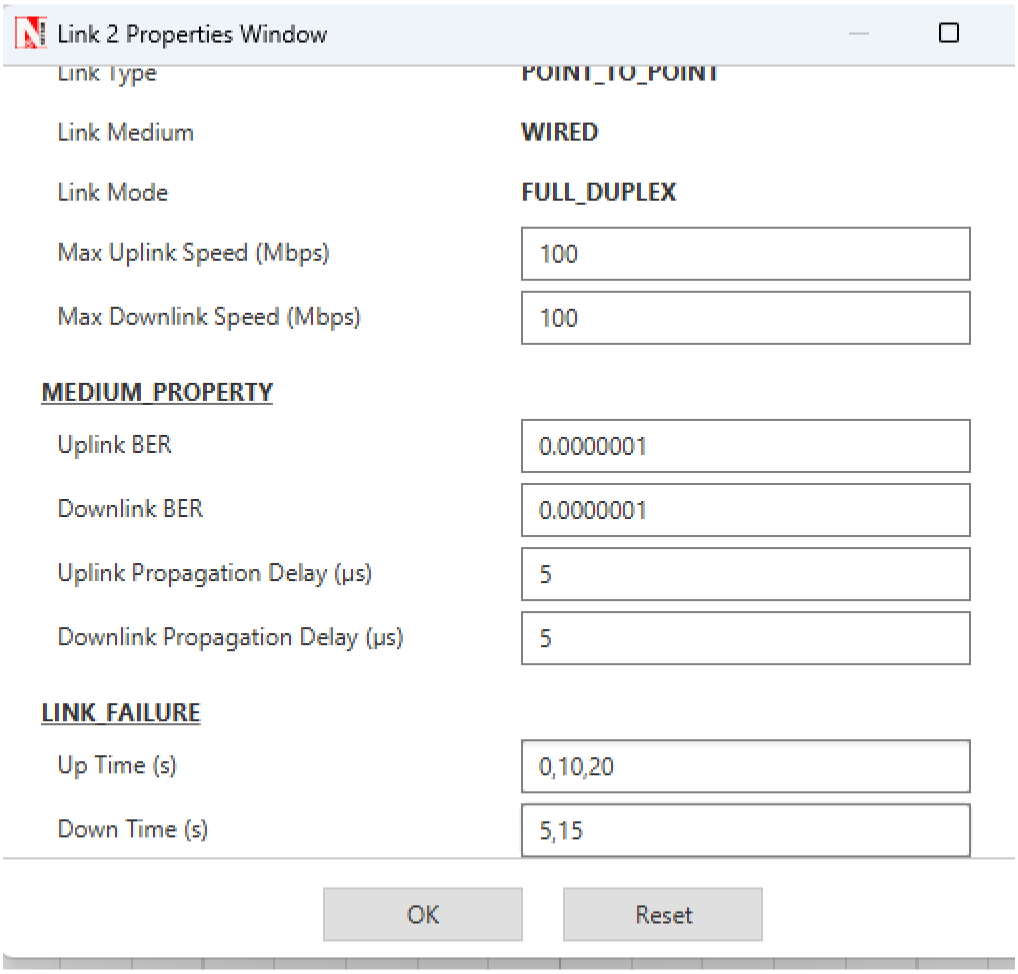
sending within a limitArp retry interval – can

retry only after the given time interval

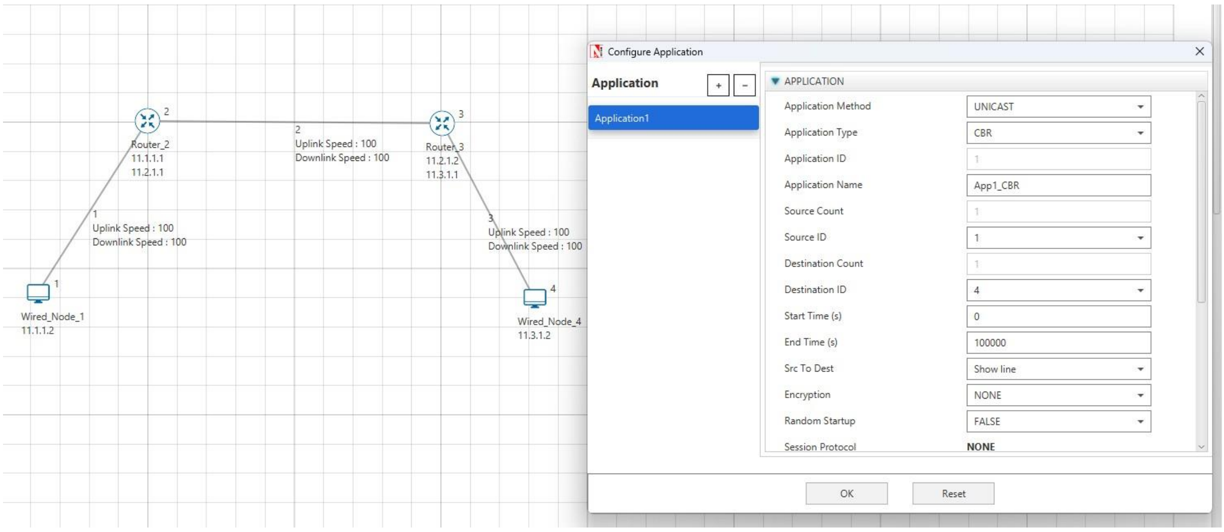

Wirednode
Wirednode
GENERAL
APPLICATION_LAYER
TRANSPORT_LAYER
NETWORK_LAYER
INTERFACE_1 (ETHERNET)
TRANSPORT_LAYER
Protocol
TCP
Congestion Control Algorithm
NEW_RENO
Congestion Plot Enabled
FALSE
Max SYN Retries
5
Acknowledgement Type
Undelayed
MSS (B)
1460
Initial SSThreshold (bytes)
65535
Time Wait Timer (s)
120
Selective ACK
FALSE
Window Scaling
FALSE
Sack Permitted
FALSE
Timestamp Option
FALSE
Protocol
UDP
OK
Reset

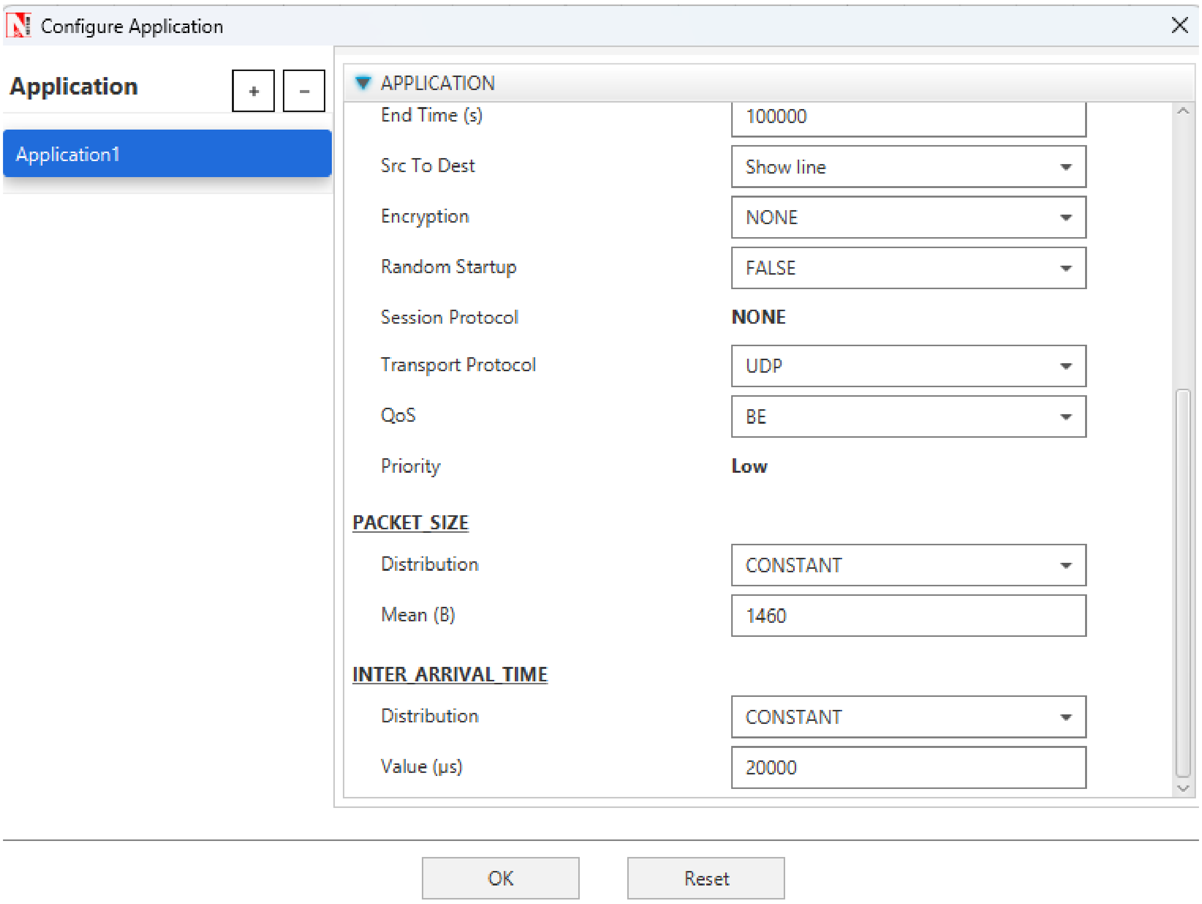

Router
Router
GENERAL
APPLICATION_LAYER
TRANSPORT_LAYER
NETWORK_LAYER
INTERFACE_1 (ETHERNET)
INTERFACE_2 (WAN)
APPLICATION_LAYER
NETWORK_LAYER
DATALINK_LAYER
PHYSICAL_LAYER
Protocol
ETHERNET
Connection Medium
WIRED
Connected To
Wired_Node_1
OK
Reset

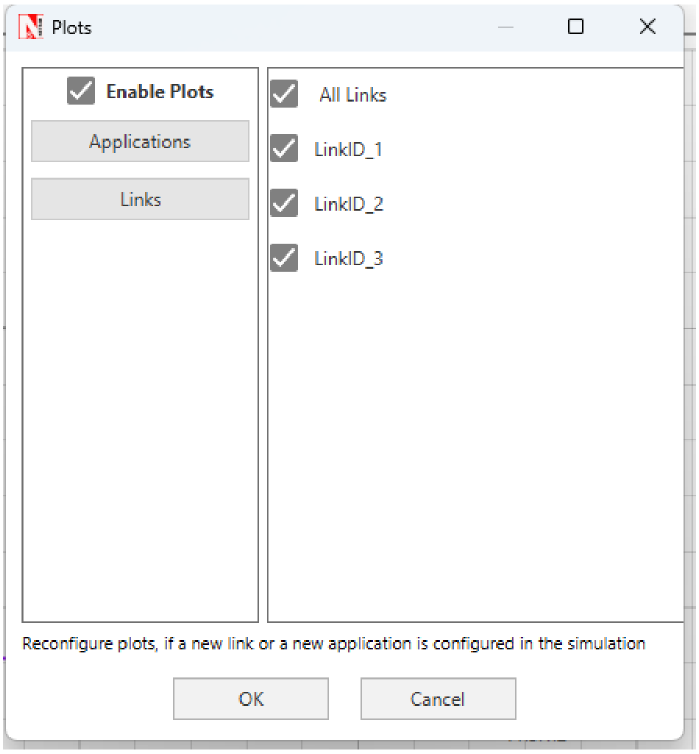

Link 2 Properties Window
Link Type
POINT_TO_POINT
Link Medium
WIRED
Link Mode
FULL_DUPLEX
Max Uplink Speed (Mbps)
100
Max Downlink Speed (Mbps)
100
MEDIUM_PROPERTY
Uplink BER
0.0000001
Downlink BER
0.0000001
Uplink Propagation Delay (µs)
5
Downlink Propagation Delay (µs)
5
LINK_FAILURE
Up Time (s)
0
Down Time (s)
100000
OK
Reset

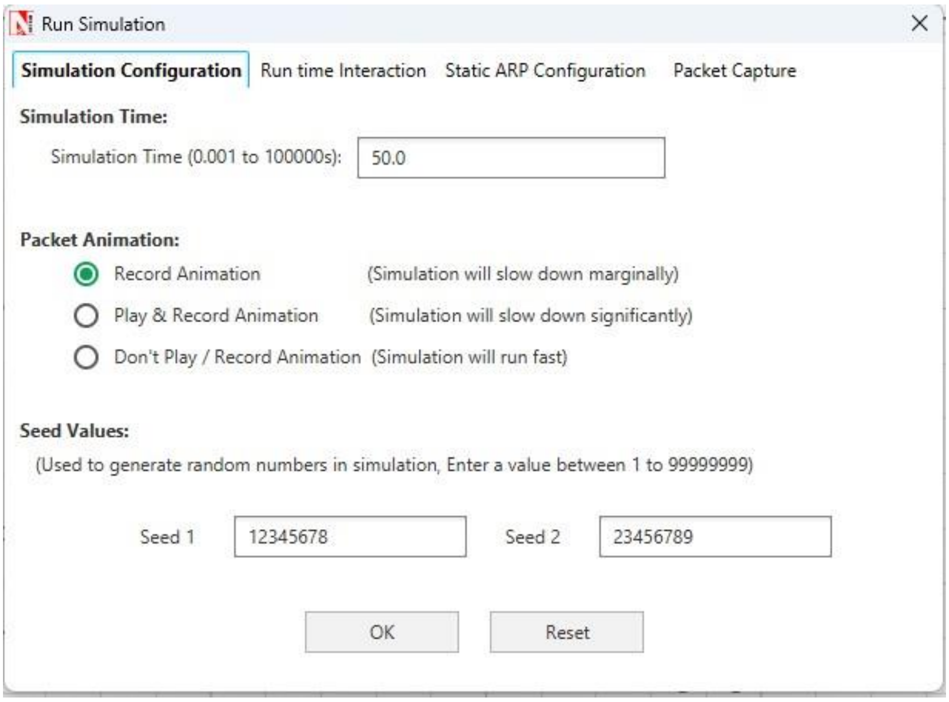

Link 2 Properties Window
Link Type
POINT_TO_POINT
Link Medium
WIRED
Link Mode
FULL_DUPLEX
Max Uplink Speed (Mbps)
100
Max Downlink Speed (Mbps)
100
MEDIUM_PROPERTY
Uplink BER
0.0000001
Downlink BER
0.0000001
Uplink Propagation Delay (µs)
5
Downlink Propagation Delay (µs)
5
LINK_FAILURE
Up Time (s)
0,10,20
Down Time (s)
5,15
OK
Reset

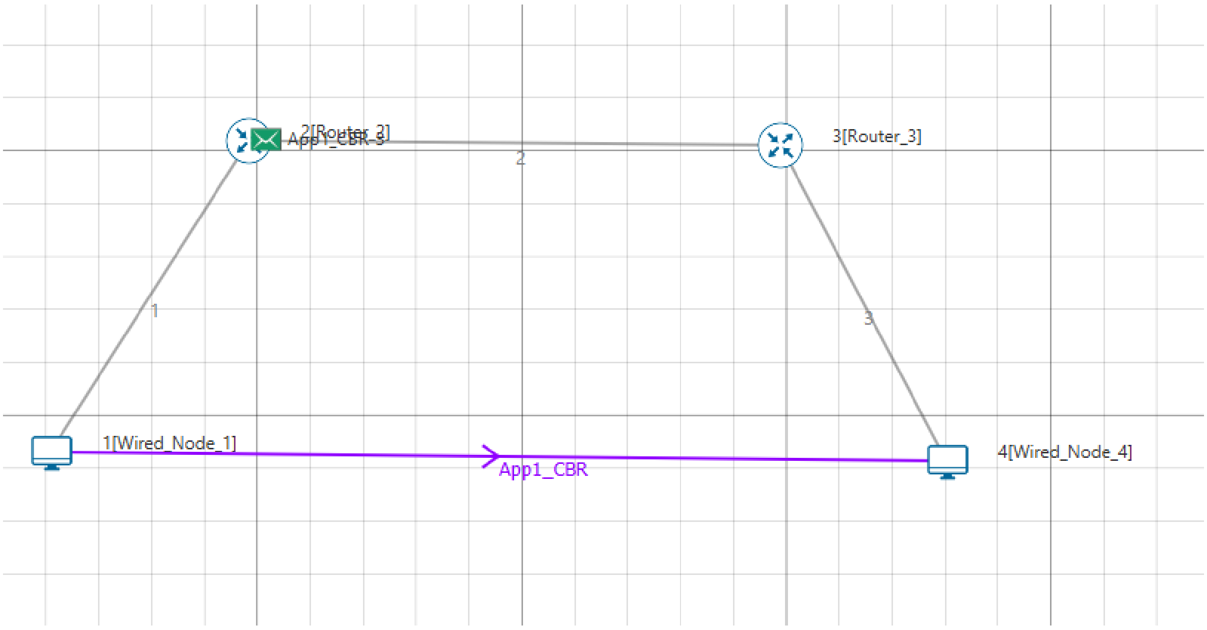
It means that from 0-5 sec the link is up, 5-10 sec the link is down, 10-15 sec the link is up and 15-20 secthe link is down



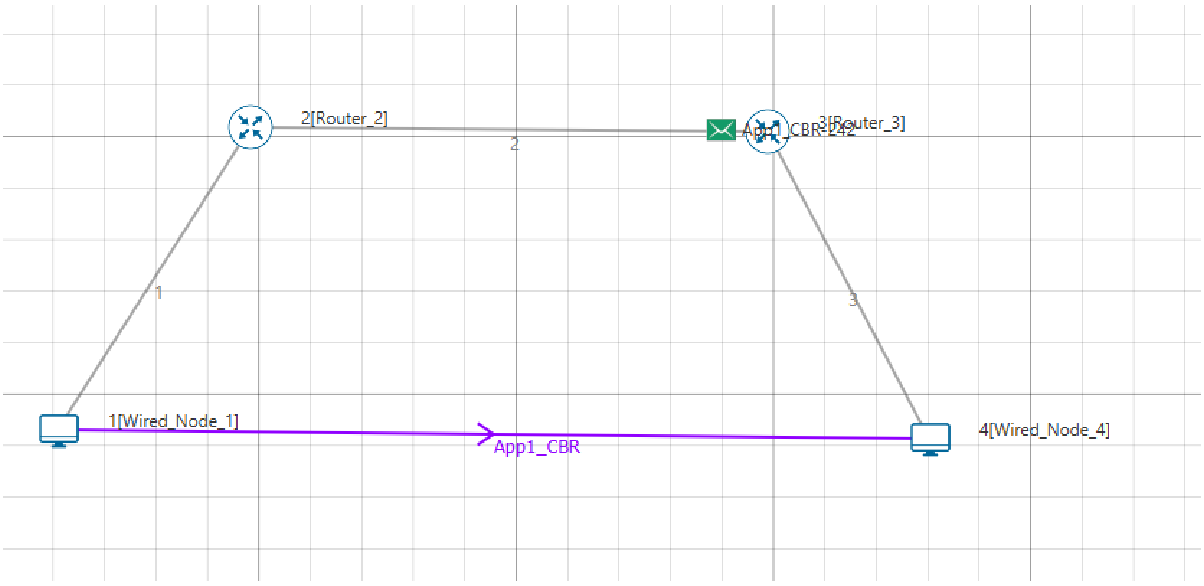



Plots
Enable Plots
Applications
Links
All Links
LinkID_1
LinkID_2
LinkID_3
Reconfigure plots, if a new link or a new application is configured in the simulation
OK
Cancel

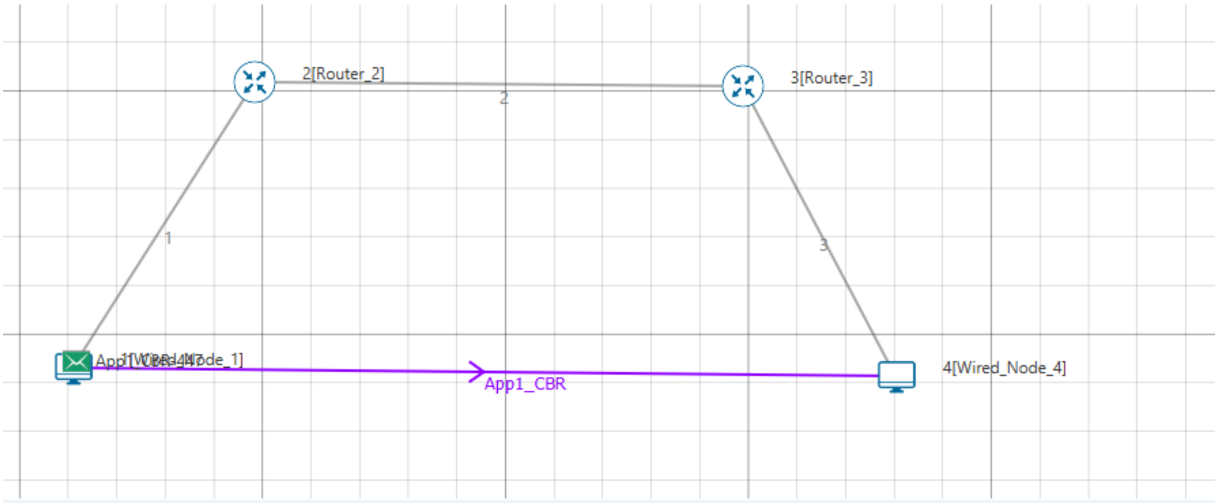

Run Simulation
Simulation Configuration
Run time Interaction
Static ARP Configuration
Packet Capture
Simulation Time:
Simulation Time (0.001 to 100000s):
50.0
Packet Animation:
Record Animation
(Simulation will slow down marginally)
Play & Record Animation
(Simulation will slow down significantly)
Don't Play / Record Animation
(Simulation will run fast)
Seed Values:
(Used to generate random numbers in simulation, Enter a value between 1 to 99999999)
Seed 1
12345678
Seed 2
23456789
OK
Reset



PACKET_ID	SEGMENT_ID	PACKET_TYPE	CONTROL_PACKET_TYP...	SOURCE_ID	DESTINATION_ID	TRANSMITTER_ID	RECEIVER_ID	APP_LAYER_ARRIVAL_T...	TRX_LAYER_ARRIVAL_T...	NW_LAYER_ARRIVAL_TI...	MAC_LAYER_ARRIVAL_...	PHY_LAYER_ARRIVA
17	0	CBR	App1_CBR	NODE-1	NODE-4	NODE-1	ROUTER-2	320000.000	320000.000	320000.000	320000.000	320000.000
16	0	CBR	App1_CBR	NODE-1	NODE-4	ROUTER-3	NODE-4	300000.000	300000.000	300250.160	300250.160	300250.160
16	0	CBR	App1_CBR	NODE-1	NODE-4	ROUTER-2	ROUTER-3	300000.000	300000.000	300126.120	300126.120	300126.120
16	0	CBR	App1_CBR	NODE-1	NODE-4	NODE-1	ROUTER-2	300000.000	300000.000	300000.000	300000.000	300000.000
15	0	CBR	App1_CBR	NODE-1	NODE-4	ROUTER-3	NODE-4	280000.000	280000.000	280250.160	280250.160	280250.160
15	0	CBR	App1_CBR	NODE-1	NODE-4	ROUTER-2	ROUTER-3	280000.000	280000.000	280126.120	280126.120	280126.120
15	0	CBR	App1_CBR	NODE-1	NODE-4	NODE-1	ROUTER-2	280000.000	280000.000	280000.000	280000.000	280000.000



After 5 sec:

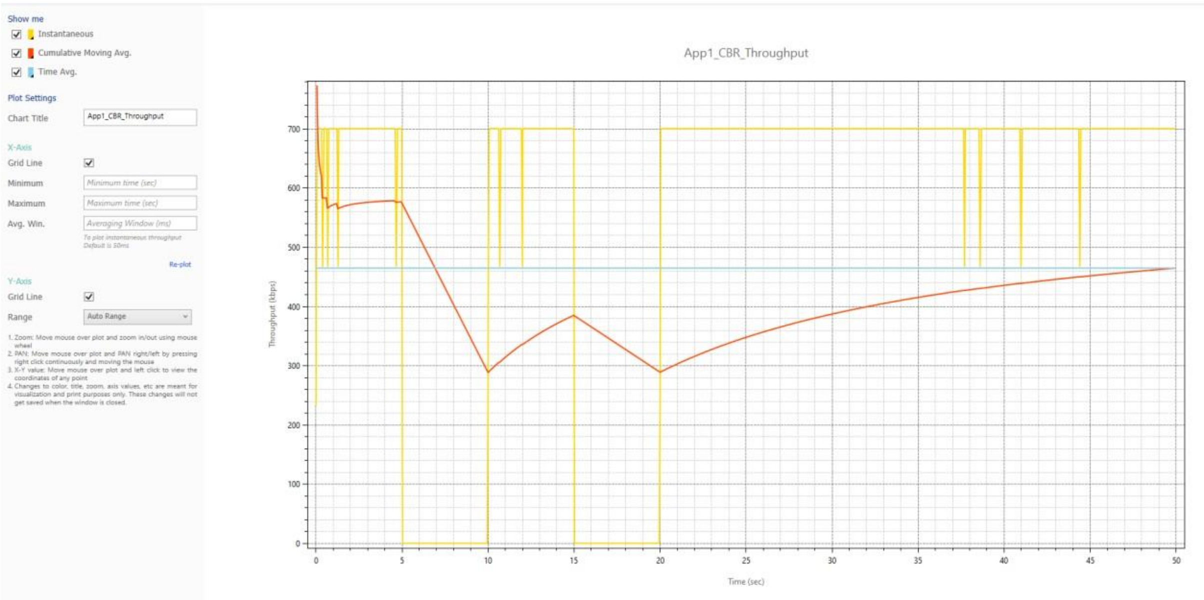


Application_Metrics_Table

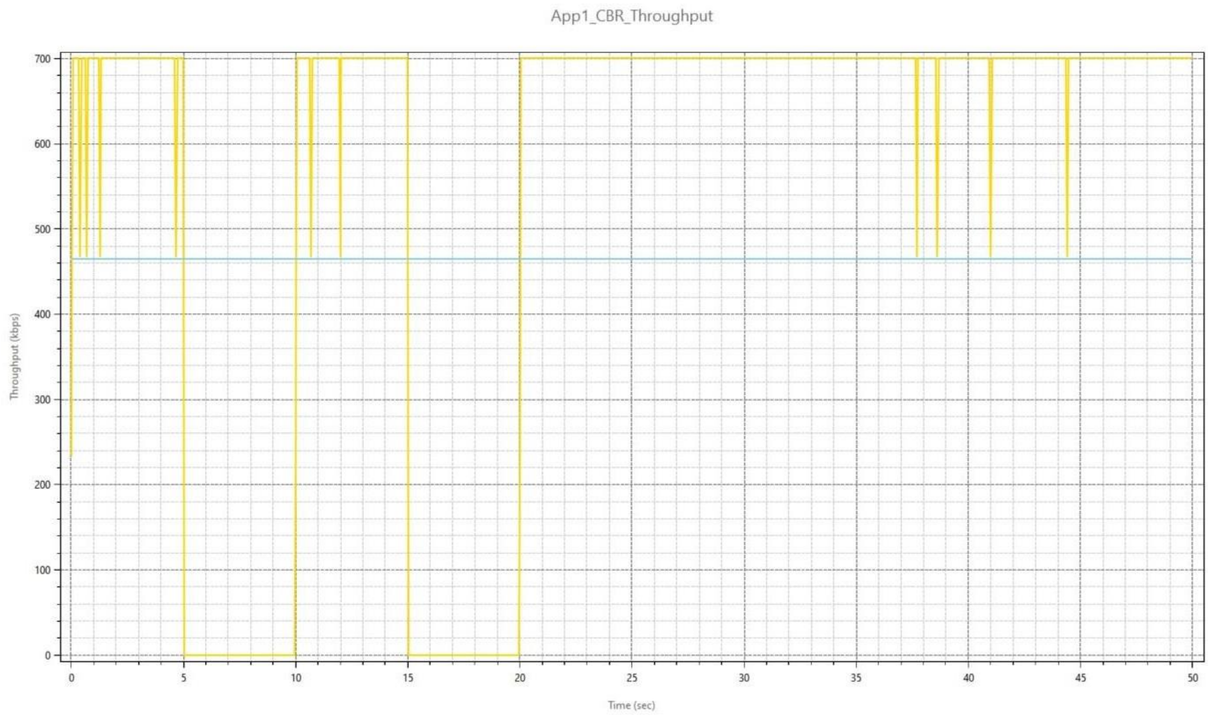
Application_Metrics

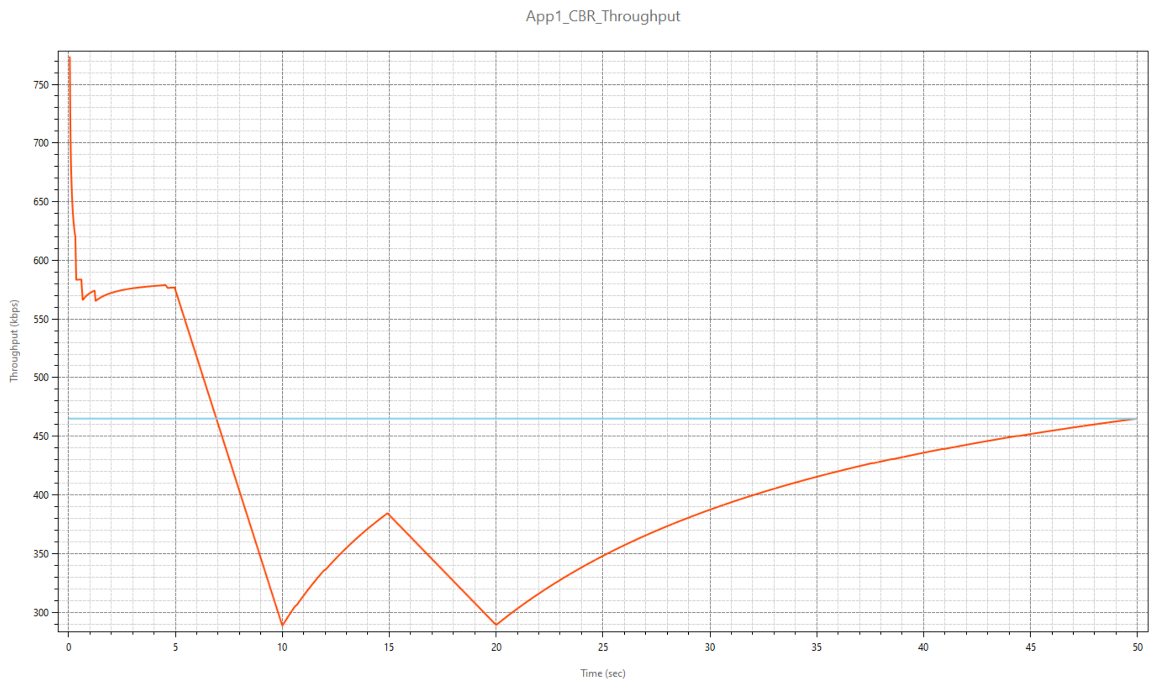
Detailed View

Application ID	Throughput Plot	Application Name	Packets Generated	Packets Received	Throughput (Mbps)	Delay (microsecond)
1	Application_Throughput_plot	App1_CBR	2500	1990	0.464864	376.304683



At 5 to 10 and 15-20 the throughput is 0 as the link is down





b) Link Failure with OSPF (Open Shortest Path First – Dynamic Routing Protocol)Design the configuration corresponding to this experiment as shown in below figure

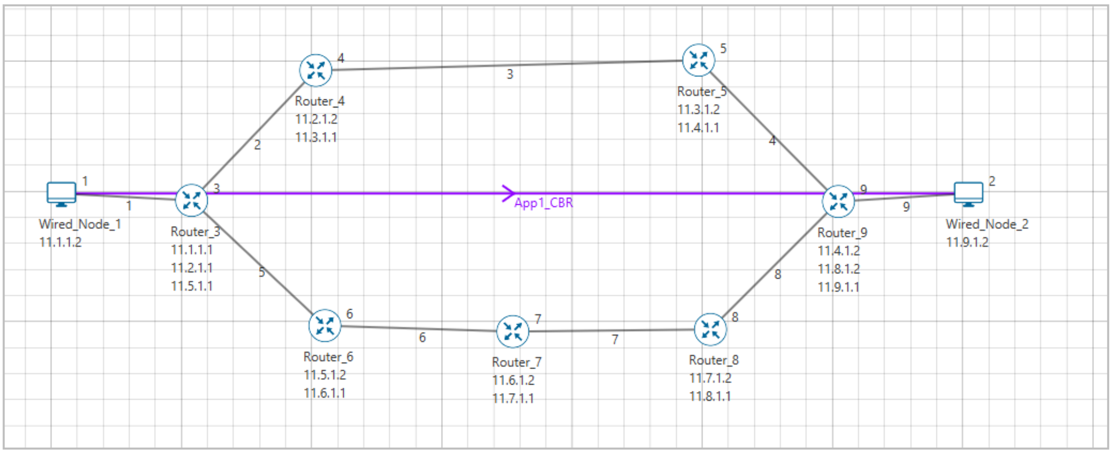


Figure 3: Network set up for studying the Link Failure with OSPF

Procedure

Without link failure: The following set of procedures were done to generate this sample:

Step 1: In the "Internetworks" library, and a network scenario is designed in NetSim comprising of 2 Wired Nodes and 7 Routers.

Step 2: By default, Link Failure Up Time is set to 0 and Down Time is set to 100000.

Step 3: Packet Trace is to be enabled in NetSim GUI. At the end of the simulation, a .csv file containing all the packet information is available for performing packet level analysis.

Step 4: Right click on the Application Flow App1 CBR and select Properties or click on the Application icon present in the top ribbon/toolbar.

A CBR Application is generated from Wired Node 1 i.e., Source to Wired Node 2 i.e., Destination with Packet Size remaining 1460 Bytes and Inter Arrival Time remaining 20000µs.

Additionally, the "Start Time(s)" parameter is set to 30, while configuring the application. This time isusually set to be greater than the time taken for OSPF Convergence (i.e., exchange of OSPF information between all the routers), and it increases as the size of the network increases.

Step 5: Transport protocol set as TCP.

Step 6: Enable the plots and run the simulation

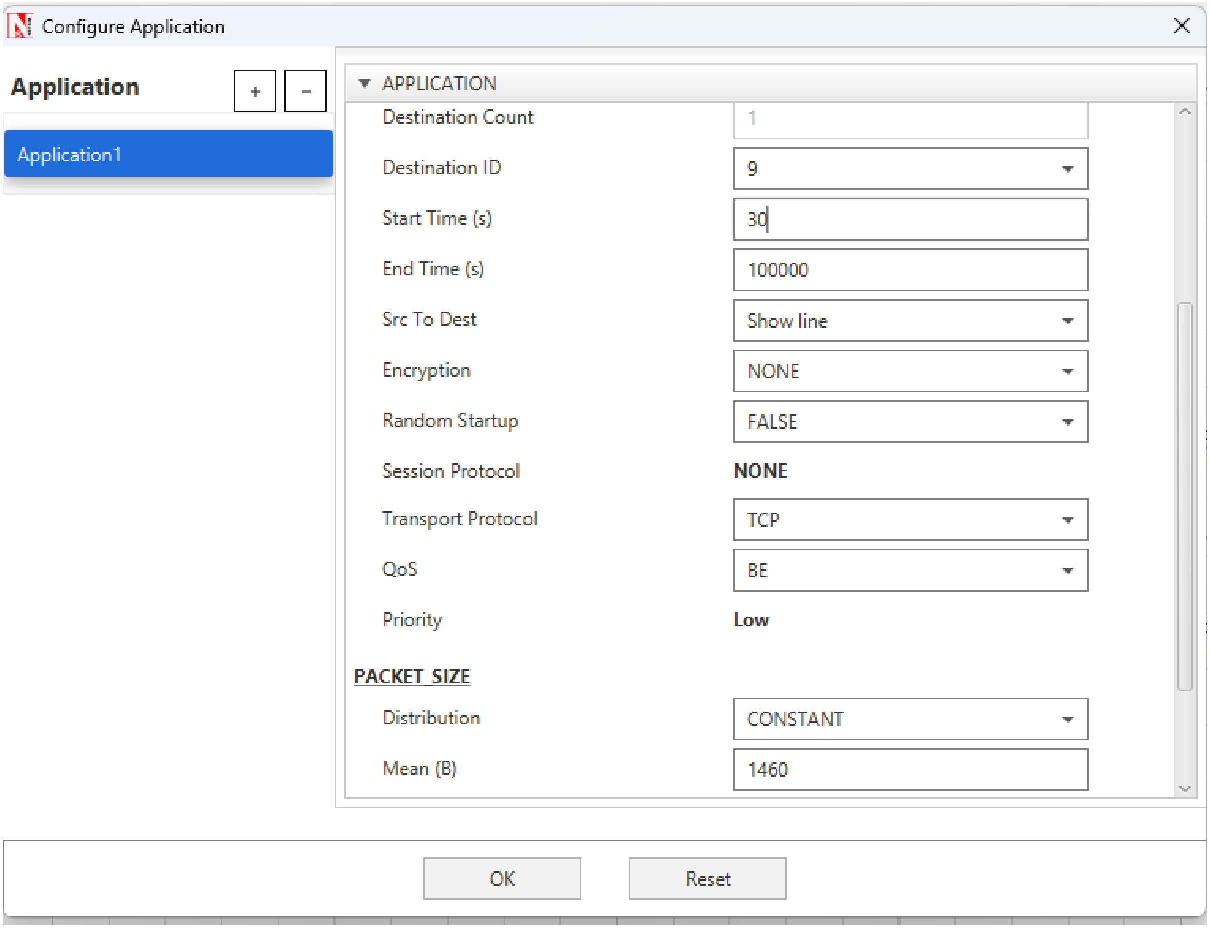
for 80 Seconds.Output

Go to NetSim Packet Animation Window, click on Play button. We can notice the following:

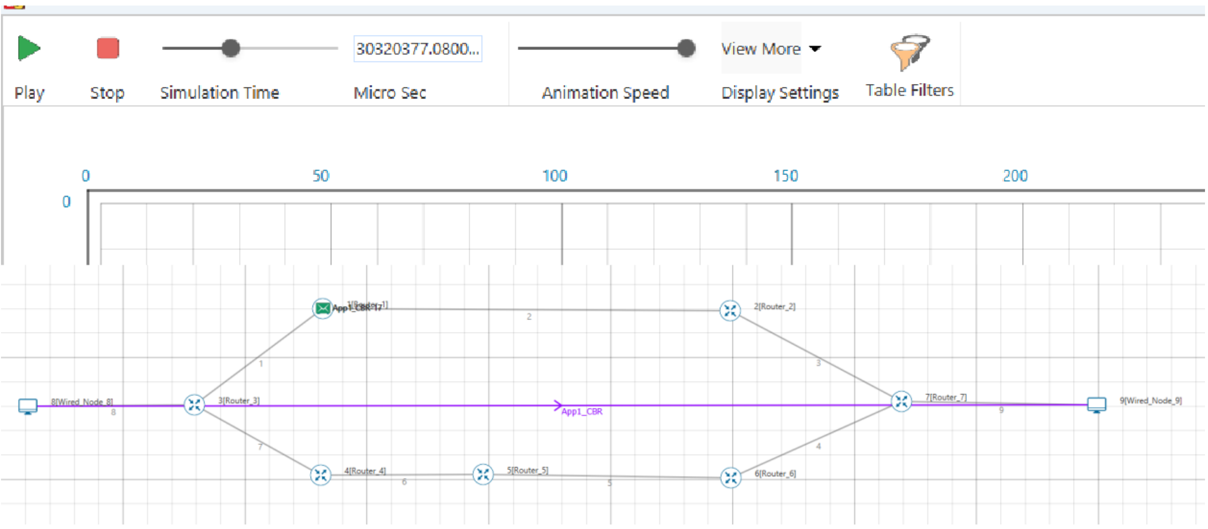
- Initially OSPF Control Packets are exchanged between all the routers.
- Once after the exchange of control packets, the data packets are sent from the source to the destination.

The packets are routed to the Destination via what path by default?

Record your inferences OSPF id decided by the number of hops



Time is started from 30 sec as time is required by the routers to learn and collect the data of all therouters



The upper path is chosen for the packet transfer

Application_Metrics_Table

Application_Metrics

Application ID	Application Name	Packets Generated	Packets Received	Throughput (Mbps)	Delay (microsecond)	Jitter (microsecond)
1	App1_CBR	2500	2500	0.584000	34898.613104	5209.182229

TCP_Metrics_Table

TCP_Metrics

Source	Destination	Segment Sent	Segment Received	Ack Sent	Ack Received	Duplicate ack r
ROUTER_1	ANY_DEVICE	0	0	0	0	0
ROUTER_2	ANY_DEVICE	0	0	0	0	0
ROUTER_3	ANY_DEVICE	0	0	0	0	0
ROUTER_4	ANY_DEVICE	0	0	0	0	0
ROUTER_5	ANY_DEVICE	0	0	0	0	0
ROUTER_6	ANY_DEVICE	0	0	0	0	0
ROUTER_7	ANY_DEVICE	0	0	0	0	0
WIRED_NODE_8	ANY_DEVICE	0	0	0	0	0
WIRED_NODE_9	ANY_DEVICE	0	0	0	0	0
WIRED_NODE_8	WIRED_NODE_9	2500	0	1	2500	3
WIRED_NODE_9	WIRED_NODE_8	0	2500	2500	1	0

Link_Metrics_Table

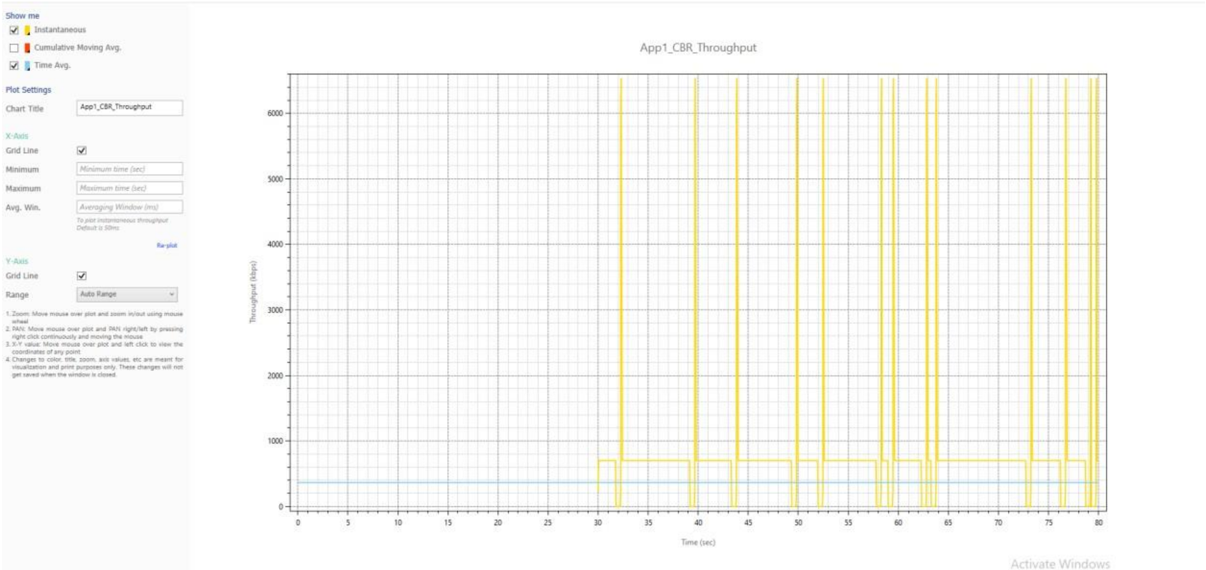
Link_Metrics

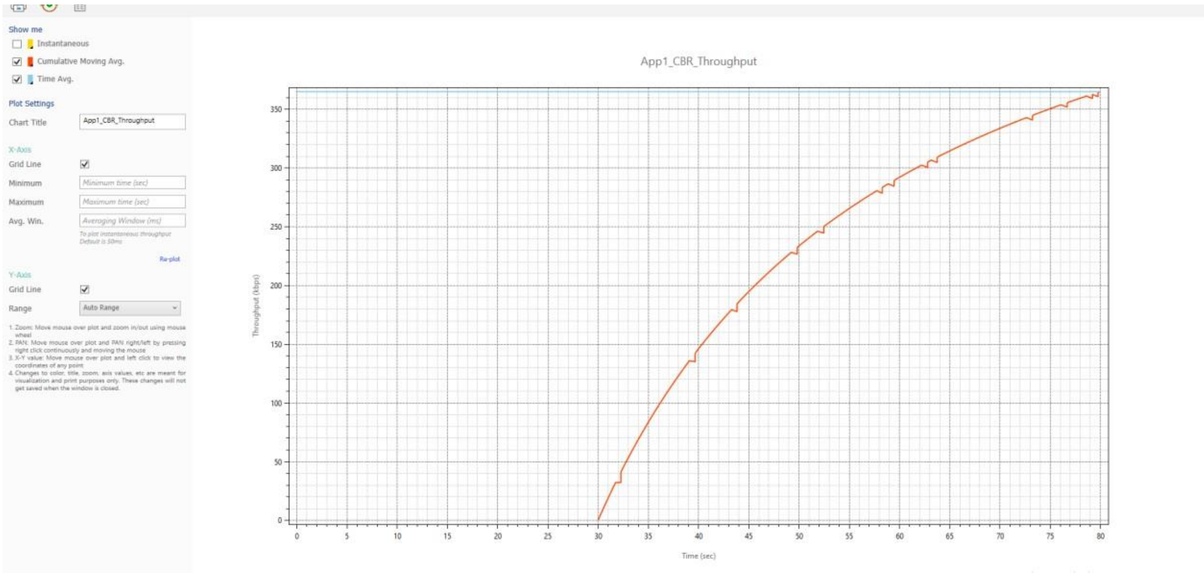
Link ID	Link Throughput Plot	Packets Transmitt...		Packets Errored		Packets Collided	
		Data	Control	Data	Control	Data	Control
All	NA	12534	12775	14	1	0	0
1	NA	2508	2541	1	0	0	0
2	NA	2507	2540	4	0	0	0
3	NA	2503	2542	1	1	0	0
4	NA	0	39	0	0	0	0
5	NA	0	36	0	0	0	0
6	NA	0	34	0	0	0	0
7	NA	0	37	0	0	0	0
8	NA	2514	2503	6	0	0	0
9	NA	2502	2503	2	0	0	0

Queue_Metrics_Table

Queue_Metrics

Device_id	Port_id	Queued_packet	Dequeued_packet	Dropped_packet
1	1	2519	2519	0
1	2	2528	2528	0
2	1	2519	2519	0
2	2	2525	2525	0
3	1	2530	2530	0
3	2	19	19	0
4	1	17	17	0
4	2	18	18	0
5	1	17	17	0
5	2	17	17	0
6	1	19	19	0
6	2	19	19	0
7	1	2520	2520	0
7	2	20	20	0





With link failure: Changes in design configuration

The following changes in settings are done from the previous sample:

Step 1: In Link 3 Properties, Link Failure Up Time is set to 0 and Down Time is set to 50. This means thatthe link would fail at 50 Seconds.

Step 2: Enable the plots and run the simulation

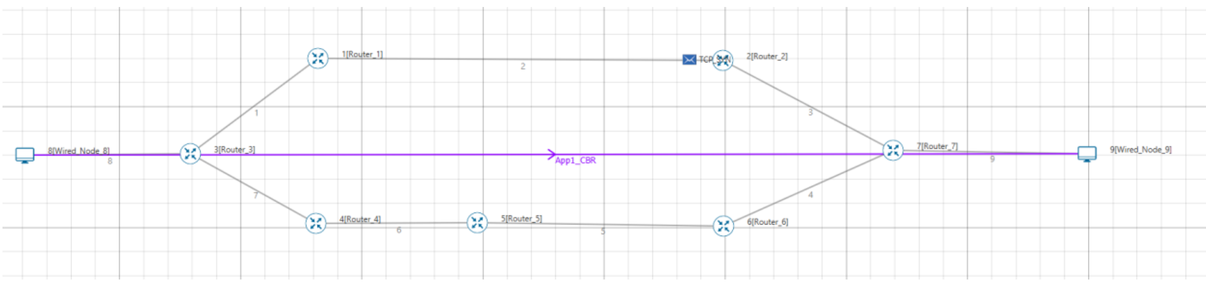
for 80 Seconds. With link failure

- We create a Link Failure in Link 3, between Router 4 and Router 5 at 50s.
- Since the packets are not able to reach the destination, the routing protocol recomputes analternate path to the Destination.

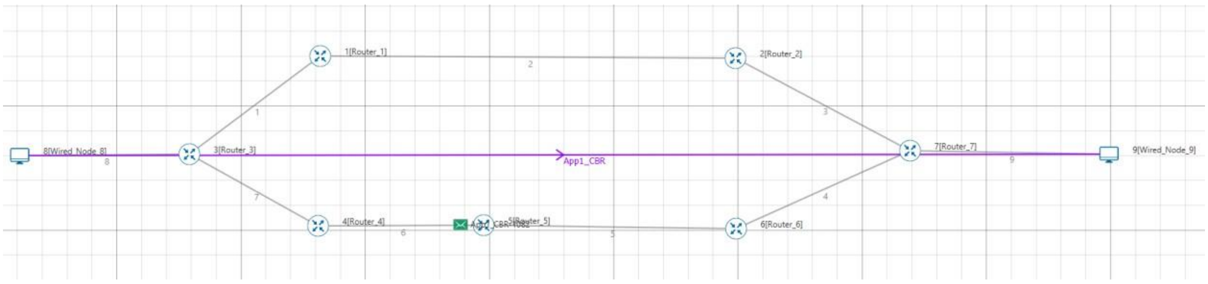
Observe the packet animation window and what happens during link failure,

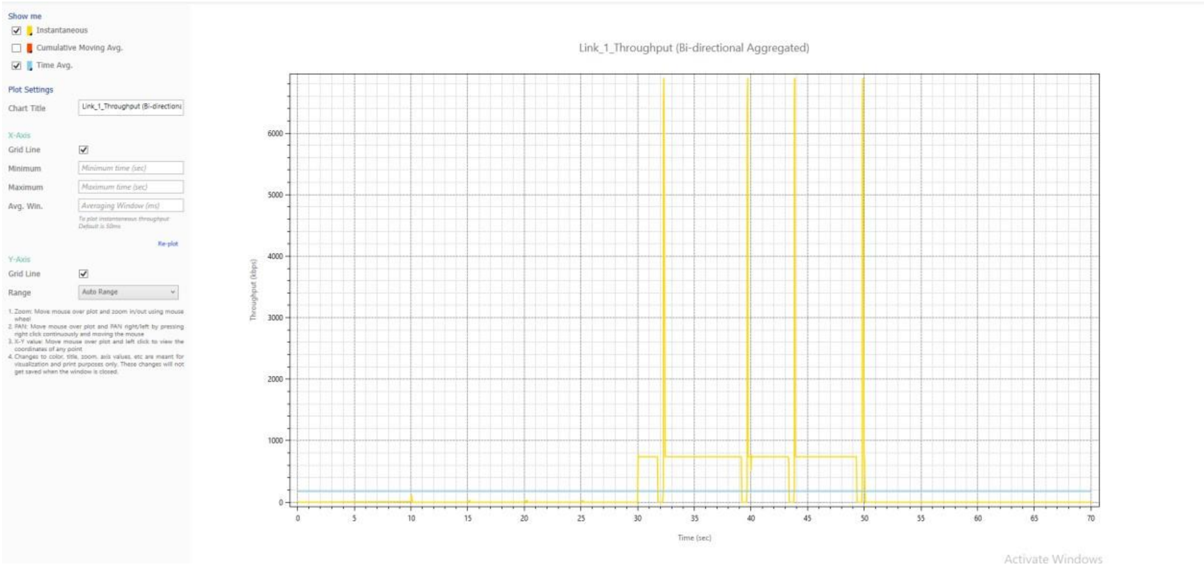
record your inferenceBefore 50 sec the packet continues in the same path as

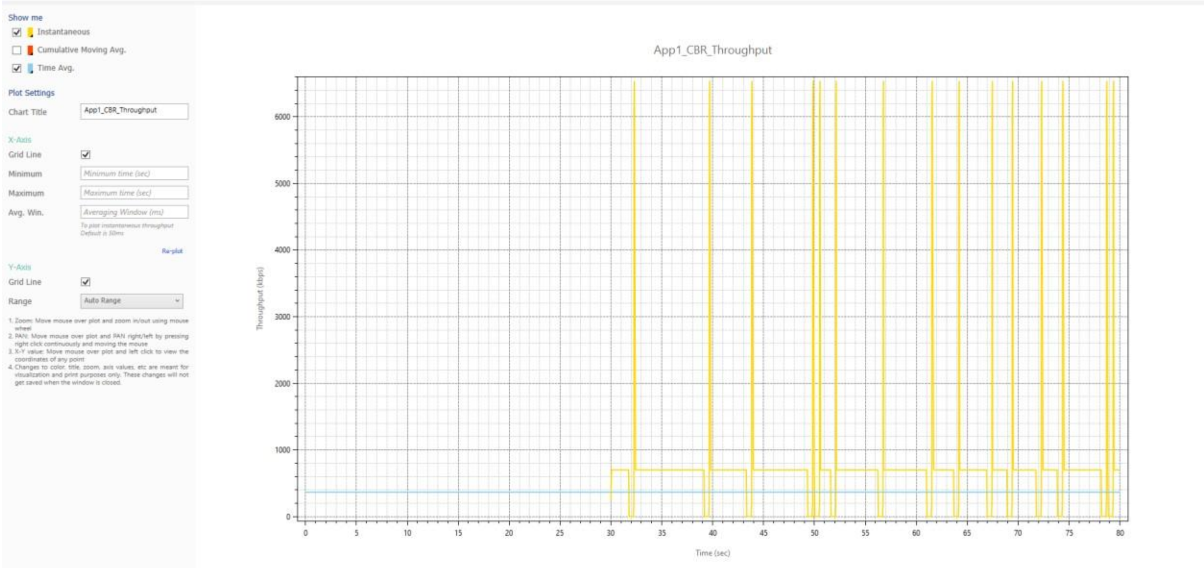
usual



After 50 sec the packet starts rerouting as the usual path link is down. now it is taking the longer route without packet loss





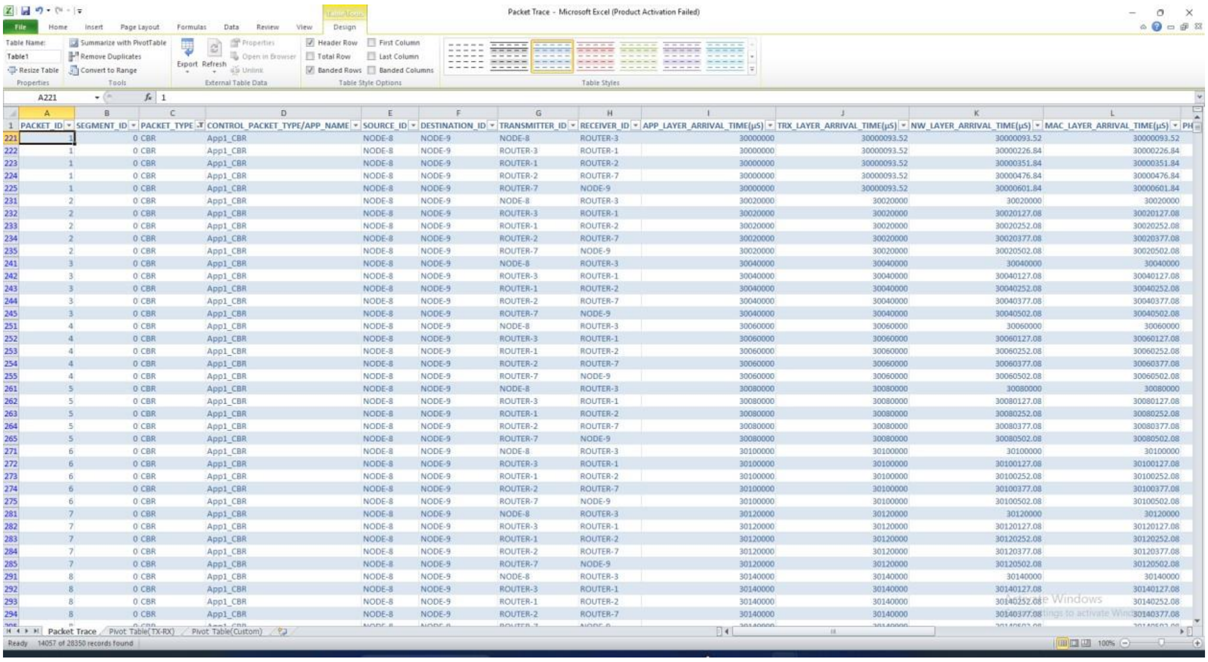


- This can also be observed in the Packet Trace.
- Go to the Results Dashboard and click on Open Packet Trace option present in the Left-Hand-Side of the window and do the following:

  Filter Control Packet Type/App Name to APP1 CBR and Transmitter ID to Router 3.

Record your results

Click on open packet trace and then in the excel sheet and for the packet type only CBR should beselected.



	PACKET_ID	SEGMENT_ID	PACKET_TYPE	CONTROL_PACKET_TYPE/APP_NAME	SOURCE_ID	DESTINATION_ID	TRANSMITTER_ID	RECEIVER_ID	APP_LAYER_ARRIVAL_TIME(µS)	TRX_LAYER_ARRIVAL_TIME(µS)	NW_LAYER_ARRIVAL_TIME(µS)	MAC_LAYER_ARRIVAL_TIME(µS)
221	1	0	CBR	App1_CBR	NODE-8	NODE-9	NODE-8	ROUTER-3	30000000	30000093.52	30000093.52	30000093.52
222	1	0	CBR	App1_CBR	NODE-8	NODE-9	ROUTER-3	ROUTER-1	30000000	30000093.52	30000226.84	30000226.84
223	1	0	CBR	App1_CBR	NODE-8	NODE-9	ROUTER-1	ROUTER-2	30000000	30000093.52	30000351.84	30000351.84
224	1	0	CBR	App1_CBR	NODE-8	NODE-9	ROUTER-2	ROUTER-7	30000000	30000093.52	30000476.84	30000476.84
225	1	0	CBR	App1_CBR	NODE-8	NODE-9	ROUTER-7	NODE-9	30000000	30000093.52	30000601.84	30000601.84
231	2	0	CBR	App1_CBR	NODE-8	NODE-9	NODE-8	ROUTER-3	30020000	30020000	30020000	30020000
232	2	0	CBR	App1_CBR	NODE-8	NODE-9	ROUTER-3	ROUTER-1	30020000	30020000	30020127.08	30020127.08
233	2	0	CBR	App1_CBR	NODE-8	NODE-9	ROUTER-1	ROUTER-2	30020000	30020000	30020252.08	30020252.08
234	2	0	CBR	App1_CBR	NODE-8	NODE-9	ROUTER-2	ROUTER-7	30020000	30020000	30020377.08	30020377.08
235	2	0	CBR	App1_CBR	NODE-8	NODE-9	ROUTER-7	NODE-9	30020000	30020000	30020502.08	30020502.08
241	3	0	CBR	App1_CBR	NODE-8	NODE-9	NODE-8	ROUTER-3	30040000	30040000	30040000	30040000
242	3	0	CBR	App1_CBR	NODE-8	NODE-9	ROUTER-3	ROUTER-1	30040000	30040000	30040127.08	30040127.08
243	3	0	CBR	App1_CBR	NODE-8	NODE-9	ROUTER-1	ROUTER-2	30040000	30040000	30040252.08	30040252.08
244	3	0	CBR	App1_CBR	NODE-8	NODE-9	ROUTER-2	ROUTER-7	30040000	30040000	30040377.08	30040377.08
245	3	0	CBR	App1_CBR	NODE-8	NODE-9	ROUTER-7	NODE-9	30040000	30040000	30040502.08	30040502.08
251	4	0	CBR	App1_CBR	NODE-8	NODE-9	NODE-8	ROUTER-3	30060000	30060000	30060000	30060000
252	4	0	CBR	App1_CBR	NODE-8	NODE-9	ROUTER-3	ROUTER-1	30060000	30060000	30060127.08	30060127.08
253	4	0	CBR	App1_CBR	NODE-8	NODE-9	ROUTER-1	ROUTER-2	30060000	30060000	30060252.08	30060252.08
254	4	0	CBR	App1_CBR	NODE-8	NODE-9	ROUTER-2	ROUTER-7	30060000	30060000	30060377.08	30060377.08
255	4	0	CBR	App1_CBR	NODE-8	NODE-9	ROUTER-7	NODE-9	30060000	30060000	30060502.08	30060502.08
261	5	0	CBR	App1_CBR	NODE-8	NODE-9	NODE-8	ROUTER-3	30080000	30080000	30080000	30080000
262	5	0	CBR	App1_CBR	NODE-8	NODE-9	ROUTER-3	ROUTER-1	30080000	30080000	30080127.08	30080127.08
263	5	0	CBR	App1_CBR	NODE-8	NODE-9	ROUTER-1	ROUTER-2	30080000	30080000	30080252.08	30080252.08
264	5	0	CBR	App1_CBR	NODE-8	NODE-9	ROUTER-2	ROUTER-7	30080000	30080000	30080377.08	30080377.08
265	5	0	CBR	App1_CBR	NODE-8	NODE-9	ROUTER-7	NODE-9	30080000	30080000	30080502.08	30080502.08
271	6	0	CBR	App1_CBR	NODE-8	NODE-9	NODE-8	ROUTER-3	30100000	30100000	30100000	30100000
272	6	0	CBR	App1_CBR	NODE-8	NODE-9	ROUTER-3	ROUTER-1	30100000	30100000	30100127.08	30100127.08
273	6	0	CBR	App1_CBR	NODE-8	NODE-9	ROUTER-1	ROUTER-2	30100000	30100000	30100252.08	30100252.08
274	6	0	CBR	App1_CBR	NODE-8	NODE-9	ROUTER-2	ROUTER-7	30100000	30100000	30100377.08	30100377.08
275	6	0	CBR	App1_CBR	NODE-8	NODE-9	ROUTER-7	NODE-9	30100000	30100000	30100502.08	30100502.08
281	7	0	CBR	App1_CBR	NODE-8	NODE-9	NODE-8	ROUTER-3	30120000	30120000	30120000	30120000
282	7	0	CBR	App1_CBR	NODE-8	NODE-9	ROUTER-3	ROUTER-1	30120000	30120000	30120127.08	30120127.08
283	7	0	CBR	App1_CBR	NODE-8	NODE-9	ROUTER-1	ROUTER-2	30120000	30120000	30120252.08	30120252.08
284	7	0	CBR	App1_CBR	NODE-8	NODE-9	ROUTER-2	ROUTER-7	30120000	30120000	30120377.08	30120377.08
285	7	0	CBR	App1_CBR	NODE-8	NODE-9	ROUTER-7	NODE-9	30120000	30120000	30120502.08	30120502.08
291	8	0	CBR	App1_CBR	NODE-8	NODE-9	NODE-8	ROUTER-3	30140000	30140000	30140000	30140000
292	8	0	CBR	App1_CBR	NODE-8	NODE-9	ROUTER-3	ROUTER-1	30140000	30140000	30140127.08	30140127.08
293	8	0	CBR	App1_CBR	NODE-8	NODE-9	ROUTER-1	ROUTER-2	30140000	30140000	30140252.08	30140252.08
294	8	0	CBR	App1_CBR	NODE-8	NODE-9	ROUTER-2	ROUTER-7	30140000	30140000	30140377.08	30140377.08

<u>Study Experiment : 1c</u> **Date: 20-08-24**

<u>Study of Basic Linux Commands</u>

Name : Thurlapati Sai Sree Praneetha

Reg. No : 22BLC1350

Faculty Name : Jaya Vignesh T

Aim:

To study and understand the basic Linux commands.

Theory:

Linux is the oft-ignored third wheel to Windows and Mac. Yes, over the past decade, the open source operating system has gained a lot of traction, but it's still a far cry from being considered popular. Yet though that may be true, Linux still earns new converts every day. It can be a traumatic experience having to go from a GUI-based operating system like Windows or Mac to one that requires command line fiddling. But if you can get over that initial hump of difficulty, you may find that Linux is surprisingly robust. Linux is case-sensitive and space-sensitive.

Linux has origins in the command line, and there can be many times when you will not be running a GUI. On some systems, such as a dedicated server, you may not have a GUI installed at all. Linux provides you with many command line tools to manipulate files.

COMMANDS

1) **<u>pwd:</u>**

It shows the current directory from the root

```
student@administrator-ThinkCentre-neo-50s-Gen-3 ~ $ pwd
/home/student
```

2) **<u>ls:</u>**

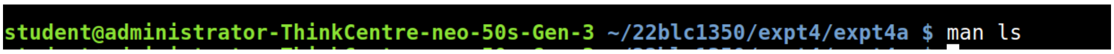
It gives the list of files and folders present in the current directory.

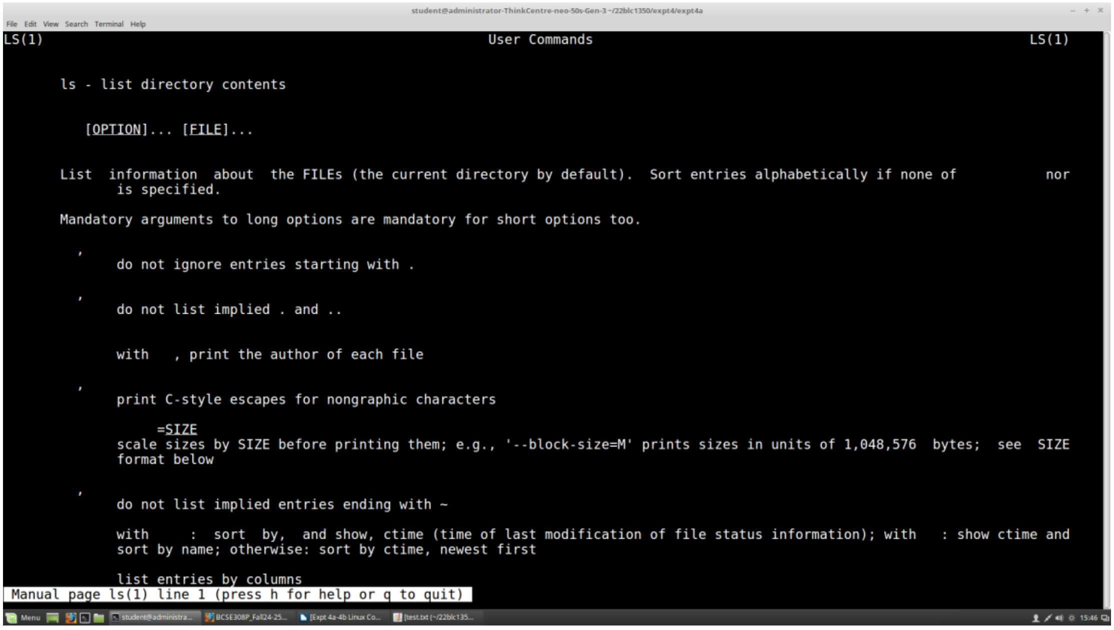
```
student@administrator-ThinkCentre-neo-50s-Gen-3 ~ $ ls
21bec1187     mglinstaller.gz
21bec1587     Music
21blc1098     nam.out
21blc1135.c   nam.tr
21blc1213     ns2ns3py_updated.txt
22BLC1401     ns-allinone-2.35.tar.gz
a.out         ns-allinone-3.37.tar.bz2
crc.c         Pictures
Desktop       Public
Documents     rudraksh
Downloads     sample.c
e1.tcl        Study Ex5a - Linux Commands_Instruction document.doc
exp77         Templates
FAT_LAB       test.txt
helllo        tracegraph202
helllo.c      tracegraph202linux.tar.gz
hello         tracemetrics-1.4.0.zip
hello.c       Videos
```

```
student@administrator-ThinkCentre-neo-50s-Gen-3 ~ $ ls -ltr
total 107492
drwxrwxr-x 4 student student     4096 Jun 22  2003 tracegraph202
-rw------- 1 student student  8247599 Dec 14  2022 mglinstaller.gz
-rw------- 1 student student 59529999 Dec 14  2022 ns-allinone-2.35.tar.gz
-rw------- 1 student student 39515477 Dec 14  2022 ns-allinone-3.37.tar.bz2
-rw------- 1 student student     4272 Dec 14  2022 ns2ns3py_updated.txt
-rw------- 1 student student   966377 Dec 14  2022 tracegraph202linux.tar.gz
-rw------- 1 student student  1638797 Dec 14  2022 tracemetrics-1.4.0.zip
drwxr-xr-x 2 student student     4096 Dec 20  2022 Videos
drwxr-xr-x 2 student student     4096 Dec 20  2022 Templates
drwxr-xr-x 2 student student     4096 Dec 20  2022 Public
drwxr-xr-x 2 student student     4096 Dec 20  2022 Music
-rw-r--r-- 1 student student     1786 Apr 13  2023 e1.tcl
-rw-r--r-- 1 student student        0 Apr 13  2023 nam.tr
-rw-r--r-- 1 student student        0 Apr 13  2023 nam.out
drwxr-xr-x 2 student student     4096 Apr 13  2023 FAT_LAB
```

3) **<u>man</u>:**

It allows users to view the reference manuals of a command or utility run in the terminal.





4) **<u>mkdir:</u>**

Used to create a new directory or document in the folders.

5) **<u>cd:</u>**

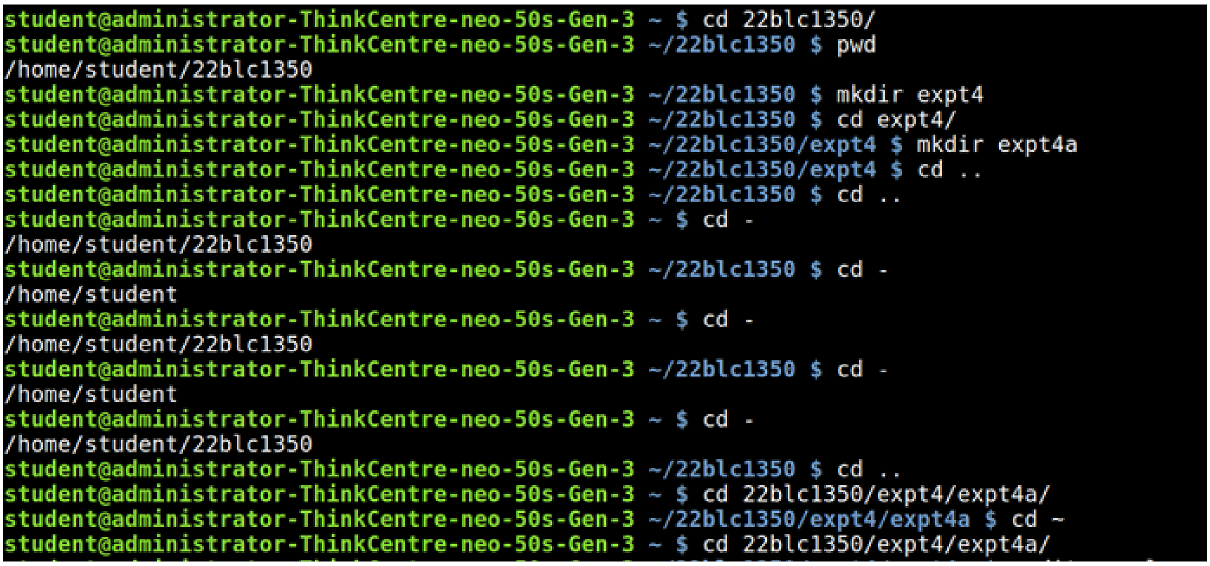
cd command is used to change the current directory and move through various document

6) **<u>cd..:</u>**

Append two dots (..) to the cd command to change to the parent of the current directory.

7) **cd ~:**

A way to quickly return to the home directory from anywhere in the filesystem



```
student@administrator-ThinkCentre-neo-50s-Gen-3 ~ $ cd 22blc1350/
student@administrator-ThinkCentre-neo-50s-Gen-3 ~/22blc1350 $ pwd
/home/student/22blc1350
student@administrator-ThinkCentre-neo-50s-Gen-3 ~/22blc1350 $ mkdir expt4
student@administrator-ThinkCentre-neo-50s-Gen-3 ~/22blc1350 $ cd expt4/
student@administrator-ThinkCentre-neo-50s-Gen-3 ~/22blc1350/expt4 $ mkdir expt4a
student@administrator-ThinkCentre-neo-50s-Gen-3 ~/22blc1350/expt4 $ cd ..
student@administrator-ThinkCentre-neo-50s-Gen-3 ~/22blc1350 $ cd ..
student@administrator-ThinkCentre-neo-50s-Gen-3 ~ $ cd -
/home/student/22blc1350
student@administrator-ThinkCentre-neo-50s-Gen-3 ~/22blc1350 $ cd -
/home/student
student@administrator-ThinkCentre-neo-50s-Gen-3 ~ $ cd -
/home/student/22blc1350
student@administrator-ThinkCentre-neo-50s-Gen-3 ~/22blc1350 $ cd -
/home/student
student@administrator-ThinkCentre-neo-50s-Gen-3 ~ $ cd -
/home/student/22blc1350
student@administrator-ThinkCentre-neo-50s-Gen-3 ~/22blc1350 $ cd ..
student@administrator-ThinkCentre-neo-50s-Gen-3 ~ $ cd 22blc1350/expt4/expt4a/
student@administrator-ThinkCentre-neo-50s-Gen-3 ~/22blc1350/expt4/expt4a $ cd ~
student@administrator-ThinkCentre-neo-50s-Gen-3 ~ $ cd 22blc1350/expt4/expt4a/
```

8) **cat :**

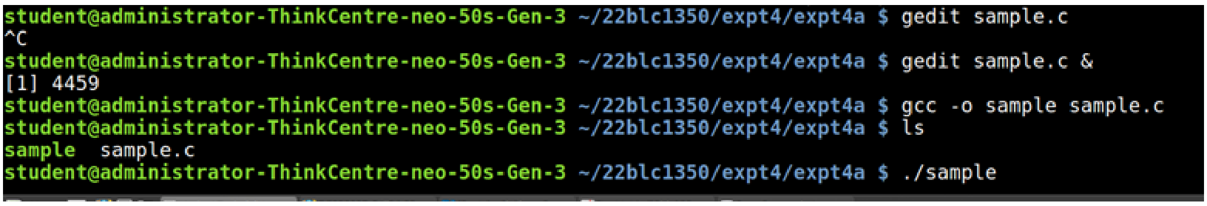
cat is most commonly used to display the contents of one or multiple text files, combine files by appending one file's contents to the end of another file, and create new files

9) **gedit <filename>:**

An easy workaround for this is to use double quotes around the path or filename in order to have the entire thing be read as a string

10) **gedit <filename> &:**

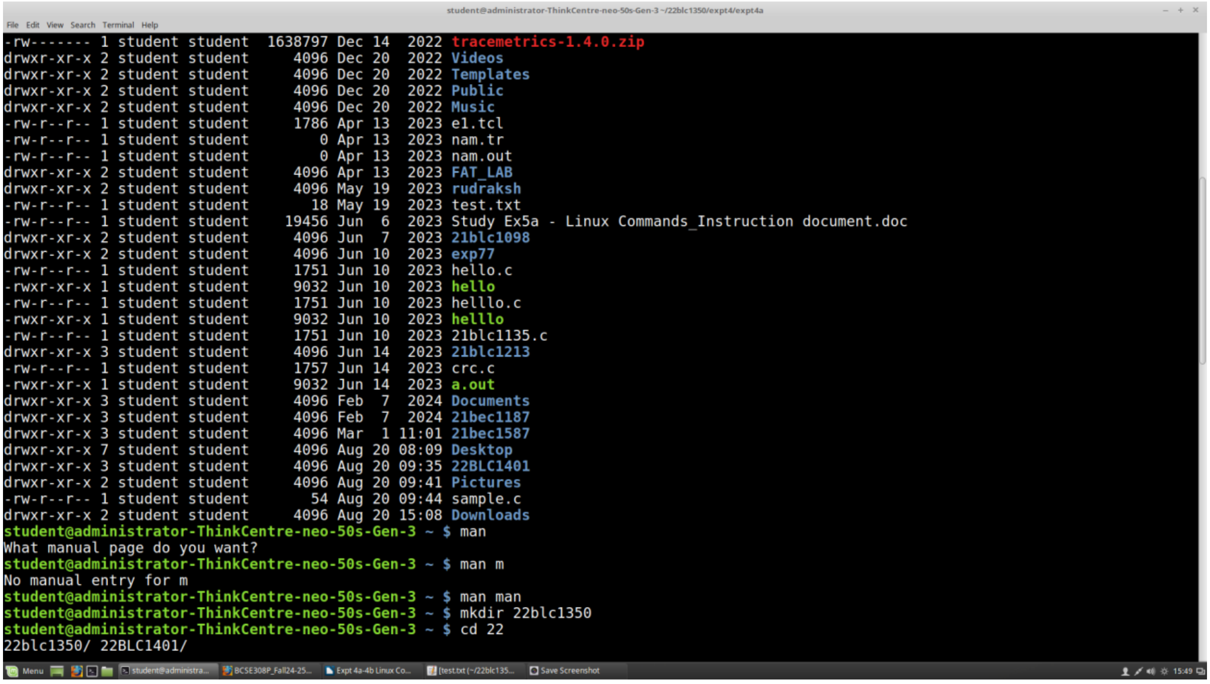
grab the filename from the user before opening it in gedit.



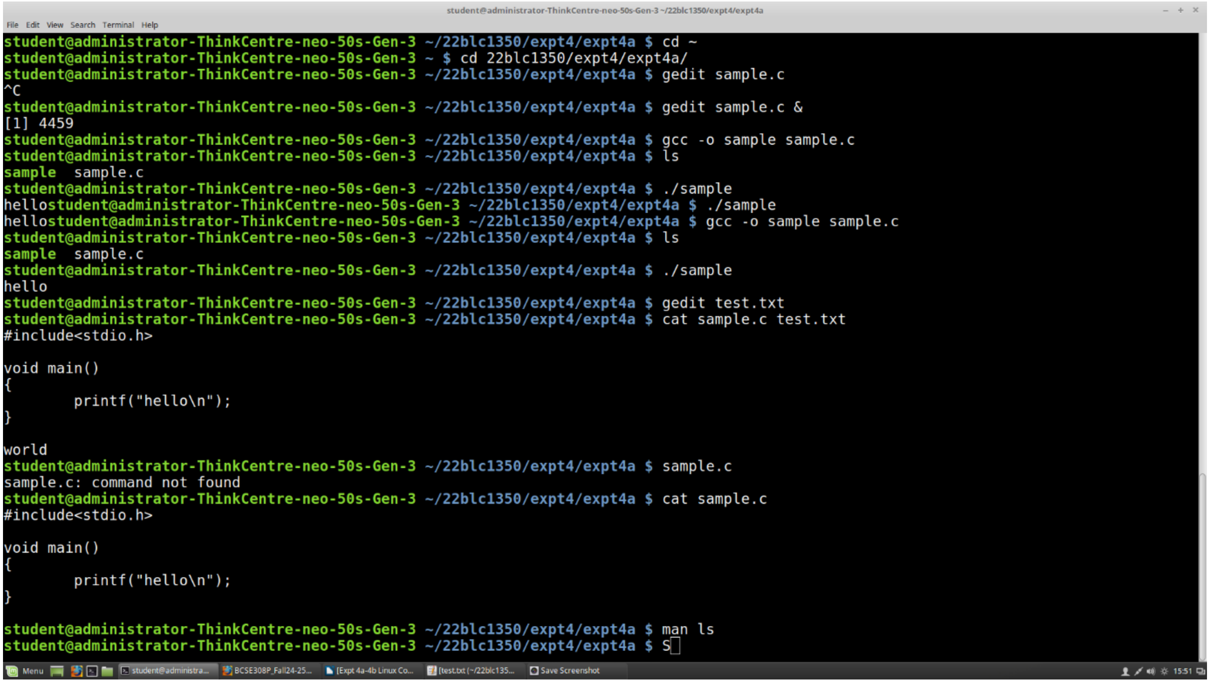
```
student@administrator-ThinkCentre-neo-50s-Gen-3 ~/22blc1350/expt4/expt4a $ gedit sample.c
^C
student@administrator-ThinkCentre-neo-50s-Gen-3 ~/22blc1350/expt4/expt4a $ gedit sample.c &
[1] 4459
student@administrator-ThinkCentre-neo-50s-Gen-3 ~/22blc1350/expt4/expt4a $ gcc -o sample sample.c
student@administrator-ThinkCentre-neo-50s-Gen-3 ~/22blc1350/expt4/expt4a $ ls
sample  sample.c
student@administrator-ThinkCentre-neo-50s-Gen-3 ~/22blc1350/expt4/expt4a $ ./sample
```

Result:

Thus a few basic linux commands are studied and practiced successfully.



```
student@administrator-ThinkCentre-neo-50s-Gen-3 ~/22blc1350/expt4/expt4a
File Edit View Search Terminal Help
-rw------- 1 student student 1638797 Dec 14  2022 tracemetrics-1.4.0.zip
drwxr-xr-x 2 student student    4096 Dec 20  2022 Videos
drwxr-xr-x 2 student student    4096 Dec 20  2022 Templates
drwxr-xr-x 2 student student    4096 Dec 20  2022 Public
drwxr-xr-x 2 student student    4096 Dec 20  2022 Music
-rw-r--r-- 1 student student    1786 Apr 13  2023 e1.tcl
-rw-r--r-- 1 student student       0 Apr 13  2023 nam.tr
-rw-r--r-- 1 student student       0 Apr 13  2023 nam.out
drwxr-xr-x 2 student student    4096 Apr 13  2023 FAT_LAB
drwxr-xr-x 2 student student    4096 May 19  2023 rudraksh
-rw-r--r-- 1 student student      18 May 19  2023 test.txt
-rw-r--r-- 1 student student   19456 Jun  6  2023 Study Ex5a - Linux Commands_Instruction document.doc
drwxr-xr-x 2 student student    4096 Jun  7  2023 21blc1098
drwxr-xr-x 2 student student    4096 Jun 10  2023 exp77
-rw-r--r-- 1 student student    1751 Jun 10  2023 hello.c
-rwxr-xr-x 1 student student    9032 Jun 10  2023 hello
-rw-r--r-- 1 student student    1751 Jun 10  2023 helllo.c
-rwxr-xr-x 1 student student    9032 Jun 10  2023 helllo
-rw-r--r-- 1 student student    1751 Jun 10  2023 21blc1135.c
drwxr-xr-x 3 student student    4096 Jun 14  2023 21blc1213
-rw-r--r-- 1 student student    1757 Jun 14  2023 crc.c
-rwxr-xr-x 1 student student    9032 Jun 14  2023 a.out
drwxr-xr-x 3 student student    4096 Feb  7  2024 Documents
drwxr-xr-x 3 student student    4096 Feb  7  2024 21bec1187
drwxr-xr-x 3 student student    4096 Mar  1 11:01 21bec1587
drwxr-xr-x 7 student student    4096 Aug 20 08:09 Desktop
drwxr-xr-x 3 student student    4096 Aug 20 09:35 22BLC1401
drwxr-xr-x 2 student student    4096 Aug 20 09:41 Pictures
-rw-r--r-- 1 student student      54 Aug 20 09:44 sample.c
drwxr-xr-x 2 student student    4096 Aug 20 15:08 Downloads
student@administrator-ThinkCentre-neo-50s-Gen-3 ~ $ man
What manual page do you want?
student@administrator-ThinkCentre-neo-50s-Gen-3 ~ $ man m
No manual entry for m
student@administrator-ThinkCentre-neo-50s-Gen-3 ~ $ man man
student@administrator-ThinkCentre-neo-50s-Gen-3 ~ $ mkdir 22blc1350
student@administrator-ThinkCentre-neo-50s-Gen-3 ~ $ cd 22
22blc1350/ 22BLC1401/
```



CONCLUSION: Therefore, we have learnt the basic commands in Linux

<u>Experiment No: 4b</u> **Date: 10/09/24**

<u>Implementation of CRC (Cyclic Redundancy Check) using C-Program to perform Error Detection mechanism</u>

Name : Thurlapati Sai Sree Praneetha
Reg. No : 22BLC1350
Faculty Name : Jayavignesh T

<u>AIM:</u>

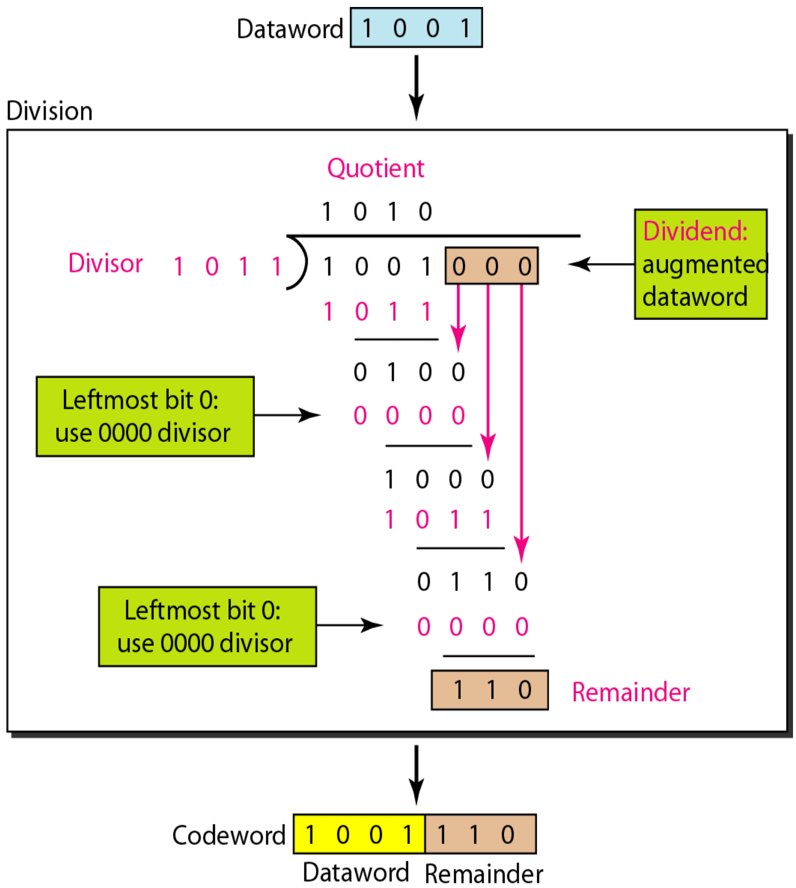
a. To Implement CRC (Cyclic Redundancy check) using C-Program to perform Error Detection mechanism

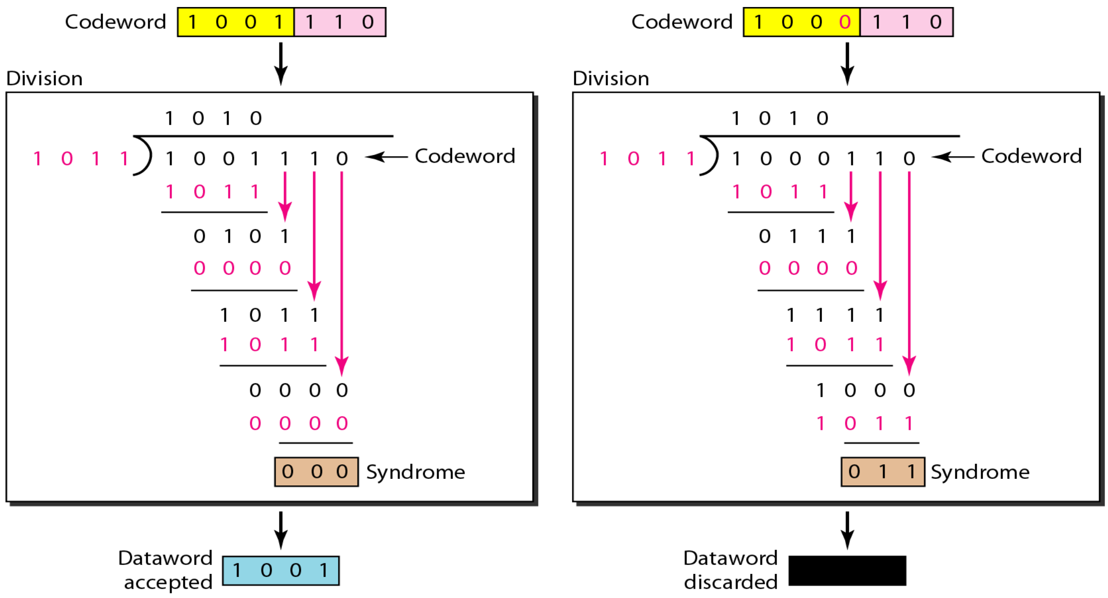
<u>SOFTWARE USED:</u>

C Programming Language, Turbo C/C++ compiler, gcc compiler (Linux)

<u>PROCEDURE:</u>

<u>Division in CRC encoder</u>





<u>Division in the CRC decoder for two cases</u>

<u>PROGRAM:</u>

<CRC Encoding Program IN C>

<CRC Decoding Program IN C>

A CRC is constructed to generate a 3 bit FCS for a 4-bit message. The generator pattern is 1011.

a. Show the generation of the codeword at the sender site
b. Show the checking of the codeword at the receiver site (assume no error).
c. Show that the detection algorithm detects error if received codeword is corrupted.

ENCODING:

CODE:

```
#include <stdio.h>
#include <string.h>

int main () {
char divisor[100], data[100];
printf("Enter the Divisor: ");
scanf("%s", divisor);
printf("Enter the DataWord: ");
scanf("%s", data);
int pd = strlen(divisor) - 1;
```

```
char padded[200];
strcpy(padded, data);
while (pd > 0) {
strcat(padded, "0");
pd--;
}
char divi[200] = "";
int j = strlen(divisor);
char current_remainder[100];
strncpy(current_remainder, padded, strlen(divisor));
current_remainder[strlen(divisor)] = '\0';
while (j <= strlen(padded)) {
if (current_remainder[0] == '1') {
strcat(divi, "1");
for (int i = 0; i < strlen(divisor); i++) {
current_remainder[i] = ((current_remainder[i] -
'0') ^ (divisor[i] - '0')) + '0';
}
}
else
{
strcat(divi, "0");1011
}
memmove(current_remainder, current_remainder + 1,
strlen(current_remainder));
if (j < strlen(padded)) {
int len = strlen(current_remainder);
current_remainder[len] = padded[j];
current_remainder[len + 1] = '\0';
}
j++;
}
char result[200];
strcpy (result, data);
```

```
        strcat (result, current_remainder);
        printf ("The CodeWord is: %s\n", result);
        return 0;
}
```

OUTPUT:

```
Output

/tmp/dtEVjV2n8X.o
Enter the Divisor: 1011
Enter the DataWord: 1001
The CodeWord is: 1001110


=== Code Execution Successful ===
```

DECODING

CODE:

```
        #include <stdio.h>
        #include <string.h>

        int main() {
        char divisor[100], data[100];
        char result[200];
        printf(“Enter the Divisor: “);
        scanf(“%s”, divisor);
        printf(“Enter the DataWord: “);
        scanf(“%s”, data);
        printf(“Enter the CodeWord: “);
        scanf(“%s”, result);
        int ptr = strlen(divisor);
        char curr_remainder[100];
```

```
strncpy(curr_remainder, result, strlen(divisor));
curr_remainder[strlen(divisor)] = '\0';
char divi[200] = "";
while (ptr <= strlen(result)) {
if (curr_remainder[0] == '1') {
strcat(divi, "1");
for (int I = 0; I < strlen(divisor); i++) {
curr_remainder[i] = ((curr_remainder[i] – '0') ^
(divisor[i] – '0')) + '0';
}
} else {
strcat(divi, "0");
}
memmove(curr_remainder, curr_remainder + 1,
strlen(curr_remainder));
if (ptr < strlen(result)) {
int len = strlen(curr_remainder);
curr_remainder[len] = result[ptr];
curr_remainder[len + 1] = '\0';
}
ptr++;
}
for (int I = 0; I < strlen(curr_remainder); i++) {
if (curr_remainder[i] == '1') {
printf("Data word is discarded due to errors");
return 0;
}
}
printf("Data word is accepted");
return 0;
}
```

OUTPUT:

- If there is an error in the received code word:

Output

```
/tmp/Sr1m0lRC5m.o
Enter the Divisor: 1011
Enter the DataWord: 1001
Enter the CodeWord: 1001101
Data word is discarded due to errors

=== Code Execution Successful ===
```

- If the received codeword is correct without any errors:

Output

```
/tmp/BqyPMTGHJ6.o
Enter the Divisor: 1011
Enter the DataWord: 1001
Enter the CodeWord: 1001110
Data word is accepted

=== Code Execution Successful ===
```

RESULT & INFERENCES:

- For encoding the data given, we use the data word which is padded with and prepare it for division by the divisor. XOR is used to perform the division. The remainder is then added to the original data to create the final codeword.
- The received codeword is divided using the same XOR method as during encoding. The remainder is checked: if it's zero, the data is error-free; if it's not, the data is corrupted. Based on the remainder, the data is either accepted or rejected.
- Thus, CRC is an effective method for determining whether the codeword received by the receiver is correct or contains errors.

Experiment 5 - Use Netsim Interactive Simulation mode and apply

the basic networking commands such as ping, route print, acl

Name : Thurlapati Sai Sree Praneetha
Reg. No : 22BLC1350
Faculty Name : Jaya Vignesh T

AIM:

Task 5) Use Netsim Interactive Simulation mode and apply the basic networking commands such as ping, route print, acl

SOFTWARE USED:

Netsim v13.1 Academic version or higher.

NetSim Interactive Simulation Theory:

NetSim allows users to interact with the simulation at runtime via a socket. User Interactions make simulation more realistic by allowing command execution to view/modify certain device parameters during runtime.

This section will demonstrate how to perform Interactive simulation for a simple network scenario. Let us consider Internetworks. To create a new scenario, go to New à Internetworks. Click & drop Wired Nodes and Router onto the Simulation Environment and link them as shown below

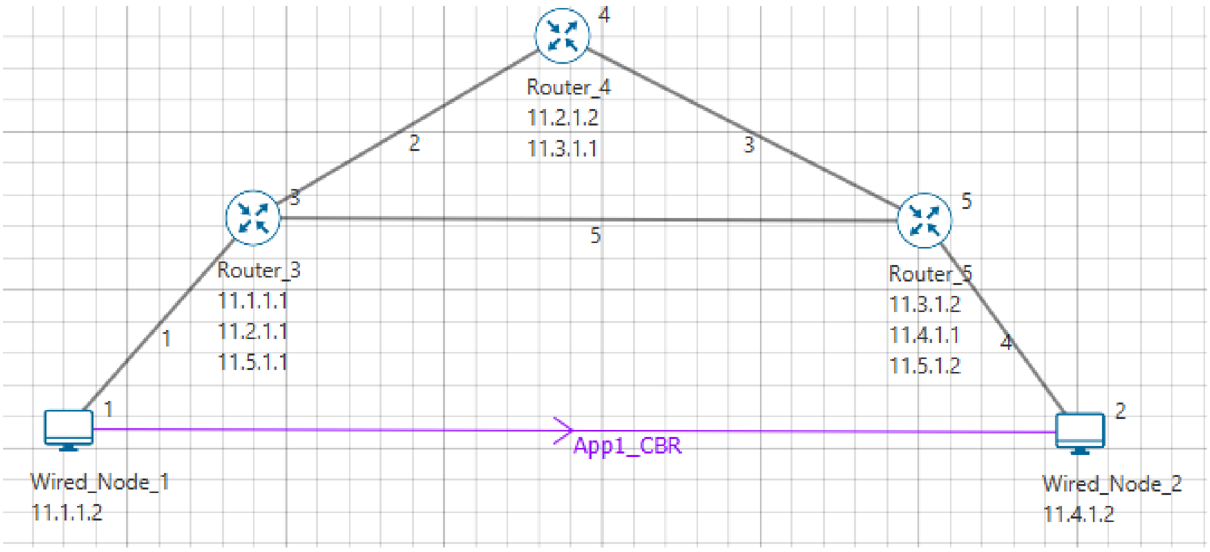


Figure 1: Network Topology

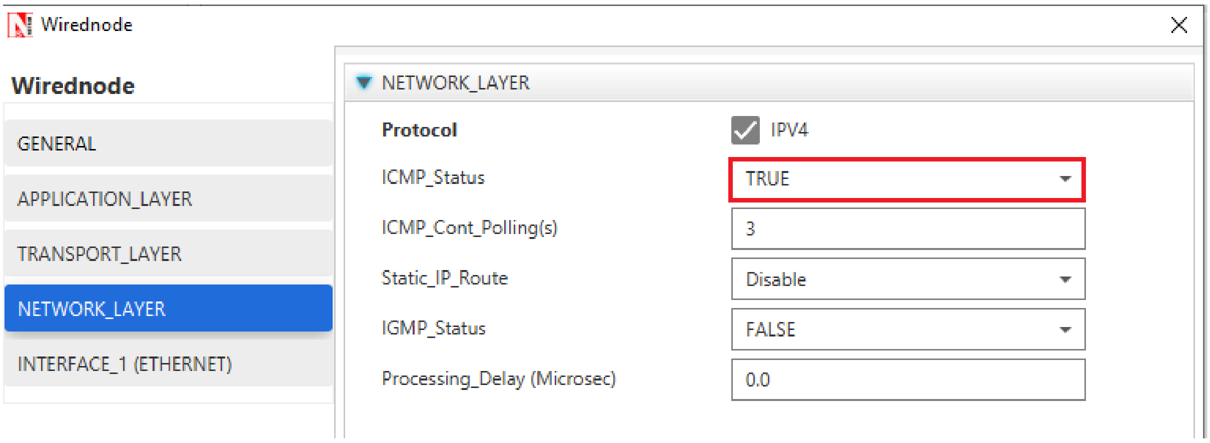
Procedure

The following set of procedures were done to generate this sample:

Step 1: A network scenario is designed in NetSim GUI comprising of 2 Wired Nodes and 3 Routers in the **"Internetworks"** Network Library.

Step 2: In the Network Layer properties of Wired Node 1, **"ICMP Status"** is set as TRUE.

Similarly, ICMP Status is set as TRUE for all the devices as shown Figure 2



Step 3: In the General properties of Wired Node 1, **Wireshark Capture** is set as Online.

Step 4: Right click on the Application Flow **App1 CBR** and select Properties or click on the Application icon present in the top ribbon/toolbar.

A CBR Application is generated from Wired Node 1 i.e., Source to Wired Node 2 i.e., Destination with Packet Size remaining 1460Bytes and Inter Arrival Time 233.6µs. Transport Protocol is set to **UDP**.

Additionally, the **"Start Time(s)"** parameter is set to 30, while configuring the application. This time is usually set to be greater than the time taken for OSPF Convergence (i.e., Exchange of OSPF information between all the routers), and it increases as the size of the network increases.

Step 5: Packet Trace is enabled in NetSim GUI. At the end of the simulation, a very large .csv file is containing all the packet information is available for the users to perform packet level analysis. Plots are enabled in NetSim GUI.

Step 6: Click on Run Simulation. Simulation Time is set to 300 Seconds and in the **Runtime Interaction** tab Figure 3, Interactive Simulation is set to True

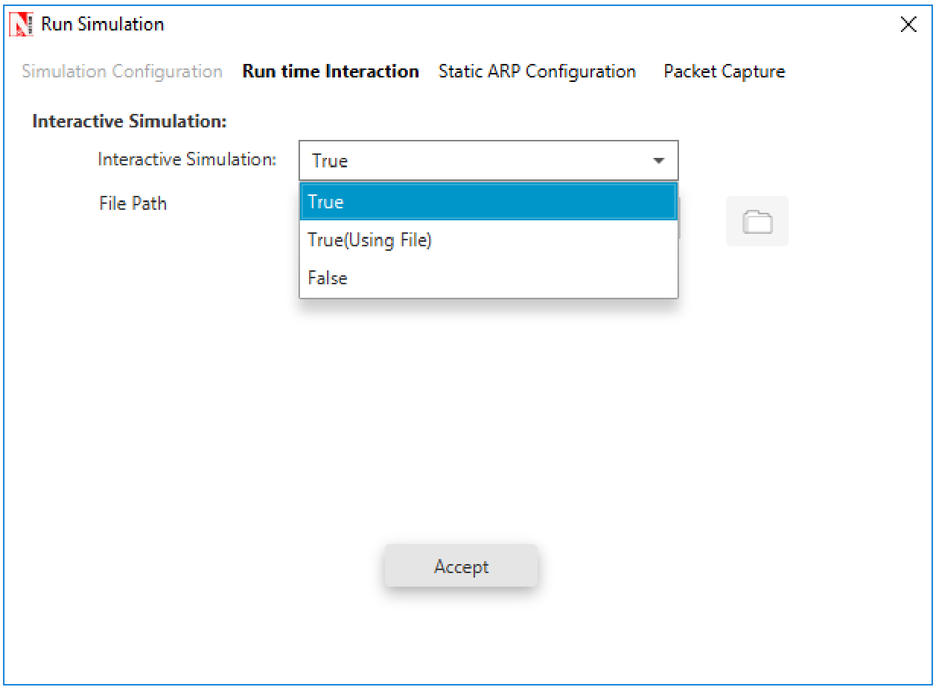


Figure 2: Run time Interaction tab set Interactive Simulation as True

- You can even set the Simulation Time as 500 sec instead (It is recommended to specify a longer simulation time to ensure that there is sufficient time for the user to execute the various commands and see the effect of that before Simulation ends) and click OK.
- Simulation (NetSimCore.exe) will start running and will display a message "waiting for first client to connect" as shown below Figure.

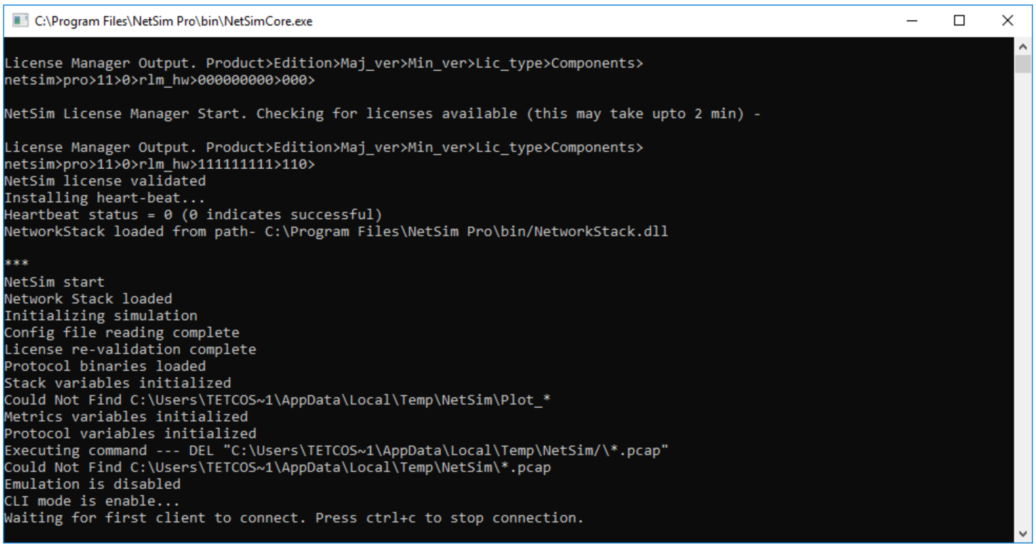


Figure: Waiting for first client to connect

After Simulation window opens, goto Network scenario and right clickon Router_3 or any other node and select NetSim Console option as shown

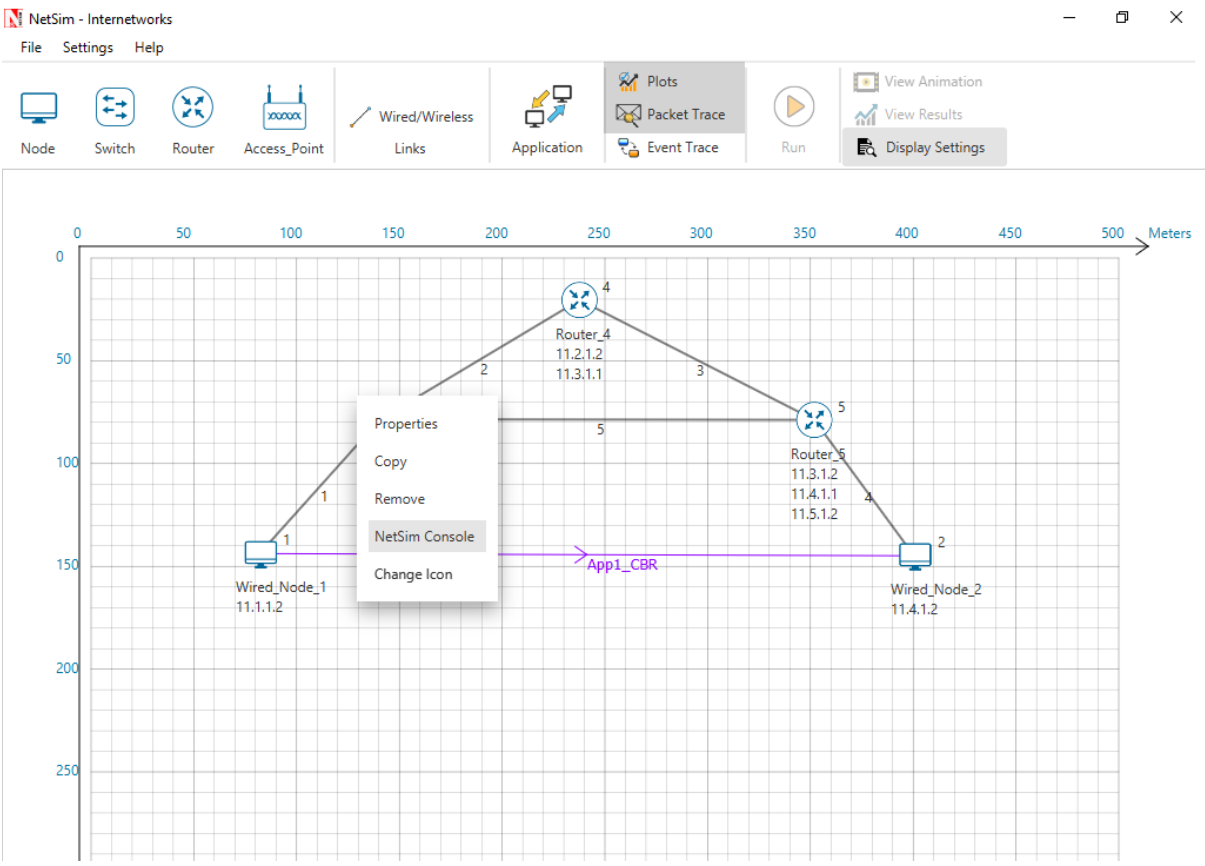


Figure: Select NetSim Console option

- Now client (NetSimCLI.exe) will start running and it will try to establish connection with NetSimCore.exe. After connection is established, the window will look similar like this shown below Figure

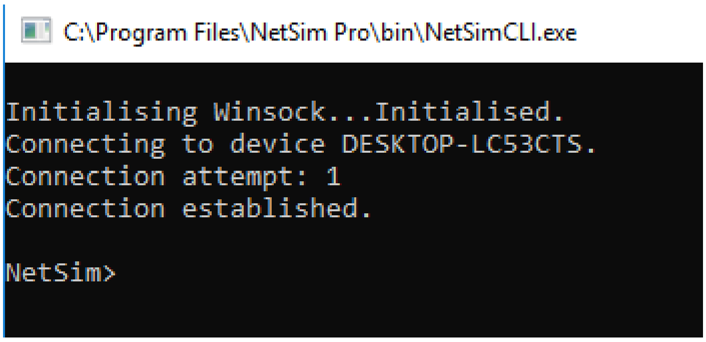


Figure: Connection is established

- After this the command line interface can be used to execute the supported commands

[Note: Commands are not a case sensitive]

Simulation specific

- Pause
- PauseAt
- Continue
- Stop
- Exit
- Reconnect

Pause: To pause the currently running simulation

PauseAt: To pause the currently running simulation with respect to particular time (Ex: To Pause simulation at 70.2 sec use command as **PauseAt 70.2**)

Continue: To start the currently paused simulation

Stop: To stop the currently running simulation (NetSimCore.exe)

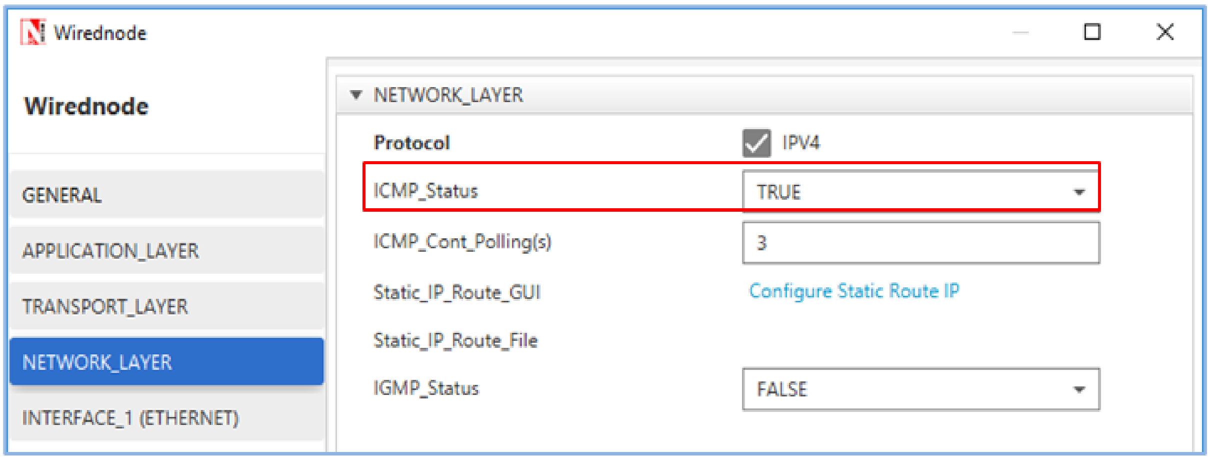
Exit: To exit from the client (NetSimCLI.exe)

Reconnect: To reconnect client (NetSimCLI.exe) to simulation (NetSimCore.exe) when we rerun simulation again

Ping Command#

- The ping command is one of the most often used networking utilities for troubleshooting network problems.
- You can use the ping command to test the availability of a networking device (usually a computer) on a network.
- When you ping a device, you send that device a short message, which it then sends back (the echo)

- If you receive a reply then the device is in Network, if you don't then the device is faulty, disconnected, switched off, incorrectly configured.
- You can use the ***ping*** cmd with an IP address or Device name.
- **ICMP_Status** should be set as True in all nodes (Wired_Node and Router)



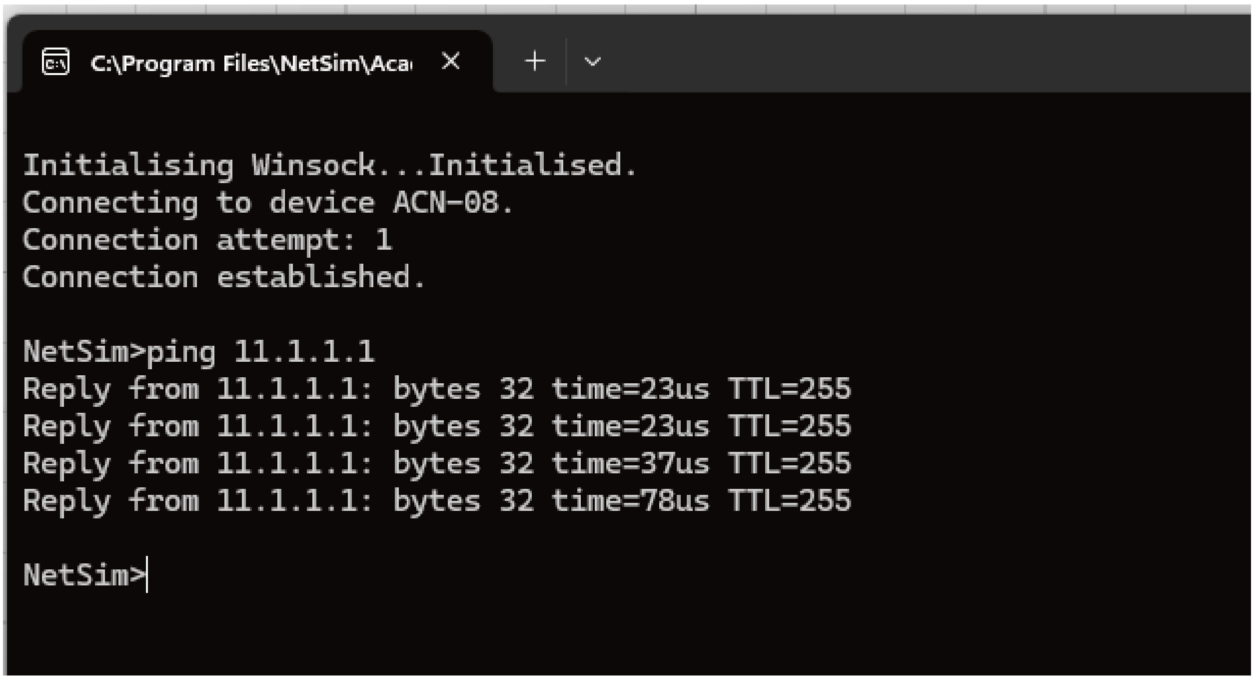
- Right click on Wired_Node_1 and go to properties. Under General > properties enable Wireshark Capture option as "Online"

<u>Perform the following tasks and observe the results</u>

ping <IP address> e.g. ping 11.4.1.2

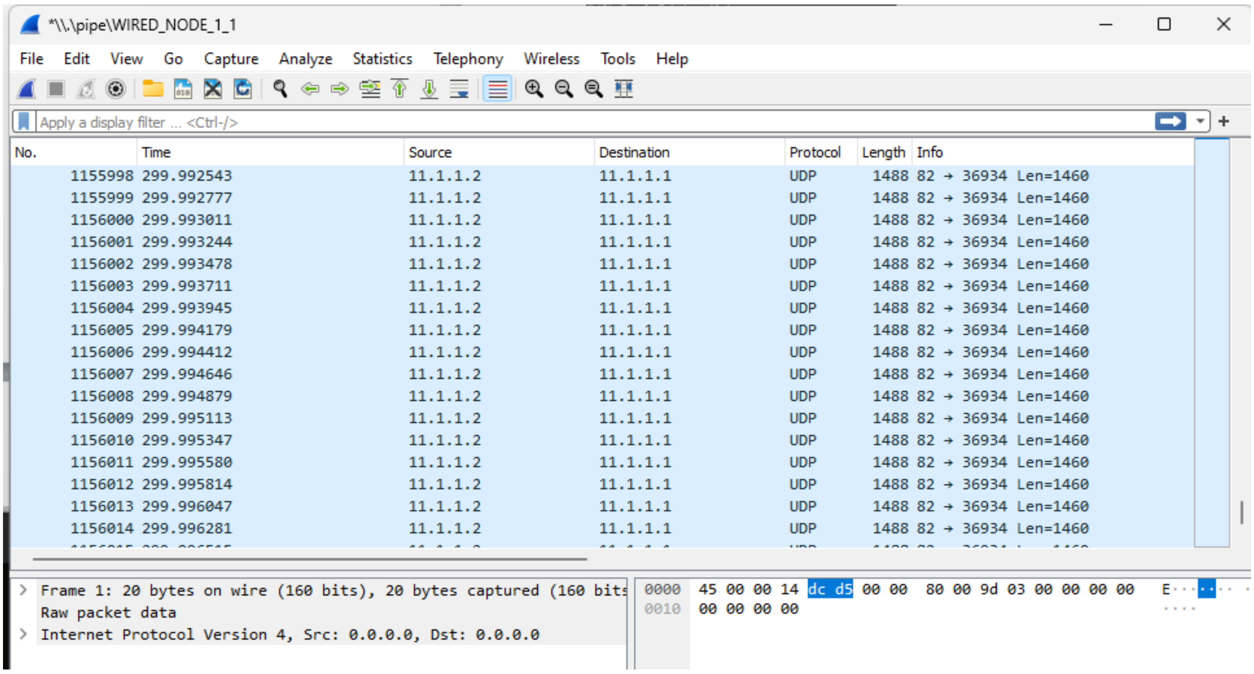
ping <NodeName> e.g. ping Wired_Node_2

and try ping to all devices

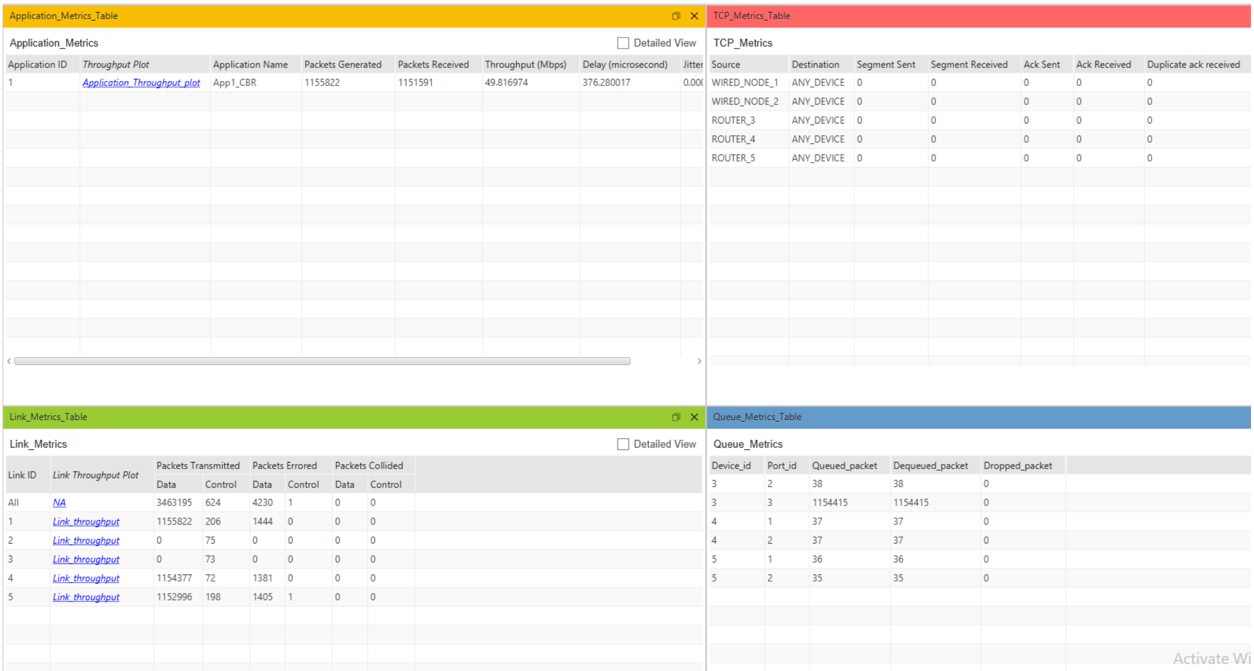


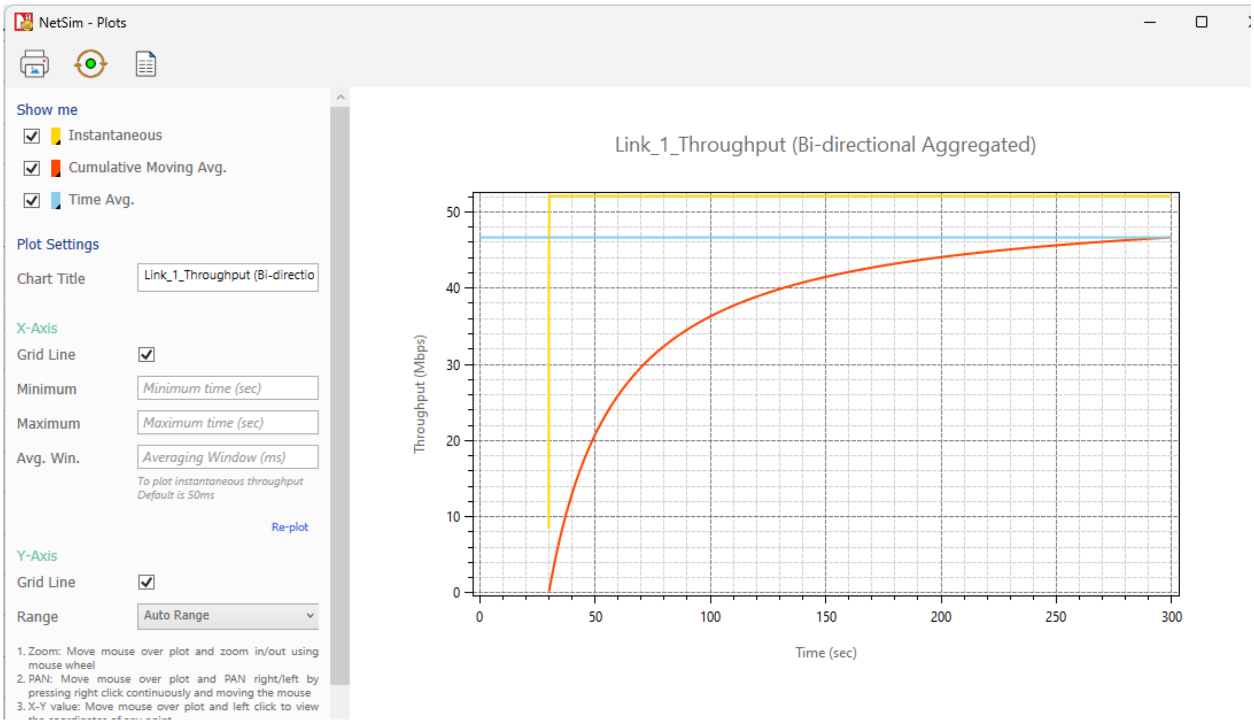
- **After simulation** Open packet trace and filter ICMP_EchoRequest and ICMP_EchoReply from CONTROL_PACKET_TYPE/APP_NAME column

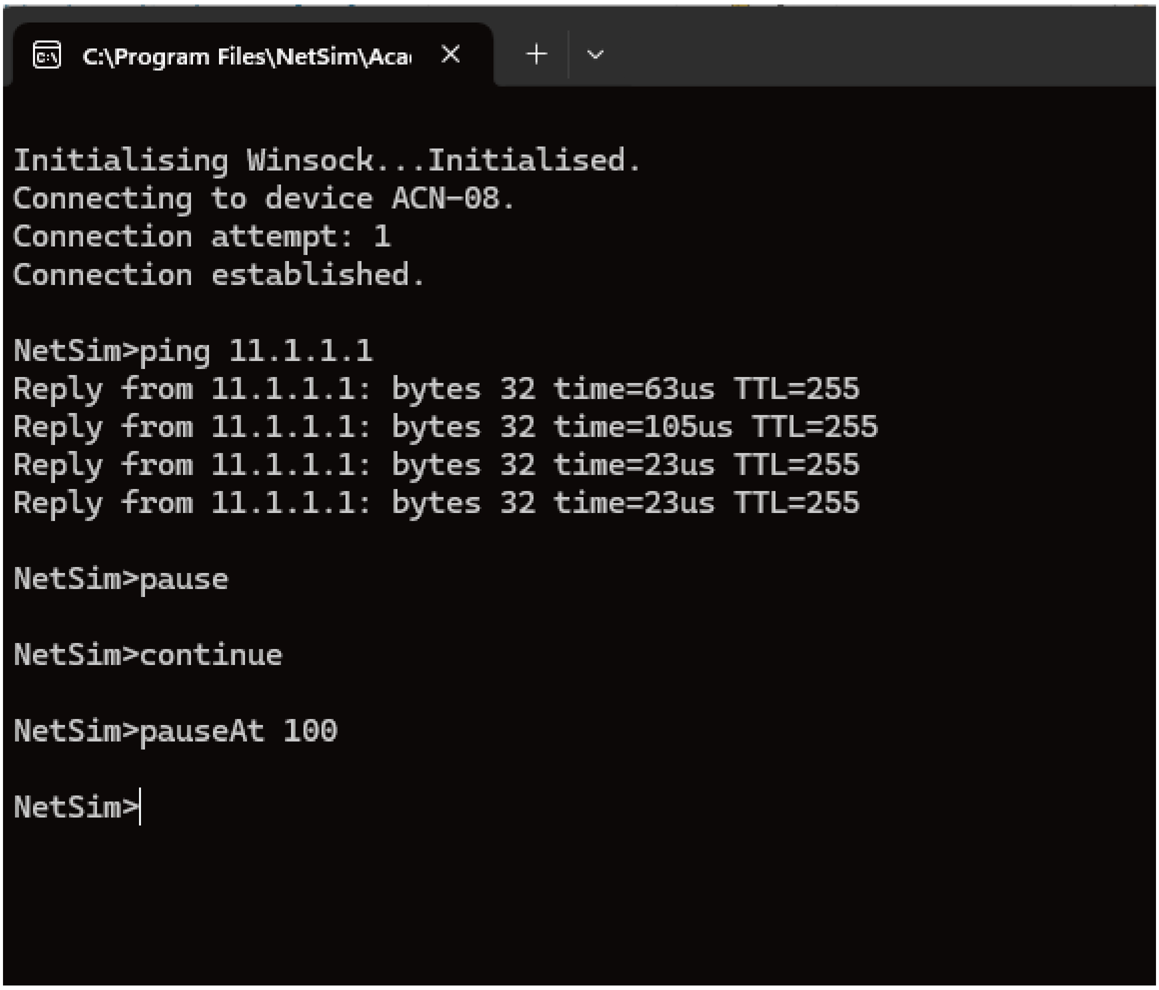
- Open Wireshark and apply filter ICMP. We can see the ping request and reply packets in Wireshark.



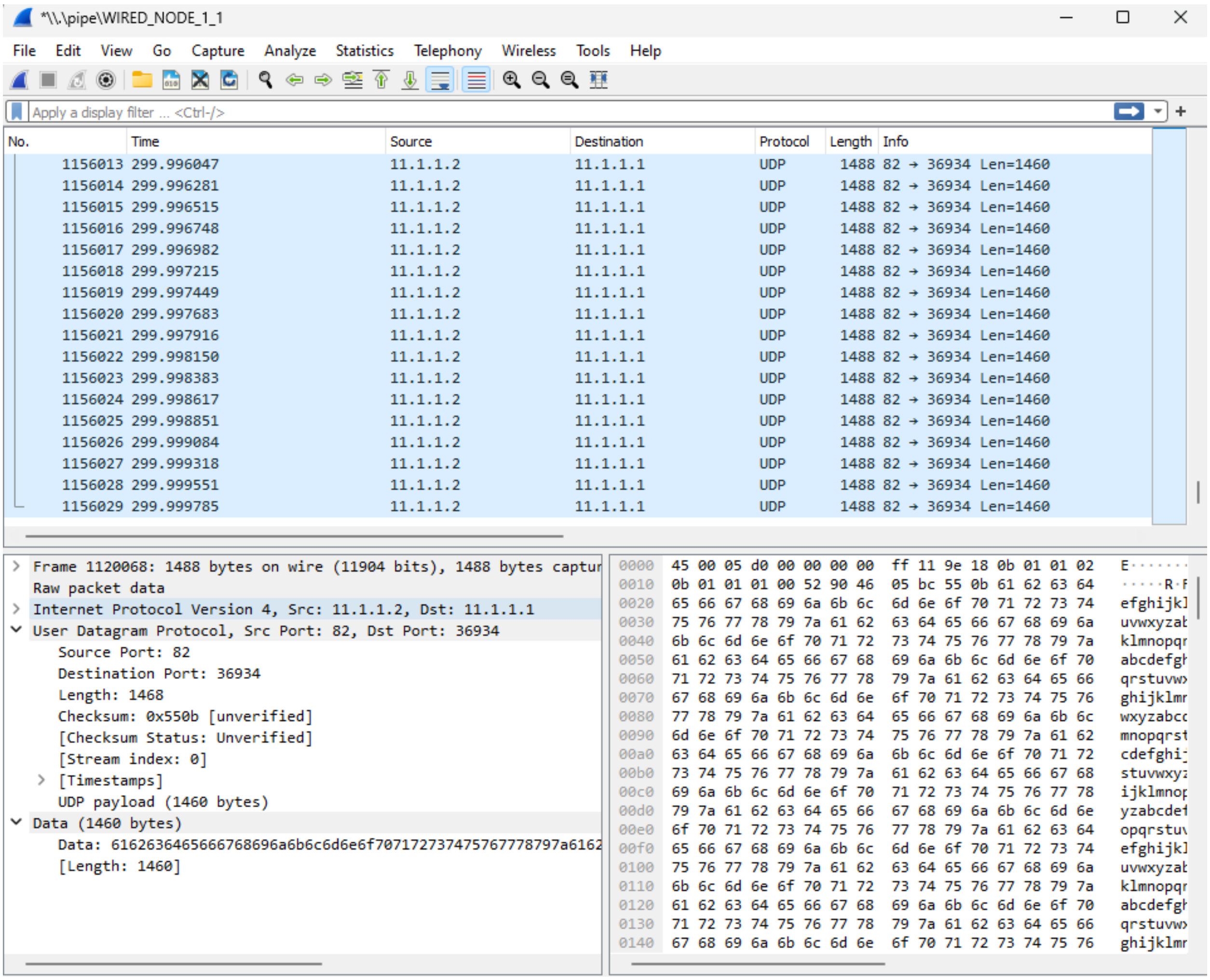
- After the running until 300^{th} sec plot screen pops up

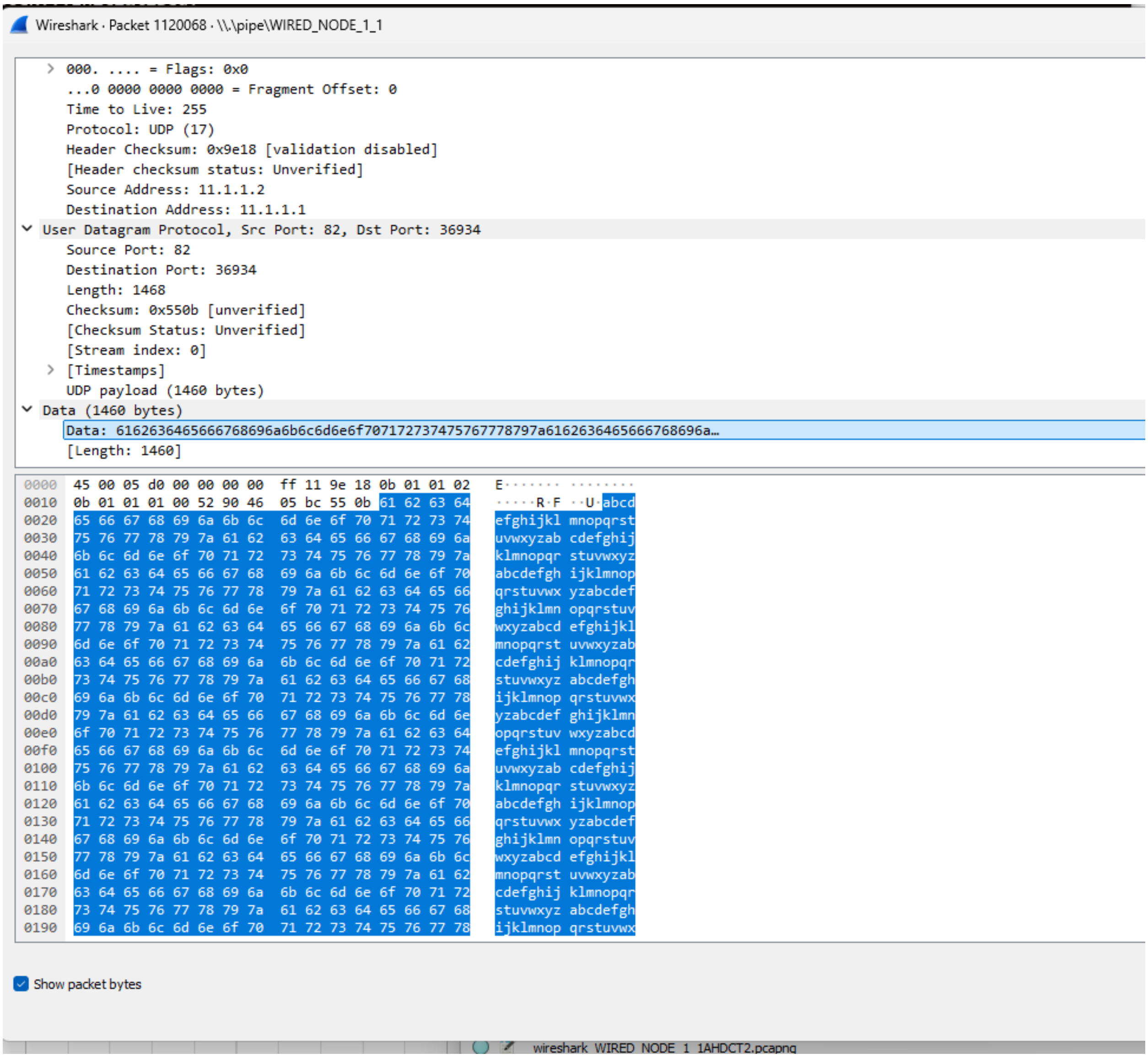






The program stops at 100 sec





Route Commands

- route print
- route delete
- route add

In order to view the entire contents of the IP routing table, use following commands **route print.**

ROUTER_3_Table

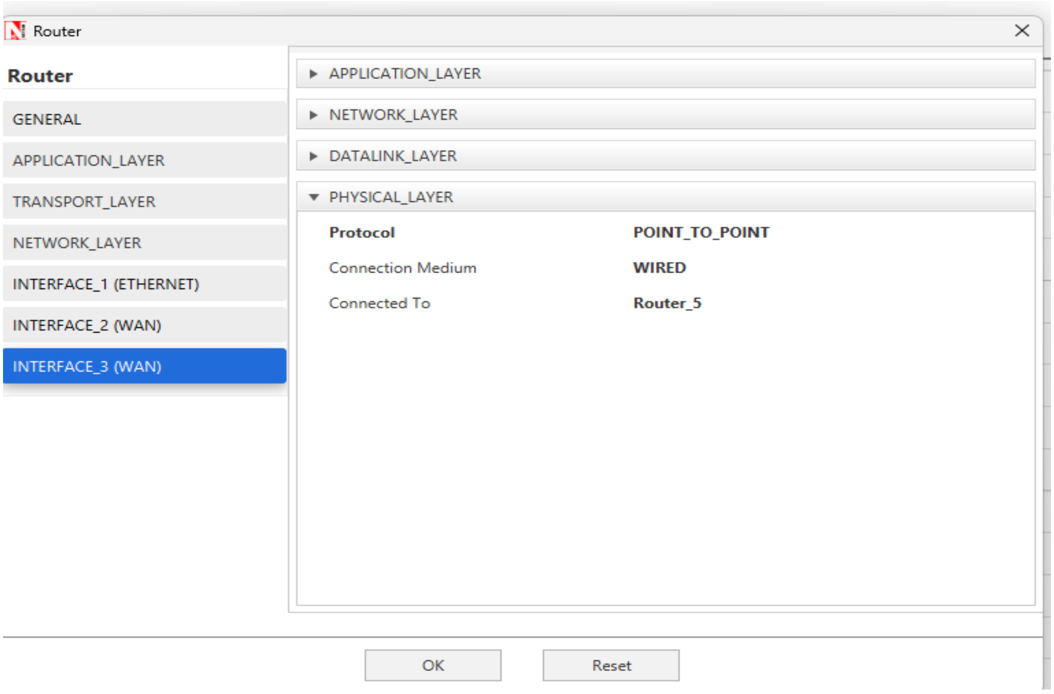
ROUTER_3

Detailed View

Network Destination	Netmask/Prefix len	Gateway	Interface
11.2.1.2	255.255.255.255	11.2.1.2	11.2.1.1
11.3.1.1	255.255.255.255	11.2.1.2	11.2.1.1
11.3.1.2	255.255.255.255	11.4.1.2	11.4.1.1
11.4.1.2	255.255.255.255	11.4.1.2	11.4.1.1
11.4.0.0	255.255.0.0	on-link	11.4.1.1
11.2.0.0	255.255.0.0	on-link	11.2.1.1
11.1.0.0	255.255.0.0	on-link	11.1.1.1
224.0.0.1	255.255.255.255	on-link	11.1.1.1 11.2.1.1 11.4.1.1
224.0.0.0	240.0.0.0	on-link	11.1.1.1 11.2.1.1 11.4.1.1
255.255.255.255	255.255.255.255	on-link	11.1.1.1

C:\Program Files\NetSim\Aca

```
Connection attempt: 1
Connection established.

NetSim>route print
===========================================

 IP Route Table
===========================================

           Network Destination  Netmask//Prefix                        Gateway                    Interface       Metric            Type
                      11.2.1.2  255.255.255.255                       11.2.1.2                     11.2.1.1          100            OSPF
                      11.3.1.1  255.255.255.255                       11.2.1.2                     11.2.1.1          100            OSPF
                      11.3.1.2  255.255.255.255                       11.4.1.2                     11.4.1.1          100            OSPF
                      11.4.1.2  255.255.255.255                       11.4.1.2                     11.4.1.1          100            OSPF
                      11.4.0.0      255.255.0.0                        on-link                     11.4.1.1          300           LOCAL
                      11.2.0.0      255.255.0.0                        on-link                     11.2.1.1          300           LOCAL
                      11.1.0.0      255.255.0.0                        on-link                     11.1.1.1          300           LOCAL
                     224.0.0.1  255.255.255.255                        on-link                     11.1.1.1          306       MULTICAST
                     224.0.0.0        240.0.0.0                        on-link                     11.1.1.1          306       MULTICAST
               255.255.255.255  255.255.255.255                        on-link                     11.1.1.1          999       BROADCAST
================================================
```



- **You will see the routing table entries with network destinations and the gateways to which packets are forwarded when they are headed to that destination. Unless you've already added static routes to the table, everything you see here will be dynamically generated.**

ACL Configuration

Routers provide basic traffic filtering capabilities, such as blocking Internet traffic, with access control lists (ACLs). An ACL is a sequential list of permit or deny statements that apply to addresses or upper-layer protocols. These lists tell the router what types of packets to: permit or deny. When using an access-list to filter traffic, a **permit** statement is used to "allow" traffic, while a **deny** statement is used to "block" traffic.

ACL Commands

- To view ACL syntax use: *acl print.*
- Before using ACL's, we must first verify that acl option enabled.
- A common way to enable ACL use command: *acl enable.*
- Enters configuration mode of ACL using: *aclconfig*
- To view ACL Table: *Print*
- To exit from ACL configuration use command: *exit*
- To disable ACL use command: *acl disable* (use this command > after exit from acl configuration)

To view ACL usage syntax use: **acl print**

[PERMIT, DENY] [INBOUND, OUTBOUND, BOTH] PROTO SRC DEST SPORT DPORT IFID

Step to Configure ACL#

- Create Network scenario as shown in below figure.

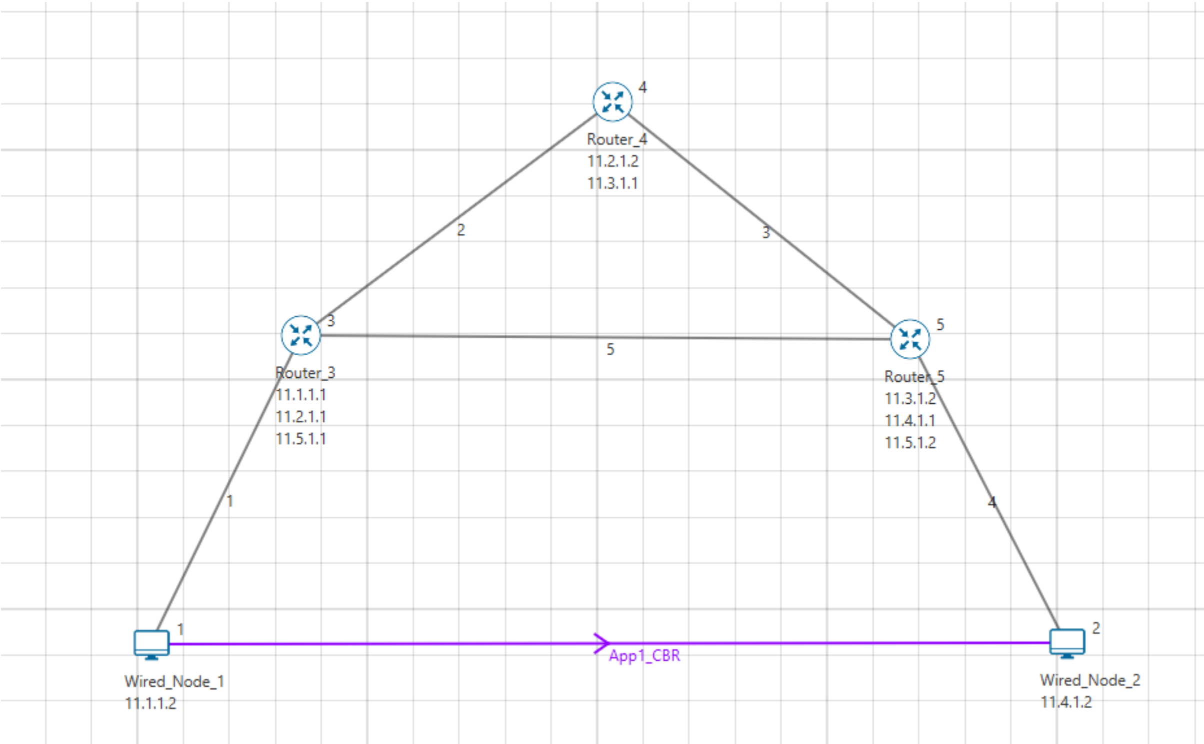


Figure: Network Scenario

- To create a new rule in the ACL use command as shown below to block UDP packet in Interface_3 of the Router_3.
- Click on the Application icon present in the top ribbon/toolbar.
- CBR Application from Wired Node 1 to Wired Node 2 with 10 MbpsGeneration Rate (Packet Size: 1460, Inter Arrival Time: 1168µs).
- Set Transport Protocol to UDP.
- Set Start Time as 30 Sec
- Click on run simulation option and In the Run time Interaction tab set Interactive Simulation as True and click on Accept.
- Set the Simulation Time as 200sec or more. Click Ok.
- Right click on Router_3 and select NetSim Console. Use the command as follows:

NetSim>acl enable

ACL is enable

NetSim>aclconfig

ROUTER_3/ACLCONFIG>acl print

Usage: [PERMIT, DENY] [INBOUND, OUTBOUND, BOTH] PROTO SRC DEST SPORT DPORT IFID

ROUTER_3/ACLCONFIG\>DENY BOTH UDP ANY ANY 0 0 3

OK!

ROUTER_3/ACLCONFIG\>print

DENY BOTH UDP ANY/0 ANY/0 0 0 3

ROUTER_3/ACLCONFIG\>exit

NetSim>acl disable

ACL is disable

NetSim>

Results

The impact of ACL rule applied over the simulation traffic can be observed in the **IP_Metrics_Table** in the simulation results window, In Router_3 number of packets blocked by firewall can be seen in the table.

- *[Note: Results will vary based on time of ACL command are executed].*

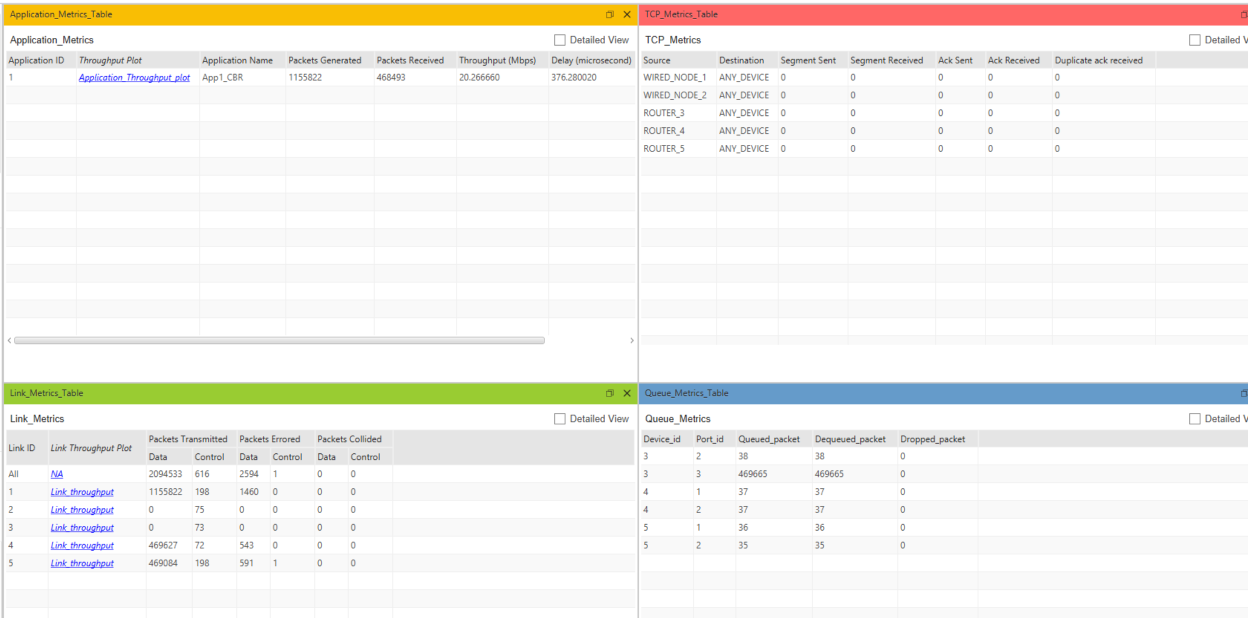
```
NetSim>acl enable
ACL is enable

NetSim>aclconfig

ROUTER_3/ACLCONFIG>acl print
Usage: [PERMIT,DENY] [INBOUND,OUTBOUND,BOTH] PROTO SRC DEST SPORT DPORT IFID

ROUTER_3/ACLCONFIG>deny both udp any any 0 0 3
OK!
ROUTER_3/ACLCONFIG>
```

```
ROUTER_3/ACLCONFIG>deny both udp any any 0 0 3
OK!
ROUTER_3/ACLCONFIG>exit

NetSim>acl disable
ACL is disable

NetSim>
```

IP_Metrics_Table

IP_Metrics — Detailed View

Device Id	Packet sent	Packet forwarded	Packet received	Packet discarded	TTL expired	Firewall blocked
1	1155921	0	0	0	0	0
2	99	0	468493	0	0	0
3	469802	1154362	72	0	0	684734
4	74	0	74	0	0	0
5	469254	469084	74	0	0	0



Application_Metrics_Table

Application_Metrics

Application ID	Throughput Plot	Application Name	Packets Generated	Packets Received	Throughput (Mbps)	Delay (microsecond)
1	Application_Throughput_plot	App1_CBR	1155822	468493	20.266660	376.280020

TCP_Metrics_Table

TCP_Metrics

Source	Destination	Segment Sent	Segment Received	Ack Sent	Ack Received	Duplicate ack received
WIRED_NODE_1	ANY_DEVICE	0	0	0	0	0
WIRED_NODE_2	ANY_DEVICE	0	0	0	0	0
ROUTER_3	ANY_DEVICE	0	0	0	0	0
ROUTER_4	ANY_DEVICE	0	0	0	0	0
ROUTER_5	ANY_DEVICE	0	0	0	0	0

Link_Metrics_Table

Link_Metrics

Link ID	Link Throughput Plot	Packets Transmitted		Packets Errored		Packets Collided	
		Data	Control	Data	Control	Data	Control
All	NA	2094533	616	2594	1	0	0
1	Link_throughput	1155822	198	1460	0	0	0
2	Link_throughput	0	75	0	0	0	0
3	Link_throughput	0	73	0	0	0	0
4	Link_throughput	469627	72	543	0	0	0
5	Link_throughput	469084	198	591	1	0	0

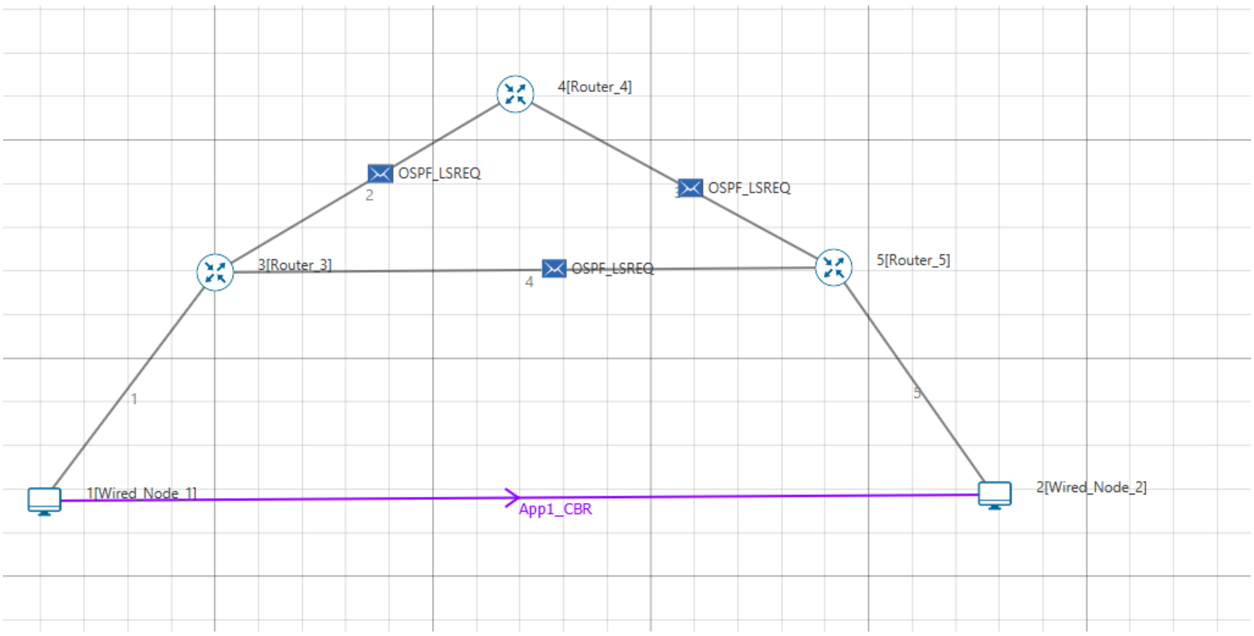
Queue_Metrics_Table

Queue_Metrics

Device_id	Port_id	Queued_packet	Dequeued_packet	Dropped_packet
3	2	38	38	0
3	3	469665	469665	0
4	1	37	37	0
4	2	37	37	0
5	1	36	36	0
5	2	35	35	0

- The impact of ACL rule applied over the simulation traffic can be observed in the Application throughput plot. Throughput graph will show a drop after ACL is set. If ACL is disabled after a while, application packets will start flowing across the router. The Application throughput plot will show a drop and increase in throughput after setting ACL and disabling ACL respectively.

FINAL CIRCUIT PATH:



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	PACKET	SEGME	PACKET	CONTR	SOURC	DESTIN	TRANS	RECEIV	APP_LA	TRX_LA	NW_LA	MAC_L	PHY_LA	PHY_LA	PHY_LA
106	1	0	CBR	App1_CBR	NODE-1	NODE-2	NODE-1	ROUTER-3	30000000	30000000	30000000	30000000	30000005	30000126	30000131
107	1	0	CBR	App1_CBR	NODE-1	NODE-2	ROUTER-3	ROUTER-5	30000000	30000000	30000131	30000131	30000131	30000250	30000255
109	1	0	CBR	App1_CBR	NODE-1	NODE-2	ROUTER-5	NODE-2	30000000	30000000	30000255	30000255	30000255	30000377	30000382

	P	Q	R	S	T	U	V	W	X	Y	Z	AA	AB	AC
1	APP_LA	TRX_LA	NW_LA	MAC_L	PHY_LA	PHY_LA	PACKET	LOCAL_	REMOT	CWND	SEQ_N	ACK_N	isSyn	isAck
106	1460	1460	1468	1488	1514	0	Successfu	N/A	N/A	N/A	N/A	N/A	N/A	N/A
107	1460	1460	1468	1488	1488	0	Successfu	N/A	N/A	N/A	N/A	N/A	N/A	N/A
109	1460	1460	1468	1488	1514	0	Successfu	N/A	N/A	N/A	N/A	N/A	N/A	N/A

When did you see the drop in throughput in the graph, since router blocks UDP packets in the plot?

After applying the ACL rule to block UDP packets on Router_3, we observe a drop in throughput in the Application throughput plot. This drop occurs around the time when the ACL rule is set and takes effect after 200 sec.

What time you see the increase in throughput when router permitted packets?

The average increase in the throughput is 20 ms compared to the running of the circuit before and after applying acl commands.

Throughput values: 49.8 milli sec, 20.66 milli sec

Packets received: 1151591, 1151589

Experiment No: 6 **Date: 24/09/24**

Study and Analyze the characteristic curve of throughput versus offered traffic for PURE ALOHA and SLOTTED ALOHA

Name : Thurlapati Sai Sree Praneetha
Reg. No : 22BLC1350
Faculty Name : Jayavignesh T

NOTE: NetSim Academic supports a maximum of 100 nodes and hence this experiment can only be limited to maximum of 100 nodes Aloha
networks

6.1 Theory

ALOHA provides a wireless data network. It is a multiple access protocol (this protocol is for allocating a multiple access channel). There are two main versions of ALOHA: pure and slotted. They differ with respect to whether time is divided up into discrete slots into which all frames must fit.

6.2 Pure Aloha

In Pure Aloha, users transmit whenever they have data to be sent. There will be collisions and the colliding frames will then be retransmitted. In NetSim's Aloha library, the sender waits a random amount of time per the exponential back-off algorithm and sends it again. The frame is discarded when the number of collisions a packet experiences crosses the “Retry Limit” - a user settable parameter in the GUI.

Let ‘‘frame time’’ denotes the amount of time needed to transmit the standard, fixed-length frame. In this experiment point, we assume that the new frames generated by the stations are modeled by a Poisson distribution with a mean of N frames per frame time. If N > 1, the nodes are generating frames at a higher rate than the channel can handle, and nearly every frame will suffer a collision. For reasonable throughput, we would expect 0 < N < 1. In addition to the new frames, the stations also generate retransmissions of frames that previously suffered collisions.

The probability of no other traffic being initiated during the entire vulnerable period is given by e^{-2G} which leads to $S = G \times e^{-2G}$ where, S is the throughput and G is the offered load. The units of S and G is frames per frame time.

G is the mean of the Poisson distribution followed by the transmission attempts per frame time, old and new combined. Old frames mean those frames that have previously suffered collisions.

The maximum throughput occurs at $G = 0.5$, with $S = \frac{1}{2e}$, which is about 0.184. In other words, the best we can hope for is a channel utilization of 18%. This result is not very encouraging, but with everyone transmitting at will, we could hardly have expected a 100% success rate.

6.3 Slotted Aloha

In slotted Aloha, time is divided up into discrete intervals, each interval corresponding to one frame. In Slotted Aloha, a node is required to wait for the beginning of the next slot in order to send the next packet. The probability of no other traffic being initiated during the entire vulnerable period is given by e^{-G} which leads to $S = G \times e^{-G}$. It is easy to compute that Slotted Aloha peaks at G = 1, with a throughput of $s = \frac{1}{e}$ or about 0.368.

6.4 Offered load and throughput calculations

Using NetSim, the attempts per packet time (G) can be calculated as follows.

$$G = \frac{Number\ of\ packets\ transmitted \times PacketTime(s)}{SimulationTime\ (s)}$$

where, G is Attempts per packet time. We derive the above formula keeping in mind that (i) NetSim's output metric, the number of packets transmitted, is nothing but the number of attempts, and (ii) since packets transmitted is computed over the entire simulation time, the number of "packet times" would be $\frac{SimulationTime(s)}{PacketTransmissionTime(s)}$, which is in the denominator. Note that in NetSim the output metric Packets transmitted is counted at link (PHY layer) level. Hence MAC layer re-tries are factored into this metric.

The throughput (in Mbps) per packet time can be obtained as follows.

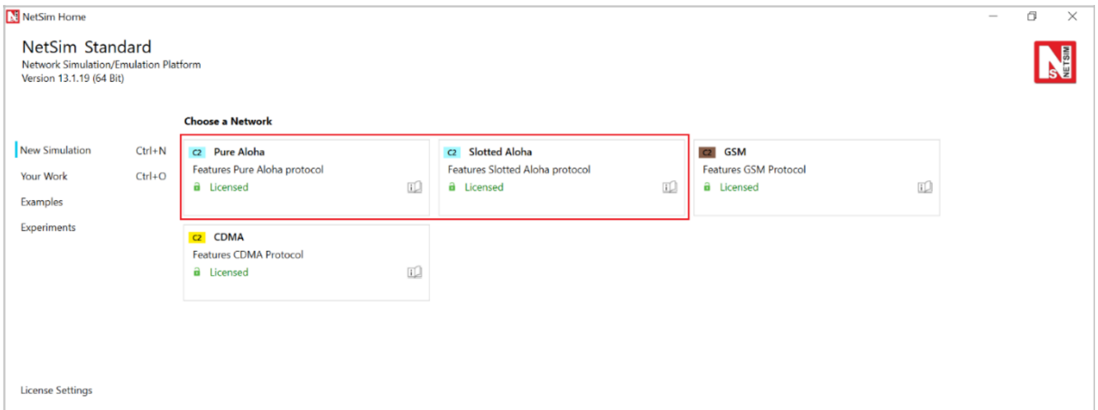
$$S = \frac{Number\ of\ packets\ successful \times PacketTime(s)}{SimulationTime\ (s)}$$

where, S = Throughput per packet time. In case of slotted aloha packet (transmission) time is equal to slot length (time). The packet transmission time is the PHY layer packet size in bits divided by the PHY rate in bits/s. Considering the PHY layer packet size as 1500B, and the PHY rate as 10 Mbps, the packet transmission time (or packet time) would be $\frac{1500\times 8}{10\times 10^6} = 1200\ \mu s$.

In the following experiment, we have taken packet size as 1460 B (Data Size) plus 28 B (Overheads) which equals 1488 B. The PHY data rate is 10 Mbps and hence packet time is equal to 1.2 milliseconds.

6.5 Network Set Up

In the Main menu select□□**New Simulation**□□ **Legacy Network**□□**Pure Aloha/ Slotted Aloha** as shown below



Create Scenario

Legacy networks come with a palette of various devices like wireless node,

Click and drop into environment

- **Click** on the **Node** icon in the tool bar and click and drop inside the grid. (*Note*: This is applicable for Pure Aloha and Slotted Aloha)

- Similarly drop **Adhoc link** and connect **Wireless Nodes** to **Adhoc links** using **Adhoc links**. (Note: A Node cannot be placed on another Node. A Node cannot float outside of the grid).

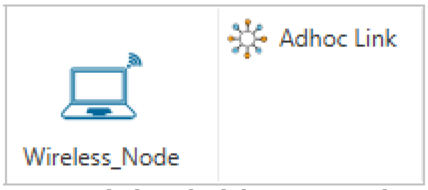
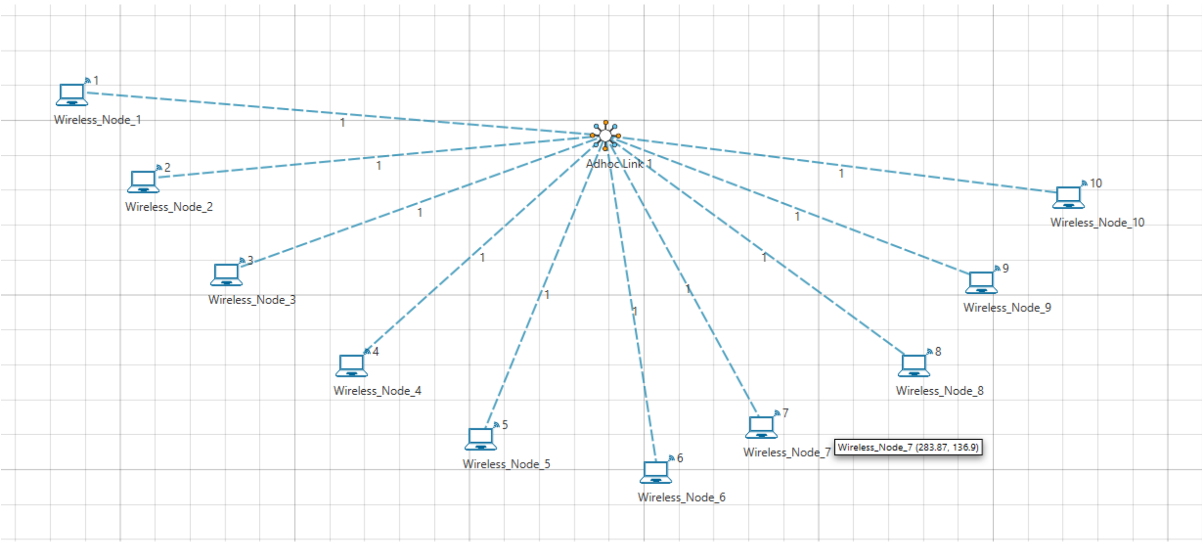


Figure 2: Pure and Slotted Aloha Device Palette in GUI

Pure Aloha: Input for 10-Nodes sample

Step 1: Drop 10 nodes (i.e., 9 Nodes are generating traffic.)



Node 2, 3, 4, 5, 6, 7, 8, 9, and 10 generates traffic. The properties of Nodes 2, 3, 4, 5, 6, 7, 8, 9, and 10 which transmits data to Node 1 are given in the below table.

Step 2: Wireless Node Properties:

Wireless Node Properties	
Interface1_Wireless (PHYSICAL_LAYER)	
Data Rate (Mbps)	10
Interface1_Wireless (DATALINK_LAYER)	
Retry_Limit	0
MAC_Buffer	FALSE
Slot Length(µs)	1200

Table 1-2: Wireless Node Properties

(**NOTE:** Slot Length(µs) parameter present only in Slotted Aloha → Wireless Node Properties → Interface_1 (Wireless))

Step 3: In Adhoc Link Properties, channel characteristics is set as **No Path Loss**.

Step 4: Application Properties:

- Right click on the Application Flow "App1 CUSTOM" and select Properties or click on the Application icon present in the top ribbon/toolbar. The properties are set according to the values given in the below Table 1-3.

Application_1 Properties		
Application Method	Unicast	
Application Type	Custom	
Source_Id	2	
Destination_Id	1	
Transport Protocol	UDP	
Packet Size	Distribution	Constant
	Value (Bytes)	1460
Inter Arrival Time	Distribution	Exponential
	Packet Inter Arrival Time (µs)	200000

Table 1-3: For Application_1 Properties

- Similarly create 8 more application, i.e., Source_Id as 3, 4, 5, 6, 7, 8, 9, 10 and Destination_Id as 1, set Packet Size and Inter Arrival Time as shown in above table.

Step 5: Plots are enabled in NetSim GUI.

Step 6: Simulation Time- 10 Seconds

NOTE: Obtain the values of Total Number of Packets Transmitted and Collided from the results window of NetSim.

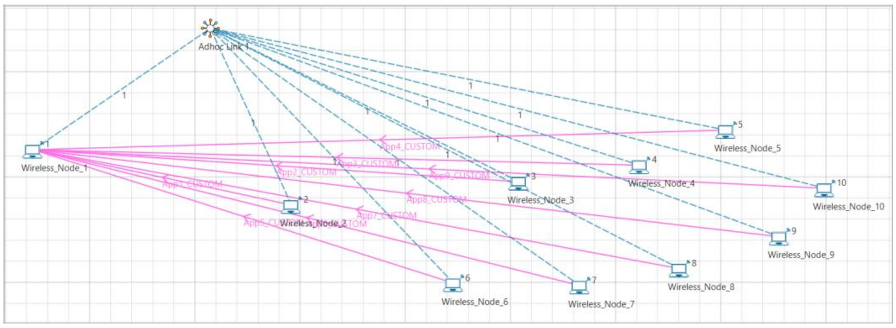
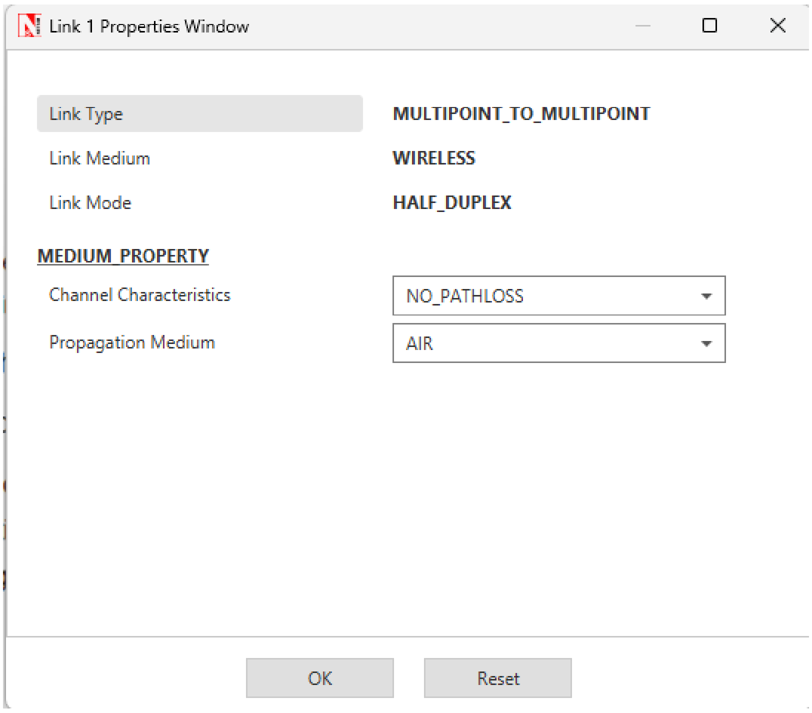
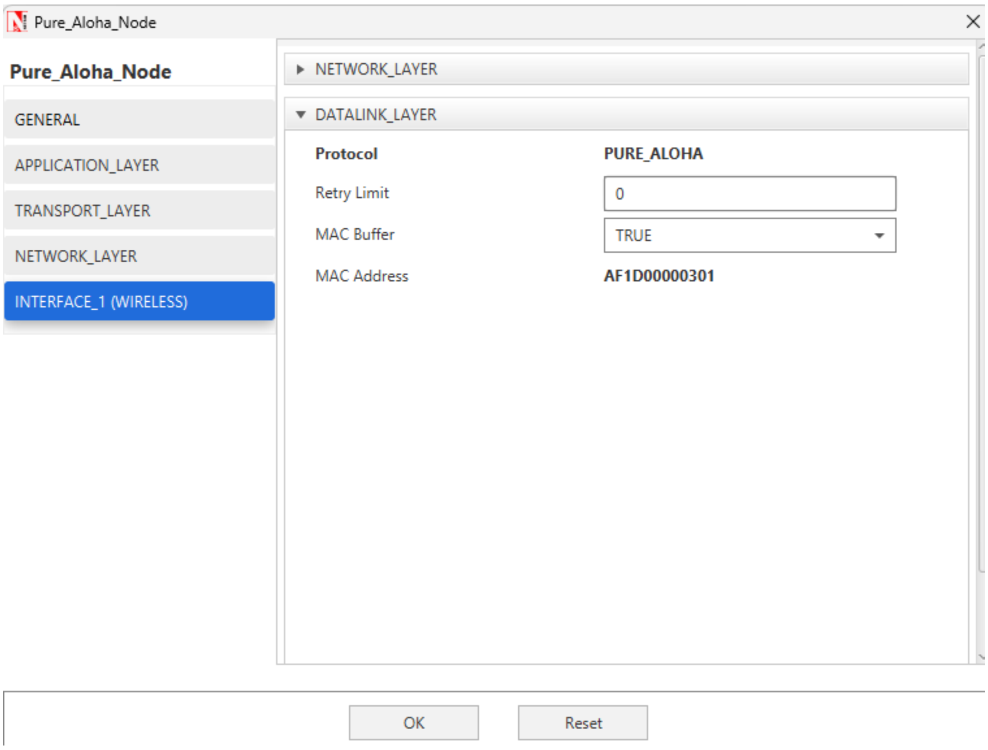


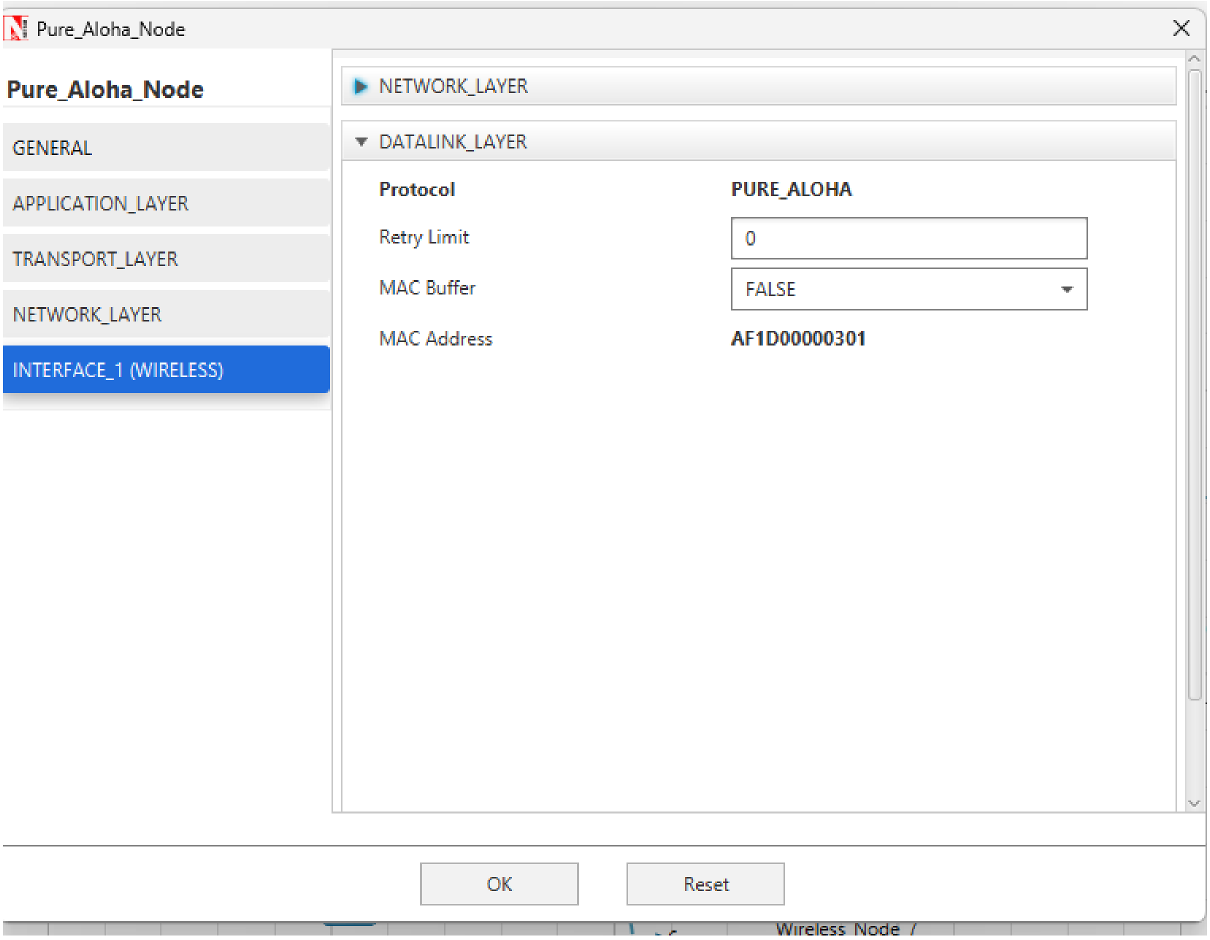
Figure: Network set up for studying the Pure aloha

After setting up the network, change the required properties in all the nodes

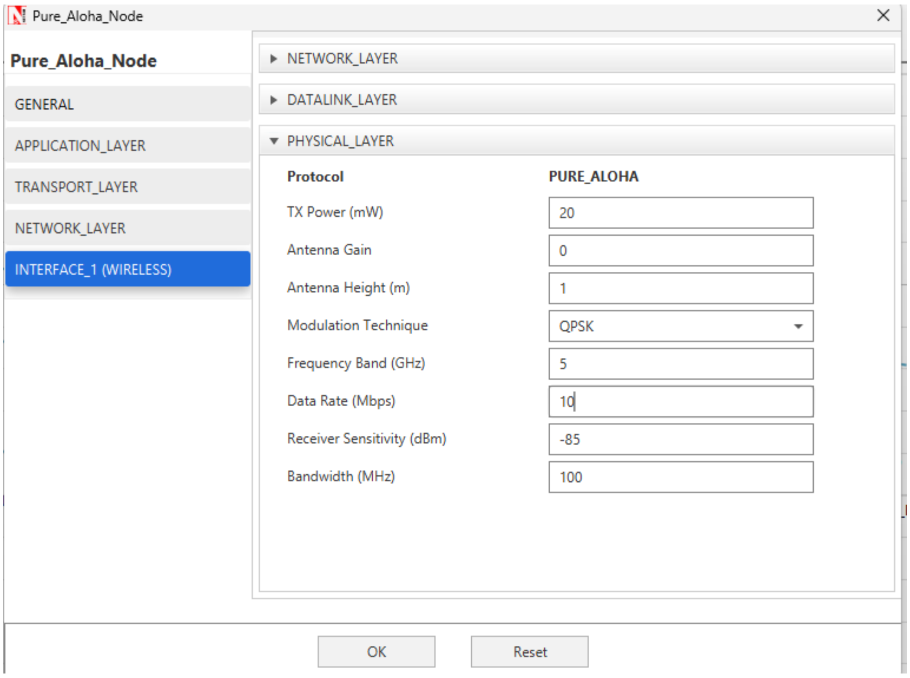




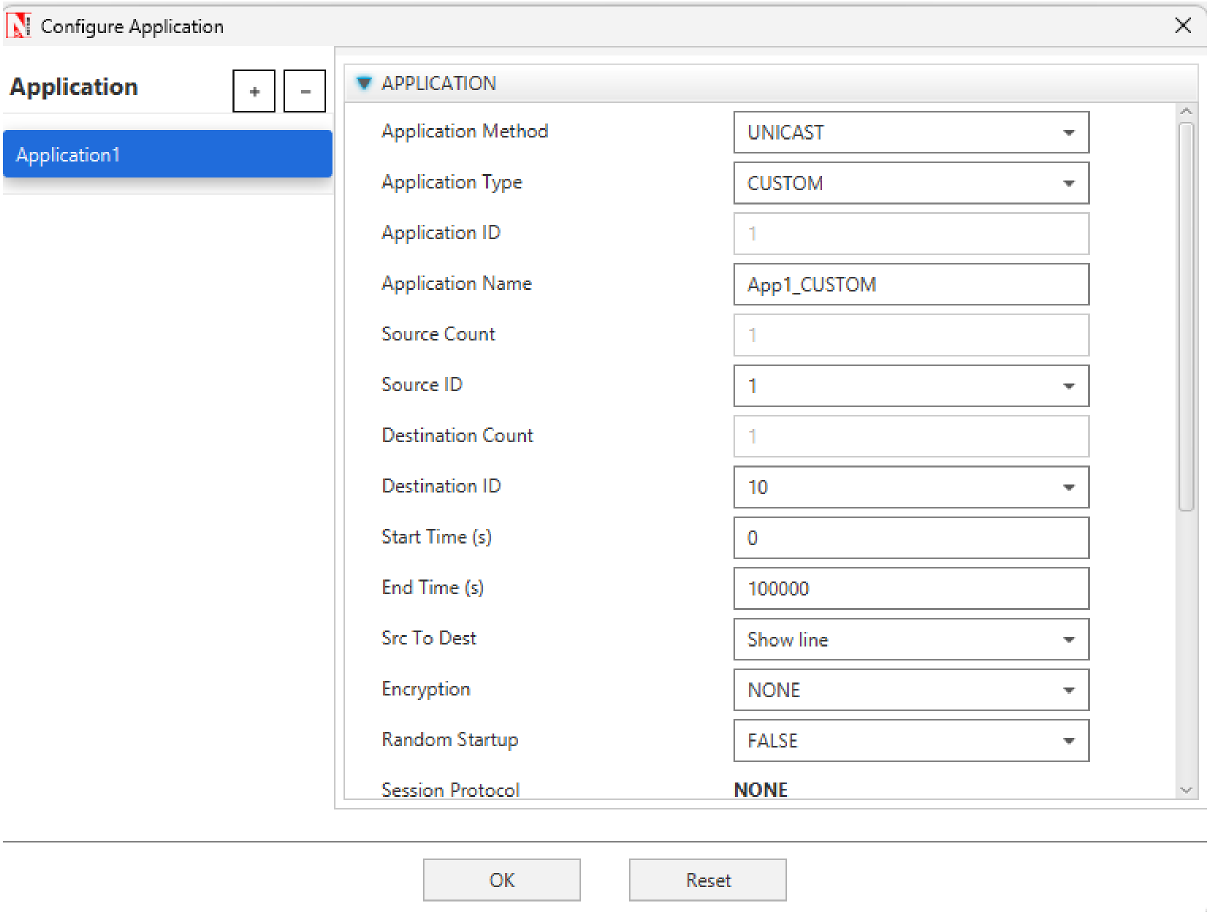
In Data link layer MAC buffer is changed to false

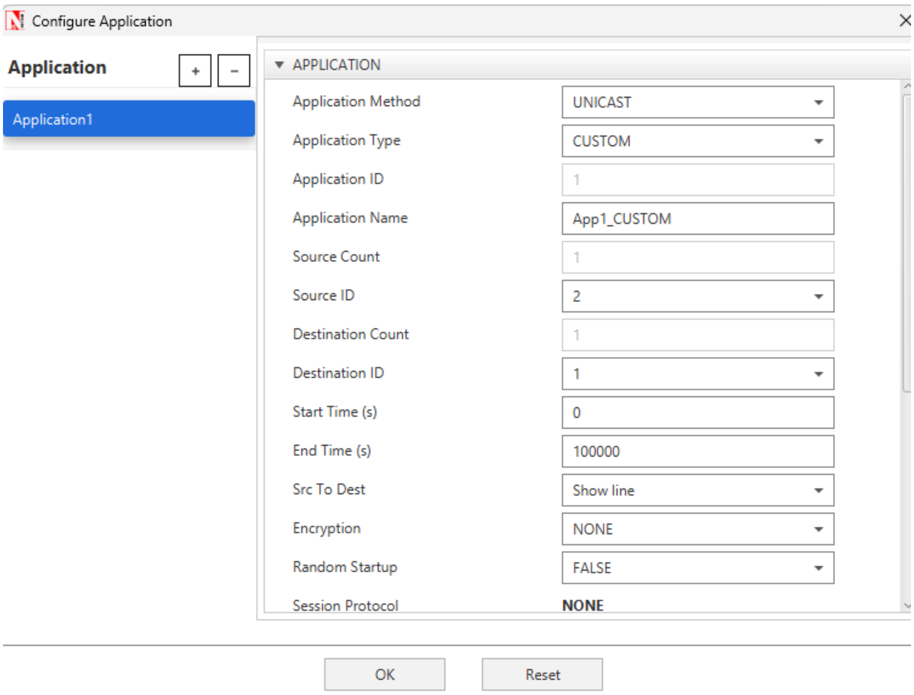


Change the data rate to 10mbps in all the wireless nodes

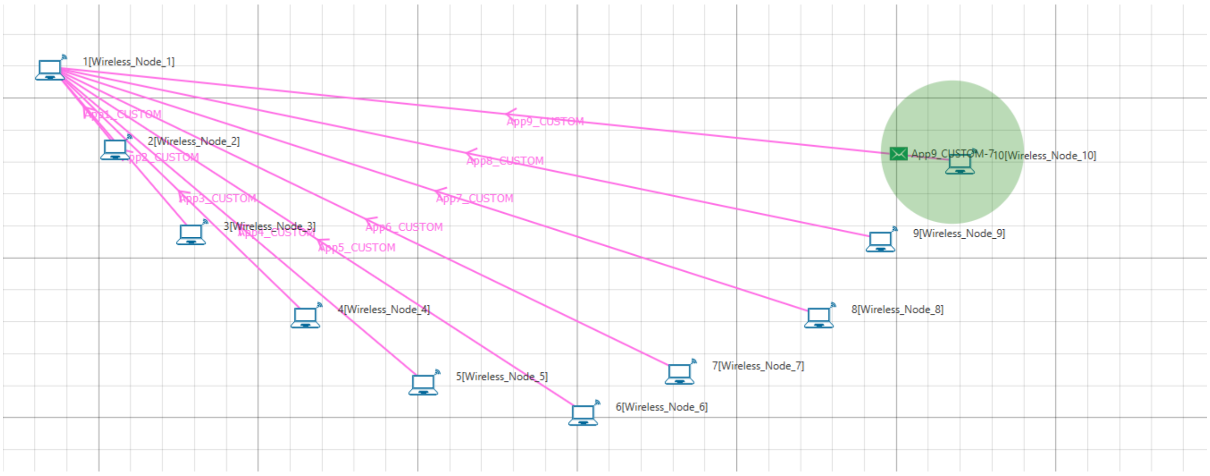


Go to application and choose custom application type, source id = 2, destination = 1





After running it for 10 sec



Link_Metrics_Table

Link_Metrics ☐ Detailed View

Link ID	Link Throughput Plot	Packets Transmit...		Packets Errored		Packets Collided	
		Data	Control	Data	Control	Data	Control
All	NA	494	0	0	0	60	0
1	Link throughput	494	0	0	0	60	0

Input for 20-Nodes sample

Step 1: Drop 20 nodes (i.e., 19 Nodes are generating traffic.)

Nodes 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, and 20 transmit data to Node 1.

Continue the experiment by increasing the number of nodes generating traffic as 29, 39, **49**, **59**, 69, **79, 89, 99** nodes

Slotted ALOHA: Input for 10-Nodes sample

Similarly repeat the experiment for Slotted Aloha with SLOT LENGTH 1200us

PLOT THE COMPARISON TABLE Output and plot graph for both the protocols

Comparison Table: The values of Total Number of Packets Transmitted and Collided obtained from the network statistics after running NetSim simulation are provided in the table below along with

Throughput per packet time& Number of Packets Transmitted per packet time. The table should have the following tabulated

Number of nodes generating traffic	Total number of Packets Transmitted	Total number of Packets Collided	Successful Packets (Packets Transmitted -Packets Collided)	Attempts per packet time(G)	Throughput per packet time(S)	Throughput per packet time. Theoretical (S = $G * e^{-2G}$)
9	494	60	434	0.059	0.052	0.052
19	978	187	791	0.011	0.094	0.092
29	1482	415	1067	0.017	0.01	0.012
59	2907	1429	1478	0.34	0.177	0.173
79	3964	2377	1587	0.47	0.19	0.183
99	4998	3468	1530	0.59	0.18	0.1807

Attempts per packet time (G) = (Total number of Packet Transmitted * 1.2ms) / 10

For 1ST iteration: 494 * 1.2/10 = 0.05928

Throughput per packet time (S) = (Total number of Successful packet transmitted * 1.2ms) / 10

For 1ST iteration: 434 * 1.2/10 = 0.052

FOR 20 NODES:

Link_Metrics_Table

Link_Metrics ☐ Detailed View

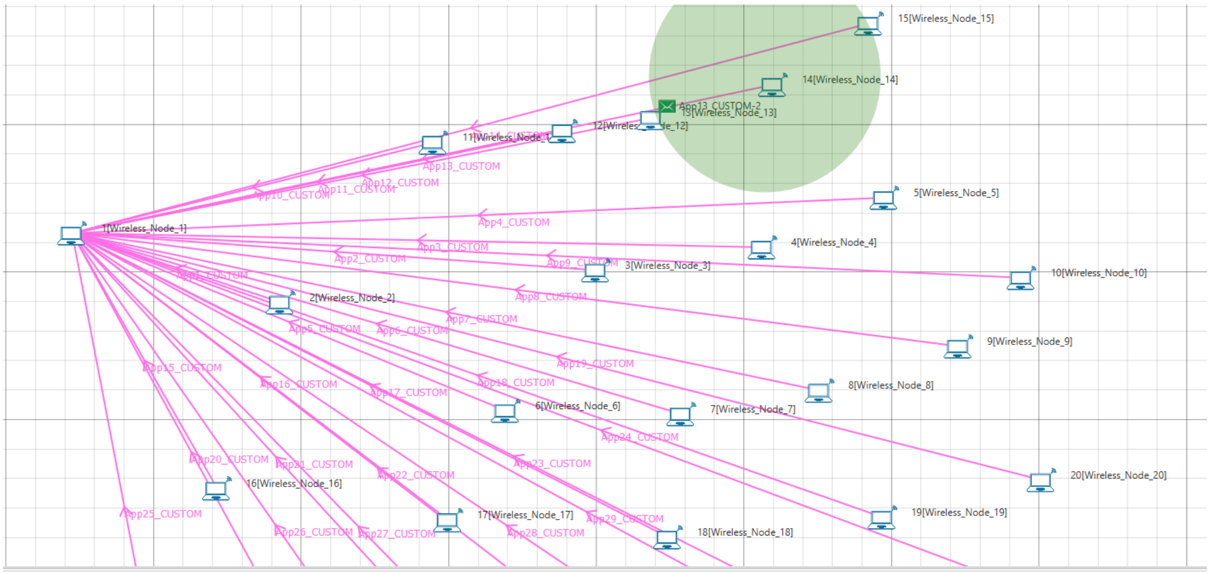
Link ID	Link Throughput Plot	Packets Transmit...		Packets Errored		Packets Collided	
		Data	Control	Data	Control	Data	Control
All	NA	978	0	0	0	187	0
1	Link_throughput	978	0	0	0	187	0

FOR 30 NODES:

Link_Metrics_Table

Link_Metrics ☐ Detailed View

Link ID	Link Throughput Plot	Packets Transmit...		Packets Errored		Packets Collided	
		Data	Control	Data	Control	Data	Control
All	NA	1482	0	0	0	415	0
1	Link_throughput	1482	0	0	0	415	0



FOR 60 NODES:

Link_Metrics_Table

Link_Metrics

☐ Detailed View

Link ID	Link Throughput Plot	Packets Transmit...		Packets Errored		Packets Collided	
		Data	Control	Data	Control	Data	Control
All	NA	2907	0	0	0	1429	0
1	Link throughput	2907	0	0	0	1429	0

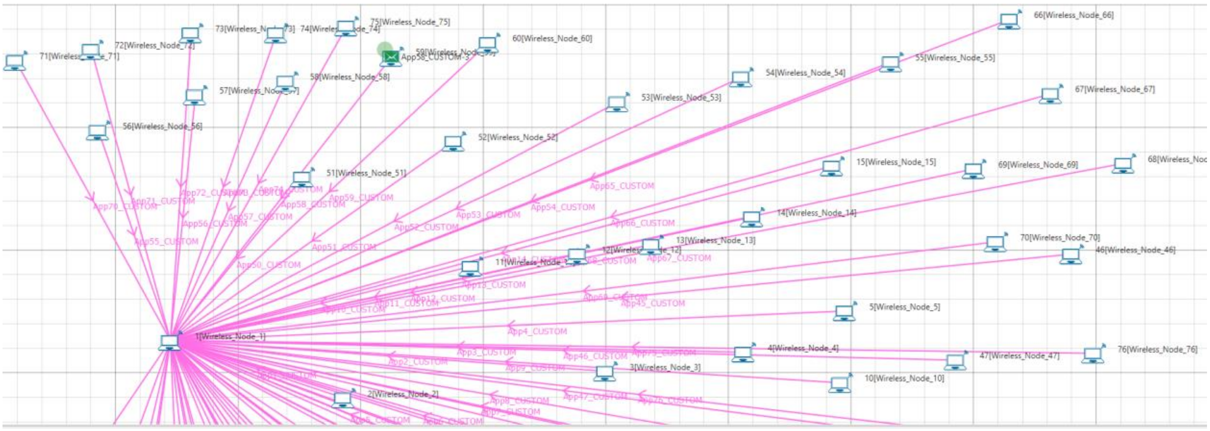
FOR 80 NODES:

Link_Metrics_Table

Link_Metrics

☐ Detailed View

Link ID	Link Throughput Plot	Packets Transmit...		Packets Errored		Packets Collided	
		Data	Control	Data	Control	Data	Control
All	NA	3964	0	0	0	2377	0
1	Link throughput	3964	0	0	0	2377	0



FOR 100 NODES:

Link_Metrics_Table

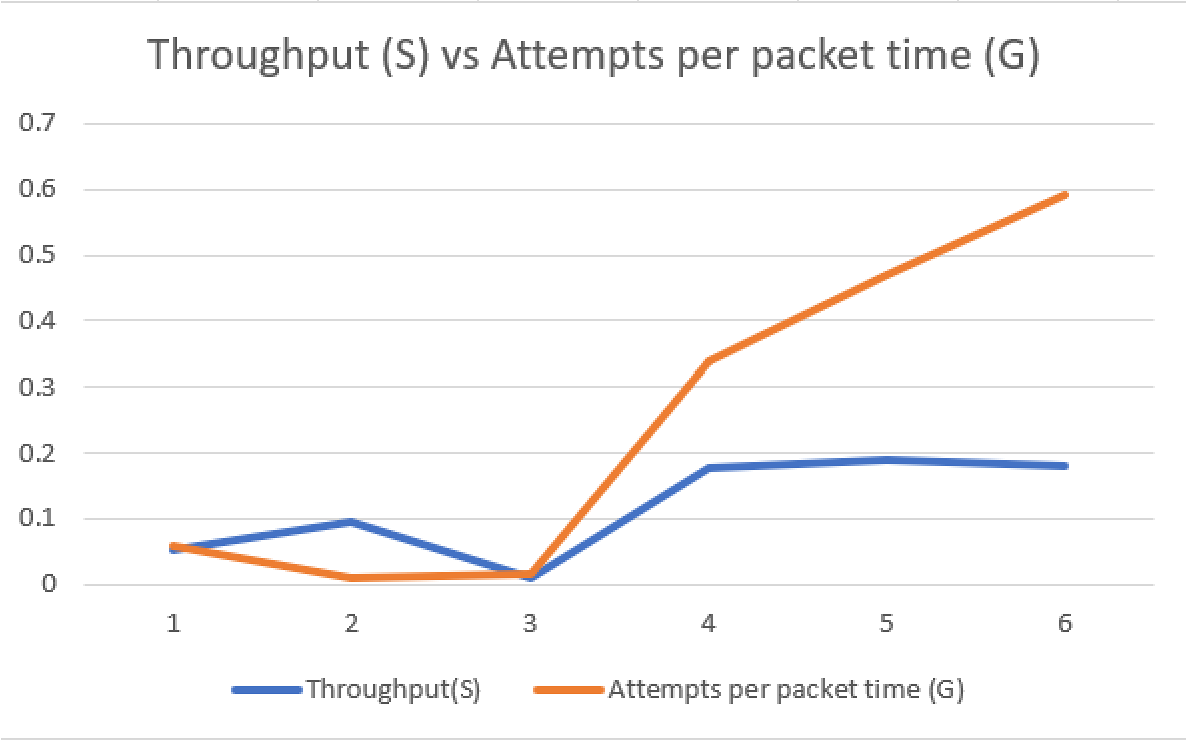
Link_Metrics

☐ Detailed View

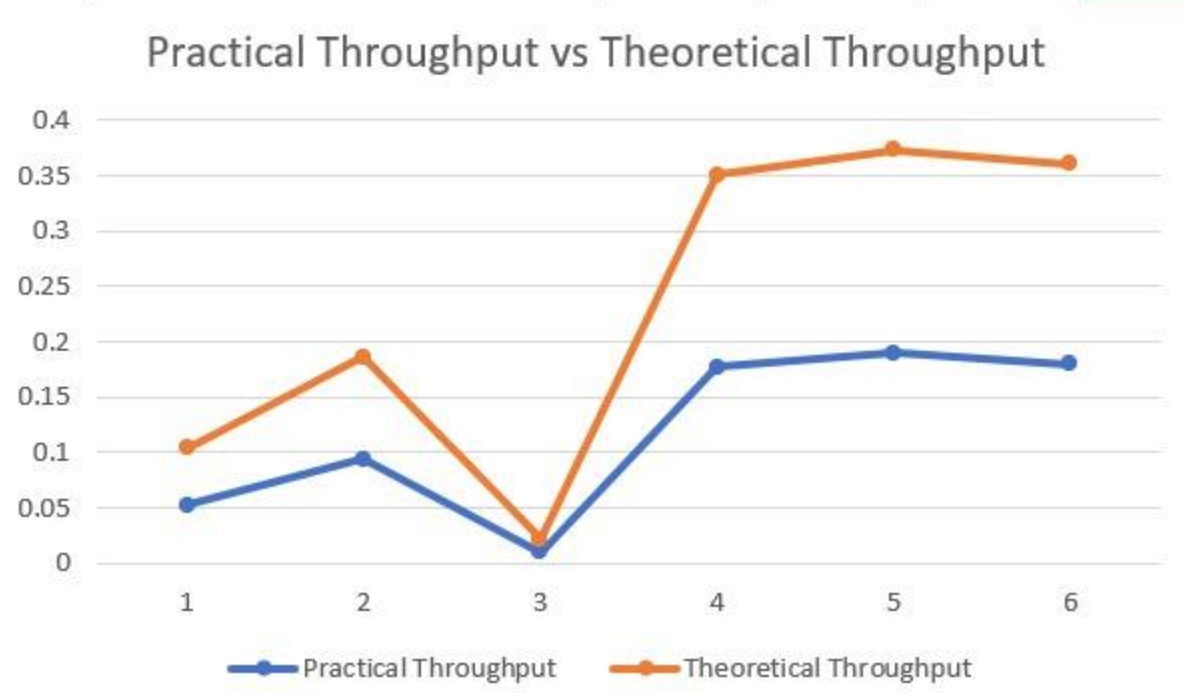
Link ID	Link Throughput Plot	Packets Transmit...		Packets Errored		Packets Collided	
		Data	Control	Data	Control	Data	Control
All	NA	4998	0	0	0	3468	0
1	Link_throughput	4998	0	0	0	3468	0

PLOT THE GRAPH WITH S AS Y-AXIS AND G AS X – AXIS:

Throughput(S)	Attempts per packet time (G)
0.052	0.059
0.094	0.011
0.01	0.017
0.177	0.34
0.19	0.47
0.18	0.59



Practical Throughput	Theoretical Throughput
0.052	0.052
0.094	0.092
0.01	0.012
0.177	0.173
0.19	0.183
0.18	0.1807



<u>RESULT & INFERENCES</u>: We calculated throughput in NETSIM and compared it with theoretical values. For Pure ALOHA, as the number of nodes increases, collisions rise, but successful transmissions also increase, peaking at 79 nodes with a throughput of 0.1904, closely matching the theoretical value of 1/(2e) or 0.18. Beyond this, throughput declines due to excessive collisions. Slotted ALOHA, by synchronizing transmissions into time slots, achieved higher throughput than Pure ALOHA, making it more efficient in high-traffic environments, while Pure ALOHA is simpler but less effective under heavy load.

Experiment No: 6 **Date: 01.10.2024**

Study and Analyze the characteristic curve of throughput versus offered traffic for PURE ALOHA and SLOTTED ALOHA

Name : Thurlapati Sai Sree Praneetha
Reg. No : 22BLC1350
Faculty Name : Dr Jaya Vignesh T

NOTE: NetSim Academic supports a maximum of 100 nodes and hence this experiment can only be limited to maximum of 100 nodes Aloha networks

6.1 Theory

ALOHA provides a wireless data network. It is a multiple access protocol (this protocol is for allocating a multiple access channel). There are two main versions of ALOHA: pure and slotted. They differ with respect to whether time is divided up into discrete slots into which all frames must fit.

6.2 Pure Aloha

In Pure Aloha, users transmit whenever they have data to be sent. There will be collisions and the colliding frames will then be retransmitted. In NetSim's Aloha library, the sender waits a random amount of time per the exponential back-off algorithm and sends it again. The frame is discarded when the number of collisions a packet experiences crosses the "Retry Limit" - a user settable parameter in the GUI.

Let ''frame time'' denotes the amount of time needed to transmit the standard, fixed-length frame. In this experiment point, we assume that the new frames generated by the stations are modeled by a Poisson distribution with a mean of N frames per frame time. If N > 1, the nodes are generating frames at a higher rate than the channel can handle, and nearly every frame will suffer a collision. For reasonable throughput, we would expect 0 < N < 1. In addition to the new frames, the stations also generate retransmissions of frames that previously suffered collisions.

The probability of no other traffic being initiated during the entire vulnerable period is given by e^{-2G} which leads to $S = G \times e^{-2G}$ where, S is the throughput and G is the offered load. The units of S and G is frames per frame time.

G is the mean of the Poisson distribution followed by the transmission attempts per frame time, old and new combined. Old frames mean those frames that have previously suffered collisions.

The maximum throughput occurs at $G = 0.5$, with $S = \frac{1}{2e}$, which is about 0.184. In other words, the best we can hope for is a channel utilization of 18%. This result is not very encouraging, but with everyone transmitting at will, we could hardly have expected a 100% success rate.

6.3 Slotted Aloha

In slotted Aloha, time is divided up into discrete intervals, each interval corresponding to one frame. In Slotted Aloha, a node is required to wait for the beginning of the next slot in order to send the next packet. The probability of no other traffic being initiated during the entire vulnerable period is given by e^{-G} which leads to $S = G \times e^{-G}$. It is easy to compute that Slotted Aloha peaks at G = 1, with a throughput of $s = \frac{1}{e}$ or about 0.368.

6.4 Offered load and throughput calculations

Using NetSim, the attempts per packet time (G) can be calculated as follows.

$$G = \frac{Number\ of\ packets\ transmitted \times PacketTime(s)}{SimulationTime\ (s)}$$

where, G is Attempts per packet time. We derive the above formula keeping in mind that (i) NetSim's output metric, the number of packets transmitted, is nothing but the number of attempts, and (ii) since packets transmitted is computed over the entire simulation time, the number of "packet times" would be $\frac{SimulationTime(s)}{PacketTransmissionTime(s)}$, which is in the denominator. Note that in NetSim the output metric Packets transmitted is counted at link (PHY layer) level. Hence MAC layer re-tries are factored into this metric.

The throughput (in Mbps) per packet time can be obtained as follows.

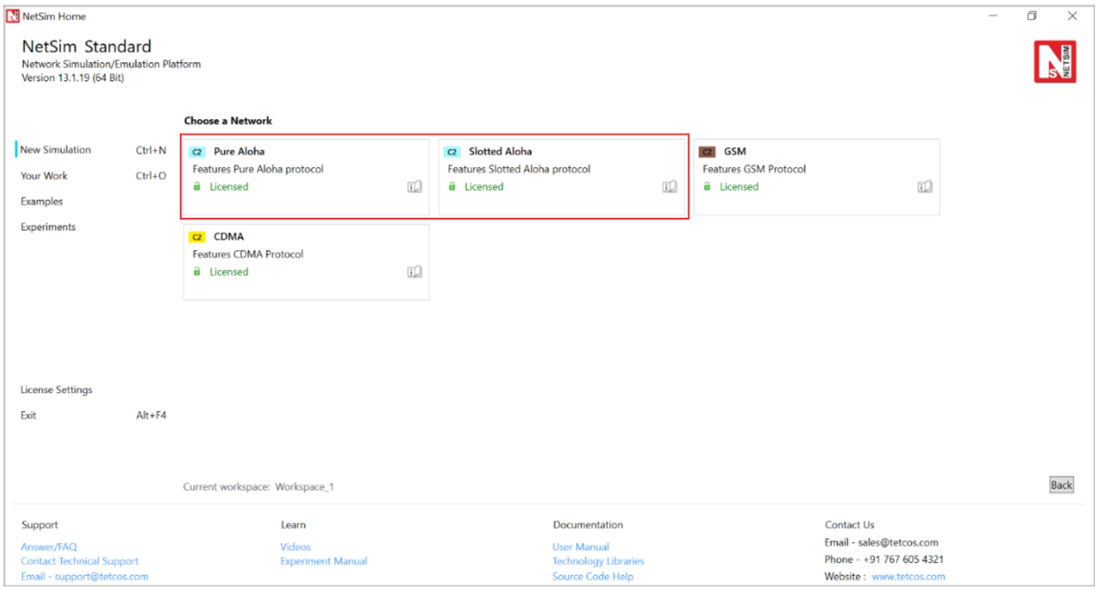
$$S = \frac{Number\ of\ packets\ successful \times PacketTime(s)}{SimulationTime\ (s)}$$

where, S = Throughput per packet time. In case of slotted aloha packet (transmission) time is equal to slot length (time). The packet transmission time is the PHY layer packet size in bits divided by the PHY rate in bits/s. Considering the PHY layer packet size as 1500B, and the PHY rate as 10 Mbps, the packet transmission time (or packet time) would be $\frac{1500\times8}{10\times10^6} = 1200\ \mu s$.

In the following experiment, we have taken packet size as 1460 B (Data Size) plus 28 B (Overheads) which equals 1488 B. The PHY data rate is 10 Mbps and hence packet time is equal to 1.2 milliseconds.

6.5 Network Set Up

In the Main menu select□□**New Simulation**□□ **Legacy Network**□□**Pure Aloha/ Slotted Aloha** as shown below



Create Scenario

Legacy networks come with a palette of various devices like wireless node,

Click and drop into environment

- **Click** on the **Node** icon in the tool bar and click and drop inside the grid. (*Note*: This is applicable for Pure Aloha and Slotted Aloha)
- Similarly drop **Adhoc link** and connect **Wireless Nodes** to **Adhoc links** using **Adhoc links**. (Note: A Node cannot be placed on another Node. A Node cannot float outside of the grid).

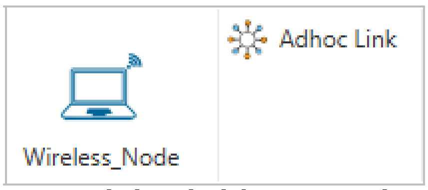


Figure 2: Pure and Slotted Aloha Device Palette in GUI

Pure Aloha: Input for 10-Nodes sample

Step 1: Drop 10 nodes (i.e., 9 Nodes are generating traffic.)

Node 2, 3, 4, 5, 6, 7, 8, 9, and 10 generates traffic. The properties of Nodes 2, 3, 4, 5, 6, 7, 8, 9, and 10 which transmits data to Node 1 are given in the below table.

Step 2: Wireless Node Properties:

Wireless Node Properties	
Interface1_Wireless (PHYSICAL_LAYER)	
Data Rate (Mbps)	10
Interface1_Wireless (DATALINK_LAYER)	
Retry_Limit	0
MAC_Buffer	FALSE
Slot Length(µs)	1200

Table 1-2: Wireless Node Properties

(**NOTE:** Slot Length(µs) parameter present only in Slotted Aloha → Wireless Node Properties → Interface_1 (Wireless))

Step 3: In Adhoc Link Properties, channel characteristics is set as **No Path Loss**.

Step 4: Application Properties:

- Right click on the Application Flow "App1 CUSTOM" and select Properties or click on the Application icon present in the top ribbon/toolbar. The properties are set according to the values given in the below Table 1-3.

Application_1 Properties		
Application Method	Unicast	
Application Type	Custom	
Source_Id	2	
Destination_Id	1	
Transport Protocol	UDP	
Packet Size	Distribution	Constant
	Value (Bytes)	1460
Inter Arrival Time	Distribution	Exponential
	Packet Inter Arrival Time (µs)	200000

Table 1-3: For Application_1 Properties

- Similarly create 8 more application, i.e., Source_Id as 3, 4, 5, 6, 7, 8, 9, 10 and Destination_Id as 1, set Packet Size and Inter Arrival Time as shown in above table.

Step 5: Plots are enabled in NetSim GUI.

Step 6: Simulation Time- 10 Seconds

NOTE: Obtain the values of Total Number of Packets Transmitted and Collided from the results window of NetSim.

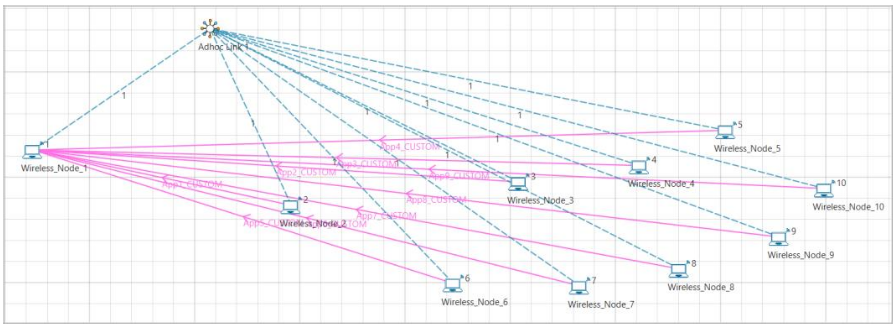


Figure: Network set up for studying the Pure aloha

Input for 20-Nodes sample

Step 1: Drop 20 nodes (i.e., 19 Nodes are generating traffic.)

Nodes 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, and 20 transmit data to Node 1.

Continue the experiment by increasing the number of nodes generating traffic as 29, 39, **49**, **59**, 69, **79, 89, 99** nodes

Slotted ALOHA: Input for 10-Nodes sample

Similarly repeat the experiment for Slotted Aloha with SLOT LENGTH 1200us

PLOT THE COMPARISON TABLE Output and plot graph for both the protocols

Comparison Table: The values of Total Number of Packets Transmitted and Collided obtained from the network statistics after running NetSim simulation are provided in the table below along with Throughput per packet time& Number of Packets Transmitted per packet time. The table should have the following tabulated

SLOTTED ALOHA

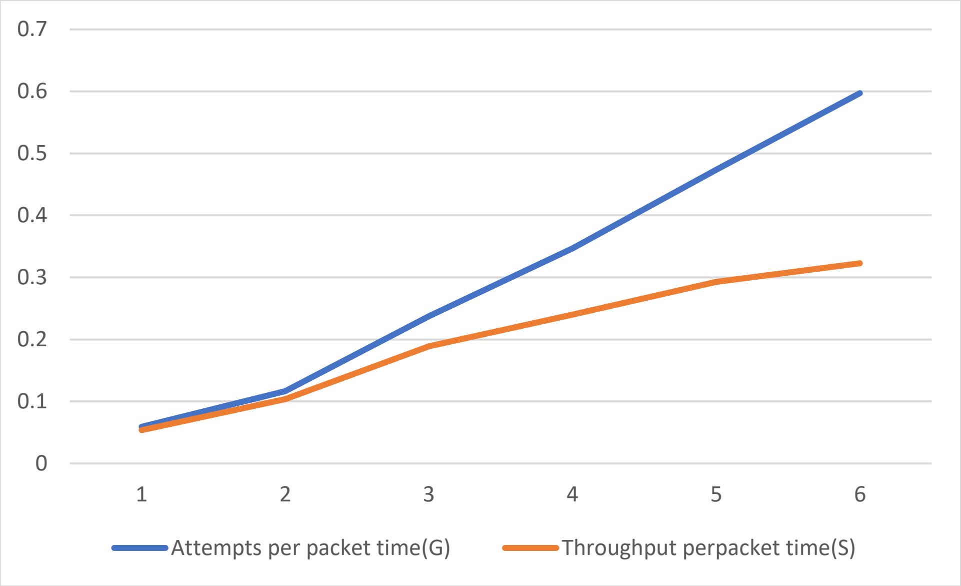
Number of nodes generating traffic	Total number of Packets Transmitted	Total number of Packets Collided	Successful Packets (Packets Transmitted -Packets Collided)	Attempts per packet time(G)	Throughput per packet time(S)	Throughput per packet time. Theoretical (S = $G * e^{-2G}$)
9	493	45	448	0.05916	0.05376	0.05255
19	974	111	863	0.11688	0.10356	0.09251
39	1981	407	1574	0.23772	0.18888	0.1477
59	2893	891	2002	0.34716	0.24024	0.17337

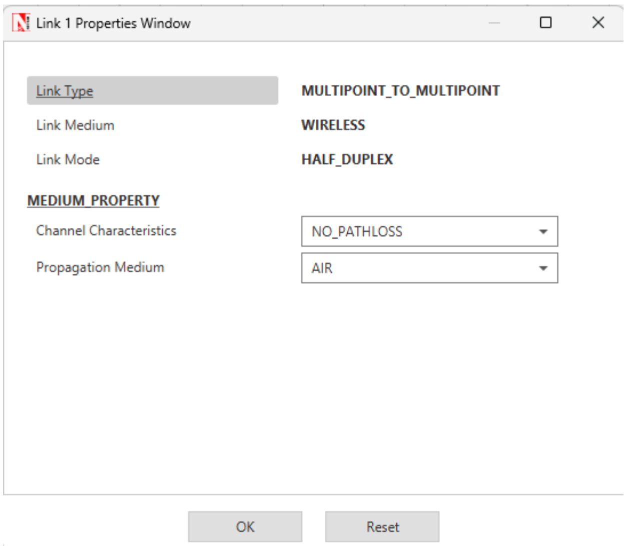
79	3946	1504	2442	0.47352	0.29304	0.18367
99	4976	2286	2690	0.59712	0.3228	0.18088

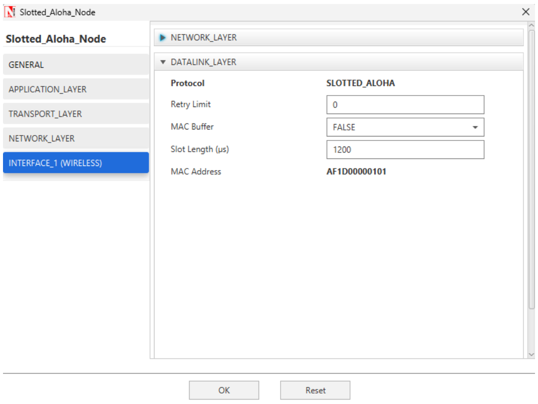
Attempts per packet time (G) = (Total number of Packet Transmitted * 1.2ms) / 10

Throughput per packet time (S) = (Total number of Successful packet transmitted * 1.2ms) / 10

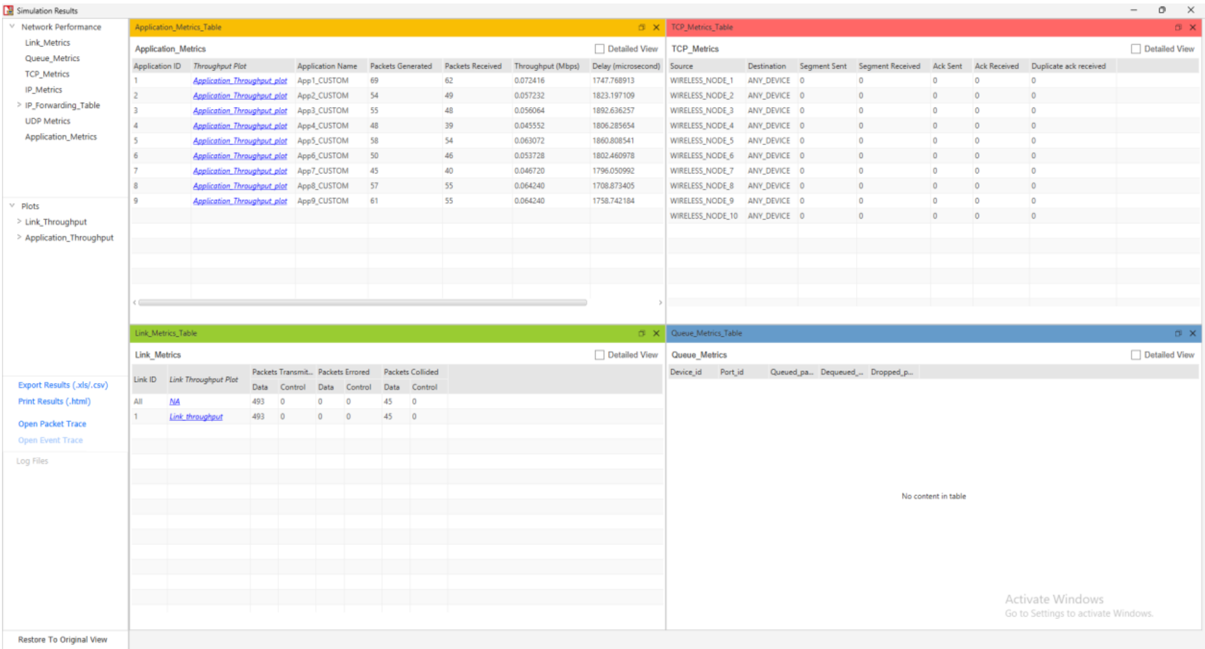
RESULT & INFERENCES:



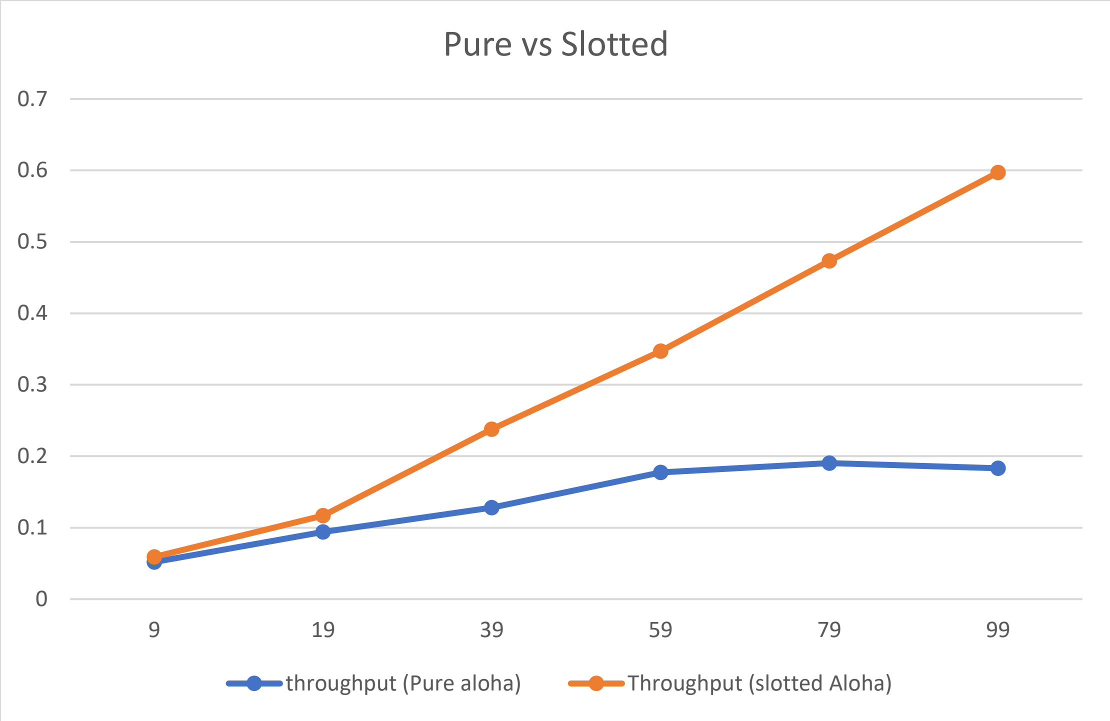




Slotted_Aloha_Node
Slotted_Aloha_Node
GENERAL
APPLICATION_LAYER
TRANSPORT_LAYER
NETWORK_LAYER
INTERFACE_1 (WIRELESS)
NETWORK_LAYER
DATALINK_LAYER
Protocol
SLOTTED_ALOHA
Retry Limit
0
MAC Buffer
FALSE
Slot Length (µs)
1200
MAC Address
AF1D00000101
OK
Reset



Simulation Results
Network Performance
Link_Metrics
Queue_Metrics
TCP_Metrics
IP_Metrics
IP_Forwarding_Table
UDP Metrics
Application_Metrics
Plots
Link_Throughput
Application_Throughput
Export Results (.xls/.csv)
Print Results (.html)
Open Packet Trace
Open Event Trace
Log Files
Restore To Original View
Application_Metrics_Table
Application_Metrics
Detailed View
Application ID
Throughput Plot
Application Name
Packets Generated
Packets Received
Throughput (Mbps)
Delay (microsecond)
1 Application_Throughput_plot App1_CUSTOM 69 62 0.072416 1747.768913
2 Application_Throughput_plot App2_CUSTOM 54 49 0.057232 1823.197109
3 Application_Throughput_plot App3_CUSTOM 55 48 0.056064 1892.636257
4 Application_Throughput_plot App4_CUSTOM 48 39 0.045552 1806.285654
5 Application_Throughput_plot App5_CUSTOM 58 54 0.063072 1860.808541
6 Application_Throughput_plot App6_CUSTOM 50 46 0.053728 1802.460978
7 Application_Throughput_plot App7_CUSTOM 45 40 0.046720 1796.050992
8 Application_Throughput_plot App8_CUSTOM 57 55 0.064240 1708.873405
9 Application_Throughput_plot App9_CUSTOM 61 55 0.064240 1758.742184
TCP_Metrics_Table
TCP_Metrics
Detailed View
Source
Destination
Segment Sent
Segment Received
Ack Sent
Ack Received
Duplicate ack received
WIRELESS_NODE_1 ANY_DEVICE 0 0 0 0 0
WIRELESS_NODE_2 ANY_DEVICE 0 0 0 0 0
WIRELESS_NODE_3 ANY_DEVICE 0 0 0 0 0
WIRELESS_NODE_4 ANY_DEVICE 0 0 0 0 0
WIRELESS_NODE_5 ANY_DEVICE 0 0 0 0 0
WIRELESS_NODE_6 ANY_DEVICE 0 0 0 0 0
WIRELESS_NODE_7 ANY_DEVICE 0 0 0 0 0
WIRELESS_NODE_8 ANY_DEVICE 0 0 0 0 0
WIRELESS_NODE_9 ANY_DEVICE 0 0 0 0 0
WIRELESS_NODE_10 ANY_DEVICE 0 0 0 0 0
Link_Metrics_Table
Link_Metrics
Detailed View
Link ID
Link Throughput Plot
Packets Transmit...
Packets Errored
Packets Collided
Data
Control
Data
Control
Data
Control
All NA 493 0 0 0 45 0
1 Link_throughput 493 0 0 0 45 0
Queue_Metrics_Table
Queue_Metrics
Detailed View
Device_id
Port_id
Queued_pa...
Dequeued_...
Dropped_p...
No content in table
Activate Windows
Go to Settings to activate Windows.



The results are based on slotted ALOHA protocol. The theoretical and practical throughput were obtained and upon comparison, an accurate simulation is ensured. We have also observed and compared the throughput values of pure ALOHA and slotted ALOHA.

Experiment No: 7 **Date: 22/10/24**

<u>Study and Analyze the router configuration, IP Packet routing process using Cisco Packet Tracer</u>

Name : Thurlapati Sai Sree Praneetha
Reg. No : 22BLC1350
Faculty Name : JayaVignesh T

Aim:

To create an internetwork of LANs and configure the router in a network with static routing and dynamic routing protocols and verify the connection using cisco packet tracer.

Tool Required:

Cisco Packet Tracer

Theory:

A device that connects different networks or subnetworks by forwarding data packets to their intended destinations. Its main functions include managing network traffic and enabling multiple devices to share an Internet connection. Routers commonly facilitate communication between local area networks (LANs) and wide area networks (WANs), which are networks that cover larger geographic areas. Typically, a single router is sufficient to connect a LAN.

Configuring a router correctly is essential for optimizing data transfer, ensuring security, preventing network congestion, and safeguarding against unauthorized access. It is a critical aspect of maintaining network performance, reliability, and overall connectivity.

<u>PROCEDURE</u>:

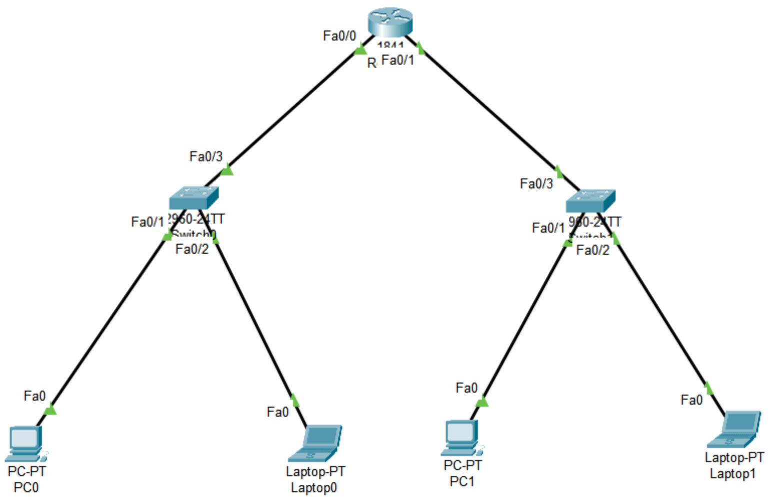
a) Launch Cisco Packet Tracer and create a new file.
b) Place three routers (model 1841), three switches, one server, and other end devices.
c) Connect the components using wires, which will initially appear red.
d) Connect the router using serial communication by adding this module to the router (WIC-2T).
e) Connect the routers using the serial DCE interfaces.
f) Configure the server's IP address: select the server, access the desktop, set the IP address,
g) Choose static configuration, set the default gateway, and specify the DNS server.

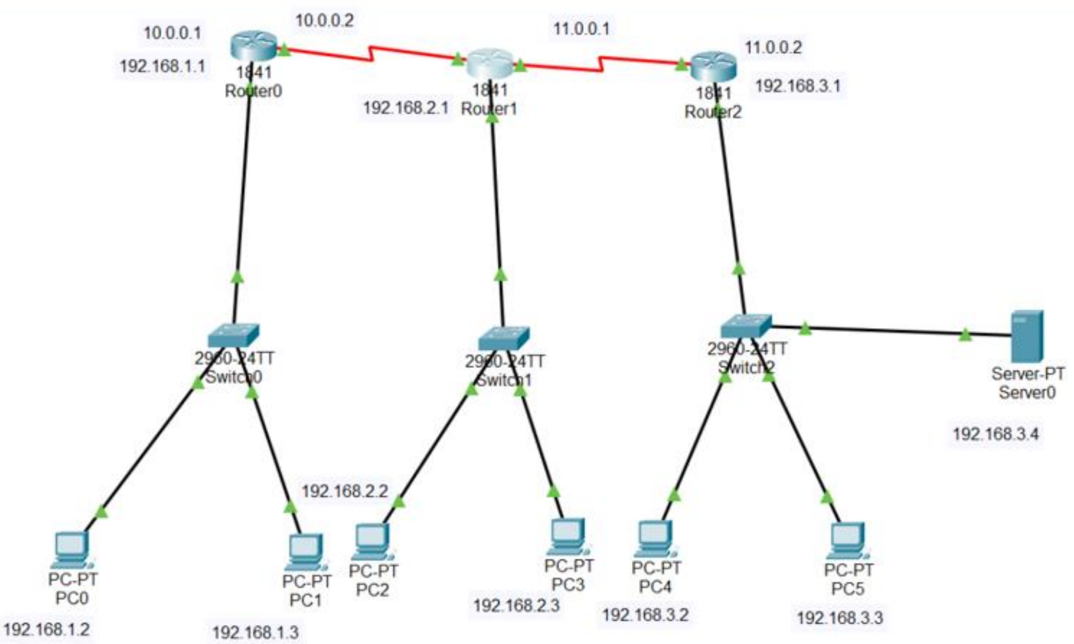
h) Configure the routers: select a router, turn off the power, add a WIC-2T module, and add static routes in routes
i) Include all server IP addresses in the router's routing table.

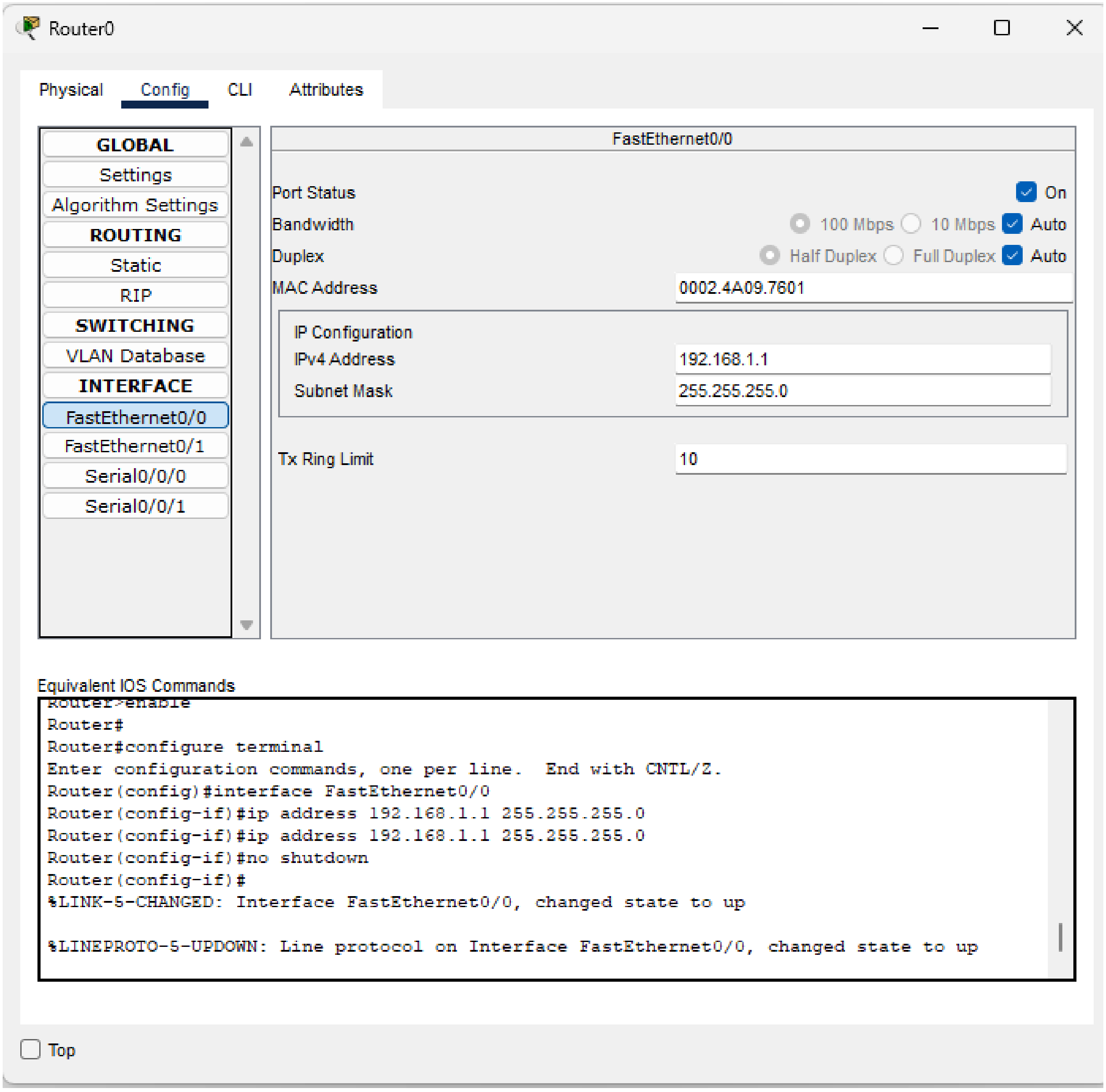
j) Configure the DNS server: select the server, enable DNS service, and specify the name and address.
k) Verify the setup: open the command prompt on any PC and ping hosted website.
l) Check if the ping response is working correctly.

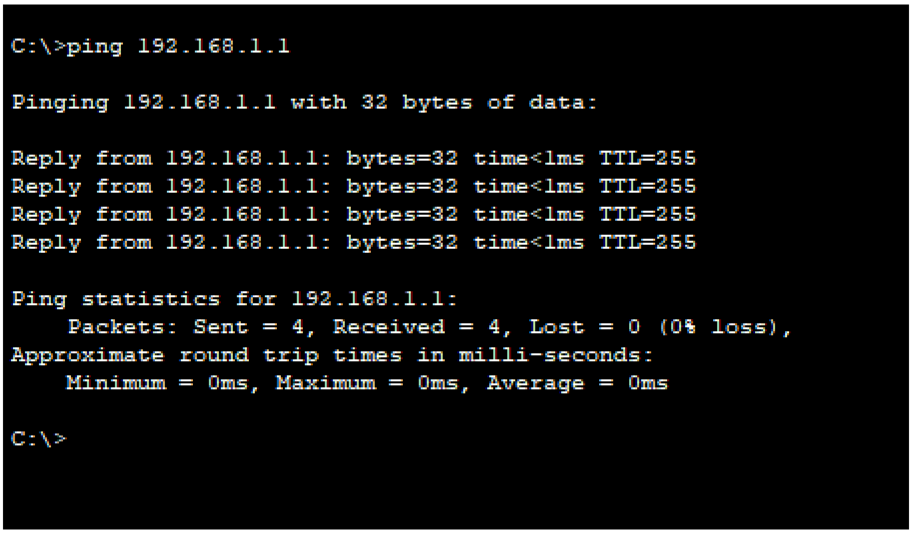
Commands to test
in PCs/Host : netstat –r
in Cisco Routers : show ip route
in PCs/Host : tracert <ip>

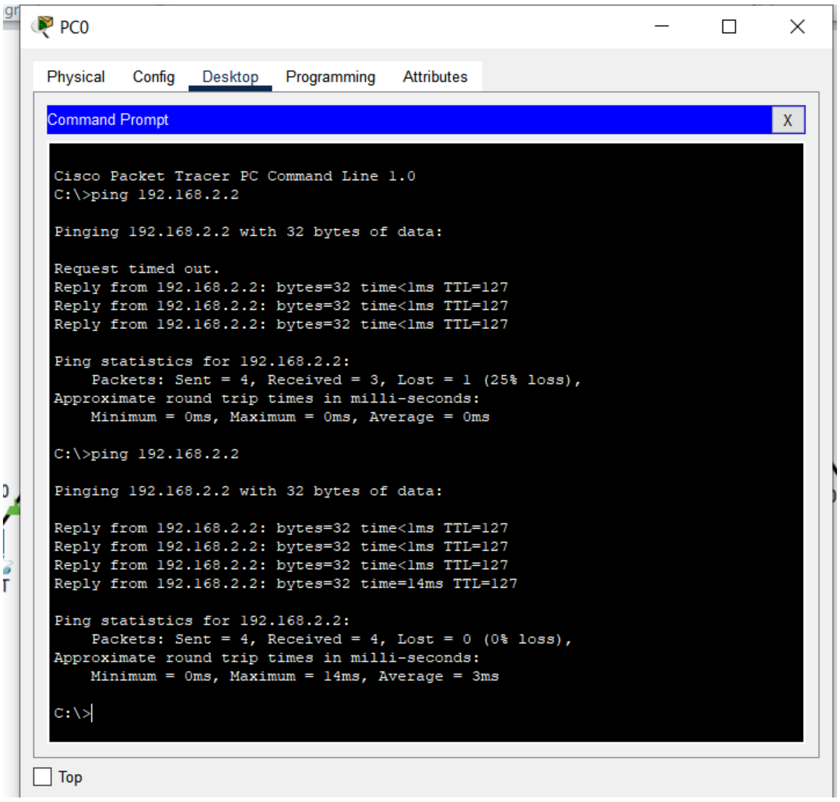
RESULT & INFERENCES:



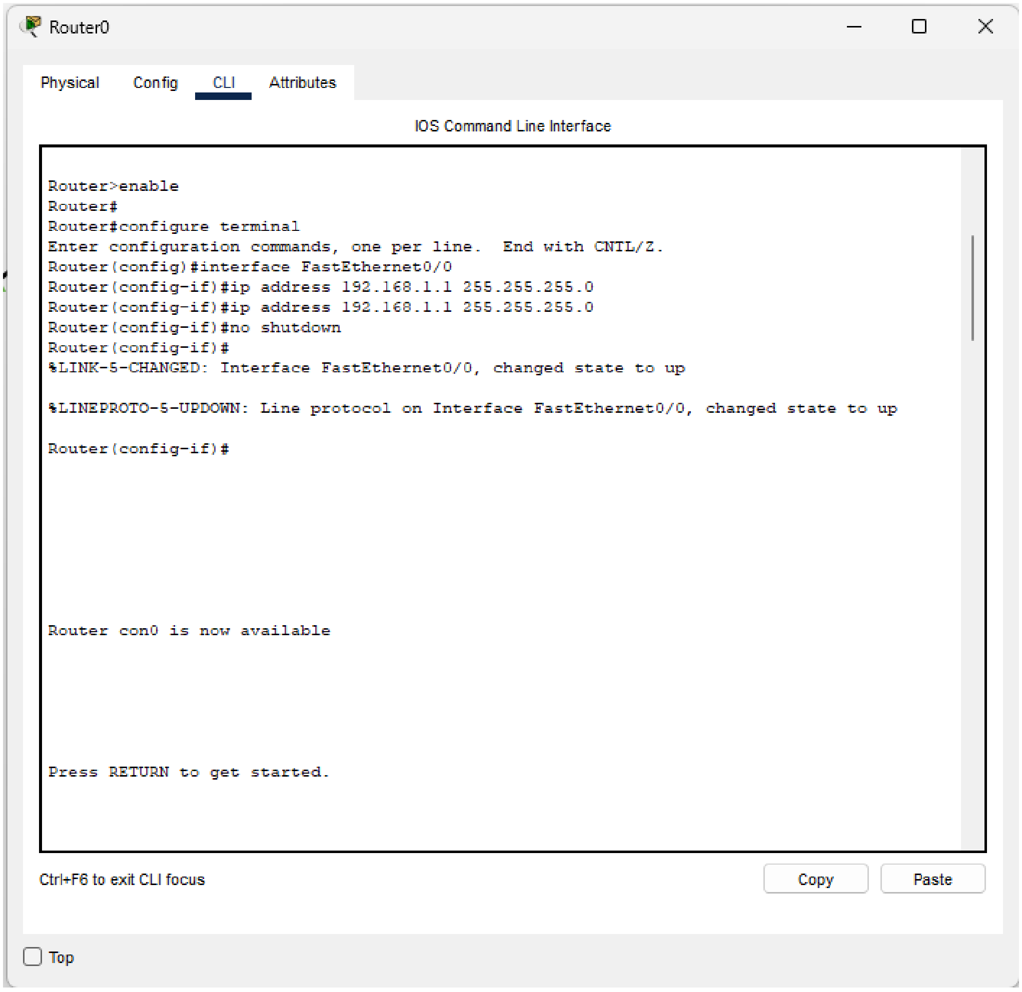



Router0
Physical
Config
CLI
Attributes
GLOBAL
Settings
Algorithm Settings
ROUTING
Static
RIP
SWITCHING
VLAN Database
INTERFACE
FastEthernet0/0
FastEthernet0/1
Serial0/0/0
Serial0/0/1
FastEthernet0/0
Port Status
On
Bandwidth
100 Mbps
10 Mbps
Auto
Duplex
Half Duplex
Full Duplex
Auto
MAC Address
0002.4A09.7601
IP Configuration
IPv4 Address
192.168.1.1
Subnet Mask
255.255.255.0
Tx Ring Limit
10
Equivalent IOS Commands
Router#
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface FastEthernet0/0
Router(config-if)#ip address 192.168.1.1 255.255.255.0
Router(config-if)#ip address 192.168.1.1 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
Top


C:\>ping 192.168.1.1
Pinging 192.168.1.1 with 32 bytes of data:
Reply from 192.168.1.1: bytes=32 time<1ms TTL=255
Reply from 192.168.1.1: bytes=32 time<1ms TTL=255
Reply from 192.168.1.1: bytes=32 time<1ms TTL=255
Reply from 192.168.1.1: bytes=32 time<1ms TTL=255
Ping statistics for 192.168.1.1:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 0ms, Maximum = 0ms, Average = 0ms
C:\>




```
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up
ip address 192.168.2.1 255.255.255.0
Router(config-if)#ip address 192.168.2.1 255.255.255.0
Router(config-if)#
Router(config-if)#
Router(config-if)#exit
Router(config)#interface FastEthernet0/0
Router(config-if)#
Router(config-if)#exit
Router(config)#interface Serial0/0/0
Router(config-if)#
Router(config-if)#exit
Router(config)#interface FastEthernet0/1
Router(config-if)#
Router(config-if)#
Router(config-if)#exit
Router(config)#interface FastEthernet0/1
Router(config-if)#
Router(config-if)#
Router(config-if)#
Router(config-if)#
Router(config-if)#exit
Router(config)# exit
Router#
%SYS-5-CONFIG_I: Configured from console by console
exit
```


Router0
Physical
Config
CLI
Attributes
IOS Command Line Interface
Router>enable
Router#
Router#configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#interface FastEthernet0/0
Router(config-if)#ip address 192.168.1.1 255.255.255.0
Router(config-if)#ip address 192.168.1.1 255.255.255.0
Router(config-if)#no shutdown
Router(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up
Router(config-if)#
Router con0 is now available
Press RETURN to get started.
Ctrl+F6 to exit CLI focus
Copy
Paste
Top

```
Router>enable
Router#show ?
  aaa                 Show AAA values
  access-lists        List access lists
  arp                 Arp table
  cdp                 CDP information
  class-map           Show QoS Class Map
  clock               Display the system clock
  controllers         Interface controllers status
  crypto              Encryption module
  debugging           State of each debugging option
  dhcp                Dynamic Host Configuration Protocol status
  dot11               IEEE 802.11 show information
  ephone              Show all or one ephone status
  file                Show filesystem information
  flash:              display information about flash: file system
  flow                Flow information
  frame-relay         Frame-Relay information
  history             Display the session command history
  hosts               IP domain-name, lookup style, nameservers, and host table
  interfaces          Interface status and configuration
  ip                  IP information
  ipv6                IPv6 information
  line                TTY line information
 --More--
```

Ctrl+F6 to exit CLI focus   Copy   Paste

Top

Router0

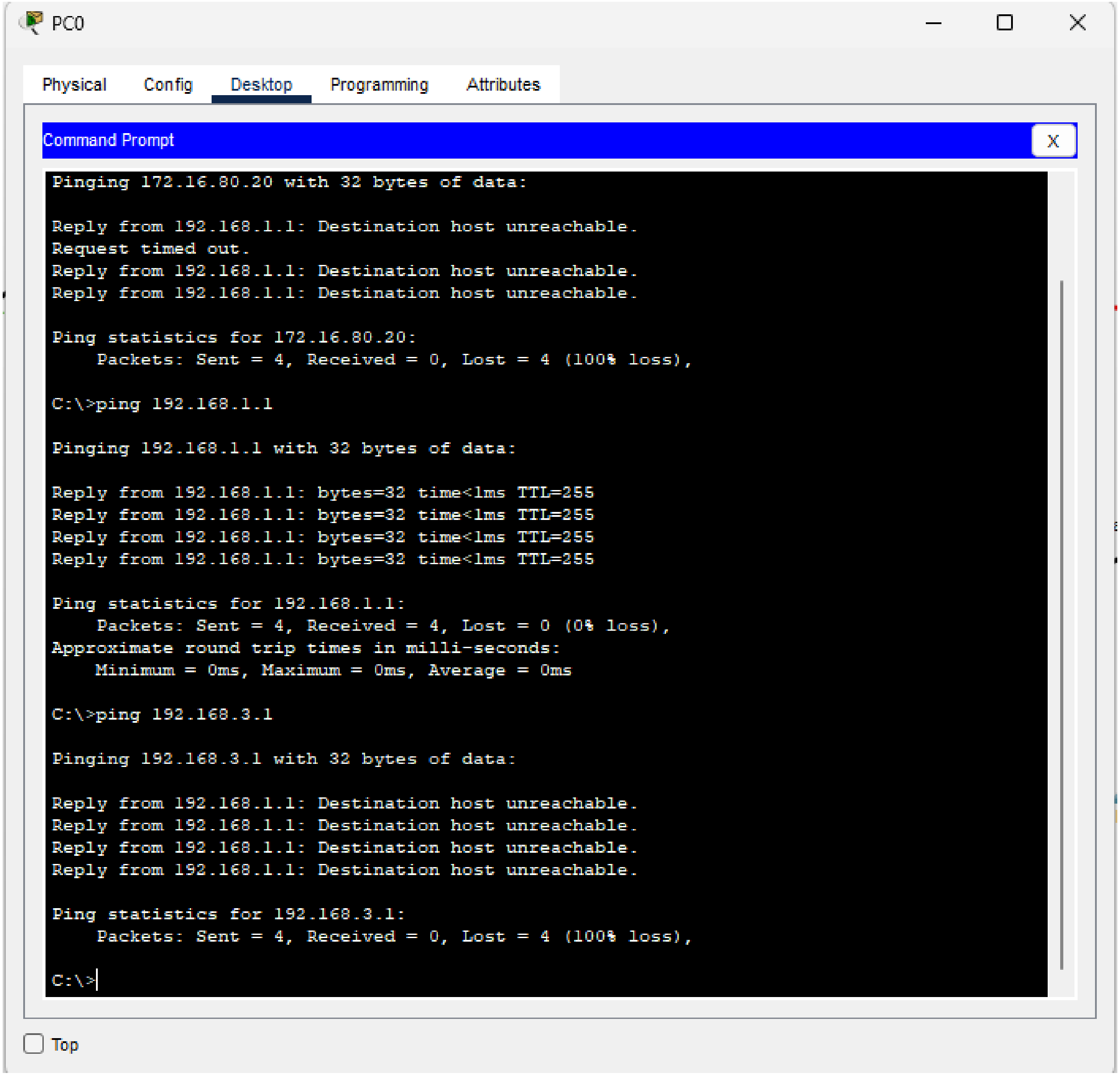
Physical   Config   CLI   Attributes

IOS Command Line Interface

```
  version             System hardware and software status
  vlan-switch         VTP VLAN status
  vtp                 Configure VLAN database
  zone                Zone Information
  zone-pair           Zone pair information
Router#show
% Incomplete command.
Router#
Router#
Router#
Router#
Router#
Router#
Router#
Router#
Router#
Router#
Router#
Router#
Router#
Router#show ip route
Codes: C - connected, S - static, I - IGRP, 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

C    192.168.1.0/24 is directly connected, FastEthernet0/0
C    192.168.2.0/24 is directly connected, FastEthernet0/1

Router#
```

Ctrl+F6 to exit CLI focus   Copy   Paste

Top


PC0
Physical
Config
Desktop
Programming
Attributes
Command Prompt
X
Pinging 172.16.80.20 with 32 bytes of data:

Reply from 192.168.1.1: Destination host unreachable.
Request timed out.
Reply from 192.168.1.1: Destination host unreachable.
Reply from 192.168.1.1: Destination host unreachable.

Ping statistics for 172.16.80.20:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>ping 192.168.1.1

Pinging 192.168.1.1 with 32 bytes of data:

Reply from 192.168.1.1: bytes=32 time<1ms TTL=255
Reply from 192.168.1.1: bytes=32 time<1ms TTL=255
Reply from 192.168.1.1: bytes=32 time<1ms TTL=255
Reply from 192.168.1.1: bytes=32 time<1ms TTL=255

Ping statistics for 192.168.1.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

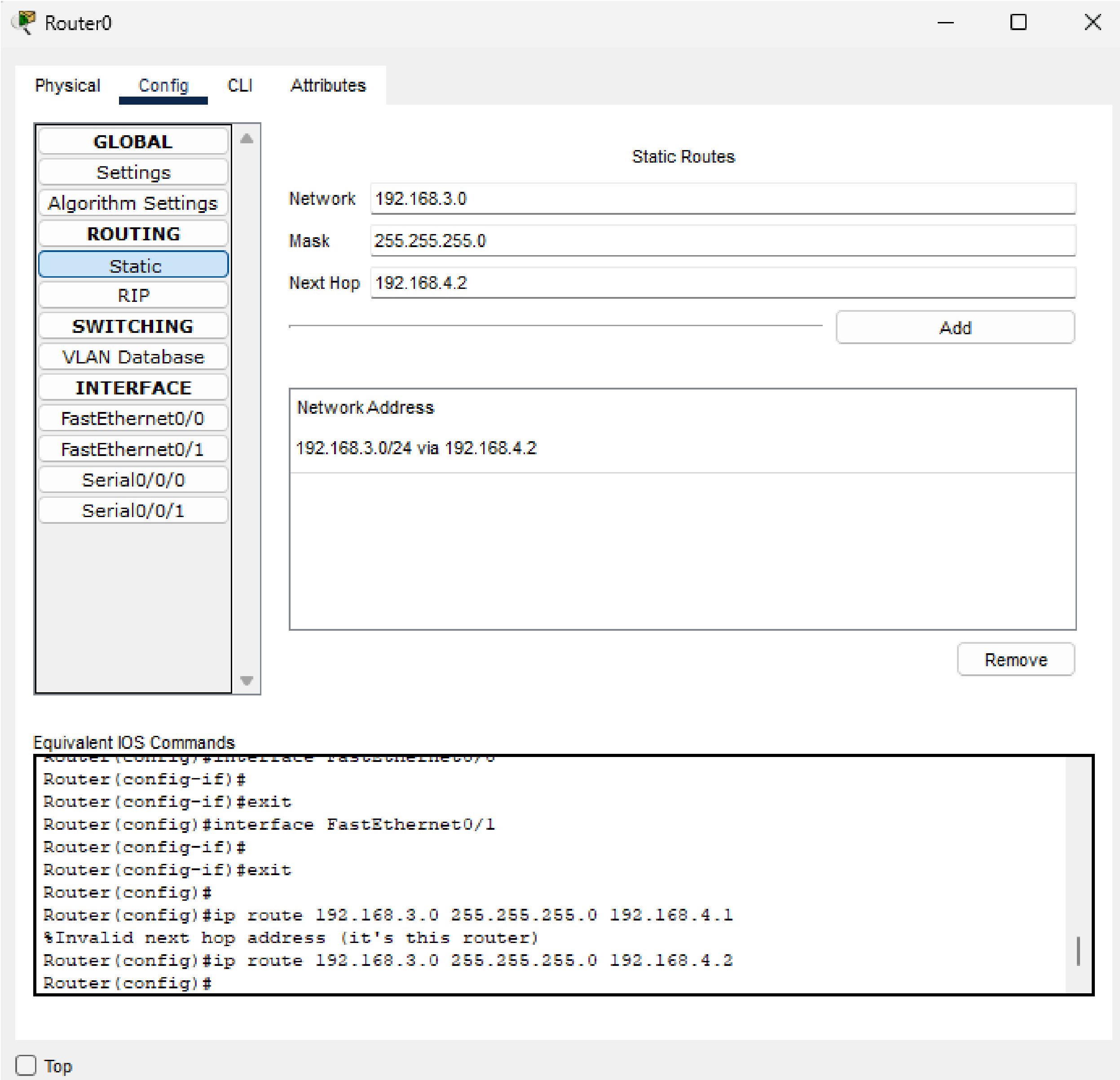
C:\>ping 192.168.3.1

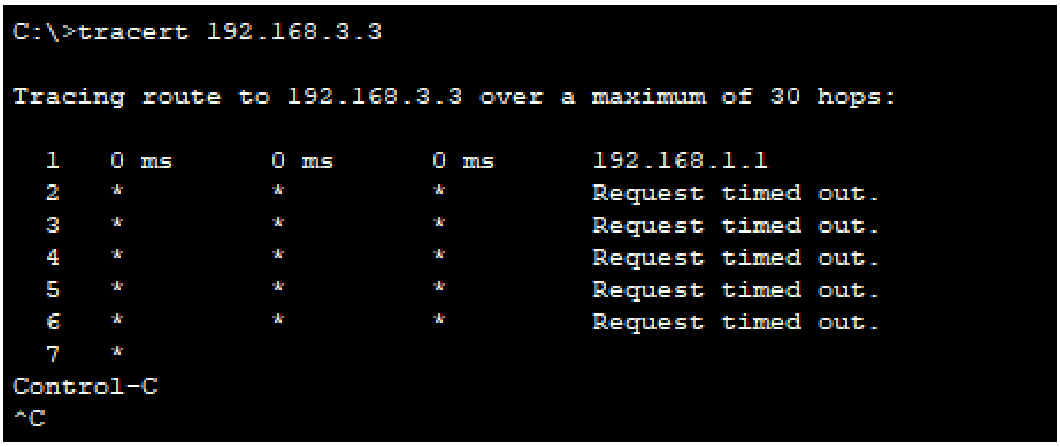
Pinging 192.168.3.1 with 32 bytes of data:

Reply from 192.168.1.1: Destination host unreachable.
Reply from 192.168.1.1: Destination host unreachable.
Reply from 192.168.1.1: Destination host unreachable.
Reply from 192.168.1.1: Destination host unreachable.

Ping statistics for 192.168.3.1:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),

C:\>
Top


Router0
Physical
Config
CLI
Attributes
GLOBAL
Settings
Algorithm Settings
ROUTING
Static
RIP
SWITCHING
VLAN Database
INTERFACE
FastEthernet0/0
FastEthernet0/1
Serial0/0/0
Serial0/0/1
Static Routes
Network 192.168.3.0
Mask 255.255.255.0
Next Hop 192.168.4.2
Add
Network Address
192.168.3.0/24 via 192.168.4.2
Remove
Equivalent IOS Commands
Router(config-if)#
Router(config-if)#exit
Router(config)#interface FastEthernet0/1
Router(config-if)#
Router(config-if)#exit
Router(config)#
Router(config)#ip route 192.168.3.0 255.255.255.0 192.168.4.1
%Invalid next hop address (it's this router)
Router(config)#ip route 192.168.3.0 255.255.255.0 192.168.4.2
Router(config)#
Top


C:\>tracert 192.168.3.3
Tracing route to 192.168.3.3 over a maximum of 30 hops:
1 0 ms 0 ms 0 ms 192.168.1.1
2 * * * Request timed out.
3 * * * Request timed out.
4 * * * Request timed out.
5 * * * Request timed out.
6 * * * Request timed out.
7 *
Control-C
^C

```
C:\>ping 192.168.3.2

Pinging 192.168.3.2 with 32 bytes of data:

Reply from 192.168.3.2: bytes=32 time=24ms TTL=126
Reply from 192.168.3.2: bytes=32 time=12ms TTL=126
Reply from 192.168.3.2: bytes=32 time=31ms TTL=126
Reply from 192.168.3.2: bytes=32 time=16ms TTL=126

Ping statistics for 192.168.3.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 12ms, Maximum = 31ms, Average = 20ms

C:\>
```

```
C:\>ping 192.168.3.4

Pinging 192.168.3.4 with 32 bytes of data:

Reply from 192.168.3.4: bytes=32 time=18ms TTL=126
Reply from 192.168.3.4: bytes=32 time=14ms TTL=126
Reply from 192.168.3.4: bytes=32 time=1ms TTL=126
Reply from 192.168.3.4: bytes=32 time=1ms TTL=126

Ping statistics for 192.168.3.4:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 18ms, Average = 8ms

C:\>
```

```
C:\>netstat -r

Route Table
===========================================================================
Interface List
0x1 ........................... PT TCP Loopback interface
0x2 ...00 16 6f 0d 88 ec ...... PT Ethernet interface
0x1 ........................... PT TCP Loopback interface
0x2 ...00 16 6f 0d 88 ec ...... PT Bluetooth interface
===========================================================================
===========================================================================
Active Routes:
Network Destination        Netmask          Gateway       Interface  Metric
          0.0.0.0          0.0.0.0      192.168.1.1     192.168.1.2       1
Default Gateway:      192.168.1.1
===========================================================================
Persistent Routes:
  None
```

CONCLUSION: Therefore, from the above experiment we have created a LAN network and configure it with static and dynamic networking protocols. We have also used and understood the importance of commands like tracert, netstat – r.

Experiment No: 8 **Date: 05-11-2024**

Design configuration and Implementation of Subnetting using Cisco Packet Tracer

Name : Thurlapati Sai Sree Praneetha
Reg. No : 22BLC1350
Faculty Name : Dr Jaya Vignesh T

Aim:

Task 1

You are granted the Class C block 192.168.60.0/24. You need to create 5 subnets each with 32 addresses each.

i) Find the first address and last address in Subnet 1

ii) Find the first address and last address in Subnet 2

iii) Find the first address and last address in Subnet 3

iv) Find the first address and last address in Subnet 4

v) Find the first address and last address in Subnet 5

vi) List the remaining unused address left in the Class C block for future allocation.

Tool Required:

Cisco Packet Tracer

Enter the answers for the above design in this box below

P.T.O

PROCEDURE:

Steps of making Subnet Planning

1.Take all required devices in cisco packet tracer.

- 8 PC
- 2 Router
- 4 Switches

2.Using cable connect all devices

- Connect three PC to one switch same for another
- Connect two router with each other and two switches to router.

4.Give IP address and default gateway to PC

- In desktop IP config give IP address in static
- In Desktop->In static->IPv4
- Also set value of default gateway (same for every two pcs which are connected with one switch is as same)
- Give subnet mask according from router to switch.

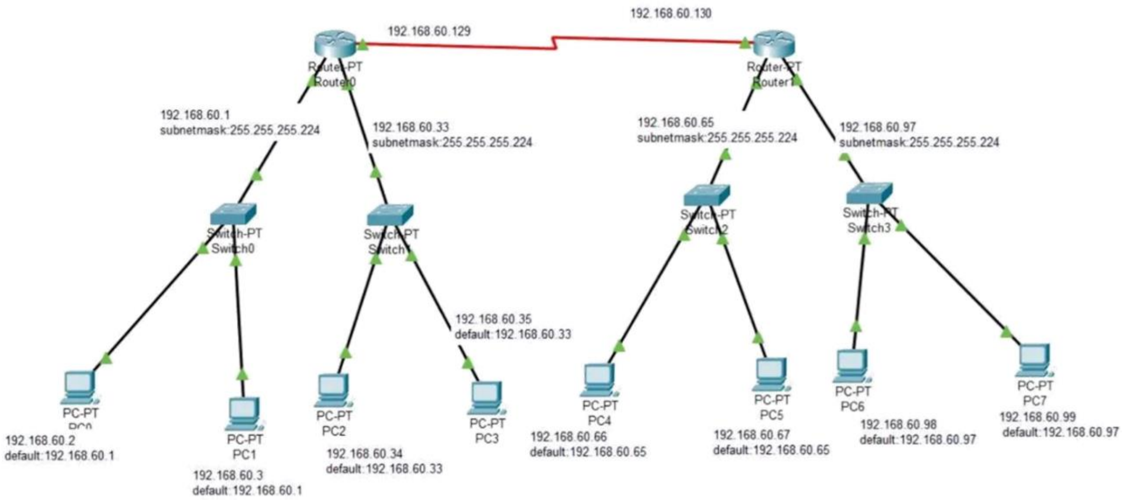
5. Enable the port of Switch according to connection

- Connect serial to serial connection between router give appropriate ip address as shown in fig.

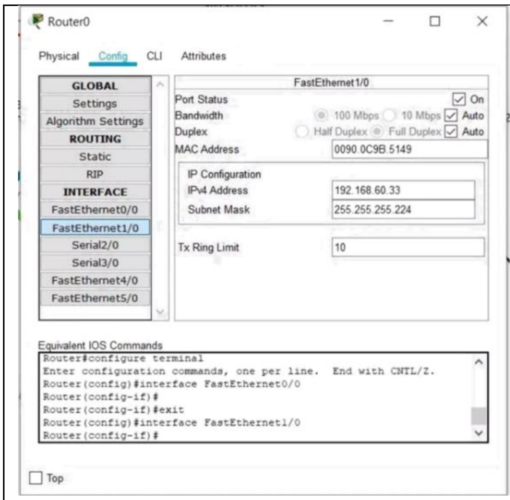
Use Router-PT which has two Fast Ethernet Ports and two serial ports

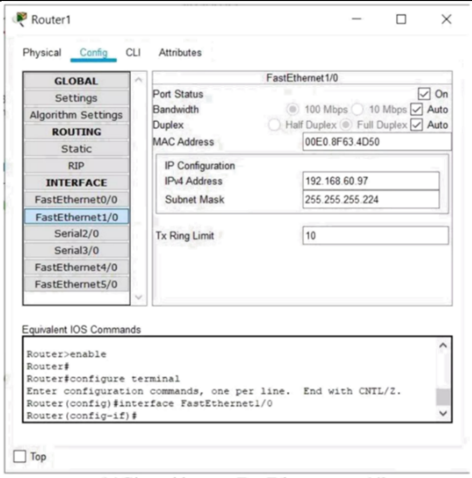
- Allocate Subnet 1 and Subnet 2 to the two LANs connected via Router 0 fast Ethernet interfaces as shown below (Left hand side)
- Allocate Subnet 3 and Subnet 4 to the two LANs connected via Router 1 fast Ethernet interfaces as shown below (Right hand side)
- **Allocate Subnet 5 to the serial interface of Router 0 and Router 1 as shown below (use the 2nd and 3rd address in that block to assign to the serial interfaces of both the routers**.

SAMPLE DESIGN AS SHOWN BELOW:

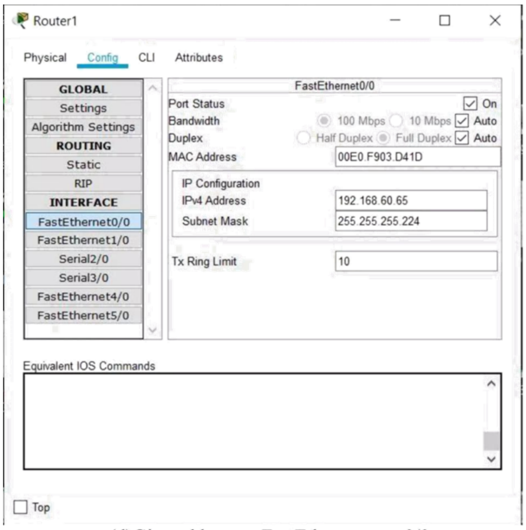


Some Sample Configurations

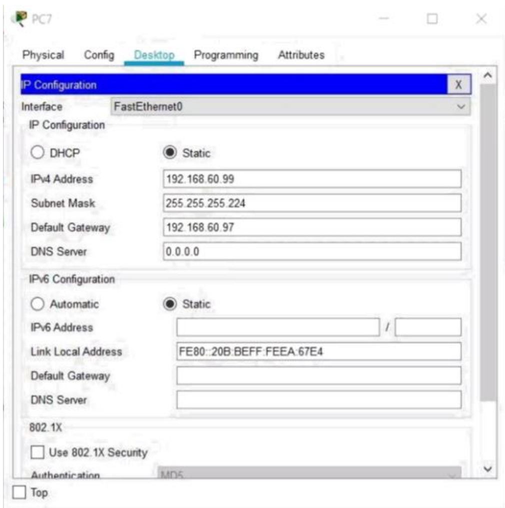




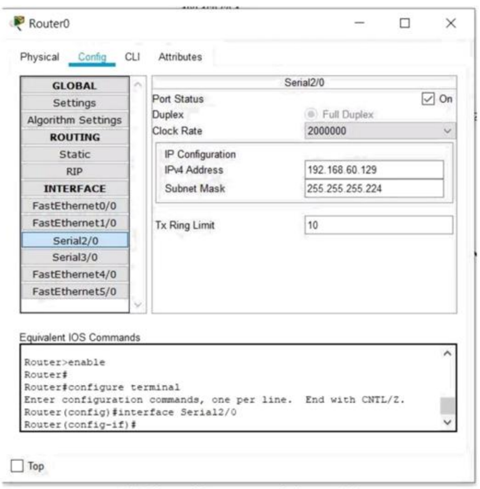
(c)Give address to FastEthernet port 1/0



(d)Give address to FastEthernet port 0/0

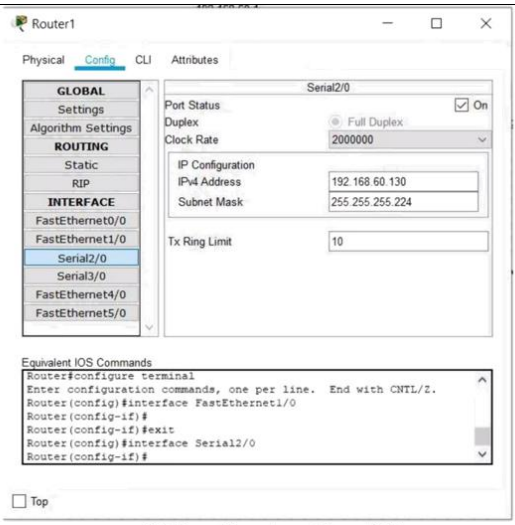


(f)give static IP to PC7



(b)Give address to serial port 2/0

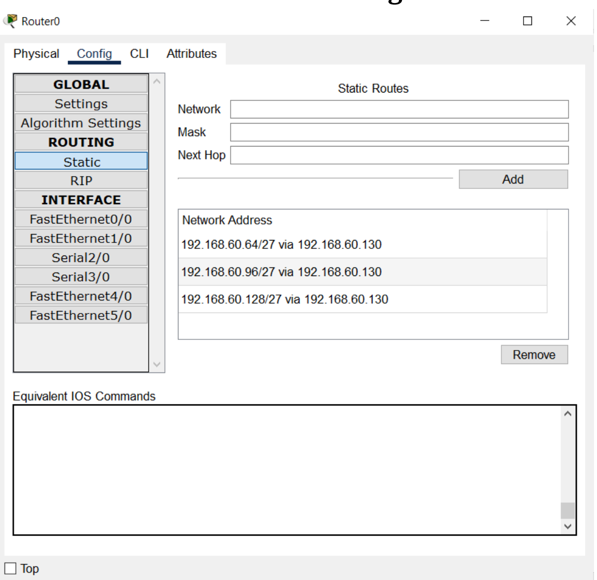
Router 0 serial interface IP configuration – Subnet 5



(n)give address to serial port 2/0

Router 1 serial interface – same Subnet 5

Add Static routes of three subnetted blocks in Router 0. **Similarly add respectively the necessary static routes in Router 1 routing table.**



RESULT & INFERENCES:

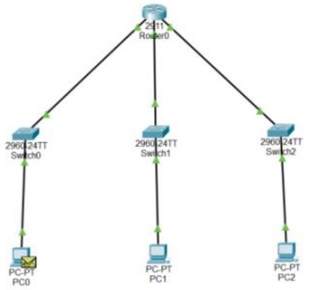
Commands to test
in PCs/Host : netstat –r
in Cisco Routers : show ip route in both routers to display the subnetted block routes
in PCs/Host : PING test and tracert <ip> to test connecting to PCs that belong to different subnets

Task 2 (Bonus)

You are granted the Class C block 192.168.10.0/24. You need to create 3 subnets such that one subnet requires 120 address, another subnet requires 60 addresses and 3rd subnet requires 30 address

i) Find the first address and last address in Subnet 1
ii) Find the first address and last address in Subnet 2

iii) Find the first address and last address in Subnet 3



1 router is alone enough for this configuration of switched LAN for each subnet. A Router 2811 model can used that has two fastethernet interfaces (additionally add one NM-1FE-TX Module in Router 2811 model to support for the 3rd Fast Ethernet interface) for configuration of 3 subnets.

RESULT & INFERENCES:

in PCs/Host: PING test and tracert <ip> to test connecting to PCs that belong to different subnets

Task 1:

```
C:\>ping 192.168.60.35

Pinging 192.168.60.35 with 32 bytes of data:

Request timed out.
Reply from 192.168.60.35: bytes=32 time<1ms TTL=127
Reply from 192.168.60.35: bytes=32 time<1ms TTL=127
Reply from 192.168.60.35: bytes=32 time<1ms TTL=127

Ping statistics for 192.168.60.35:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Task 2:

```
C:\>ping 192.168.60.98

Pinging 192.168.60.98 with 32 bytes of data:

Request timed out.
Reply from 192.168.60.98: bytes=32 time<1ms TTL=127
Reply from 192.168.60.98: bytes=32 time<1ms TTL=127
Reply from 192.168.60.98: bytes=32 time<1ms TTL=127

Ping statistics for 192.168.60.98:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>
```

Task 3:

```
C:\>ping 192.168.60.66

Pinging 192.168.60.66 with 32 bytes of data:

Reply from 192.168.60.1: Destination host unreachable.
Reply from 192.168.60.1: Destination host unreachable.
Reply from 192.168.60.1: Destination host unreachable.
Reply from 192.168.60.1: Destination host unreachable.

Ping statistics for 192.168.60.66:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

Task 4

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.60.98

Pinging 192.168.60.98 with 32 bytes of data:

Reply from 192.168.60.33: Destination host unreachable.
Reply from 192.168.60.33: Destination host unreachable.
Reply from 192.168.60.33: Destination host unreachable.
Reply from 192.168.60.33: Destination host unreachable.

Ping statistics for 192.168.60.98:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

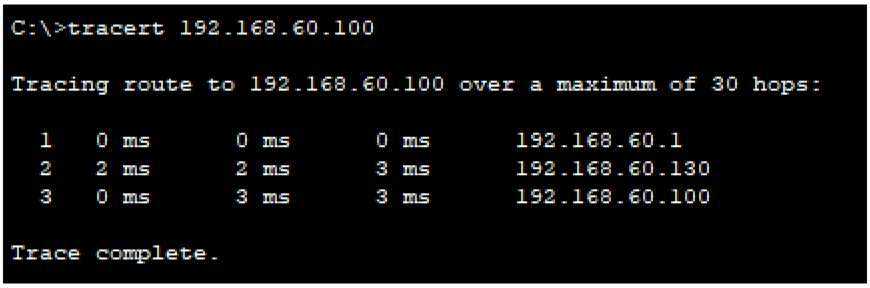
Task 5:

```
C:\>ping 192.168.60.1

Pinging 192.168.60.1 with 32 bytes of data:

Reply from 192.168.60.1: bytes=32 time<1ms TTL=255
Reply from 192.168.60.1: bytes=32 time<1ms TTL=255
Reply from 192.168.60.1: bytes=32 time<1ms TTL=255
Reply from 192.168.60.1: bytes=32 time<1ms TTL=255

Ping statistics for 192.168.60.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Task 6:

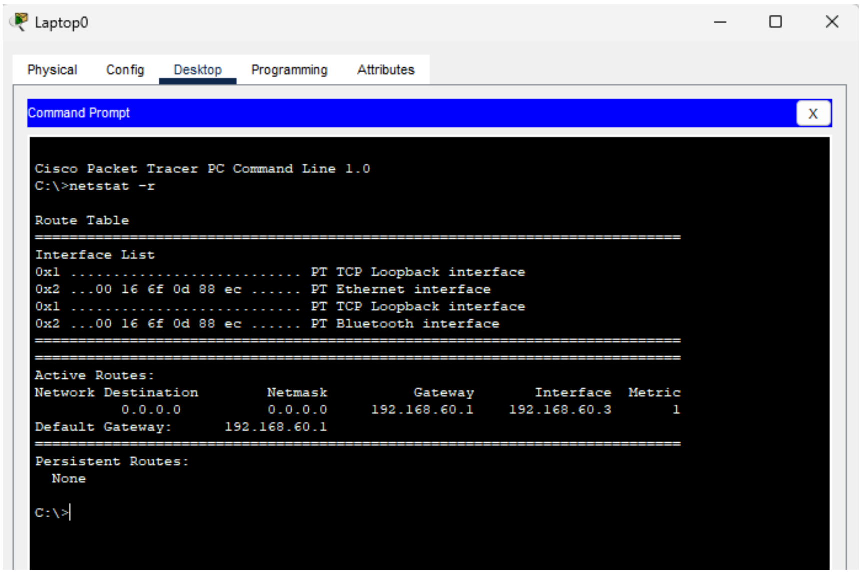
```
C:\>ping 192.168.60.129

Pinging 192.168.60.129 with 32 bytes of data:

Reply from 192.168.60.129: bytes=32 time<1ms TTL=255
Reply from 192.168.60.129: bytes=32 time<1ms TTL=255
Reply from 192.168.60.129: bytes=32 time<1ms TTL=255
Reply from 192.168.60.129: bytes=32 time<1ms TTL=255

Ping statistics for 192.168.60.129:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Task 7:



```
C:\>tracert 192.168.60.100

Tracing route to 192.168.60.100 over a maximum of 30 hops:

  1   0 ms      0 ms      0 ms      192.168.60.1
  2   2 ms      2 ms      3 ms      192.168.60.130
  3   0 ms      3 ms      3 ms      192.168.60.100

Trace complete.
```

Task 8:




```
Cisco Packet Tracer PC Command Line 1.0
C:\>netstat -r

Route Table
===========================================================================
Interface List
0x1 ........................... PT TCP Loopback interface
0x2 ...00 16 6f 0d 88 ec ...... PT Ethernet interface
0x1 ........................... PT TCP Loopback interface
0x2 ...00 16 6f 0d 88 ec ...... PT Bluetooth interface
===========================================================================
===========================================================================
Active Routes:
Network Destination        Netmask          Gateway       Interface  Metric
          0.0.0.0          0.0.0.0     192.168.60.1    192.168.60.3       1
Default Gateway:      192.168.60.1
===========================================================================
Persistent Routes:
  None

C:\>
```

Experiment No: 8a **Date: 12-11-24**

Study and analyze and routing table formation of Interior routing protocol, i.e., Routing Information Protocol (RIP) using Cisco Packet Tracer

Name : Thurlapati Sai Sree Praneetha
Reg. No : 22BLC1350
Faculty Name : JayaVignesh T

Aim:

To create an internetwork of LANs and configure the router in a network with dynamic routing RIP and verify the connection using cisco packet tracer and repeat the analysis with Netsim and record the performance metrics.

Tool Required:

Cisco Packet Tracer & Netsim

Theory:

RIP is intended to allow hosts and gateways to exchange information for computing routes through an IP-based network. RIP is a distance vector protocol which is based on Bellman-Ford algorithm. This algorithm has been used for routing computation in the network.

Distance vector algorithms are based on the exchange of only a small amount of information using RIP messages.

Each entity (router or host) that participates in the routing protocol is assumed to keep information about all of the destinations within the system. Generally, information about all entities connected to one network is summarized by a single entry, which describes the route to all destinations on that network. This summarization is possible because as far as IP is concerned, routing within a network is invisible. Each entry in this routing database includes the next router to which datagram's destined for the entity should be sent. In addition, it includes a "metric" measuring the total distance to the entity.

Distance is a somewhat generalized concept, which may cover the time delay in getting messages to the entity, the dollar cost of sending messages to it, etc. Distance vector algorithms get their name from the fact that it is possible to compute optimal routes when the only information exchanged is the list of these distances. Furthermore, information is only exchanged among entities that are adjacent, that is, entities that share a common network

PROCEDURE:

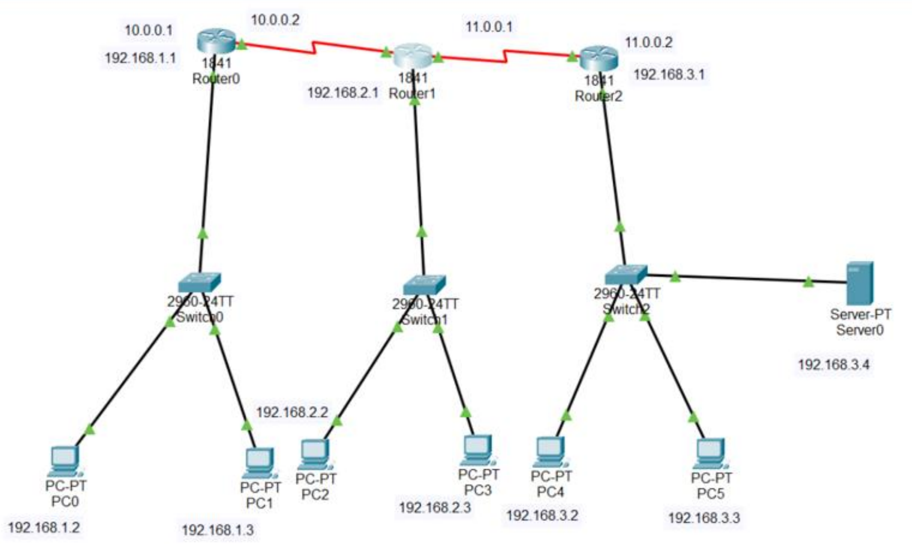
a) Launch Cisco Packet Tracer and create a new file.
b) Place three routers (model 1841), three switches, one server, and other end devices.
c) Connect the components using wires, which will initially appear red.
d) Connect the router using serial communication by adding this module to the router (WIC-2T).

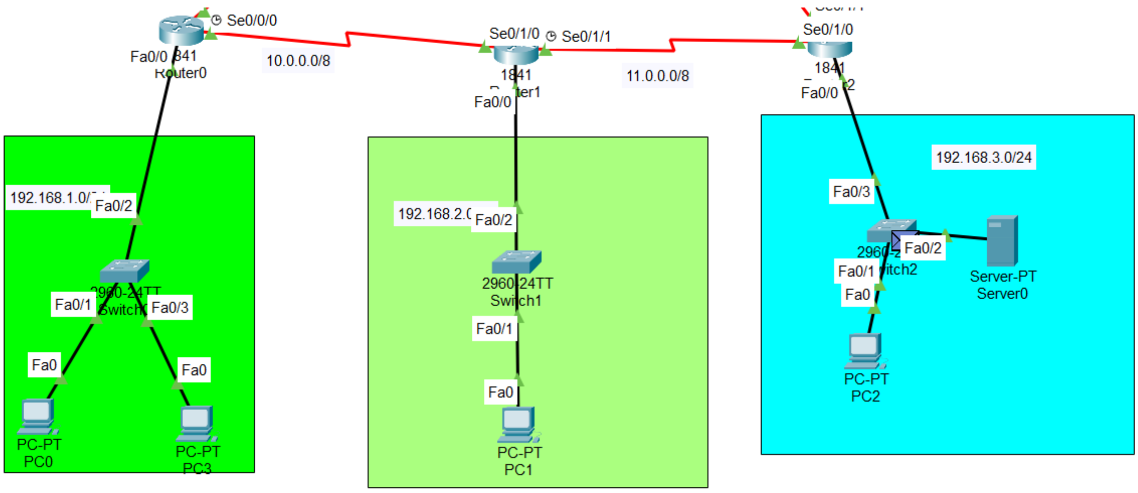
e) Connect the routers using the serial DCE interfaces.
f) Configure the server's IP address: select the server, access the desktop, set the IP address,
g) Choose static configuration, set the default gateway, and specify the DNS server.
h) Configure the routers: select a router, turn off the power, add a WIC-2T module, and **enable RIP by adding the directly connected networks**
i) Include all server IP addresses in the router's routing table.
j) Configure the DNS server: select the server, enable DNS service, and specify the name and address.
k) Verify the setup: open the command prompt on any PC and ping hosted website.
l) Check if the ping response is working correctly.

Commands to test and check the routing table to learn the networks via RIP protocol in Cisco Routers : show ip route

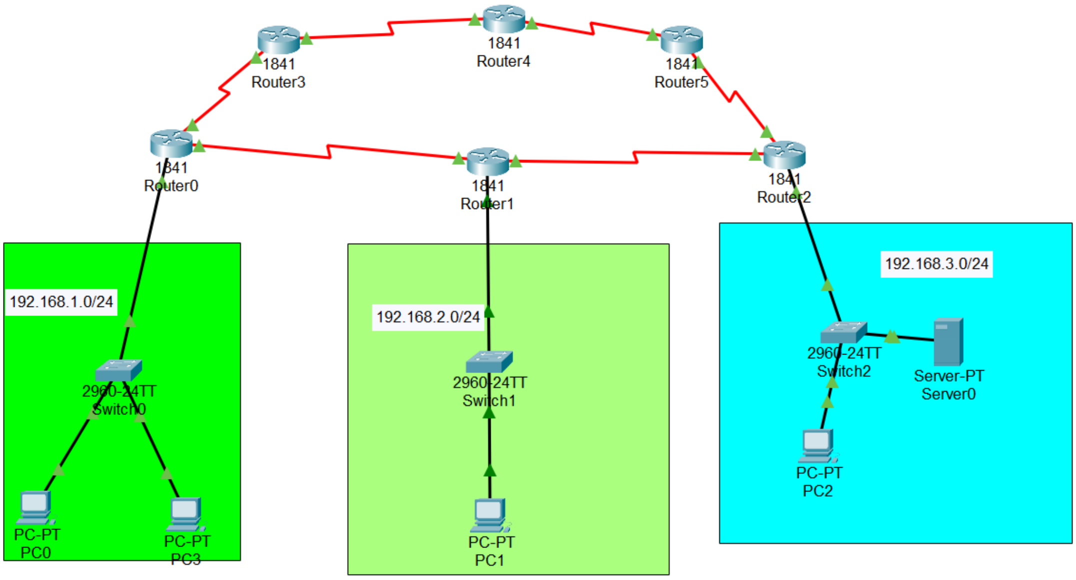
in PCs/Host : tracert <ip>

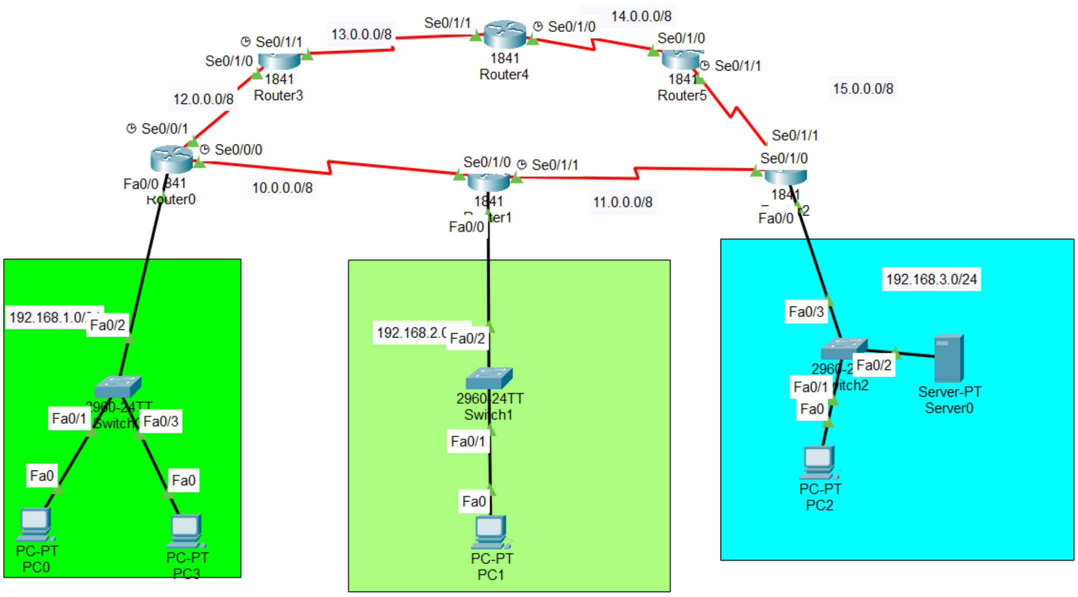
Get into Simulation Mode and test it



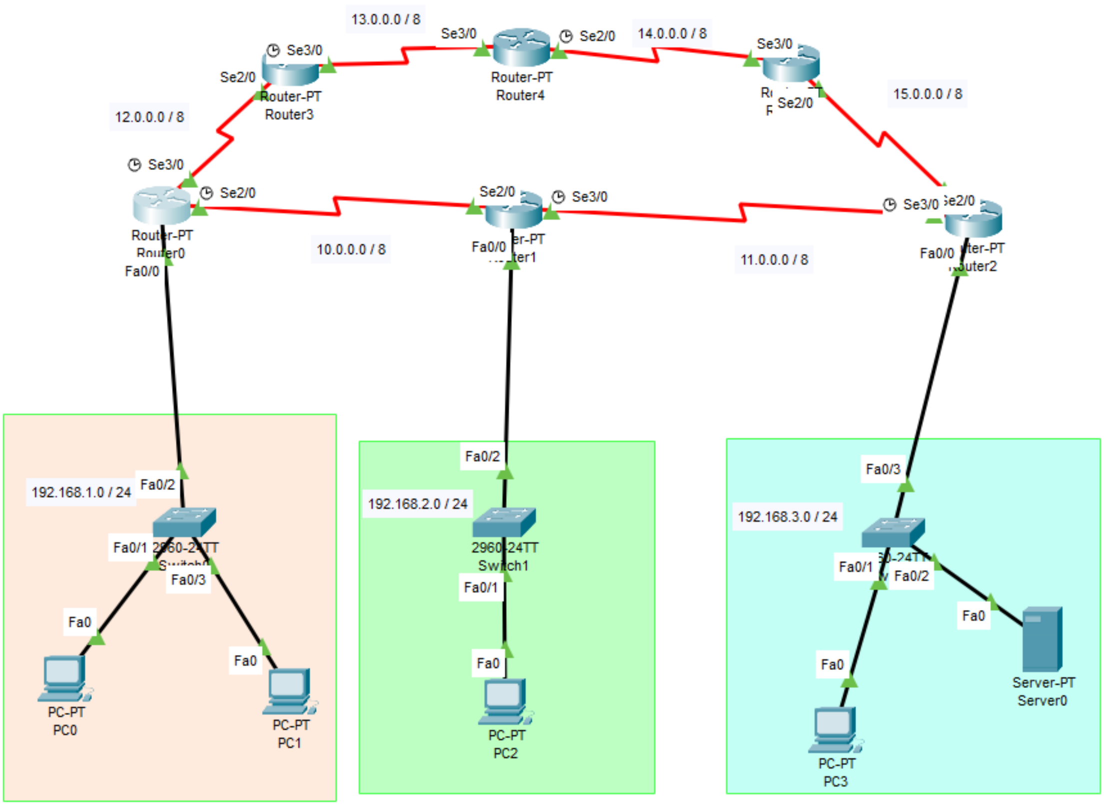


Expand the WAN networks and repeat the process





Trace the route the packet takes for reaching from any network to any network .Provide your inferences on RIP

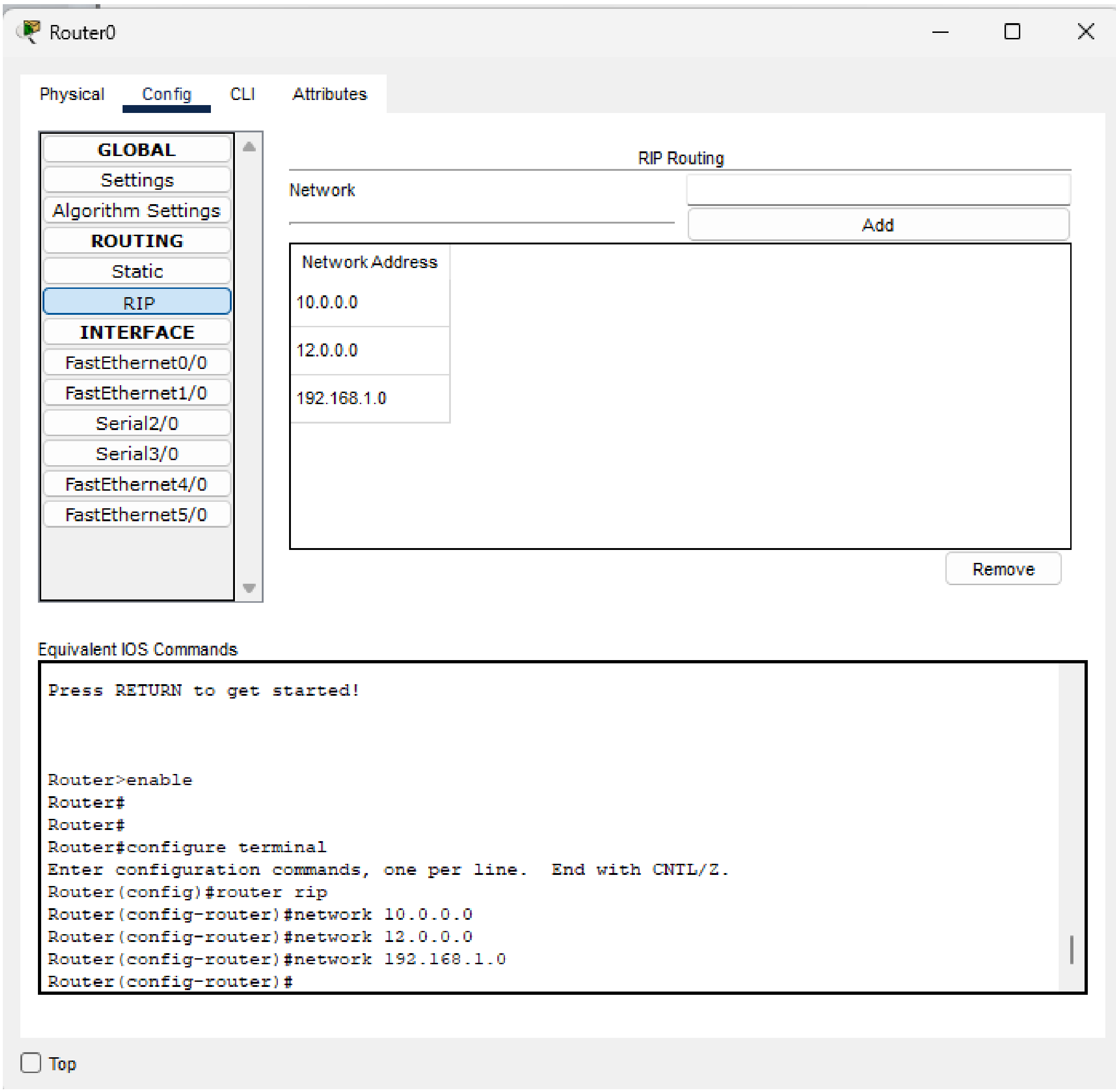


IP ROUTE OF ROUTER 0 BEFORE APPLYING RIP:

```
Router0
Physical    Config    CLI    Attributes
IOS Command Line Interface
%Error parsing filename (Bad file number)
Router#
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface Serial2/0
Router(config-if)#
?Bad filename
%Error parsing filename (Bad file number)
Router(config-if)#ex
Router(config)#sh ip route
                ^
% Invalid input detected at '^' marker.

Router(config)#ex
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#
Router#
Router#show ip route
Codes: C - connected, S - static, I - IGRP, 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

C    10.0.0.0/8 is directly connected, Serial2/0
C    12.0.0.0/8 is directly connected, Serial3/0
C    192.168.1.0/24 is directly connected, FastEthernet0/0

Router#
Ctrl+F6 to exit CLI focus                                   Copy    Paste
```

Router# debug ip rip (Shows the transactions or logs happening in rip)

APPLYING RIP TO ALL THE ROUTERS:



Router0

Physical Config CLI Attributes

IOS Command Line Interface

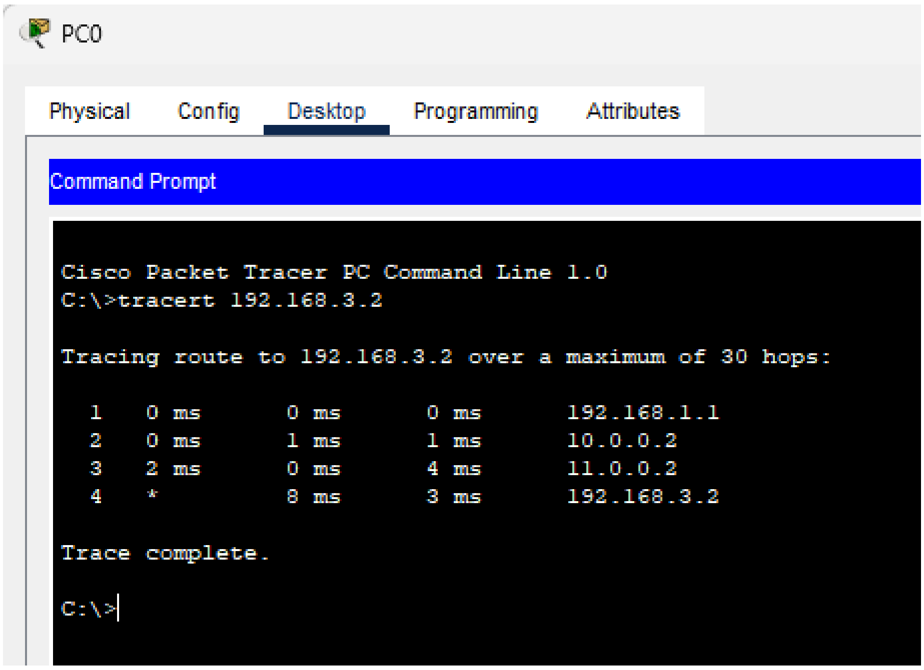
```
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#router rip
Router(config-router)#network 10.0.0.0
Router(config-router)#network 12.0.0.0
Router(config-router)#network 192.168.1.0
Router(config-router)#
Router(config-router)#exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#show ip
% Incomplete command.
Router#show ip route
Codes: C - connected, S - static, I - IGRP, 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

C    10.0.0.0/8 is directly connected, Serial3/0
R    11.0.0.0/8 [120/1] via 10.0.0.2, 00:00:25, Serial3/0
C    12.0.0.0/8 is directly connected, Serial2/0
R    13.0.0.0/8 [120/1] via 12.0.0.2, 00:00:21, Serial2/0
R    14.0.0.0/8 [120/2] via 12.0.0.2, 00:00:21, Serial2/0
R    15.0.0.0/8 [120/2] via 10.0.0.2, 00:00:25, Serial3/0
C    192.168.1.0/24 is directly connected, FastEthernet0/0
R    192.168.2.0/24 [120/1] via 10.0.0.2, 00:00:25, Serial3/0
R    192.168.3.0/24 [120/2] via 10.0.0.2, 00:00:25, Serial3/0

Router#
```

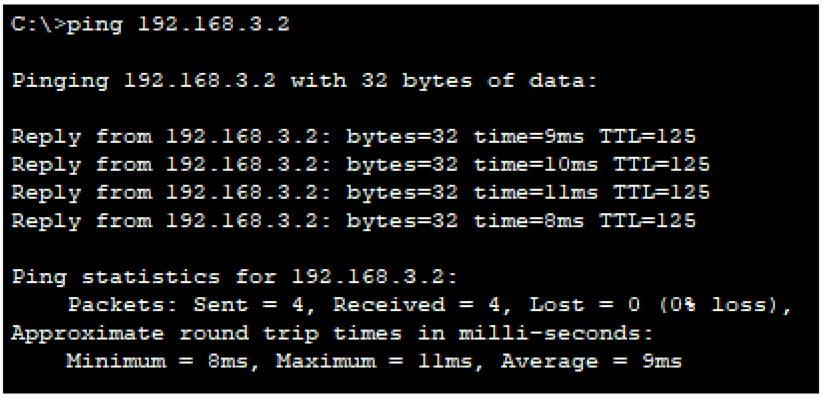
Ctrl+F6 to exit CLI focus

Copy Paste

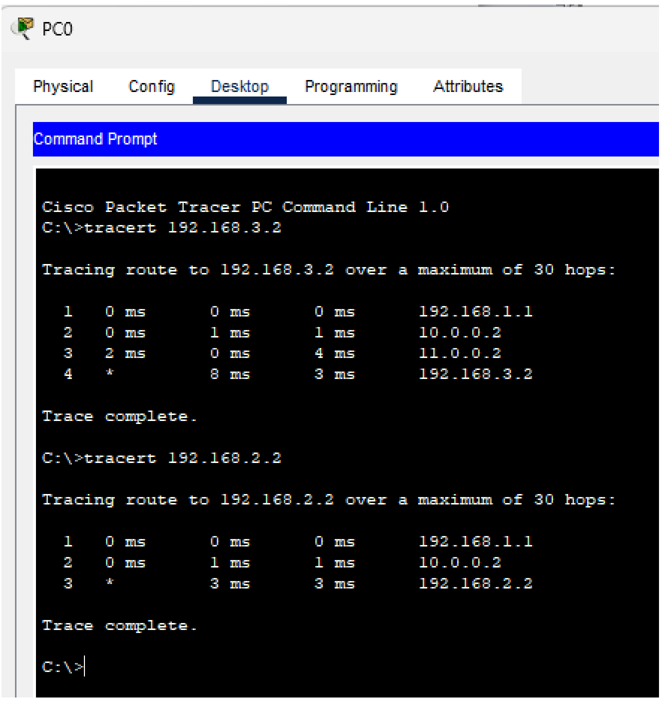
Top

PINGING FROM ONE END DEVICE TO ANOTHER:

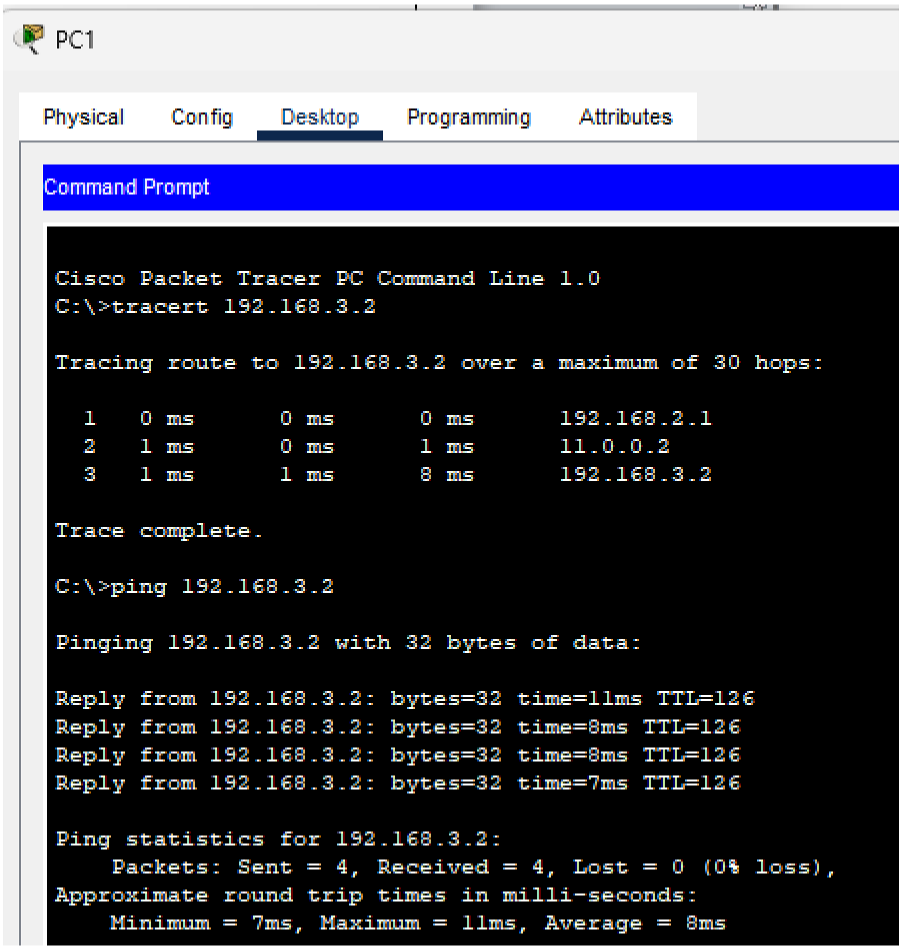




LAN 1 TO LAN 2:



LAN 2 TO LAN 3



Inference:
RIP enables dynamic routing in small to medium-sized networks, using hop count as a metric to determine the best path. It automatically updates routing tables every 30 seconds, allowing routers to adapt to network changes. However, it has scalability limitations due to its reliance on hop count and periodic updates. While it’s easy to configure, RIP has slower convergence compared to other protocols. The trace route tests show RIP’s ability to find efficient paths between networks. Overall, RIP is suitable for simpler networks, but less efficient in larger, more complex setups.

CONCLUSION: Therefore, the above lab experiments are performed and I was able to clearly understand how to use NETSIM, Cisco Packet tracer and Linux and their basic commands. The tools were also helpful in configuring networks.