NetSim®

Accelerate Network R & D

Emulator

A Network Simulation & Emulation Software

By

TETCOS LLP

The information contained in this document represents the current view of TETCOS LLP on the issues discussed as of the date of publication. Because TETCOS LLP must respond to changing market conditions, it should not be interpreted to be a commitment on the part of TETCOS LLP, and TETCOS LLP cannot guarantee the accuracy of any information presented after the date of publication.

This manual is for informational purposes only.

The publisher has taken care in the preparation of this document but makes no expressed or implied warranty of any kind and assumes no responsibility for errors or omissions. No liability is assumed for incidental or consequential damages in connection with or arising out of the use of the information contained herein.

Warning! DO NOT COPY

Contact us at
TETCOS LLP
214, 39th A Cross, 7th Main, 5th Block Jayanagar,
Bangalore - 560 041, Karnataka, INDIA.
Phone: +91 80 26630624
E-Mail: sales@tetcos.com
Visit: www.tetcos.com

Table of Contents

1 Introduction

A network simulator mimics the behaviour of networks but cannot connect to real networks. NetSim Emulator enables users to connect NetSim simulator to real hardware and interact with live applications.

- NetSim emulator is an IP based, data plane, flow-through emulator. This means:
- It can interact with IP based devices.
- It can emulate data place functionality and not control plane functionality.
- The source and destination for traffic should be external. A virtual device within NetSim cannot be a source or sink for traffic.

1.1 Emulation: How Simulation interacts with the real world

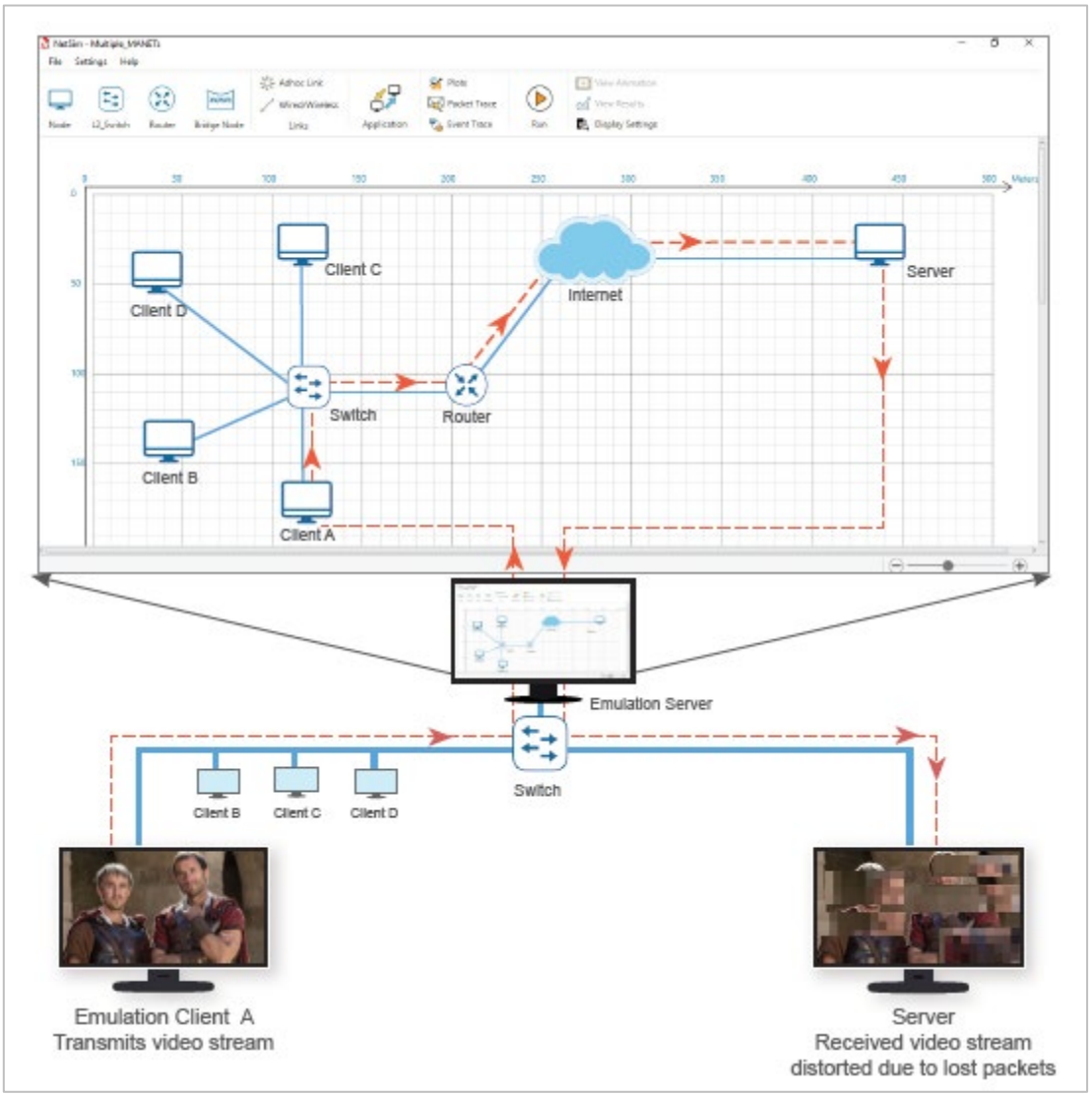


Figure 1-1: Simulator interacting with the real world

A real PC (running NetSim Emulation Client) sends live traffic to the PC (running NetSim Emulation Server). Whenever a packet arrives at the interface of server, this packet is “converted” into a simulation packet and sent from a source node (user selectable) in the simulated network (user configurable) to a destination node (again user selectable). Upon receipt of this packet at the destination, the packet is then “re-converted” and sent back to a real PC destination node (running NetSim Emulation Client). The real packet thus undergoes network effects such as delay, loss, error etc. created virtually by NetSim Simulator.

2 Emulation Set-up

The set-up to run emulation would be to have a minimum of three (3) PC's. One would be the real source, the second would run NetSim emulation server, and the third would be the real destination.

Prerequisite for NetSim Emulation: Enabling IP routing in windows.

IP Routing is the process that allows data to cross over a network of computers rather than just one. Routing is often disabled by default in Windows, to check whether IP routing enabled or not.

Open Command Prompt (cmd.exe) type **ipconfig /all**. It will show if IP Routing Enabled: **Yes\ No**. If IP Routing Enabled is set to **No** we may have to manually enable IP routing as shown Figure 2-1.

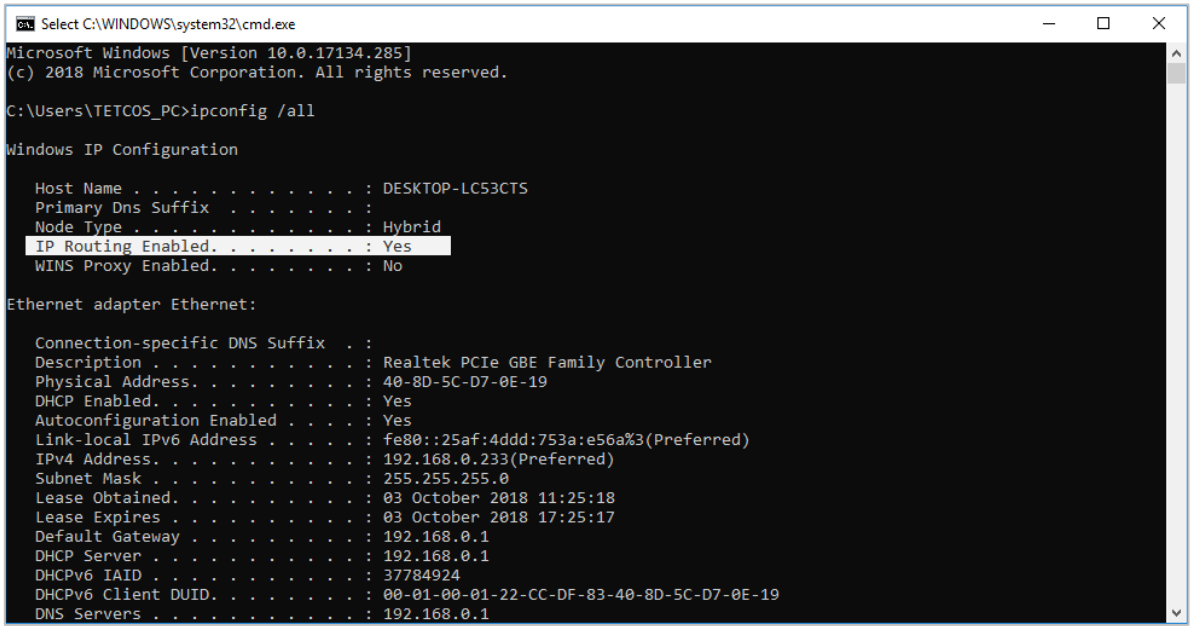


Figure 2-1: IP Routing Enabled

Steps to enable IP routing in windows:

1. Open the start menu, and type REGEDIT32.EXE into the search box. Hit enter. You can also click on "Run" and type REGEDIT to open it.
2. Navigate to the **HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters \IPEnableRouter setting**
3. Right click and select Modify. Change **0 to 1** and click OK then exit the editor as shown below Figure 2-2.

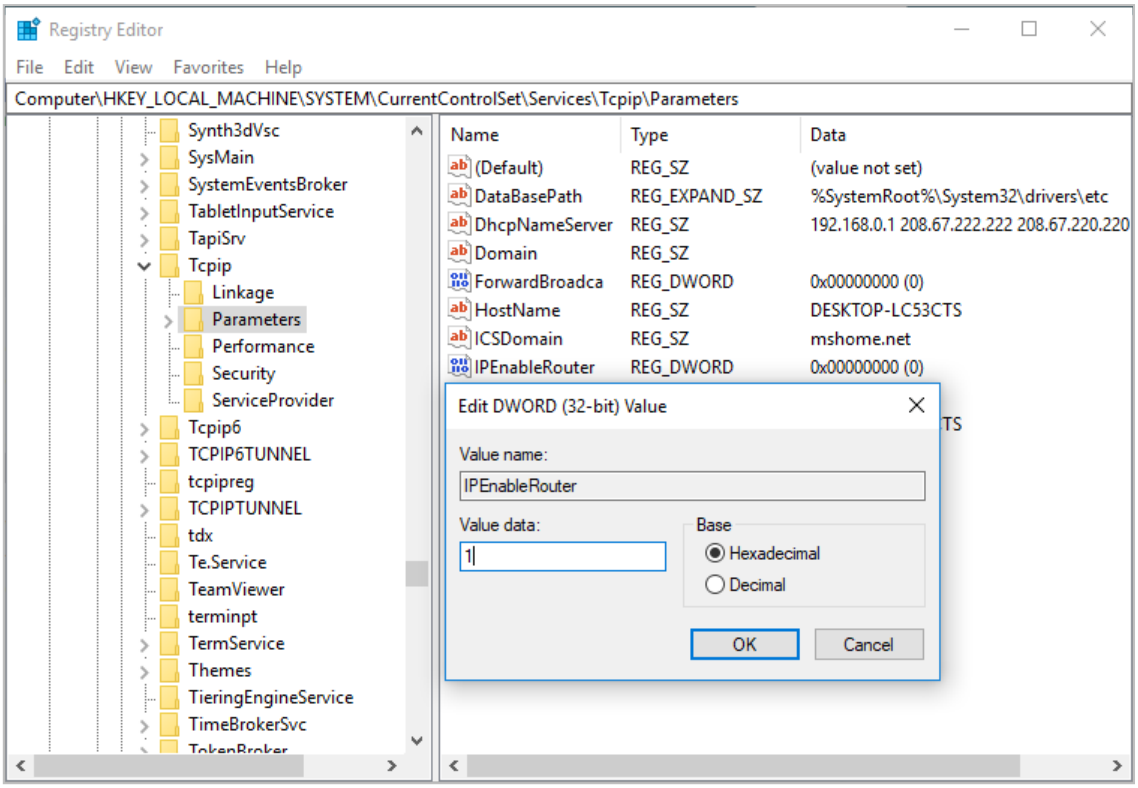


Figure 2-2: IP Enable Router value is set to 1

4. Restart the system and check if IP Enabled Router is set to Yes by using the command **ipconfig /all**

2.1 Setting up the NetSim Server

Note: Never use NetSim License server as Source or destination for emulation application. Also, if the license server is running on Virtual Machine (VM) than any VM on that physical system cannot be used as source or destination. If used, this will lead to license check out problems in the license server.

- Run NetSim in Administrative Mode (Right Click on NetSim.exe → Run as Administrator).
- User has to open any Stack based Network (Any network except Legacy Networks and Cellular Network) in NetSim with Emulation.
- Create a network scenario of your choice (refer application examples provided) and set the Application properties as shown below Figure 2-3.

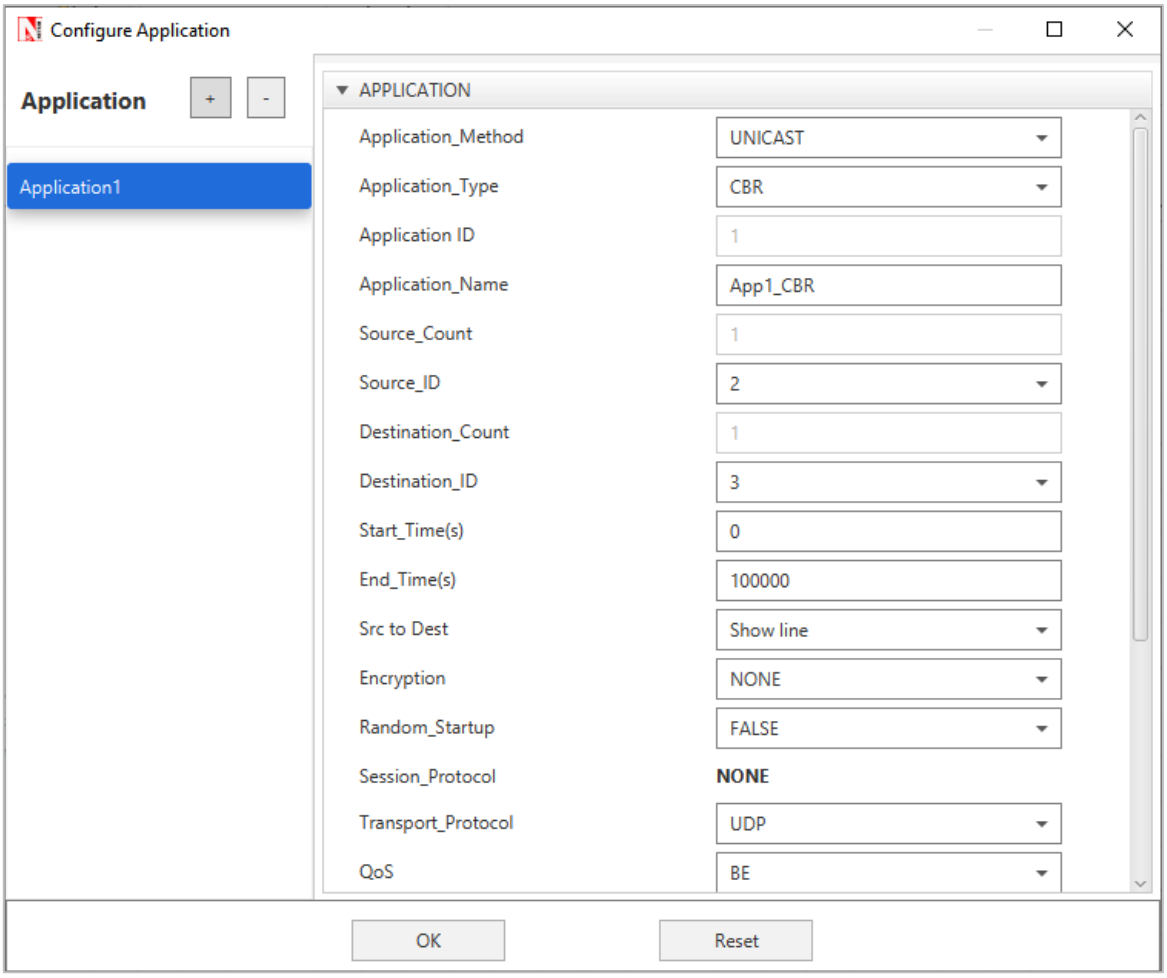


Figure 2-3: Application properties window

- In the Application Properties, set Application Type as “EMULATION”. Assign real Source IP address and Destination IP address in the respective fields. Then Click OK.
- Set the Simulation Time as how long you want to perform the Emulation in Real World. **Do not run the simulation until setting up Emulation in the Client system**.

NOTE: If the Emulation Server is located in a different subnet from clients

- User has to configure the router settings of the real-world network so as to allow the packets to be transmitted to the Emulation Server.
- For Example, if we consider a sample real world network scenario where the Emulation clients and server are located in different subnets as shown Figure 2-4.

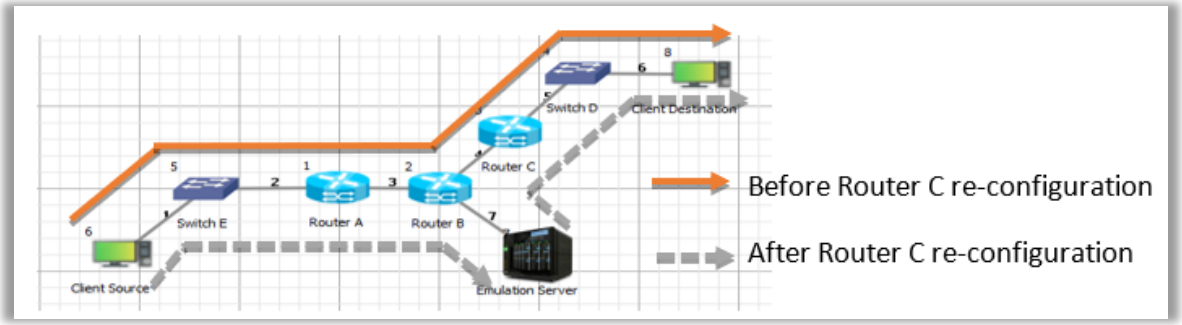


Figure 2-4: Emulation Server is located in a different subnet from clients

- Routing table of router 3 needs to be configured such that any packet having Source Address as IP Address of Node 6(Client Source) and Destination Address as IP Address of Node 8(Client Destination) must be routed to Emulation Server. NetSim configuration will ensure that the packet is re-injected with destination set to the appropriate IP Address (set in the application properties)

2.2 Setting up the Client systems (Real Source and Destination system)

The client systems which are sources of real traffic can be connected to NetSim emulator by resetting the gateway. NetSim Emulator supports both Windows and Linux clients. Once the gateway for the client system is set as the NetSim Emulator PC then traffic from the clients will go via NetSim Emulator PC. The steps involved in configuring the gateway will vary based on the operating system used.

2.2.1 Configuring Windows clients

The following steps can be used to configure the gateway IP address in systems running windows operating system.

- Open command prompt in administrative mode as shown Figure 2-5.

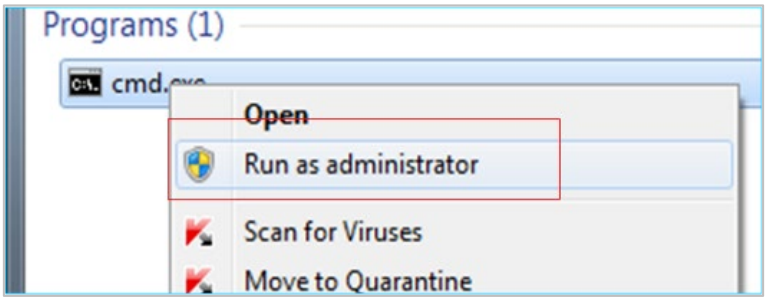


Figure 2-5: Run Command prompt in administrative mode

- Type command,

  route delete <Network Address>

Then press Enter key. You will get “OK”. For example, if your IP address is 192.168.0.4 and the subnet mask is 255.255.255.0 then the network address is 192.168.0.0 (Got by performing a bitwise AND of the IP Address and the subnet mask)

- Type command

  route add <Network Address**>mask 255.255.255.0 <**IP Address where NetSim Emulation server is running**> metric 1**

Here the subnet mask is taken as 255.255.255.0). After execution, you will get “OK”.

- Type command

  netstat –r

To check if the IP configuration is done or not.

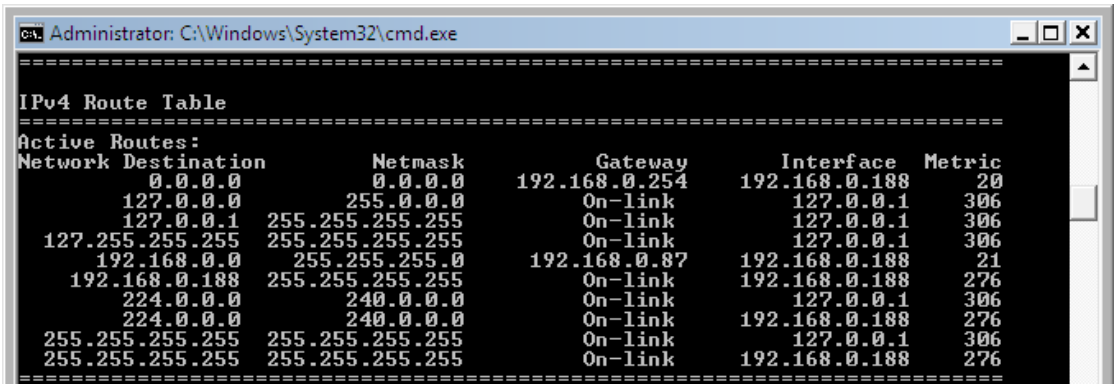


Figure 2-6: Route Table

Note that in the above screenshot, for the network 192.168.0.0, the gateway address assigned is 192.168.0.87(Address of the system where NetSim Emulation Server is running).

2.2.2 Configuring Linux clients running RHEL 7

Go to the Wired Settings option in the Network Adapter Icon.

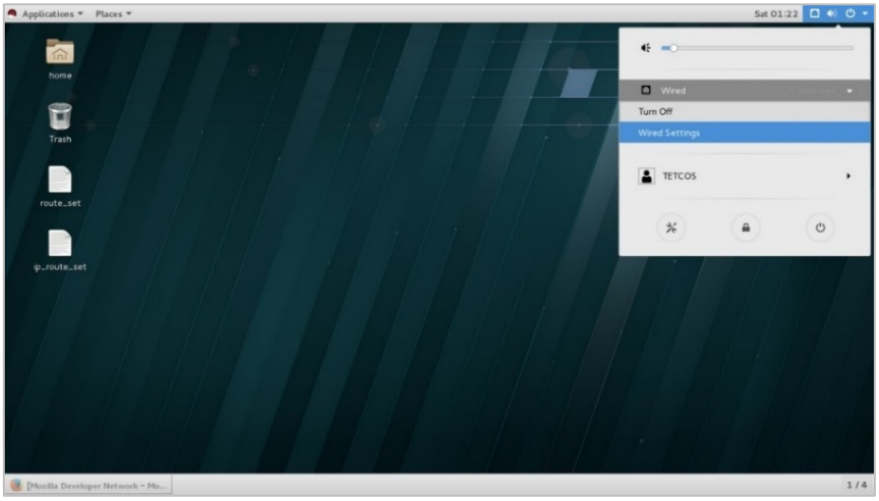


Figure 2-7: Wired Settings option in the Network Adapter Icon

In the IPV4 settings, set static IP Address to the machine and specify the Emulation Server IP as the Gateway IP.

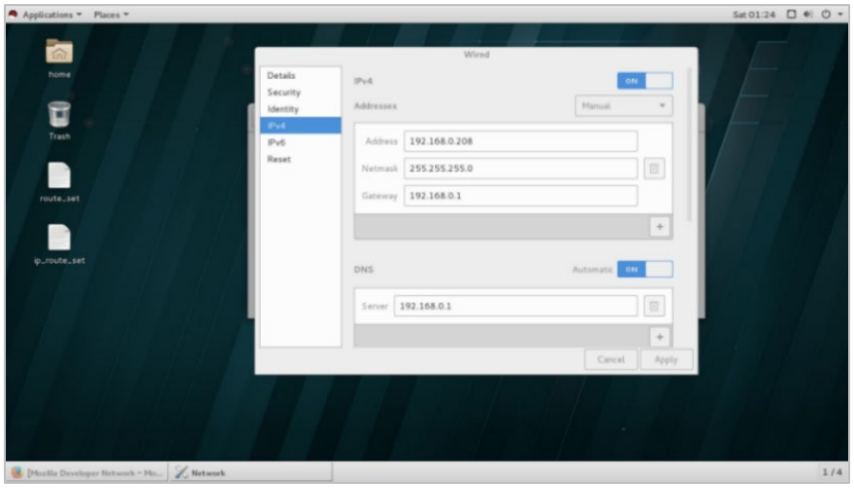


Figure 2-8: Set static IP Address to the machine and specify the Emulation Server IP as the Gateway IP

Example: If 192.168.0.141 is the IP of the system where Emulation Server is running. This is specified as the gateway IP.

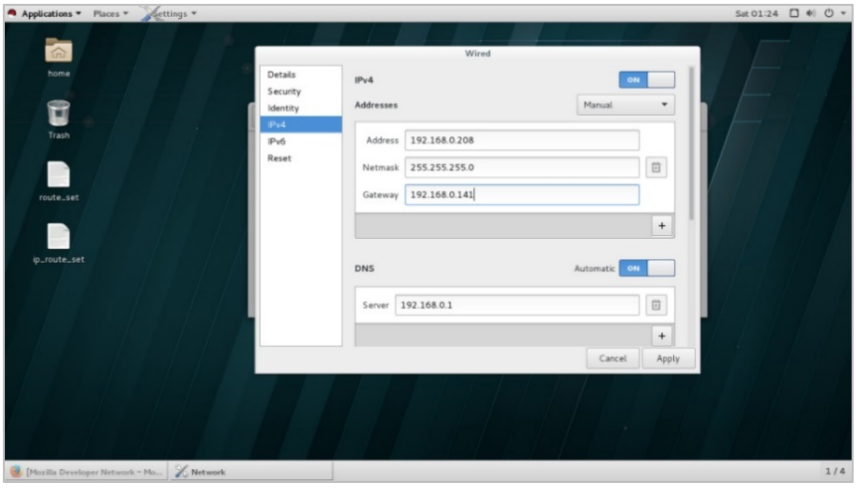


Figure 2-9: Setting up Gateway to Emulation Server IP:192.168.0.141

Turn off Automatic DNS.

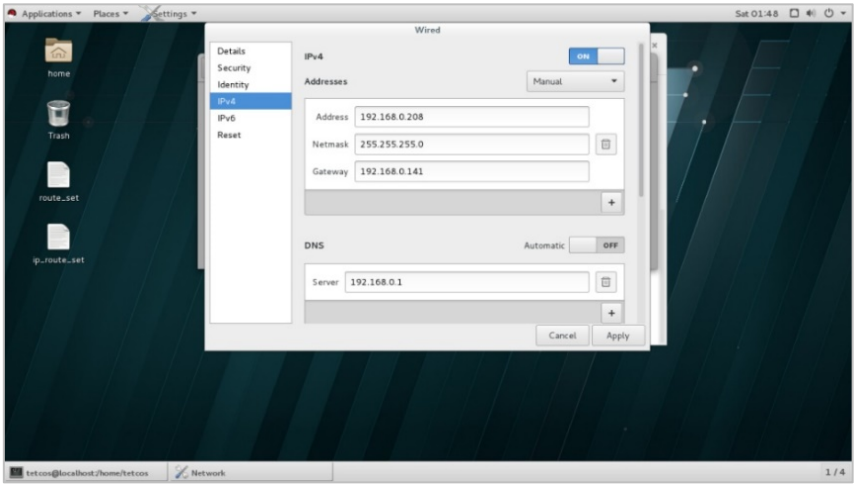


Figure 2-10: Turn off Automatic DNS

Turn off and on the Network Adapter

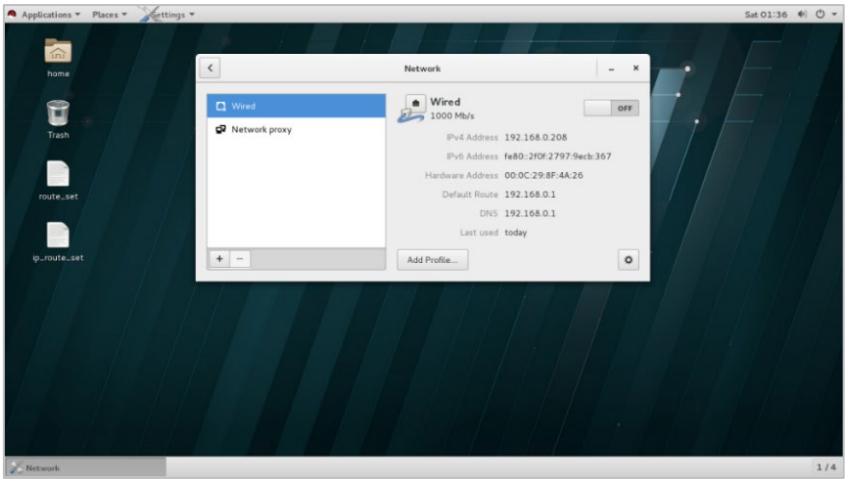


Figure 2-11: Turn off Network Adapter

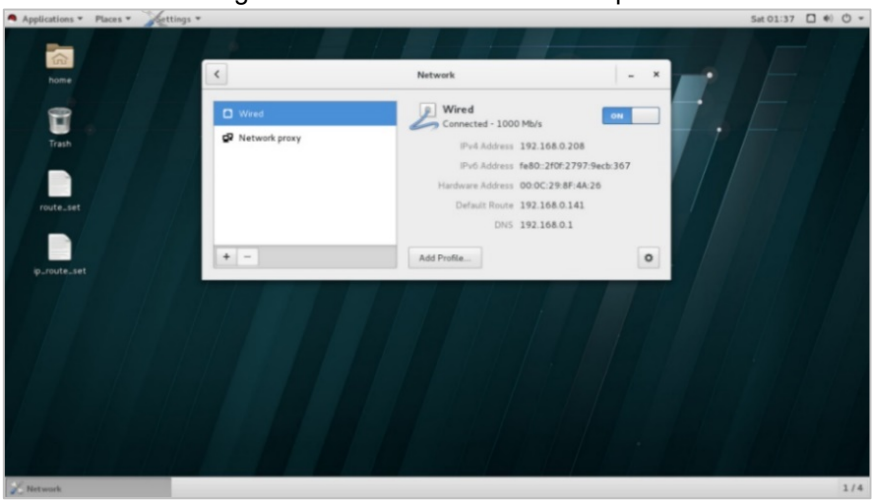


Figure 2-12: Turn ON Network Adapter

Open terminal window

Type command

su

This is to switch to root user.

Enter the root password

Type command

ip route

This is to check the default route

It will now show the default via <Emulation server IP>

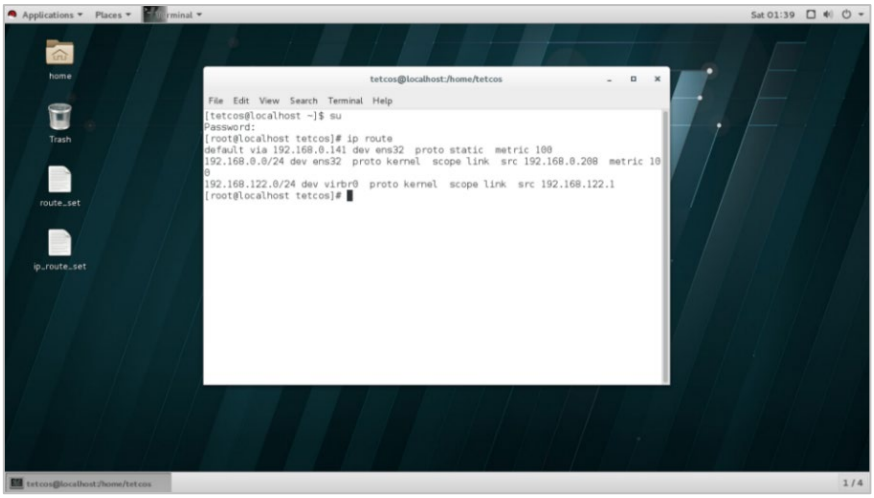


Figure 2-13: Default route

Type command

ip route del <Network Address>

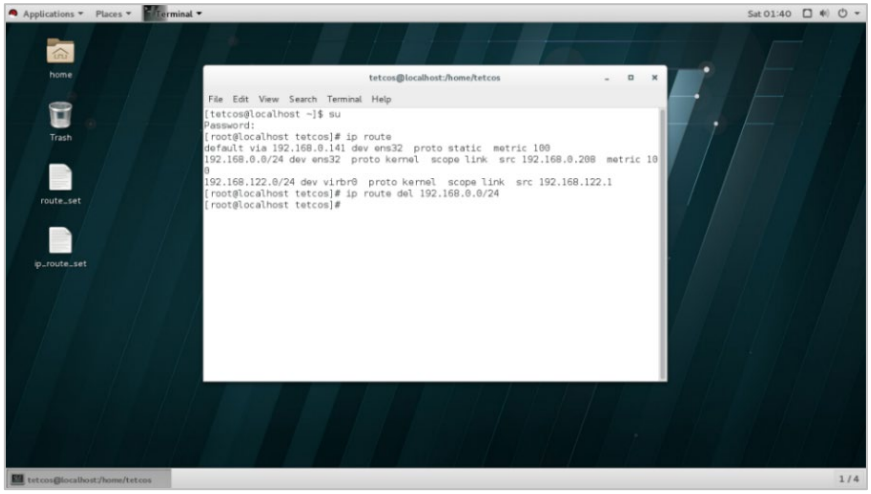


Figure 2-14: ip route del 192.168.0.0/24

Example:

ip route del 192.168.0.0/24

Type command

ip route

This is to check if the IP configuration is done.

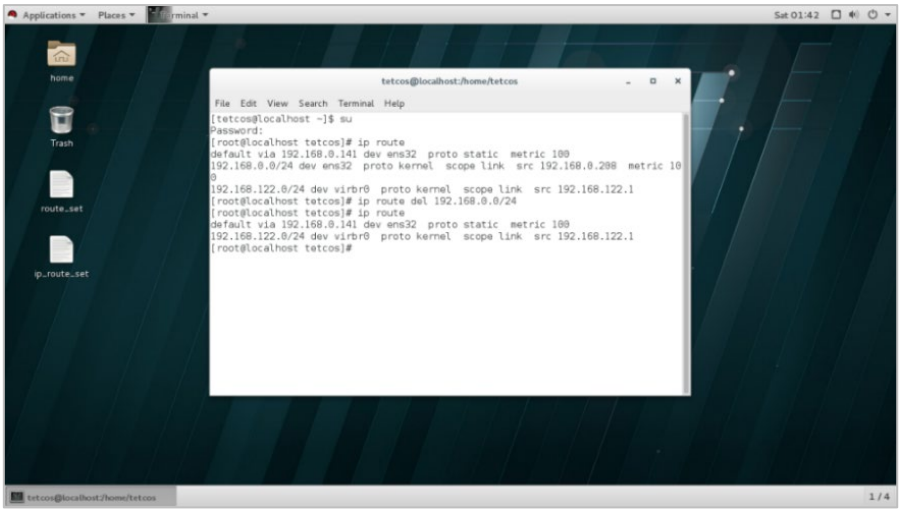


Figure 2-15: IP configuration is done

2.2.3 Configuring Linux clients running Ubuntu

In Ubuntu Environment, if you want to set NetSim Emulator as the gateway, you can use command line tool in your client systems such as

- ip command – show / manipulate routing, devices, policy routing and tunnels.
- route command – show / manipulate the IP routing table.
- Save routing information to a configuration file so that after reboot you get same default gateway.

Any of the following methods can be used for this purpose:

In this example we have considered the NetSim Emulator IP address as 10.244.1.55

- ip command to set a default router to 10.244.1.55
  Login as the root and type:
  # ip route add default via 10.244.1.55
  OR
  $ **sudo ip route add default via 10.244.1.55**
- route command to set a default router to 10.244.1.55
  Login as the root and type:
  # route add default gw 10.244.1.55
  OR
  $ **sudo route add default gw 10.244.1.55**
- Save routing information to a configuration file /etc/network/interfaces
  Open /etc/network/interfaces file
  # vi /etc/network/interfaces
  OR
  $ **sudo vi /etc/network/interfaces**

Find eth0 or desired network interface and add following option

gateway 10.244.1.55

Save and close the file. Restart networking:

 # /etc/init.d/networking restart

OR

$ sudo /etc/init.d/networking restart

After setting static routes through NetSim Emulator, you can use **'traceroute <destination ip>'** command in the client systems to check if the packets are sent via the NetSim emulator.

2.2.4 Configuring Raspberry Pi clients

Open Raspberry PI terminal and apply **"sudo su"**

- Apply **"nano /etc/sysctl.conf"** command and edit the file by adding the following comment

  net.ipv4.ip_forward=1
- To save and Exit

  [Ctrl] + X, then chose yes or no
- Apply "nano /etc/sysctl" command
- Then add the following comments

  a. **IP_DYNIP="no"**

  b. **IP_TCP_SYNCOOKIES="yes"**

  c. **IP_FORWARD="yes"**
- Follow step 3

```
nohook lookup-hostname

interface eth0
inform 192.168.0.124
static routers=192.168.0.53
static domain_name_servers=192.168.0.1
static domain_search=192.168.0.1
```

Figure 2-16: Print interface ID

- Apply "nano /etc/dhcpcd.conf**"**

  a. change the "static routers" to NetSim Server IP as shown in the below image
- Apply **"route"** command

```
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         192.168.0.53    0.0.0.0         UG    202    0        0 eth0
192.168.0.0     *               255.255.255.0   U     202    0        0 eth0
```

Figure 2-17: Route Print

- Apply "ip r del <network ip>/24"

```
root@tetcos:/etc# ip r del 192.168.0.0/24
root@tetcos:/etc# route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         192.168.0.53    0.0.0.0         UG    202    0        0 eth0
```

Figure 2-18: ip r del 192.168.0.0/24

Example: ip r del 192.168.0.0/24

- Apply “ping <any ip within the network>”. Example: ping 192.168.0.202

```
root@tetcos:/etc# ping 192.168.0.202
PING 192.168.0.202 (192.168.0.202) 56(84) bytes of data.
From 192.168.0.53: icmp_seq=1 Redirect Network(New nexthop: 192.168.0.202)
64 bytes from 192.168.0.202: icmp_seq=1 ttl=64 time=0.758 ms
From 192.168.0.53: icmp_seq=2 Redirect Network(New nexthop: 192.168.0.202)
64 bytes from 192.168.0.202: icmp_seq=2 ttl=64 time=0.749 ms
From 192.168.0.53: icmp_seq=3 Redirect Network(New nexthop: 192.168.0.202)
64 bytes from 192.168.0.202: icmp_seq=3 ttl=64 time=0.742 ms
From 192.168.0.53: icmp_seq=4 Redirect Network(New nexthop: 192.168.0.202)
64 bytes from 192.168.0.202: icmp_seq=4 ttl=64 time=0.701 ms
From 192.168.0.53: icmp_seq=5 Redirect Network(New nexthop: 192.168.0.202)
```

Figure 2-19: Pinging to 192.168.0.202

2.3 Setting up the network for client systems communicating across the network

Devices communicating across networks can be connected to NetSim Emulator. This is achieved by connecting the system running NetSim to the routers which connect to the client machines taking part in communication.

2.3.1 System Configuration

Connecting devices across network to NetSim emulator involves configurations at the router and in the NetSim Emulator system. There is no configuration required in the client systems communicating across the network, unlike the case of emulation within the same network.

- The System running NetSim Emulator will require two or more Network Interface Cards (NICs) to connect to different networks.
- Static routes should be set to route packets to specific Network Interfaces as they come in.
- Static routes should be set for routing any external network traffic to the interface that connects to NetSim emulator.

Consider the network shown in the figure below **Figure 2-20**.

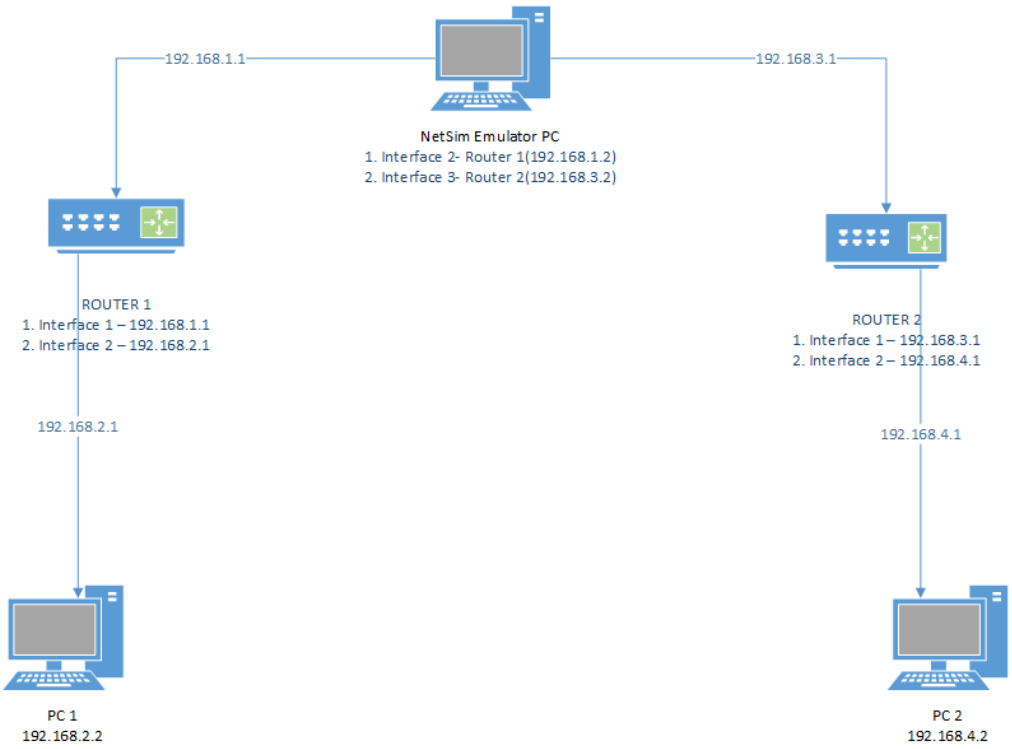


Figure 2-20: LAB Setup

PC 1 and PC 2 are connected to Router 1 and Router 2 respectively. NetSim Emulation PC connects to Router 1 in one of its interface and Router 2 in the other.

To send packets exchanged between PC1 and PC2 via NetSim Emulator, following settings are to be done:

Router 1

Route is added to send any packet to PC 2(192.168.4.2), to NetSim Emulator interface that is connected to it. (Interface with IP 192.168.1.2)

Router 2

Route is added to send any packet to PC 1(192.168.2.2), to NetSim Emulator interface that is connected to it. (Interface with IP 192.168.3.2)

NetSim Emulator

Route is added to send any packet to PC1(192.168.2.2), to its first interface (192.168.1.2)

Route is added to send any packet to PC2(192.168.4.2), to its second interface (192.168.3.2)

After performing the above settings, packets from PC 1 to PC 2 will take the following route:

PC 1 -> NetSim Emulator Interface 1 -> NetSim Emulator Interface 2 -> PC 2

And vice versa for packets from PC 2 to PC 1.

2.4 Performing Multicast Emulation

IP multicast is a method of sending Internet Protocol (IP) datagrams to a group of interested receivers in a single transmission. It is the IP-specific form of multicast and is used for streaming media and other network applications.

NetSim emulator allows sending real traffic via the virtual network modelled in its environment. The real packets will experience the effect of the network conditions modelled in NetSim's virtual network in addition to the real network. This is usually achieved by setting the system hosting NetSim emulator as the gateway in the systems that are the sources of traffic.

Consider the following network setup where there are two systems running JPERF client and server applications and one system running NetSim emulator.

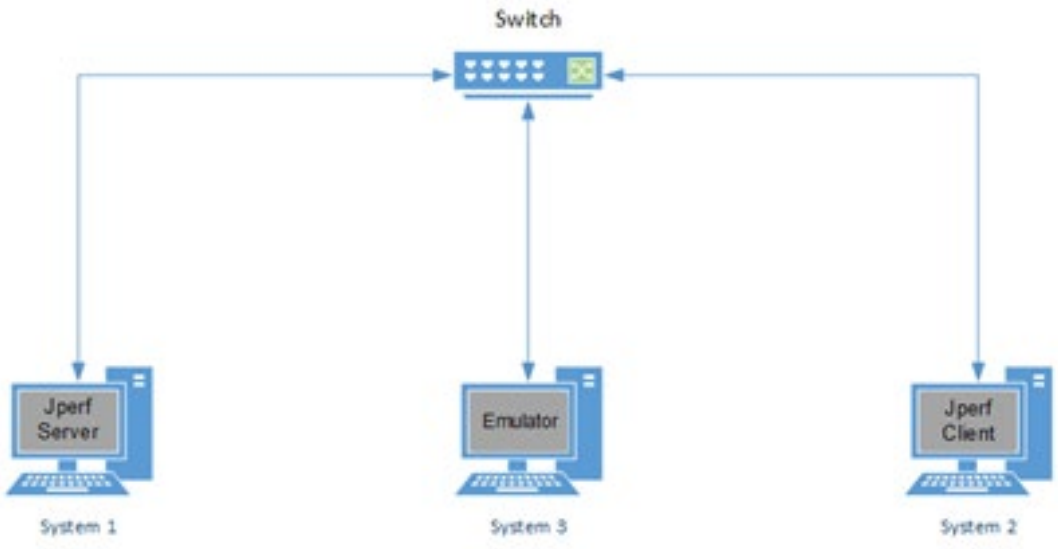


Figure 2-21: JPERF client - server and NetSim Emulator connected to Switch

In the above network the gateway/nexthop of System 1 (and system 2 if traffic is bi-directional) needs to be set as System 3 for packets to reach NetSim Emulator prior to reaching the destination. NetSim can then take care of sending the real packets over the virtual network modelled in its design environment.

However, this works only in case of simple application models such as Unicast. In case of application models such as Multicast NetSim offers different techniques to ensure that the packets go through the emulated network prior to reaching the destination devices.

NetSim provides a Multicast Emulation Client application specifically to perform multicast emulation. This application can be found in the install directory of NetSim within as shown below:

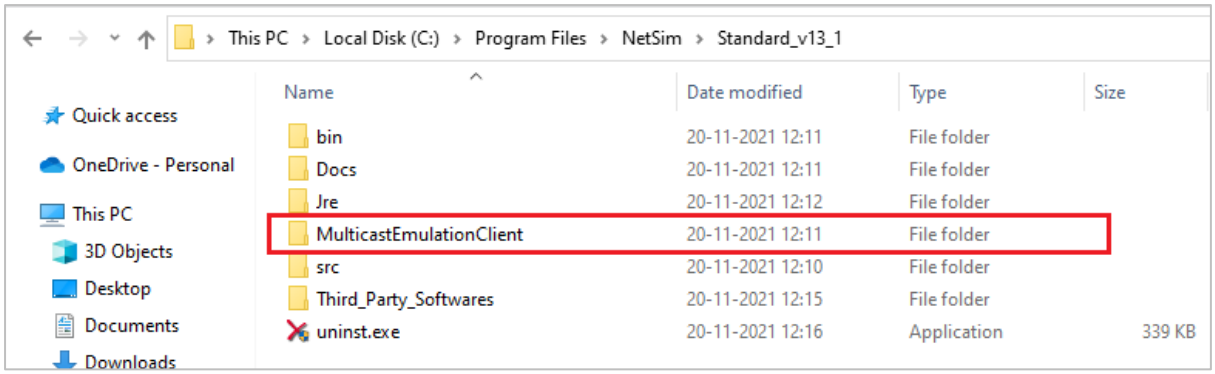


Figure 2-22: NetSim Installation Directory

Following is an example related to performing multicast emulation in NetSim. This example involves 4 real PC's

System Setup:

- SYSTEM 1 - NetSim Emulator (IP:192.168.0.34)
- SYSTEM 2 - Source (iperf Client) (IP:192.168.0.11)
- SYSTEM 3 - Destination 1 (iperf server 1) (IP:192.168.0.36)
- SYSTEM 4 - Destination 2 (iperf server 2) (IP: 192.168.0.19)
- All 4 systems must be in a single network.

System 1:

1. Install NetSim Standard version including Emulator License
2. Run NetSim in Administrative Mode, open Internetworks create a scenario as shown below Figure 2-23.

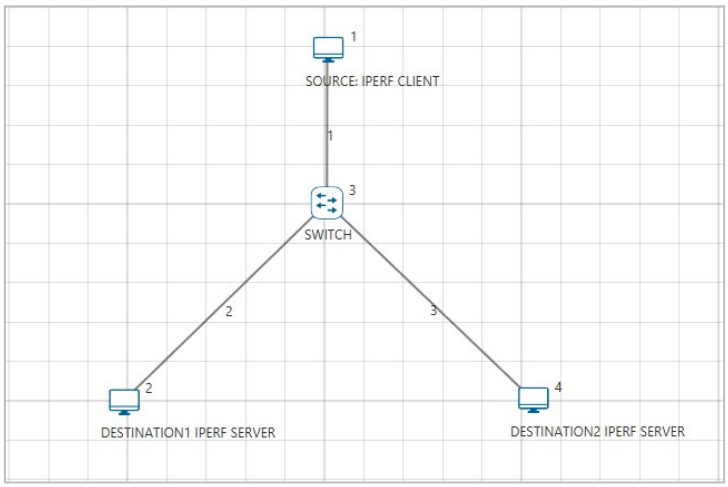


Figure 2-23: Network Topology

3. Create Multicast Emulation with following properties:
   - Application Method: Multicast
   - Application Type: Emulation

- Device Count: Count of devices which act as multicast server and client. 3 in this case (excludes the system hosting NetSim emulator, includes the Iperf client and two Iperf servers)
- Device ID: The ID's of the virtual devices representing the multicast clients and servers.
- Multicast Dest Address: The multicast IP address to which nodes are listening.
- Device Real IP: The IP Address of all the systems involved in Multicast (Excluding the system running NetSim emulator)
- Destination Real IP: The multicast IP Address
- Source Port: Can be set to 0.
- Destination Port: The port to which the server is listening.

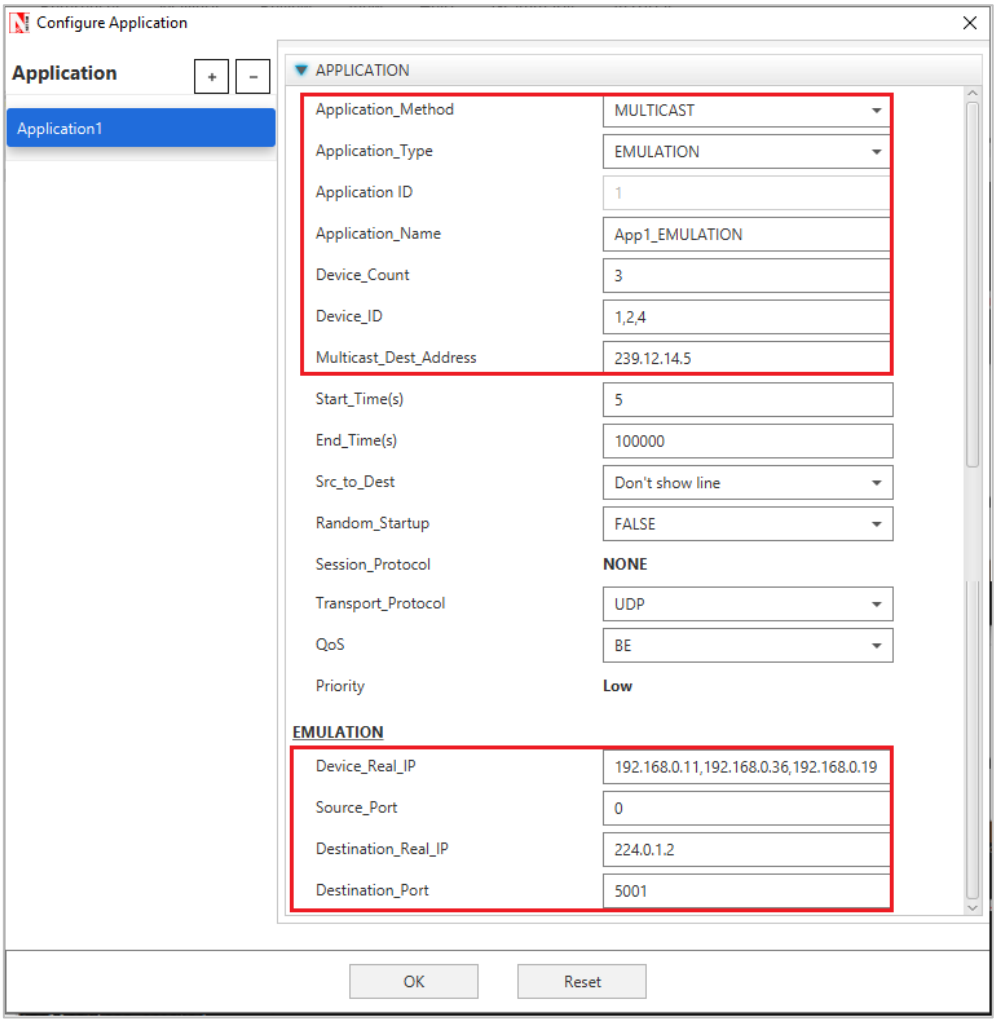


Figure 2-24: Application Configuration Window

4. Configure all other systems in the Network and the perform simulation.
5. Set IGMP status as true under Network Layer of all nodes.
6. Run the simulation.

System 2:

1. Copy Multicast Emulation Client folder from NetSim installation directory to the current system.

   E.g.: C:\Program Files\NetSim Standard\MulticastEmulationClient
2. Run iperf with the following command in Admin Mode:

   iperf -c 224.0.1.2 -u -i 1 -t 1000
3. Run multicastEmulationclient in client system in command prompt in admin mode:

Command:

MulticastEmulationClient.exe <Client system Ip> <Emulation server Ip> <Multicast Ip> <port>

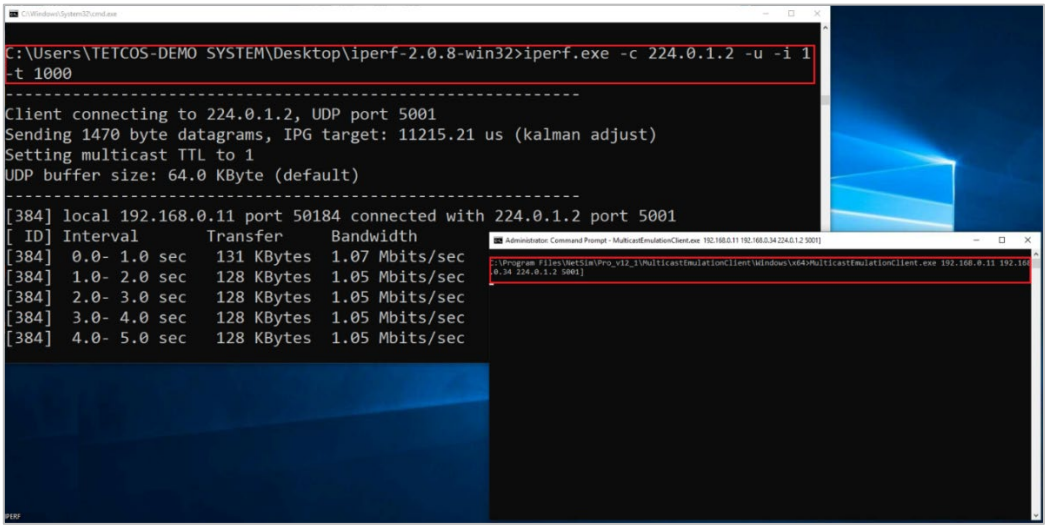


Figure 2-25: Iperf- Multicast command for system 2

System 3 and System 4:

1. Copy Multicast Emulation Client folder from NetSim installation directory to the current system.

   E.g.: C:\Program Files\NetSim Standard\MulticastEmulationClient
2. Run jperf with the following command in Administrative Mode:

   iperf -s -B 224.0.1.2 -u -i 1
3. Run multicastEmulationclient in client system command prompt in admin mode:

Command:

MulticastEmulationClient.exe <Client system Ip> <Emulation server Ip> <Multicast Ip> <port>

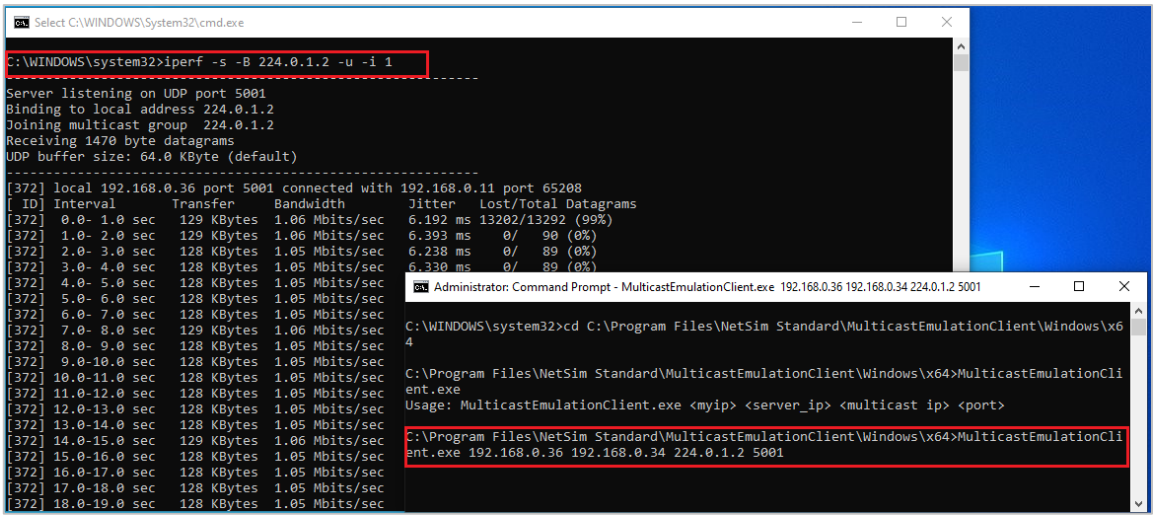


Figure 2-26: Iperf- Multicast command for system 3 and 4

Now perform Step 5 of System 1 i.e. Run the simulation in System 1. User will be asked to Enter an option to choose local IP while simulation. Please enter the option that displays the server-IP given as <multicast ip> for MulticastEmulationClient.exe. In this example its 192.168.0.34.

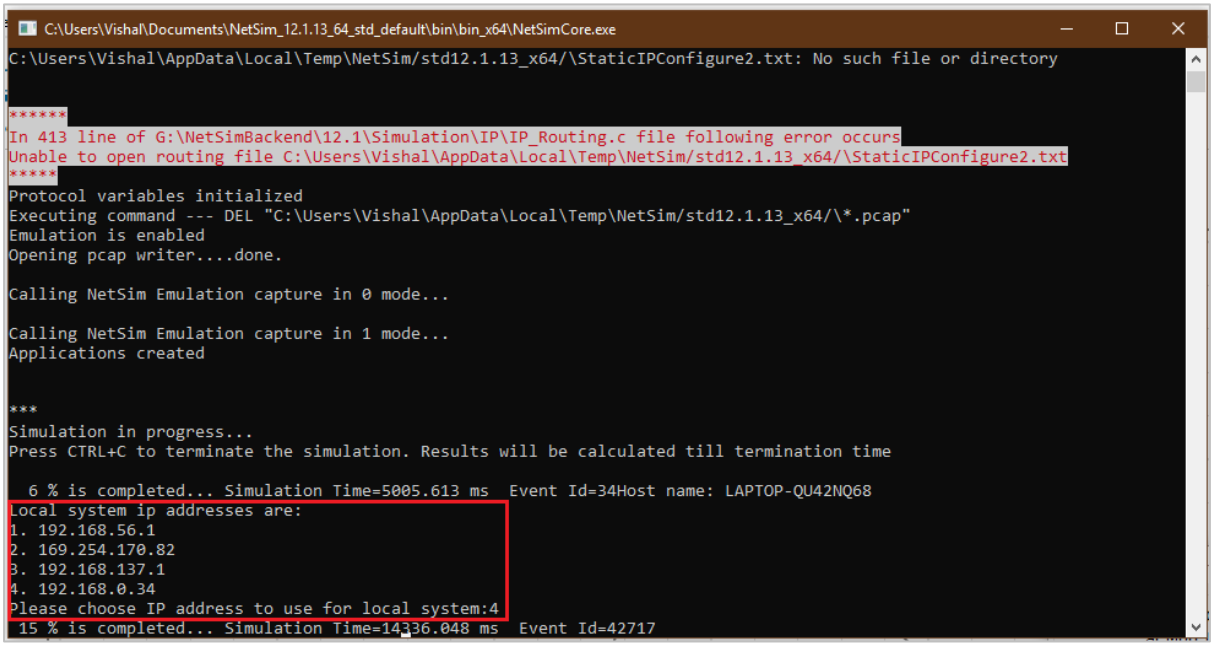


Figure 2-27: Choose Multicast IP during Emulation

On successful completion of Simulation, NetSim provides performance metrics in the Simulation Results dashboard.

2.5 Setting up the network for Database PostgreSQL Emulation

Devices running Database Management Systems such as PostgreSQL can be connected to NetSim Emulator. This is achieved by connecting the systems running the database server, the database web client and the system running NetSim Emulator to a switch.

2.5.1 System Configuration

Connecting devices involved in database application to NetSim emulator involves configurations at both the systems that run the database server and the web client.

- Static routes should be set to route all outgoing packets to the system running NetSim Emulator.

Consider the network shown in the figure below Figure 2-28.

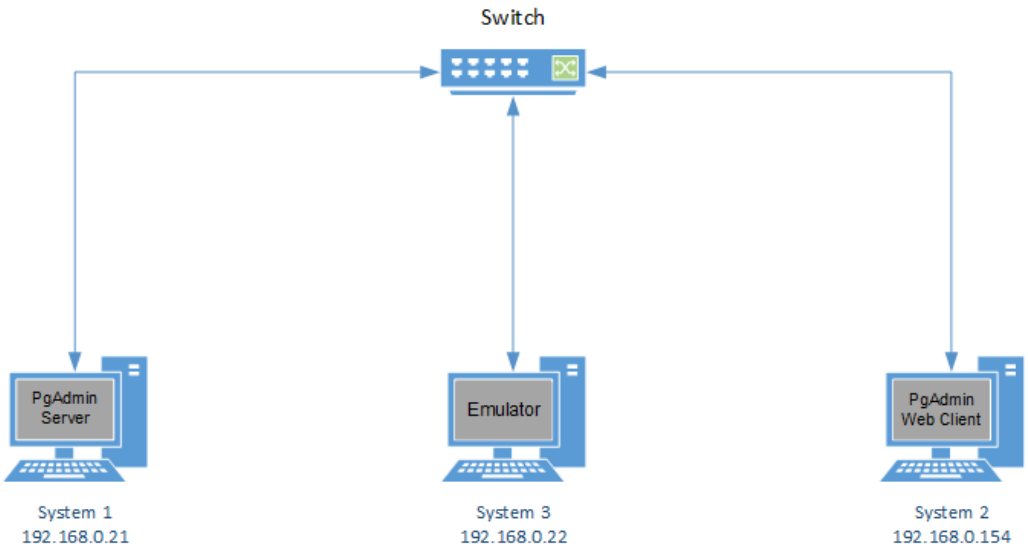


Figure 2-28: PgAdmin web client and server with NetSim Emulator to switch

System 1, System 2 and System 3 are connected to a L2 switch and are part of the same network.

To send packets exchanged between System 1 and System 2 via System 3(NetSim Emulator), following settings are to be done:

System 1

Route is added to send any packet to System 2(192.168.0.154), to NetSim Emulator interface that is connected to it. (Interface with IP 192.168.0.22)

System 2

Route is added to send any packet to System 1(192.168.0.21), to NetSim Emulator interface that is connected to it. (Interface with IP 192.168.0.22)

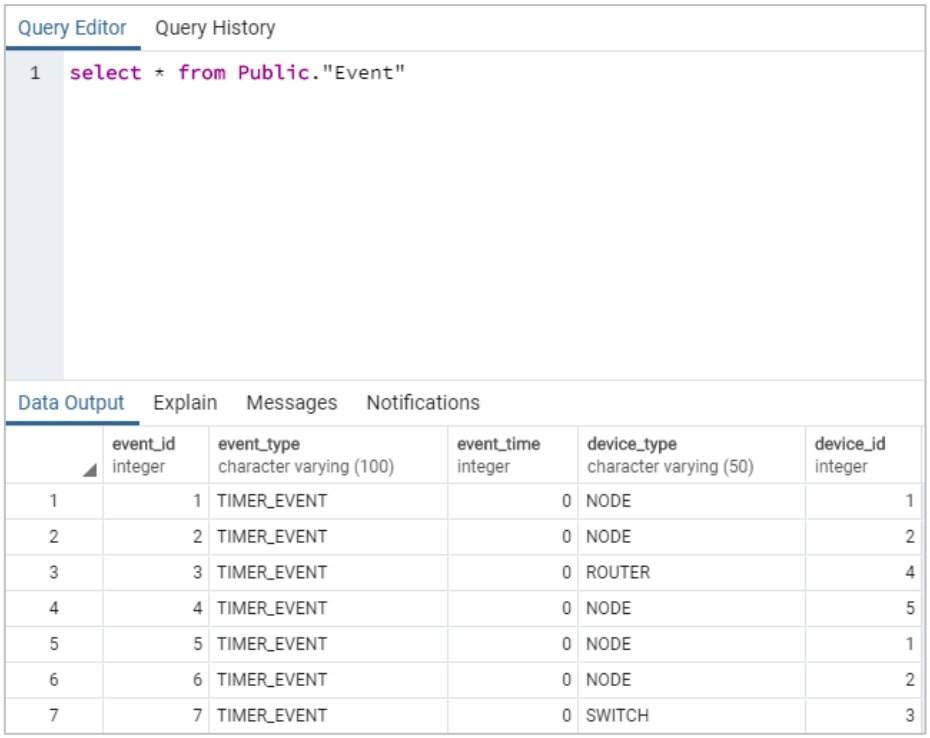
After performing the above settings, packets from PC 1 to PC 2 will take the following route:

System 1(PgAdmin Server) -> **System 3**(NetSim Emulator)-> **System 2**(PgAdmin WebClient)

And vice versa for packets from System 2 to System 1.

2.5.2 Steps to Start generating Network Traffic

- Start PgAdmin application in System 1 and create a database with 100000 Records in it.
- Now open PgAdmin application in System 2 and connect to the database created in System 1. In the example shown below, we are attempting to retrieve the records from a large csv file.



	event_id integer	event_type character varying (100)	event_time integer	device_type character varying (50)	device_id integer
1	1	TIMER_EVENT	0	NODE	1
2	2	TIMER_EVENT	0	NODE	2
3	3	TIMER_EVENT	0	ROUTER	4
4	4	TIMER_EVENT	0	NODE	5
5	5	TIMER_EVENT	0	NODE	1
6	6	TIMER_EVENT	0	NODE	2
7	7	TIMER_EVENT	0	SWITCH	3

Figure 2-29: PgAdmin application in System 2

2.5.3 Steps to configure application for NetSim Emulation

- Once the database connection is established, In System 3, create a simple scenario in internetworks as shown below:

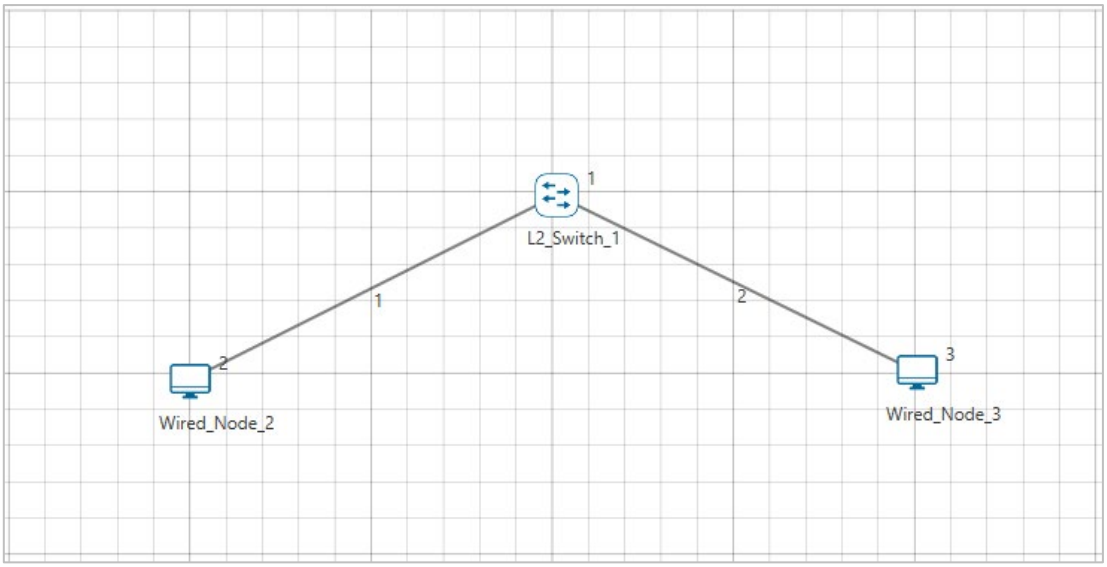


Figure 2-30: Network Topology

- Configure an Emulation Application from Wired Node 2 to Wired Node 3 as shown below:

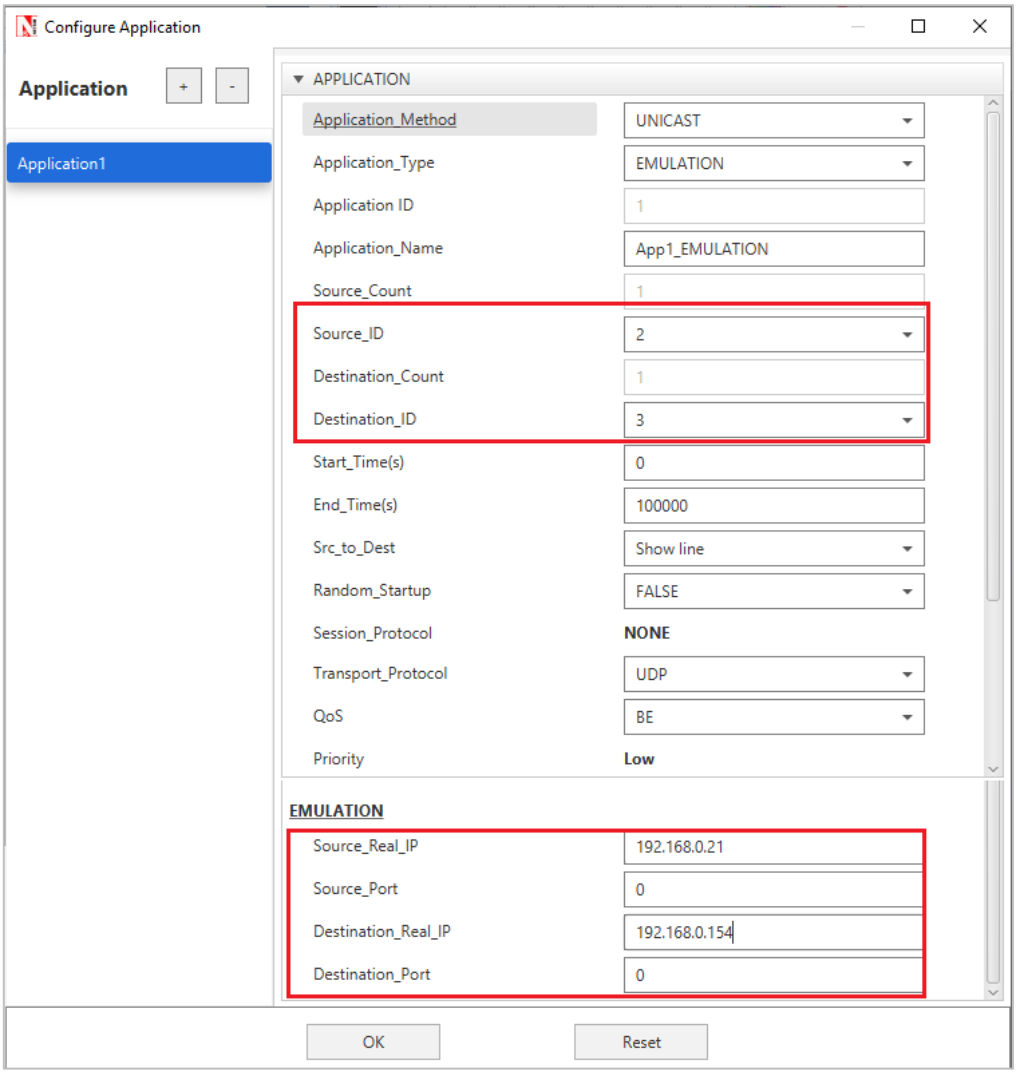


Figure 2-31: Application properties window for Application 1

- Configure an Emulation Application from Wired Node 3 to Wired Node 2 as shown below:

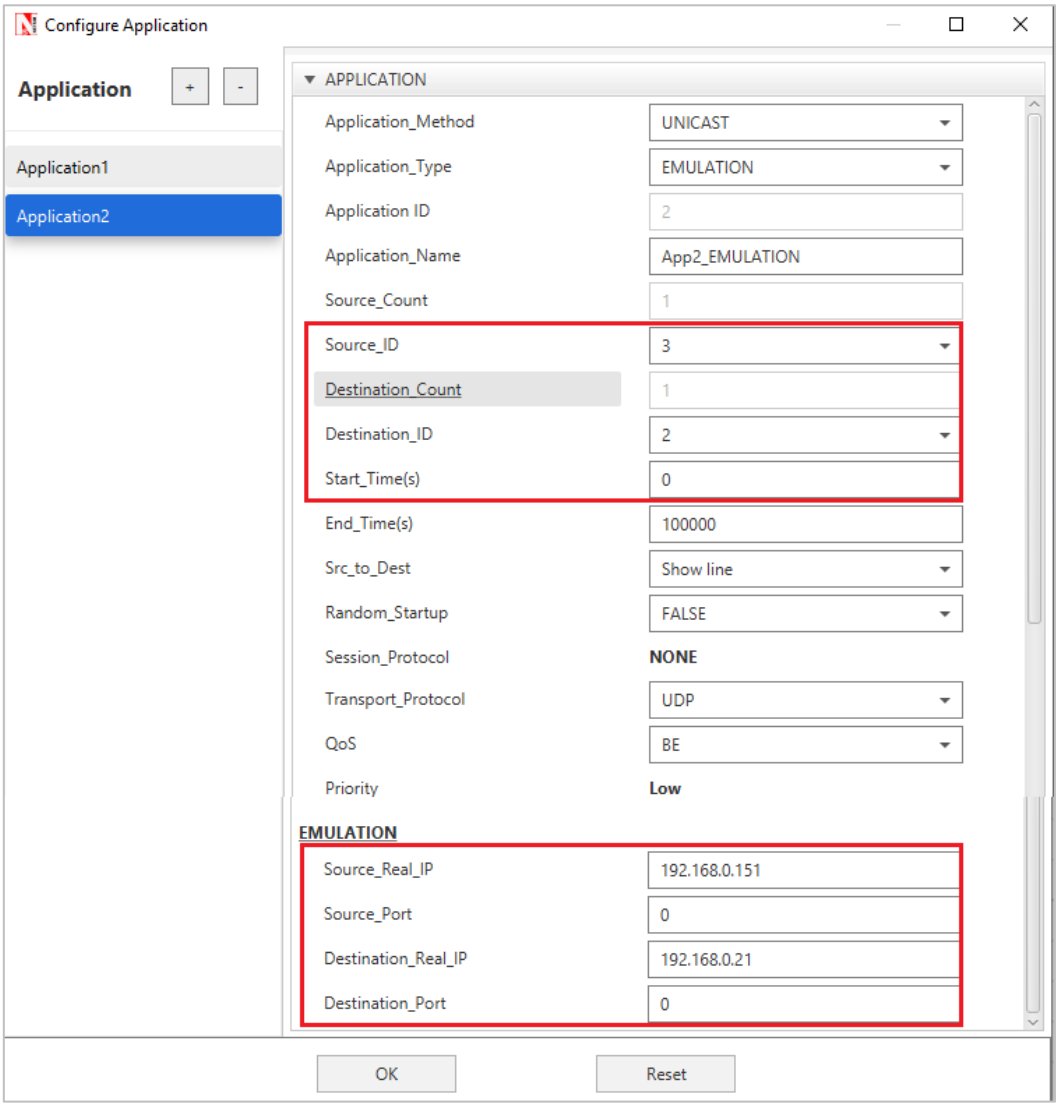


Figure 2-32: Application properties window for Application 2

The above settings will ensure that packets from System 1 to System 2 and vice versa will be sent via NetSim Emulator.

2.5.4 Results and Analysis

- Start Simulation in NetSim
- Perform any query on the database while NetSim simulation is running.
- Analyse the impact of flow of the packets through the network designed in NetSim based on the time taken for the query response.
- The pcap log files such as All Network Packets Capture, Dispatched to Emulator, Reinjected from Emulator, Not Dispatched to Emulator that are accessible from the NetSim results dashboard after the simulation is over, can be used for further analysis.

2.6 Setting up the network for GeoServer Application Emulation

Devices running Database Management Systems such as PostgreSQL can be connected to NetSim Emulator. This is achieved by connecting the systems running the database server, the database web client and the system running NetSim Emulator to a switch.

2.6.1 System Configuration

Connecting devices involved in database application to NetSim emulator involves configurations at both the systems that run the database server and the web client.

- Static routes should be set to route all outgoing packets to the system running NetSim Emulator.

Consider the network shown in the figure below Figure 2-33.

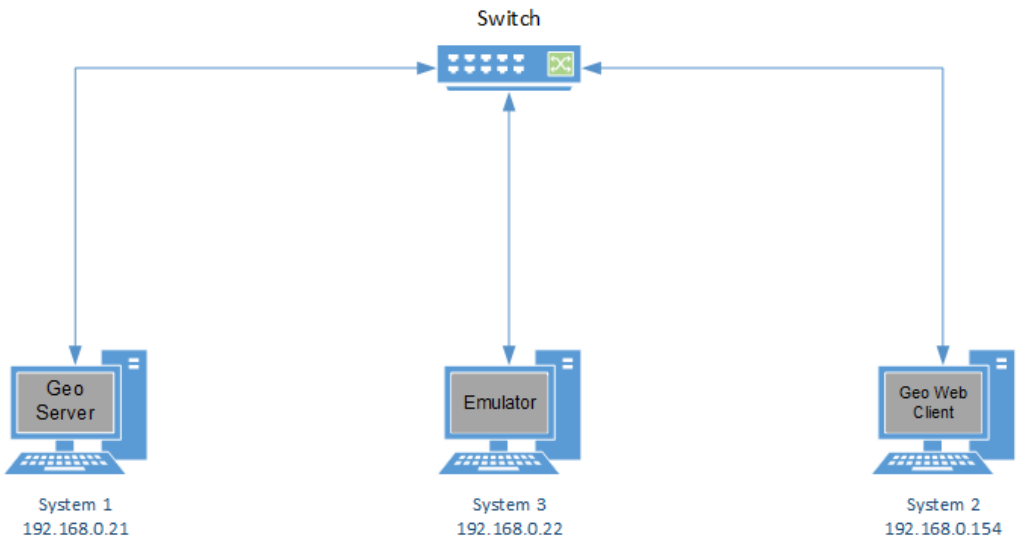


Figure 2-33: Geo web client and server applications nd server with NetSim Emulator to switch

System 1, System 2 and System 3 are connected to a L2 switch and are part of the same network.

To send packets exchanged between System 1 and System 2 via System 3(NetSim Emulator), following settings are to be done:

System 1

Route is added to send any packet to System 2(192.168.0.154), to NetSim Emulator interface that is connected to it. (Interface with IP 192.168.0.22)

System 2

Route is added to send any packet to System 1(192.168.0.21), to NetSim Emulator interface that is connected to it. (Interface with IP 192.168.0.22)

After performing the above settings, packets from PC 1 to PC 2 will take the following route:

System 1(Geo Server) -> **System 3**(NetSim Emulator)-> **System 2**(Geo WebClient)

And vice versa for packets from System 2 to System 1.

2.6.2 Steps to Start generating Network Traffic

- Start GeoServer application in System 1
- Now access GeoServer via web browser in System 2 by specifying IP address: GeoServerPort//Geoserver/browser. Eg:192.168.0.21:2020/Geoserver/browser

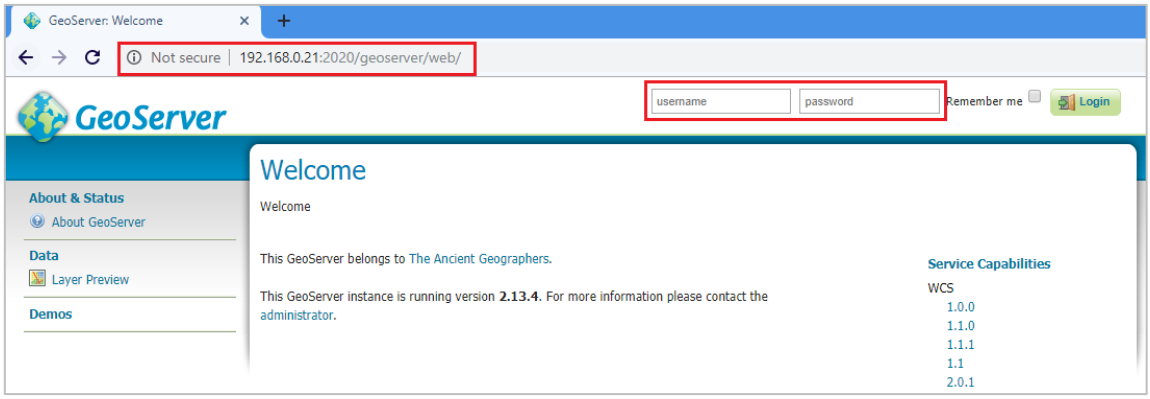


Figure 2-34: GeoServer application in System 2

2.6.3 Steps to configure application for NetSim Emulation

- Once the database connection is established, In System 3, create a simple scenario in internetworks as shown below Figure 2-35.

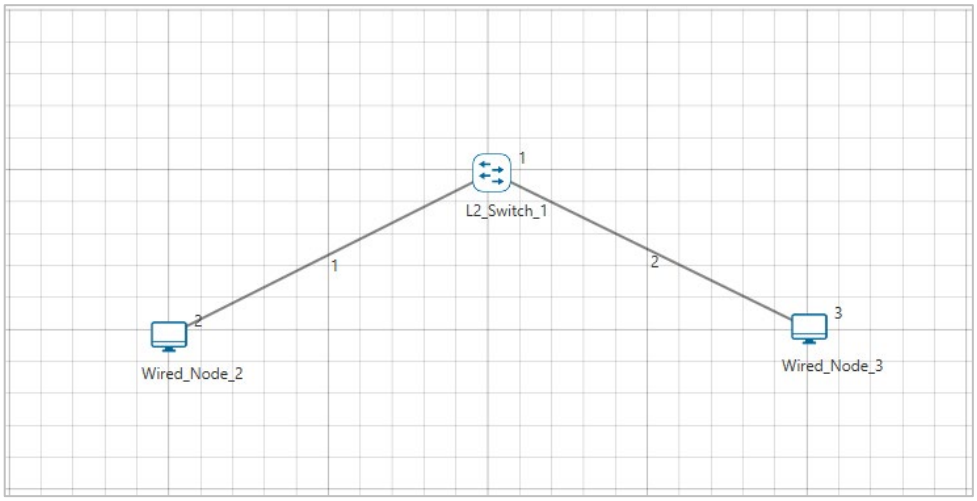


Figure 2-35: Network Topology

- Configure an Emulation Application from Wired Node 2 to Wired Node 3 as shown below:

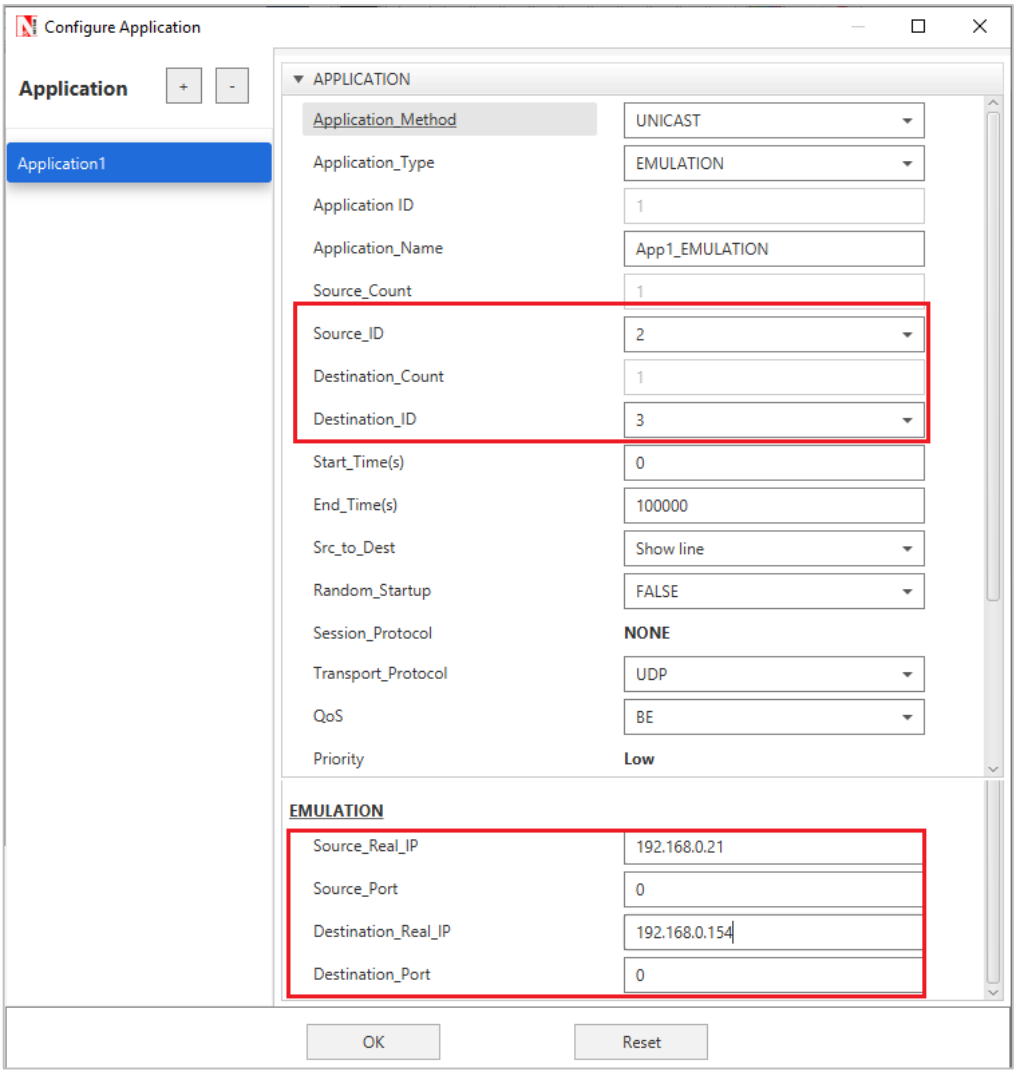


Figure 2-36: Application properties window for Application 1

- Configure an Emulation Application from Wired Node 3 to Wired Node 2 as shown below:

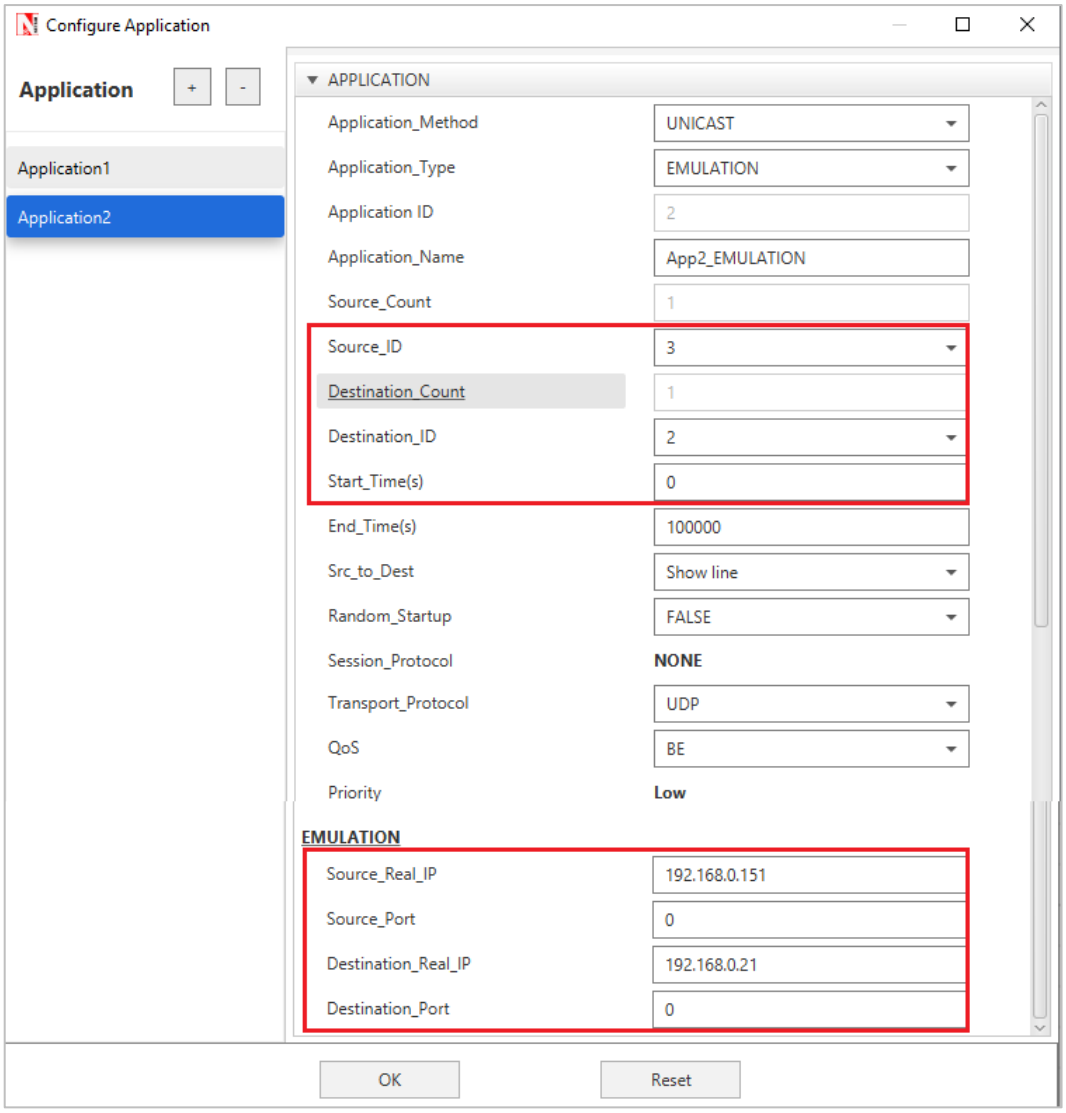


Figure 2-37: Application properties window for Application 2

The above settings will ensure that packets from System 1 to System 2 and vice versa will be sent via NetSim Emulator.

2.6.4 Results and Analysis

- Start Simulation in NetSim
- Try to access specific regions of the map by zooming in and other operations to generate live network traffic.
- Analyse the impact of flow of the packets through the network designed in NetSim based on the time taken for the map to load.
- The pcap log files such as All Network Packets Capture, Dispatched to Emulator, Reinjected from Emulator, Not Dispatched to Emulator that are accessible from the NetSim results dashboard after the simulation is over, can be used for further analysis.

2.7 Setting multiple Virtual Machines (VM) to act as Nodes for Emulation

2.7.1 VMs sharing the same network as the host

A computer on which one or more virtual machines are running is defined as a Host Machine. Each virtual machine is called a Guest Machine. In this scenario, we have 3 VMs running in a Host Machine – VM1, VM2 and VM3. Users can run NetSim License server in any system connected to the network in which Host Machine is running.

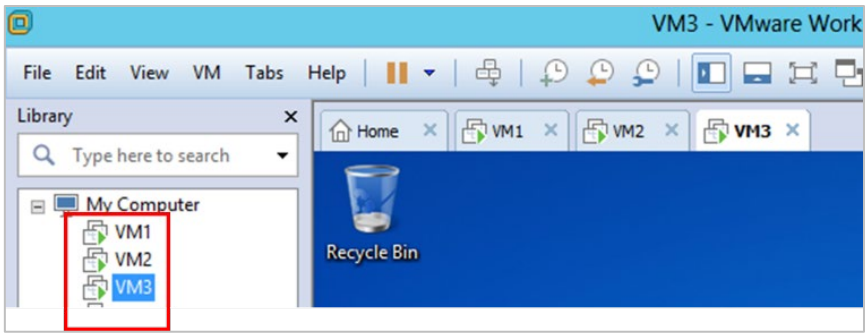


Figure 2-38: Multiple Virtual Machines (VM)

Now right click on each VM and select Settings. Click on Network Adapter, and select "**Bridged: Connected directly to the physical network**". Also enable the "**Replicate Physical network connection state**".

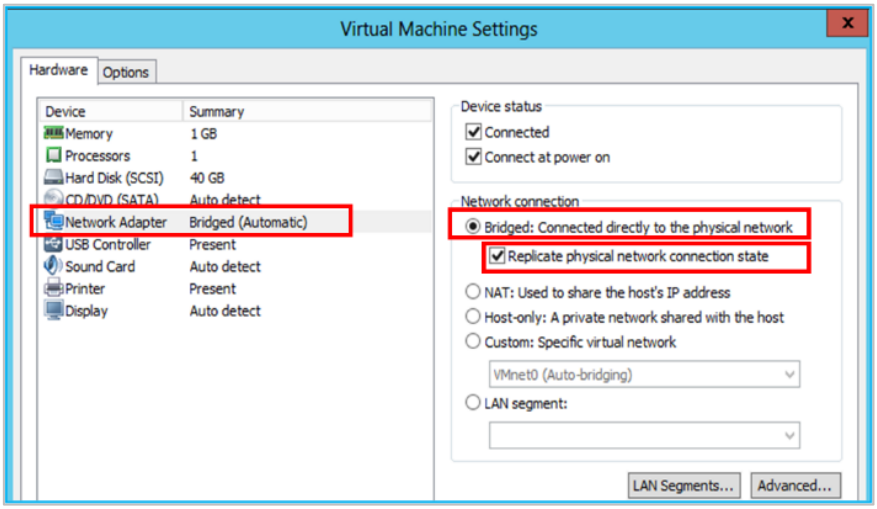


Figure 2-39: Multiple Virtual Machines setting window

An advantage of this technique is that, if the license server is running in another system, connected to the same network as the original host, then NetSim running in the VM can obtain the licenses.

2.8 VMs sharing a network but insulated from the host network

A computer on which one or more virtual machines are running is defined as a Host Machine. Each virtual machine is called a Guest Machine. In this scenario, we have 3 VMs running in a Host Machine – VM1, VM2 and VM3. NetSim License server is running in one of these 3 VMs.

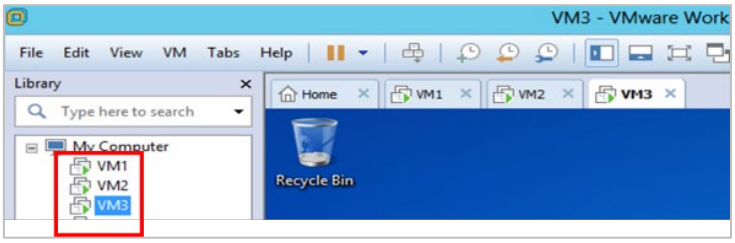


Figure 2-40: NetSim License server is running in one VM

If user needs to create an internal network which is segregated from host network, follow the steps

1. Right click on each VM and select Settings.
2. Click on Network Adapter, and select “Custom: Specific Virtual network”
3. Select “VMnet8 (NAT)”

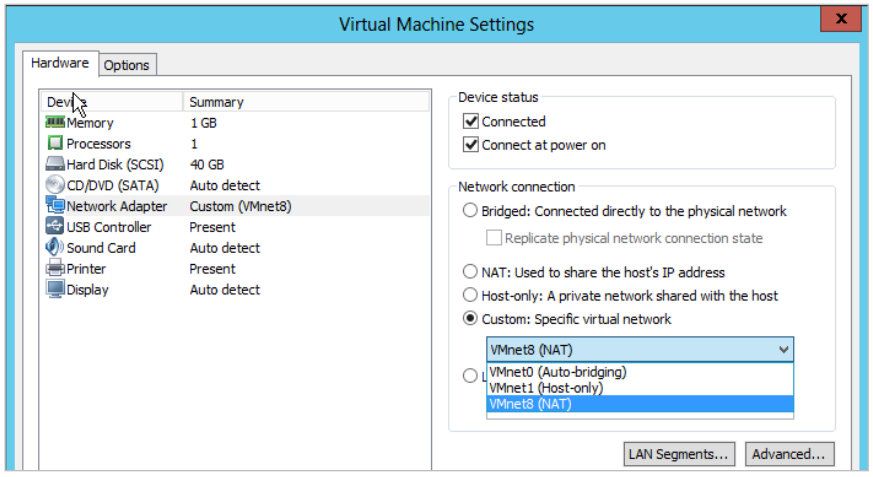


Figure 2-41: Select Specific Virtual network to VMnet8 (NAT)

By default, a network address is assigned to this segregated network by VMware. To configure this IP address, go to EDIT → Virtual Network Editor

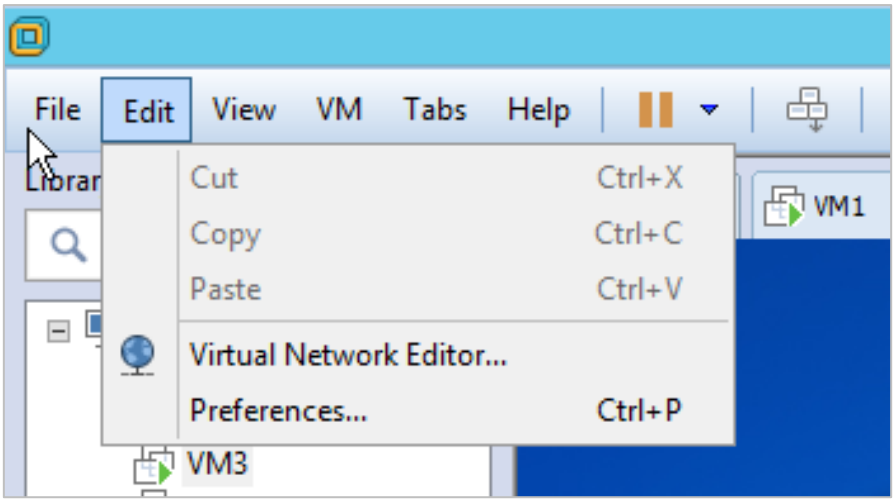


Figure 2-42: Select option to Virtual Network Editor in VM

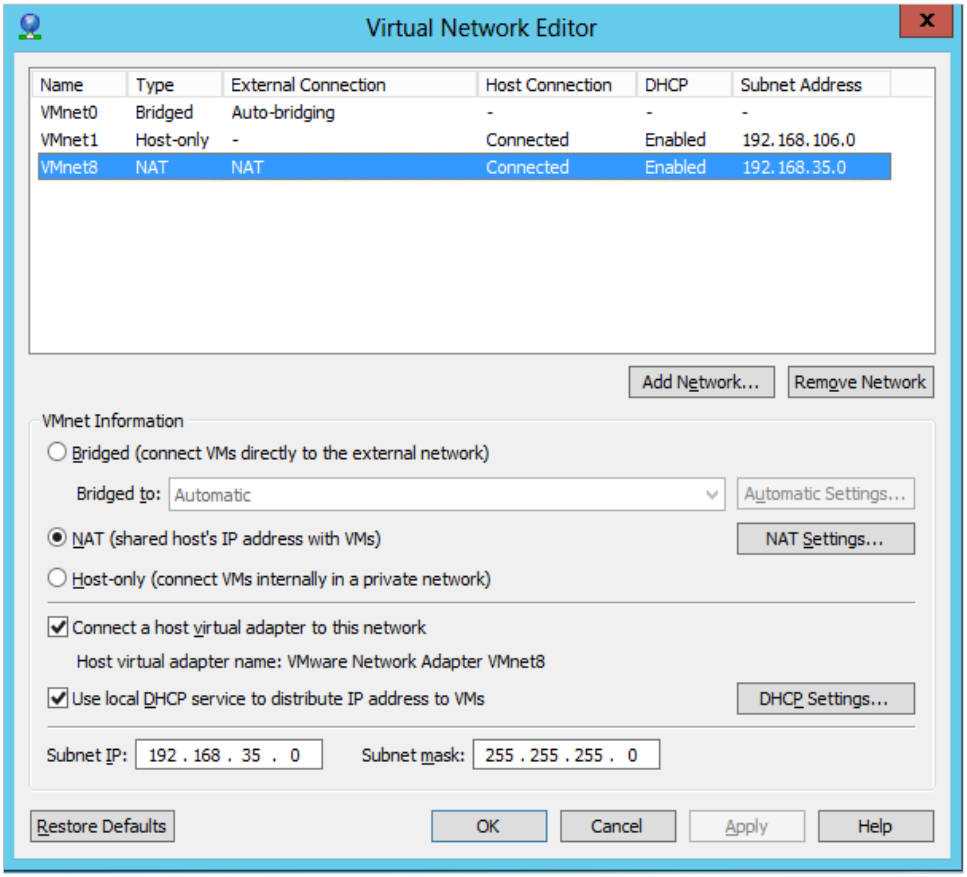


Figure 2-43: Modify the Subnet IP and Subnet Mask in Virtual Network Editor

User can modify the Subnet IP and Subnet Mask to suit their own preference.

The disadvantage of this technique is that, if the license server must compulsorily run in the VM for NetSim to obtain the licenses.

2.9 NetSim Emulator interfacing with Kubernetes Clients

Objective: Kubernetes Client running on the docker generates application traffic destined for Kubernetes Client. This traffic is sent through NetSim Emulator in which an equivalent virtual network is created. Thereby test the performance of your real application when transported over the virtual network.

Installation

For installation and running Kubernetes and Docker on Windows please refer

https://support.tetcos.com/support/solutions/articles/14000124321-how-to-run-netsim-on-windows-container-

https://www.youtube.com/watch?v=cgYOpw5XLtk

2.9.1 Traffic flow diagram

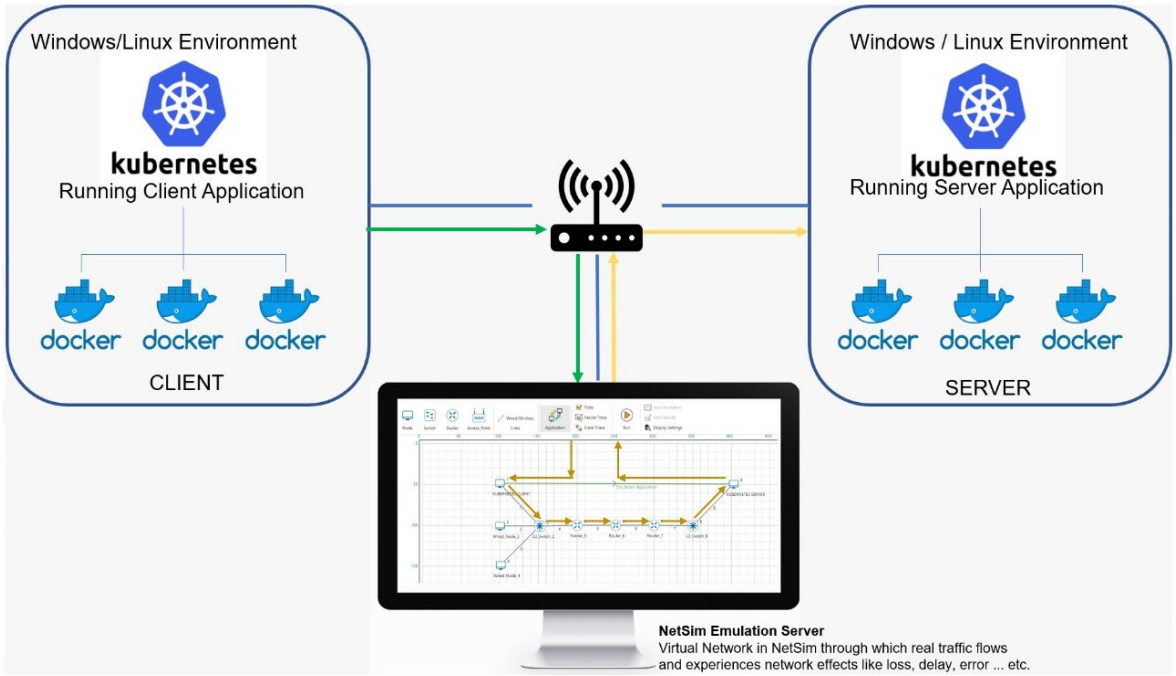


Figure 2-44: Traffic flow diagram

2.9.2 Lab Setup

- 3 Systems
  - Kubernetes Clients
    - Pods IP Address: 10.10.10.1
    - Host IP Address: 192.168.0.100
  - Kubernetes Server
    - Pods IP Address: 10.10.10.2
    - Host IP Address: 192.168.0.102
  - NetSim Emulation server (IP Address: 192.168.0.101)

- Kubernetes Client/server applications running on respective system. They be any application. In this example, we run the jperf application on client and server.
- Static routing in the lab setup.
    - CLIENT POD:
        - Run CLI docker in administrator mode and add a route to Server Pod. This must be in such way that the traffic generated from the Client Pod must flow through the Emulation server.
        - Command Windows: route add 10.10.10.2 MASK 255.255.255.255 192.168.0.101 METRIC 1
    - SERVER POD:
        - Run CLI docker in administrator mode and add a route to Client Pod. This must be in such way that the traffic generated from the Server Pod must flow through the Emulation server.
        - Command Windows: route add 10.10.10.1 MASK 255.255.255.255 192.168.0.101 METRIC 1

2.9.3 Steps to configure the application for NetSim Emulation.

- Once the Client Pod and Server Pod connection is established, In NetSim Emulation server, create a simple scenario in internetworks as shown below Figure 2-45.

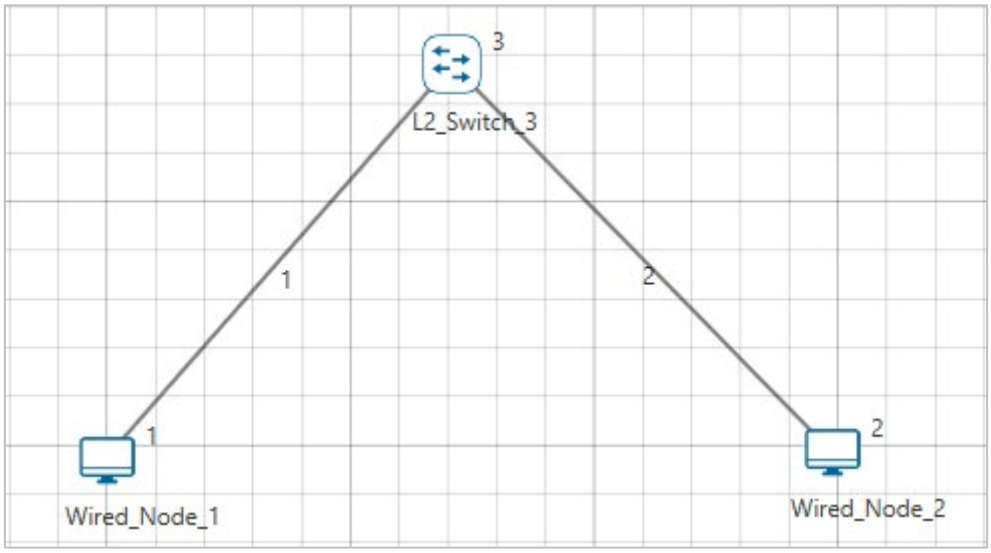


Figure 2-45: An example network topology with 2 nodes mapped to a Kubernetes client and a server. Note that the virtual network topology can have any number (typically < 500) of nodes. And can be mapped to any number of external devices.

- Configure an Emulation Application from Wired Node 1 to Wired Node 2 as shown below:

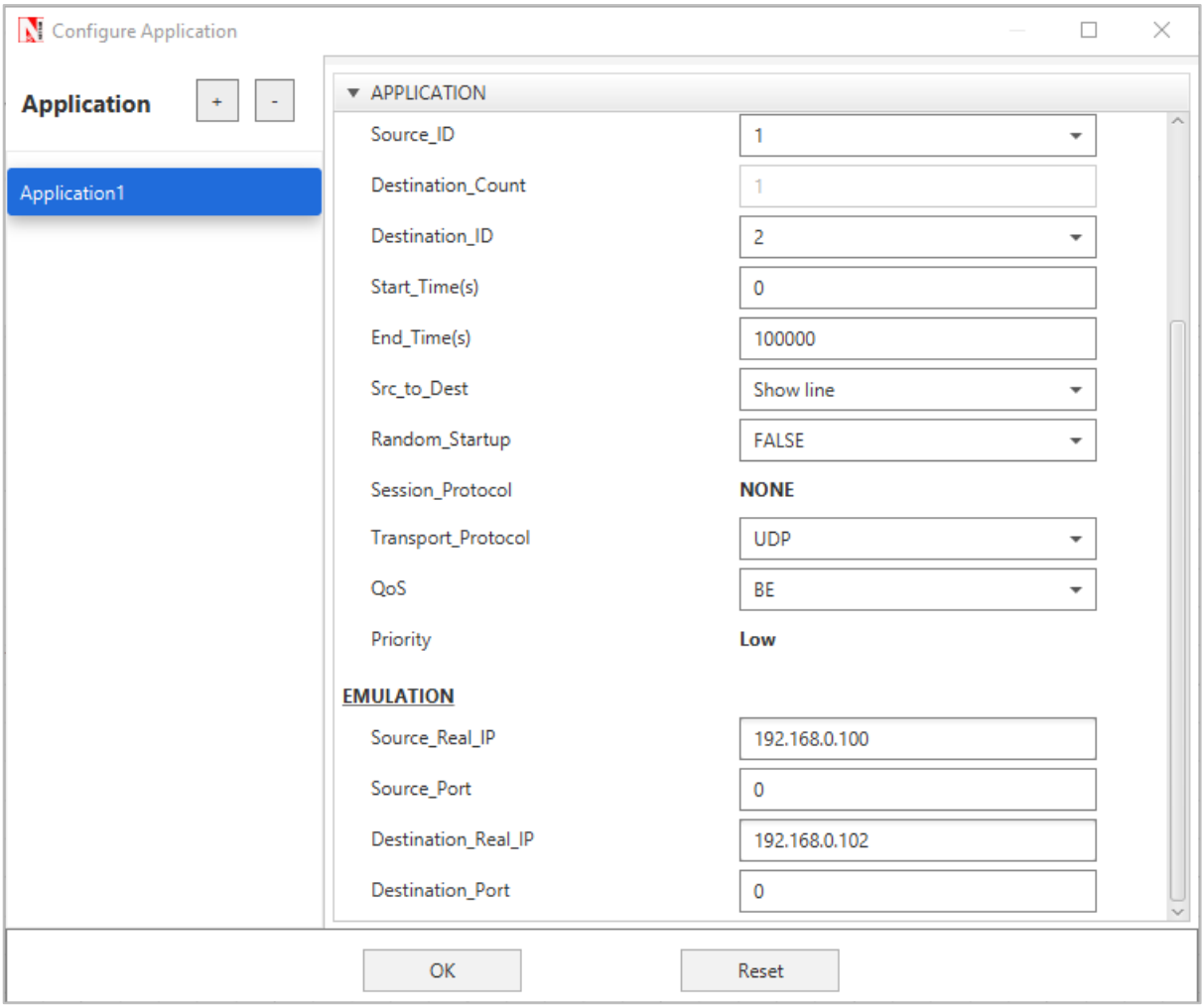


Figure 2-46: Application properties window for Application 1

The above settings will ensure that packets from System 1 to System 2 and vice versa will be sent via NetSim Emulator.

2.9.4 Results and Analysis

- Start Simulation in NetSim
- Analyze the impact of the flow of the packets through the virtual network designed in NetSim
- The pcap log files such as All Network Packets Capture, Dispatched to Emulator, Reinjected from Emulator, Not Dispatched to Emulator that are accessible from the NetSim results dashboard after the simulation is over, can be used for further analysis.

3 Model Features

3.1 Working of an Emulation Application in NetSim

Note: ***The following explanation is provided assuming that you have performed all necessary configuration required to divert network traffic via the system running NetSim Emulator. (This is explained in section 9 of the User Manual.***

The following parameters are specific to Emulation Application in NetSim:

Source_Real_IP

Source_Port

Destination_Real_IP

Destination_Port

Unlike Simulation, if users want to connect real devices running live applications to the simulator, then Emulation component is required. The Emulation Application in the traffic generator allows users to pump in real traffic into the Simulator.

The real application is mapped using the source and destination IP addresses that we set in the Emulation Application.

Various combination of Emulation Parameters are as follows:

Device Specific Emulation:

Example 1:

SOURCE_REAL_IP = 192.168.0.151

SOURCE_PORT = 0

DESTINATION_REAL_IP = 192.168.0.202

DESTINATION_PORT = 0

Dispatches all packets with the source real IP 192.168.0.151 and destination real IP as 192.168.0.202, into the Simulator.

Example 2:

SOURCE_REAL_IP = 192.168.0.151

SOURCE_PORT = 0

DESTINATION_REAL_IP = 0.0.0.0

DESTINATION_PORT = 0

Dispatches all packets from source real IP 192.168.0.151 regardless of whatever is the destination real IP, into the Simulator.

Example 3:

SOURCE_REAL_IP = 0.0.0.0

SOURCE_PORT = 0

DESTINATION_REAL_IP = 192.168.0.202

DESTINATION_PORT = 0

Dispatches all packets to destination real IP 192.168.0.202 regardless of whatever is the source real IP, into the Simulator.

Example 4:

SOURCE_REAL_IP = 0.0.0.0

SOURCE_PORT = 0

DESTINATION_REAL_IP = 0.0.0.0

DESTINATION_PORT = 0

Dispatches all packets that are reaching the Emulator Device regardless of whatever is the source or destination, into the Simulator.

Application Specific Emulation

Example 1:

SOURCE_REAL_IP = 192.168.0.151

SOURCE_PORT = 5004

DESTINATION_REAL_IP = 192.168.0.202

DESTINATION_PORT = 6245

Dispatches all packets with the source real IP 192.168.0.151, source Port No 5004, destination real IP as 192.168.0.202 and destination Port No 6245 into the Simulator.

Emulation Specific Metrics

On running an Emulation Application Users can optionally obtain the following log files which are Wireshark compatible .pcap files:

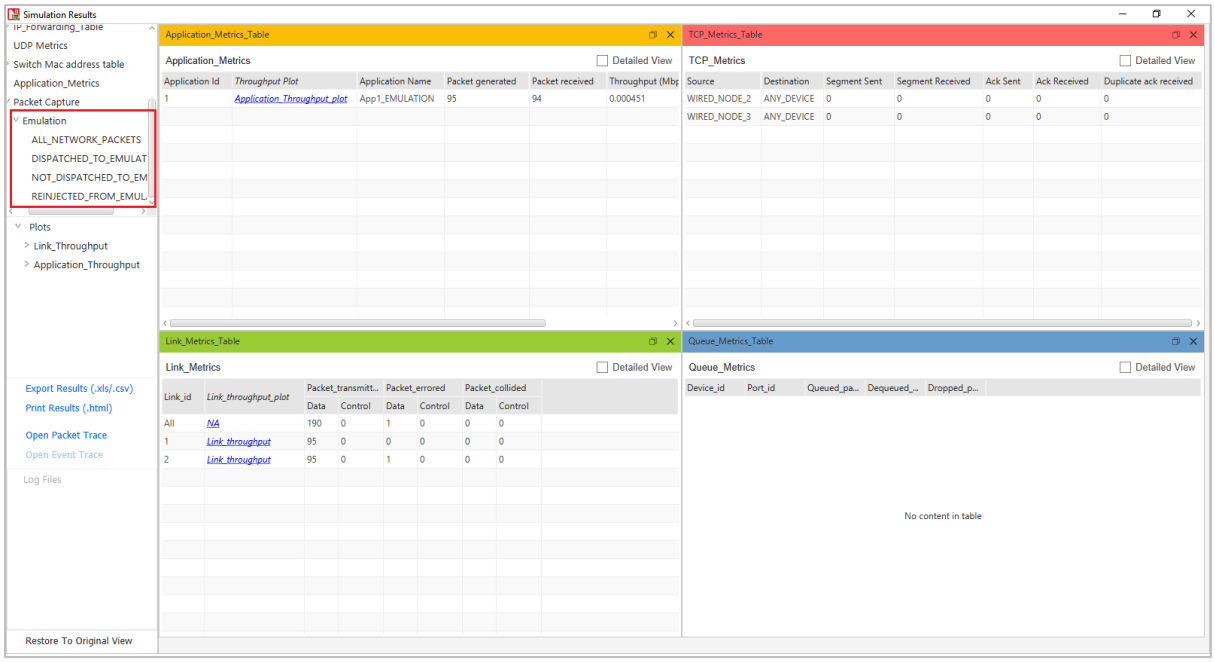


Figure 3-1: Different Emulation Specific Metrics in Result window

All_Network_Packets - Log of all packets flowing via the system running NetSim Emulator.

Dispatched_To_Emulator - Log of packets for which were sent to NetSim based on Emulation Application is configured in NetSim.

Reinjected_From_Emulator - Log of packets that successfully reached the virtual destination node in NetSim Simulator and was re-injected into the network.

Not_Dispatched_To_Emulator - Log of packets flowing via the system running NetSim emulator but not dispatched to emulator (All_Network_Packets minus Disptached_To_Emulator)

3.1.1 Delay measurement when pinging through NetSim Emulator

Pinging through NetSim emulator takes only one direction delay, if you have set only one application with Ping Source IP and ping Destination IP. This is because PING is a two way application and constitutes PING_REQUEST and PING_REPLY. For ping to take round trip delay users must configure two Emulation Applications, one for forward PING_REQUEST and other for the reverse PING_REPLY.

For example: If you are running a ping from the IP 192.168.0.151 to an IP 192.168.0.202 the time take will normally be around 1ms.

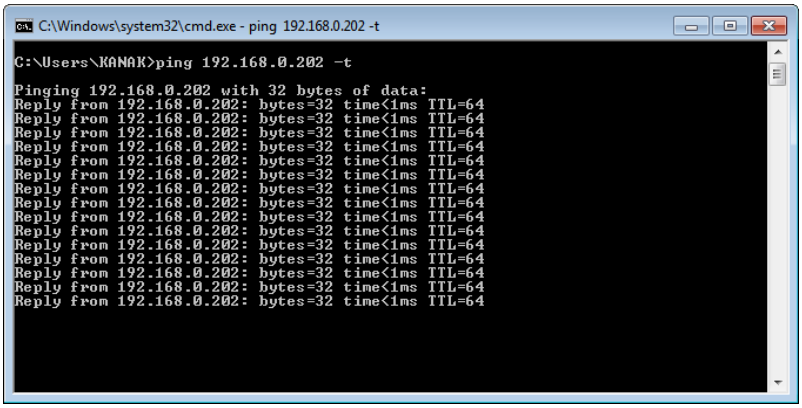


Figure 3-2: Pinging from one device to other device and total time taken by 1ms

Now we create a network scenario in NetSim similar to the screenshot shown below **Figure 3-3**.

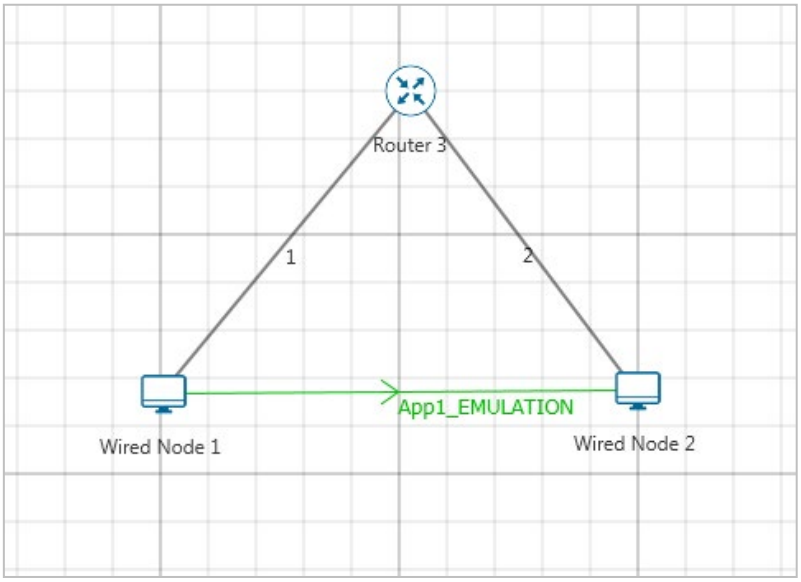


Figure 3-3: Network Topology with Emulation Application

We reset the propagation delay in both the wired links to 5 µs.

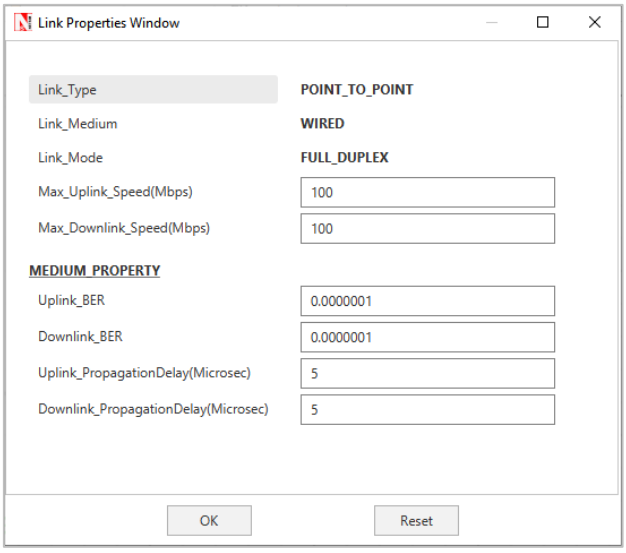


Figure 3-4: Wired Link properties window

We configure an Emulation application between the wired nodes with the source and destination real IP specified, as shown below:

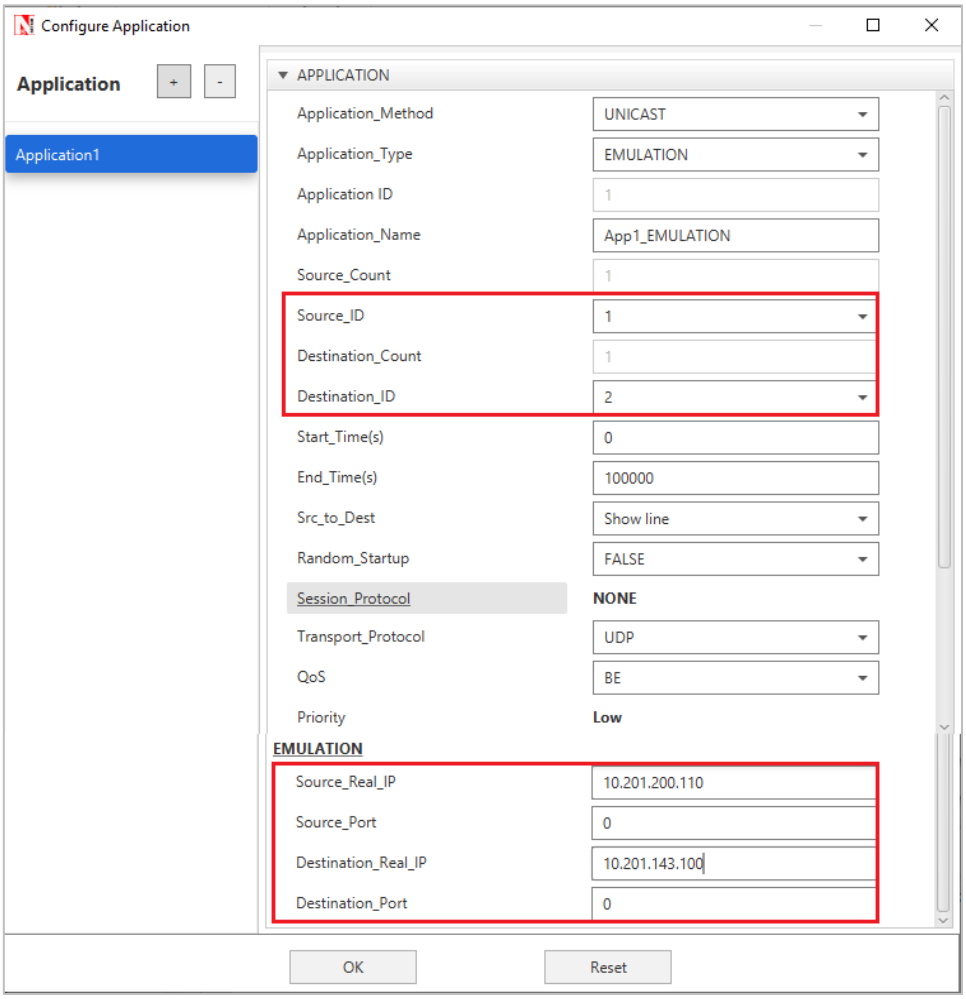


Figure 3-5: Application properties window for Application 1

On running the simulation, you will observe the variation in the time taken to get the ping reply in the source system, as shown below:

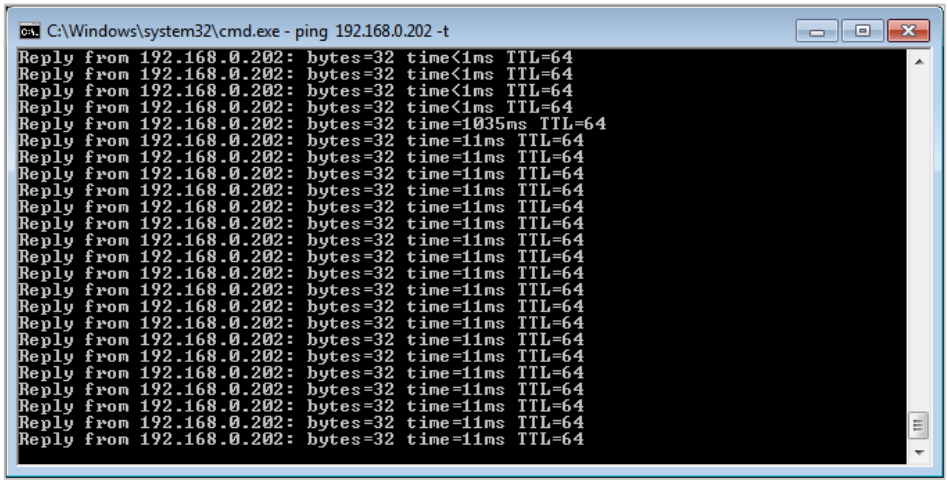


Figure 3-6: Pinging from one device to other device and total time taken by 11ms and including 10µs additional delay for both the links

Ping packets has experienced an additional delay of 10µs which is a sum of the delay in both the links.

The additional delay experienced by ping packets is not 20µs because, the application that we have configured applies to only the Ping Request Packets which has the Source IP as 192.168.0.151 and Destination IP as 192.168.0.202.

The Ping Reply Packets has the Source IP as 192.168.0.202 and Destination IP as 192.168.0.151, for which we have not configured any application.

For the ping to take the round trip delay, we will have to configure one more application for the reverse traffic. On adding an application for the reverse traffic as shown below:

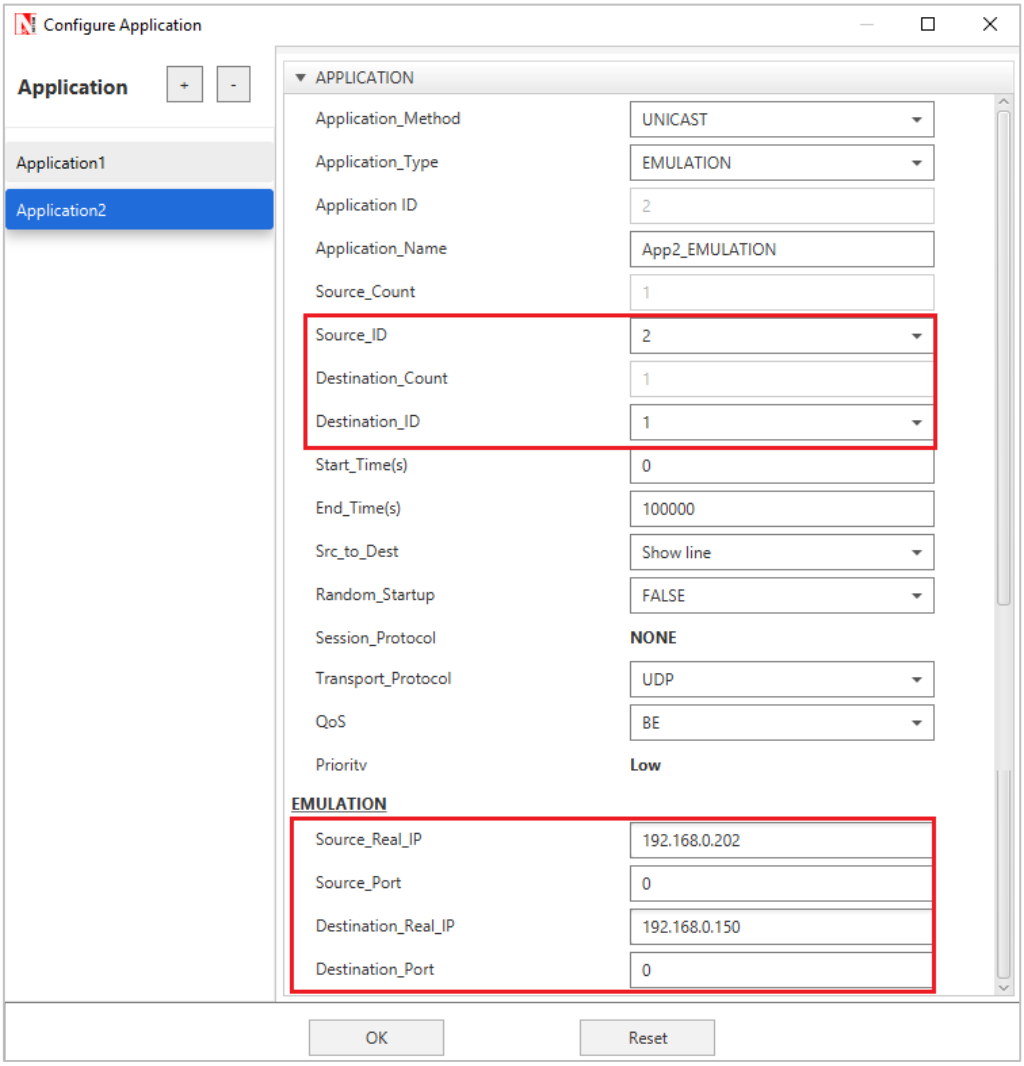


Figure 3-7: Application properties window for Second application

We will now be able to see round trip delay being experienced by the PING application, as shown below:

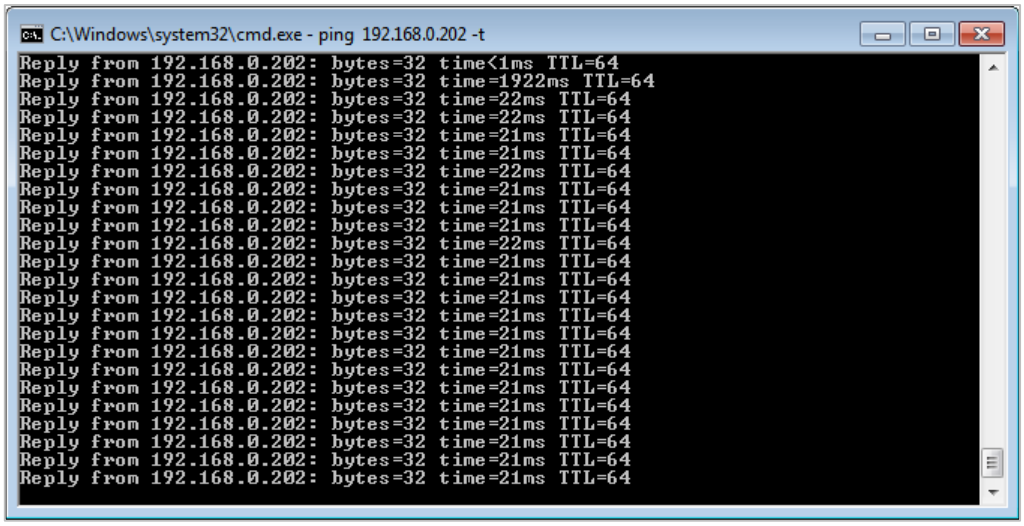


Figure 3-8: Total round-trip time taken by 20ms

Ping experiences an additional overall delay of 20ms, which is the sum of the delays experienced by Ping Request and Ping Reply (10ms + 10ms).

3.2 Jitter in NetSim Emulations

Jitter is defined as a variation in the delay of received packets. Let us suppose at the sending side, packets are sent in a continuous stream with the packets spaced evenly apart. Due to network congestion, improper queuing, or configuration errors, this steady stream can become lumpy, or the delay between each packet can vary instead of remaining constant. This variation in delay is 'jitter'. While there are many ways of measuring this variation, in NetSim 'jitter' is measured as the statistical variance of delay. Variance is defined as the square of deviation from the mean.

3.2.1 Introducing Jitter using Background traffic

Background traffic can be used to test the performance of applications when link bandwidth is consumed by other traffic. It can also be used to induce jitter for testing real-time applications.

The Background traffic in NetSim can be modelled as a Poisson process in which bursts of data of a fixed size are transmitted at an average rate such that the link will be occupied at the specified link utilization rate. Because it is a random process, over short periods the actual background traffic link utilization rate may vary from the configured value. The rate of arrival of background traffic frames affects the jitter. Larger number of background packets induce greater jitter in competing traffic. In NetSim, the way to increase the number of background packets arriving is to reduce the inter-arrival time of that application, as explained in the link https://tetcos.freshdesk.com/support/solutions/articles/14000067807-how-do-i-introduce-jitter-in-netsim-simulations-emulations-

4 Featured Examples

4.1 Example 1: PING (One way Communication)

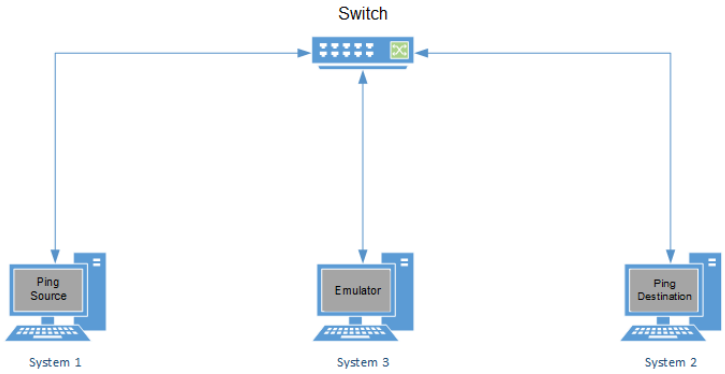


Figure 4-1: One way Communication

Steps at Emulation Server

1. Run NetSim in Administrative Mode and create a basic network Scenario in any stack based protocol (Any network except Legacy Networks, Wireless Sensor Network, and Cellular Network) in NetSim. Screenshot of a sample scenario in Internetworks is shown below Figure 4-2.

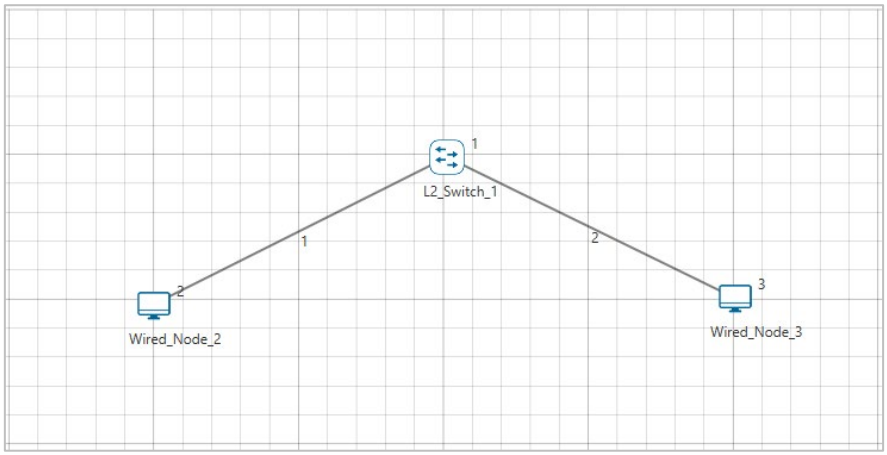


Figure 4-2: Network Topology

- Go to Properties of Link1 and Link2 and set Uplink and Downlink Delay to 5000µs. Click on the Application icon present on the ribbon and set properties.
- In the Application Type select Emulation.
- Select Source and Destination ID according to the network scenario and change the Source and Destination IP address according to the IP address of the real system.

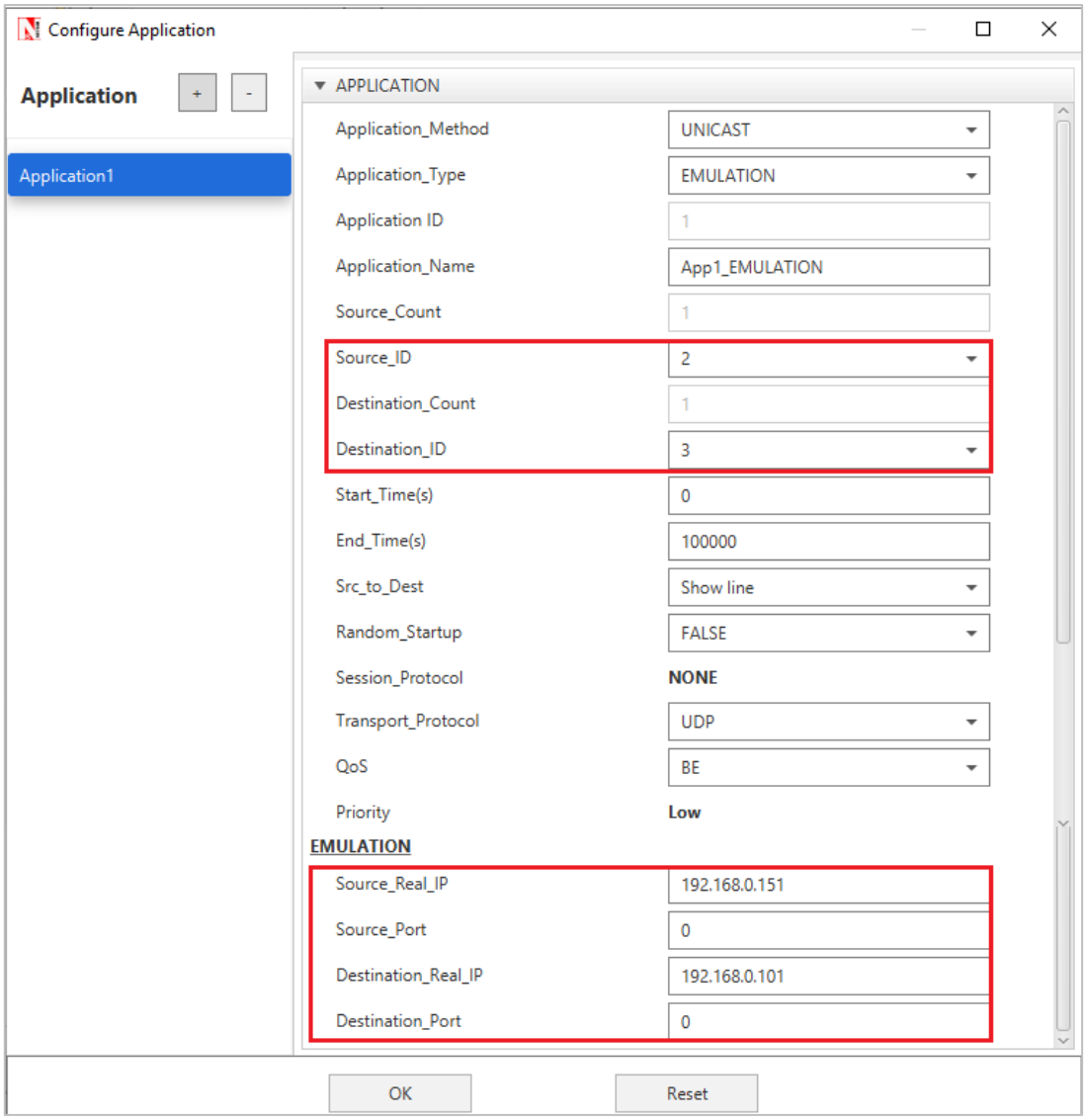


Figure 4-3: Application properties window

2. Provide the Simulation Time as how long you want the Emulation to be performed. Make sure client system(s) are ready and then click Run Simulation.

Steps at Source PC

- Before running simulation, start pinging the Destination from Source using command "ping <Destination_IP> –t" and note down the time duration.

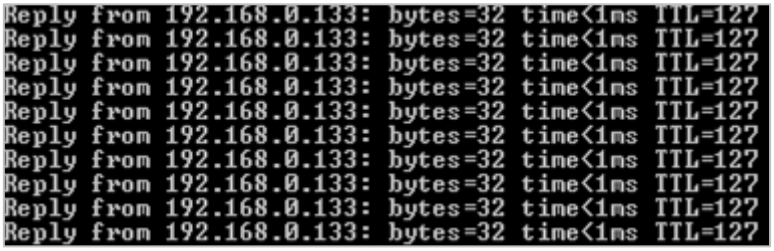


Figure 4-4: Before running simulation, total time taking by 1ms

- Follow steps as provided before in "Emulation Set-up: Setting up the NetSim Client".

- Perform the steps at Emulation Server as provided and simulate. During simulation, ping the destination system. You will notice that the present time duration is higher than the earlier ping results. This is because the network created in NetSim has link propagation delay. Also, Wireshark (if installed) will automatically start capturing the packets as soon as Emulation Server starts simulation.

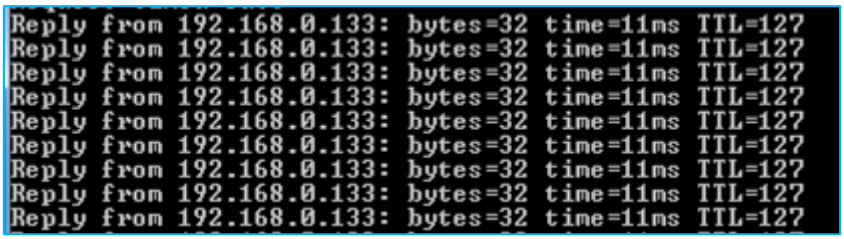


Figure 4-5: After running simulation, total time taking by 11ms

(NOTE: In case if no ping messages can be sent from source to destination, disable windows firewall and try again.)

- The impact of the link propagation delay in NetSim Emulator is seen on a real packet.

4.2 Example 1: PING (Two-way Communication)

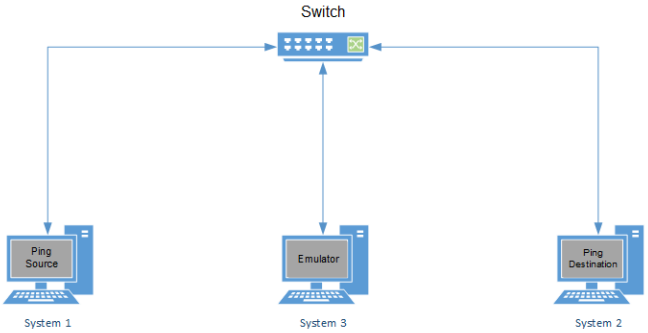


Figure 4-6: Two-way Communication

In PING (Two-way communication), almost all the steps are same as PING (One way communication), except that in NetSim Emulation server there will be two applications instead of one. One Application will be directed from Source to Destination node, while the other application will be directed from Destination to Source node.

The difference caused in the network behaviour is that in the first case (PING -One way communication), the PING reply packets were not routed via NetSim Emulator. But in the second case (PING -Two-way communication), the PING reply to packets will be routed via NetSim Emulator, thereby the total delay will be approximately 21milliseconds.

4.3 Example 2: Video (One way Communication)

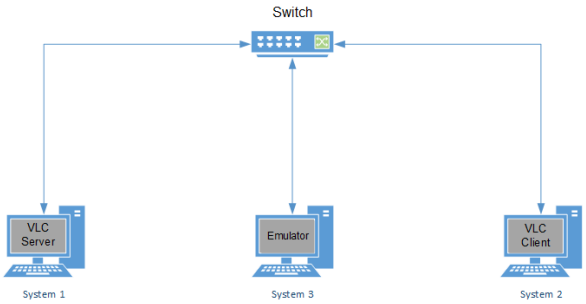


Figure 4-7: Video One-way Communication

Steps at NetSim Emulation Server

1. Run NetSim in Administrative Mode and create a basic network Scenario in any stack-based protocol (Any network except Legacy Networks, Wireless Sensor Network, and Cellular Network) in NetSim. Screenshot of a sample scenario in Internetworks is shown below Figure 4-8.

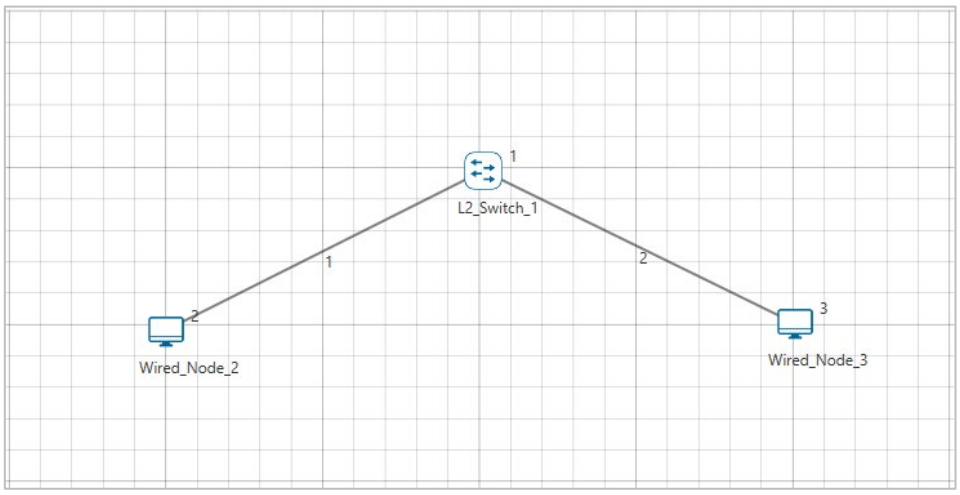


Figure 4-8: Network Topology

- Click on the Application icon present on the ribbon and set properties.
- In the Application Type select Emulation.
- Select Source and Destination ID according to the network scenario and change the Source and Destination IP address according to the IP Address of the real system and click OK.

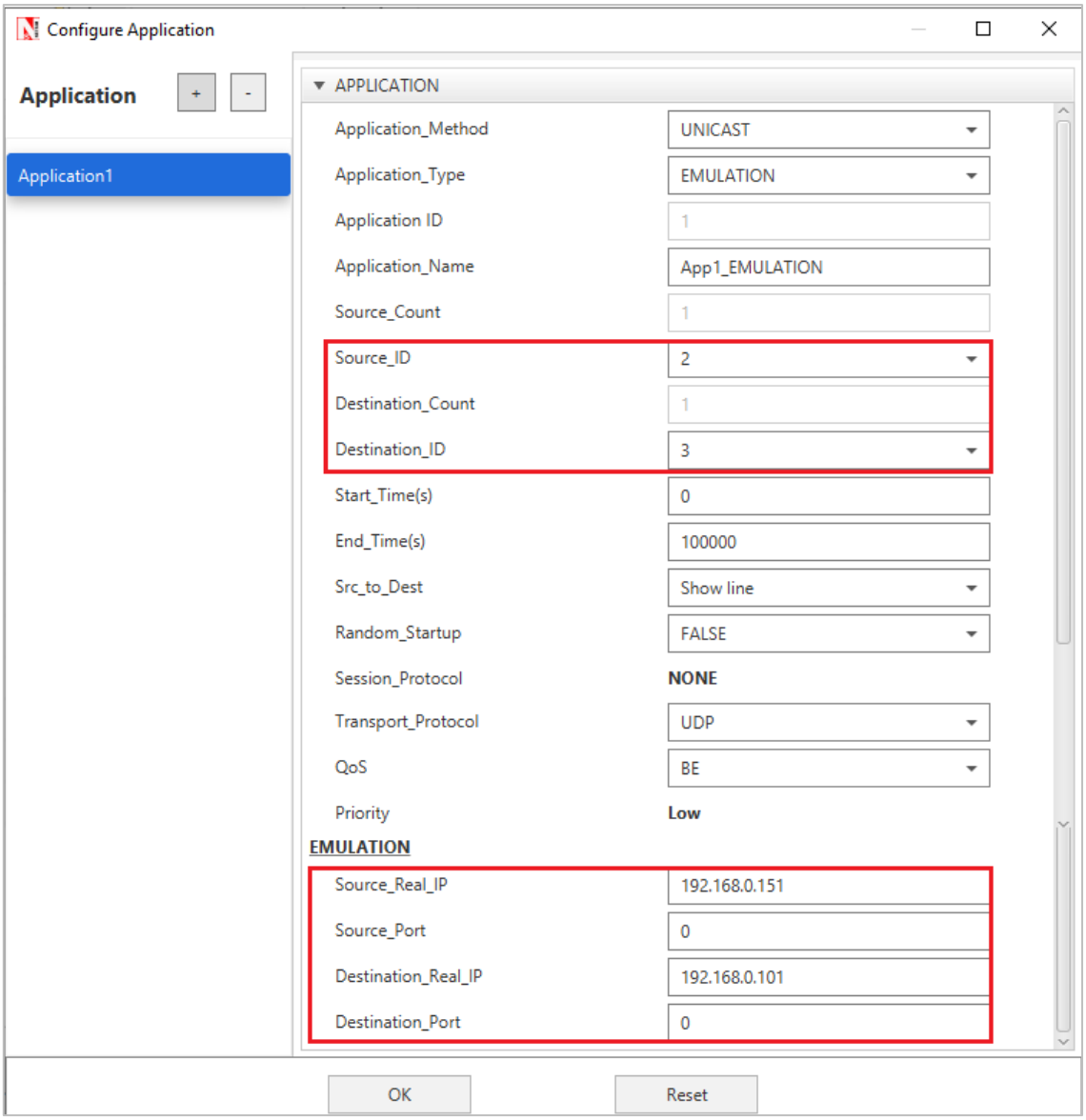


Figure 4-9: Application properties window

2. Provide the Simulation Time as how long you want the Emulation to be performed. Make sure client system(s) are ready and then click Run Simulation.

During Simulation you will notice a change in the quality of the video being played in the destination PC. This is because the network created in NetSim has errors / delays etc in the links. The impact of this loss / jitter / delay etc in NetSim Emulator is seen on a real video stream.

Steps at Source PC

1. Follow steps as provided before in “Running Emulation via GUI → Setting up the NetSim Client”. Then open VLC Media player → Click Media menu → Select Stream Option.
2. Click add button then select the video which you want to play.

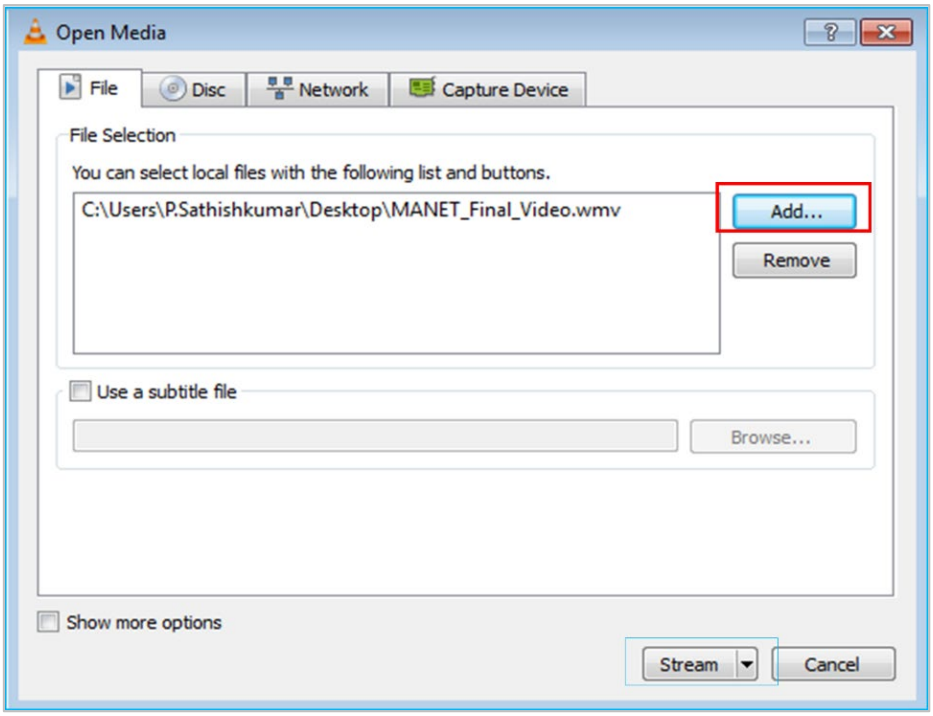


Figure 4-10: Select a video using Add option in VLC media

3. Click on Stream Option. Then click next button.
4. Enable the display locally checkbox. Then select the RTP / MPEG Transport Stream from the drop-down list as shown in the below screen shot

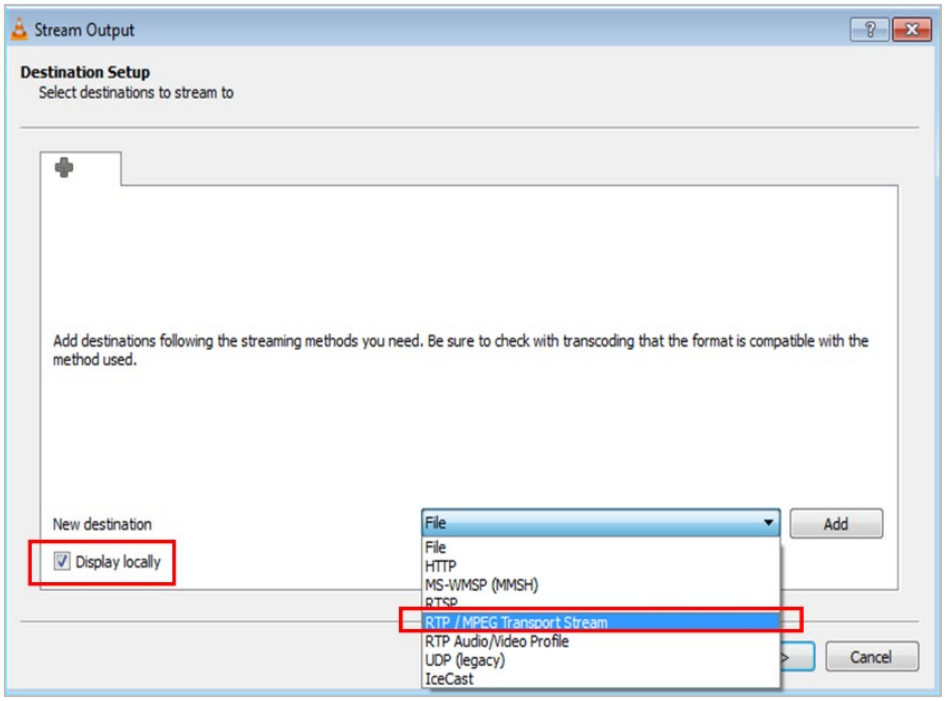


Figure 4-11: Select the RTP / MPEG Transport Stream from the drop-down list in VLC media

5. Click on Add Button. Then enter the Destination IP address in the Address field and enter a stream name (user defined) and click next button.

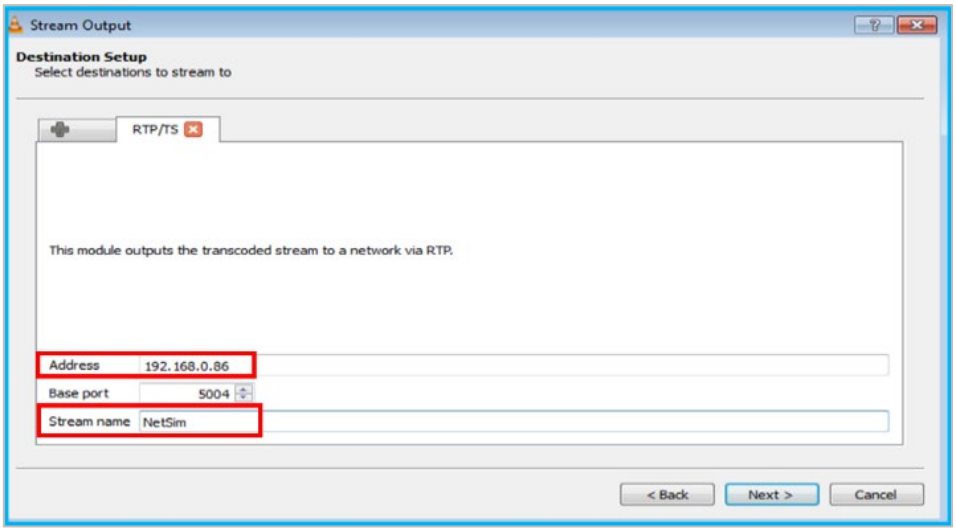


Figure 4-12: Enter the Destination IP address in the Address field and enter a stream name (user defined)

6. Select Video –MPEG-2 + MPGA (TS) option from the drop-down list as shown in the below screen shot. Then click next button.

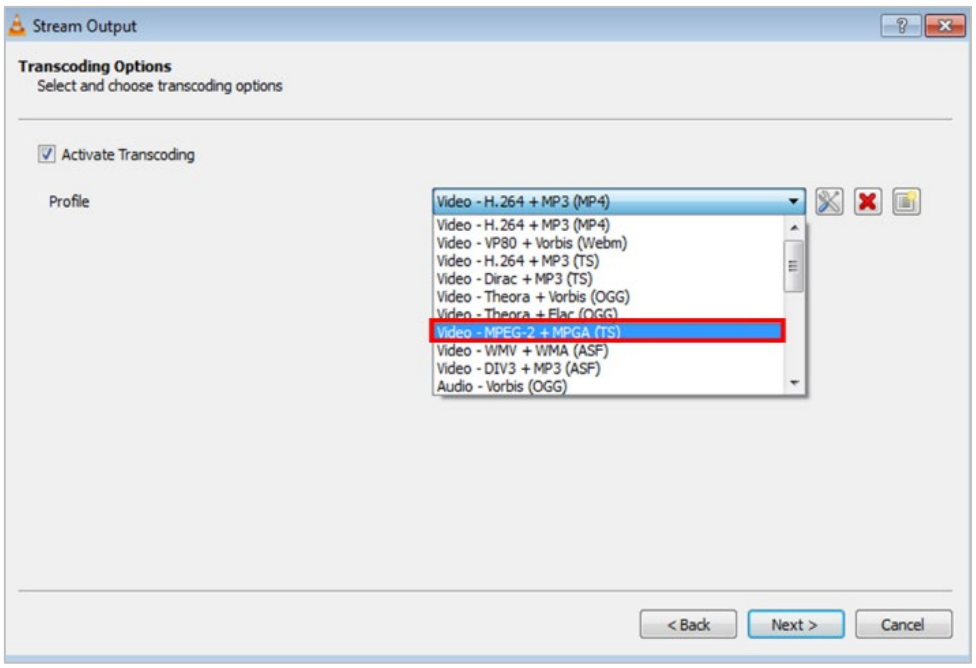


Figure 4-13: Select Video –MPEG-2 + MPGA (TS) option from the drop-down list

7. Perform all the steps at Emulation Server and then click on Stream button. Also, Wireshark (if installed) will automatically start capturing the packets as soon as Emulation Server starts simulation.

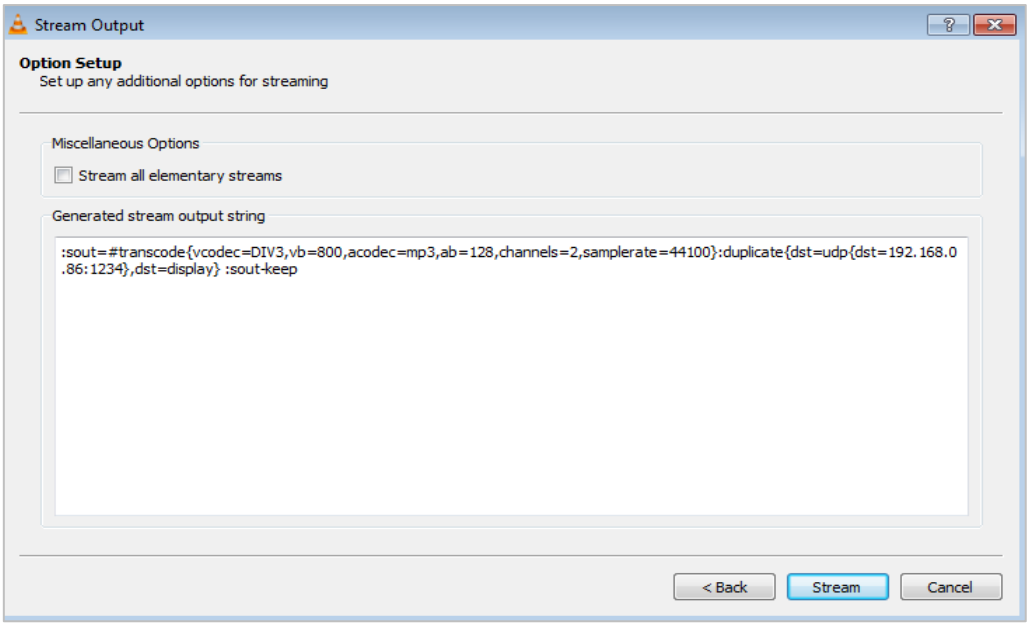


Figure 4-14: Perform all the steps at Emulation Server and then click on Stream button

Steps at Destination PC

1. Follow steps as provided before in “Running Emulation via GUI–Setting up the NetSim Client”. After performing all the steps at Source PC and NetSim Emulation Server, open VLC Media Player→ Click on Toggle Playlist icon as shown in the below screenshot Figure 4-15.

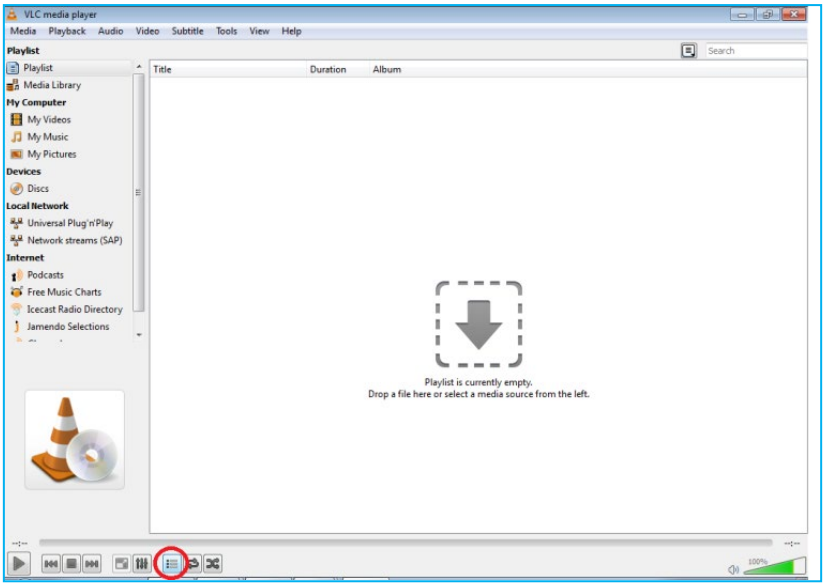


Figure 4-15: Select Toggle Playlist icon

Toggle button is circled in red at the bottom of the screen shot

2. Double click on Network Stream (SAP) under local network. Then right click and play on the stream name that appears on the screen.

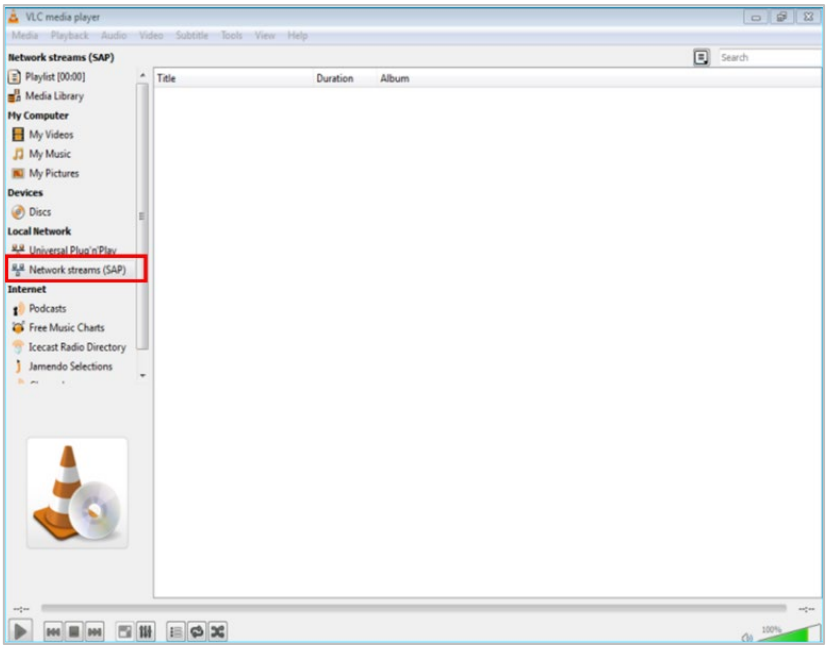


Figure 4-16: Double click on Network Stream (SAP) under local network

3. In the streamed video, you will notice a change in the quality of the video being played in the destination PC. Also, Wireshark (if installed) will automatically start capturing the packets as soon as Emulation Server starts simulation.

Figure 4-17: Change in the quality of the video being played in the destination PC

4.4 Example 3: File Transfer using FileZilla (One-way)

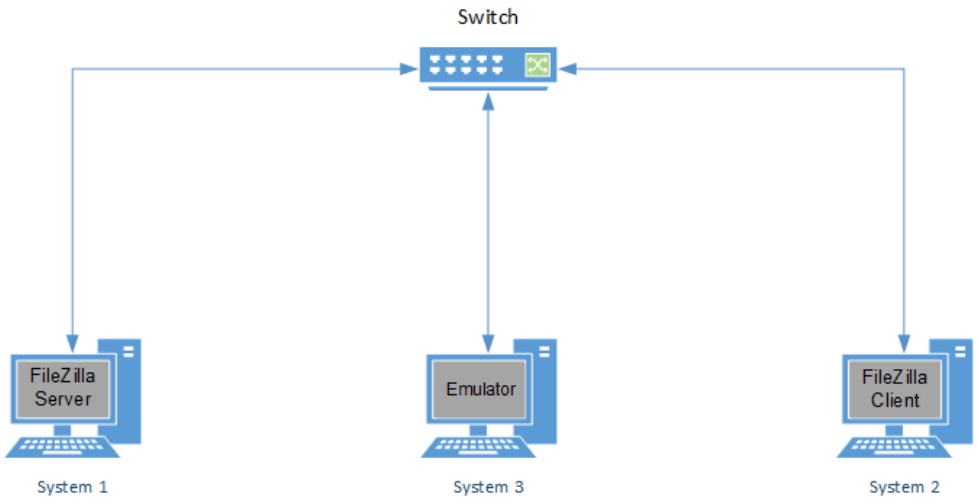


Figure 4-18: File Transfer using FileZilla (One-way communication)

Steps at Destination PC

1. Follow steps as provided before in "**Emulation Set-up: Setting up the NetSim Client"**. Run FileZilla Server software. Create a group by going to Edit → Groups → Select "General" under Page: → Click Add in Groups → Give Any Name (Ex: Admin) and click ok.

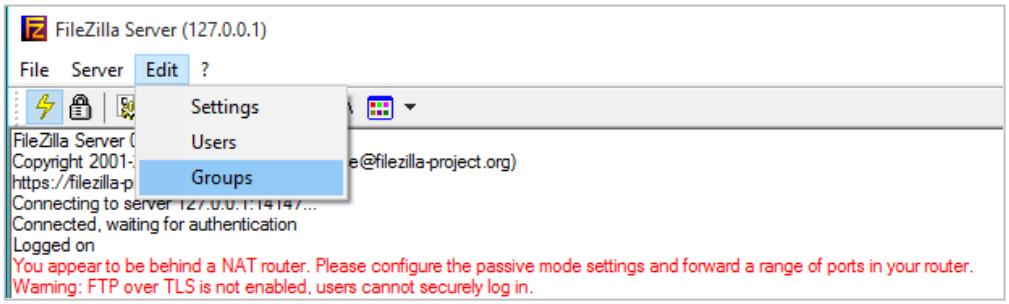


Figure 4-19: Give Any Name to group in edit option in FileZilla server

2. Go to Edit → User → General → Click Add in User → Give Any Name (Ex: User1) and Select Group what you given in Group Setting (In this case, we provide "Admin") and click ok.

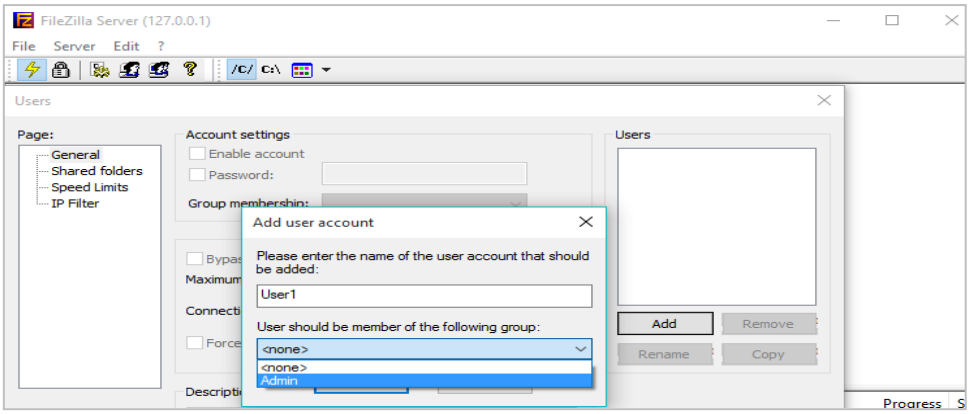


Figure 4-20: Add name to Users field in FileZilla server

3. In Account Setting, select Enable account and set password and click ok.

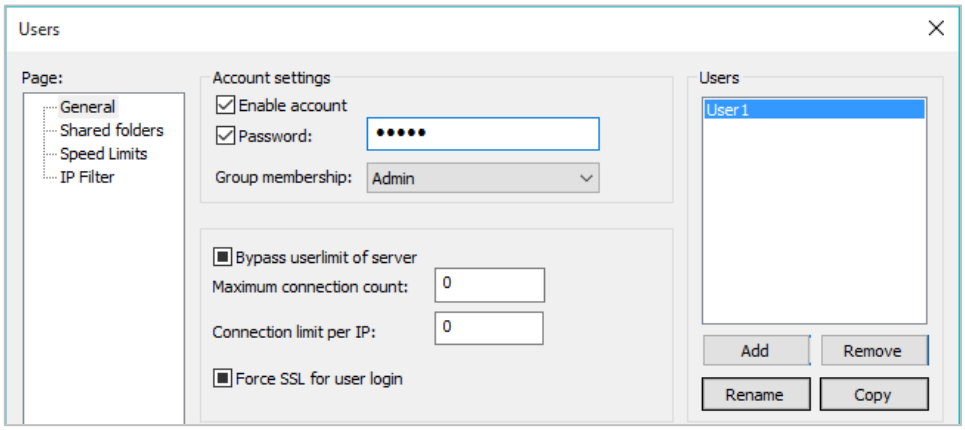


Figure 4-21: Enable account and set password in account setting window

4. Go to Shared folder → Add Folder to share (EX: FTP_FILES from Desktop) → Select all the Files and Directories Permissions and set that folder as Home Directory by selecting "Set as Home Dir". Click OK.

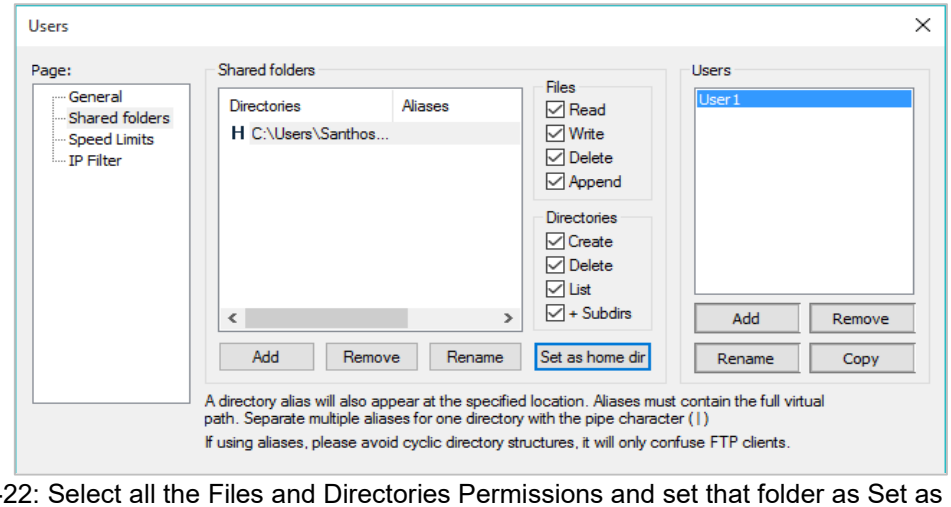


Figure 4-22: Select all the Files and Directories Permissions and set that folder as Set as Home Dir

Steps at Source PC

Follow steps as provided before in "Emulation Set-up: Setting up the NetSim Client". Run FileZilla Client software. Enter the Host Name(Server System ip (EX: 192.168.0.133)) and

Give the User, Password that we created in Server side and give Port No = 21. Run Emulation server and click Quick Connect. Drag and drop files from Local Site to Remote Site.

Steps at NetSim Emulation Server

1. Run NetSim in Administrative Mode and create a basic network Scenario in any stack-based protocol (Any network except Legacy Networks, Cellular Networks, and Wireless Sensor Networks) in NetSim. A sample scenario in Internetworks is performed as shown with link speed set to 1 Mbps.

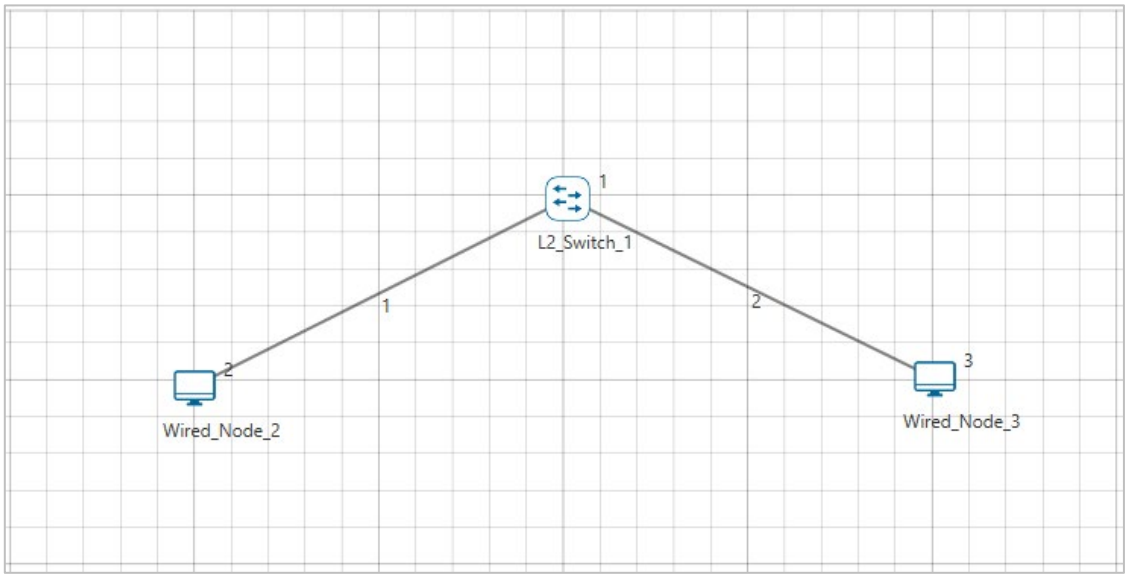


Figure 4-23: Network Topology

- Click on the Application icon present on the ribbon and set properties.
- In the Application Type select Emulation.
- Select Source and Destination ID according to the network scenario and change the Source and Destination IP address according to the IP Address of the real system and click OK.

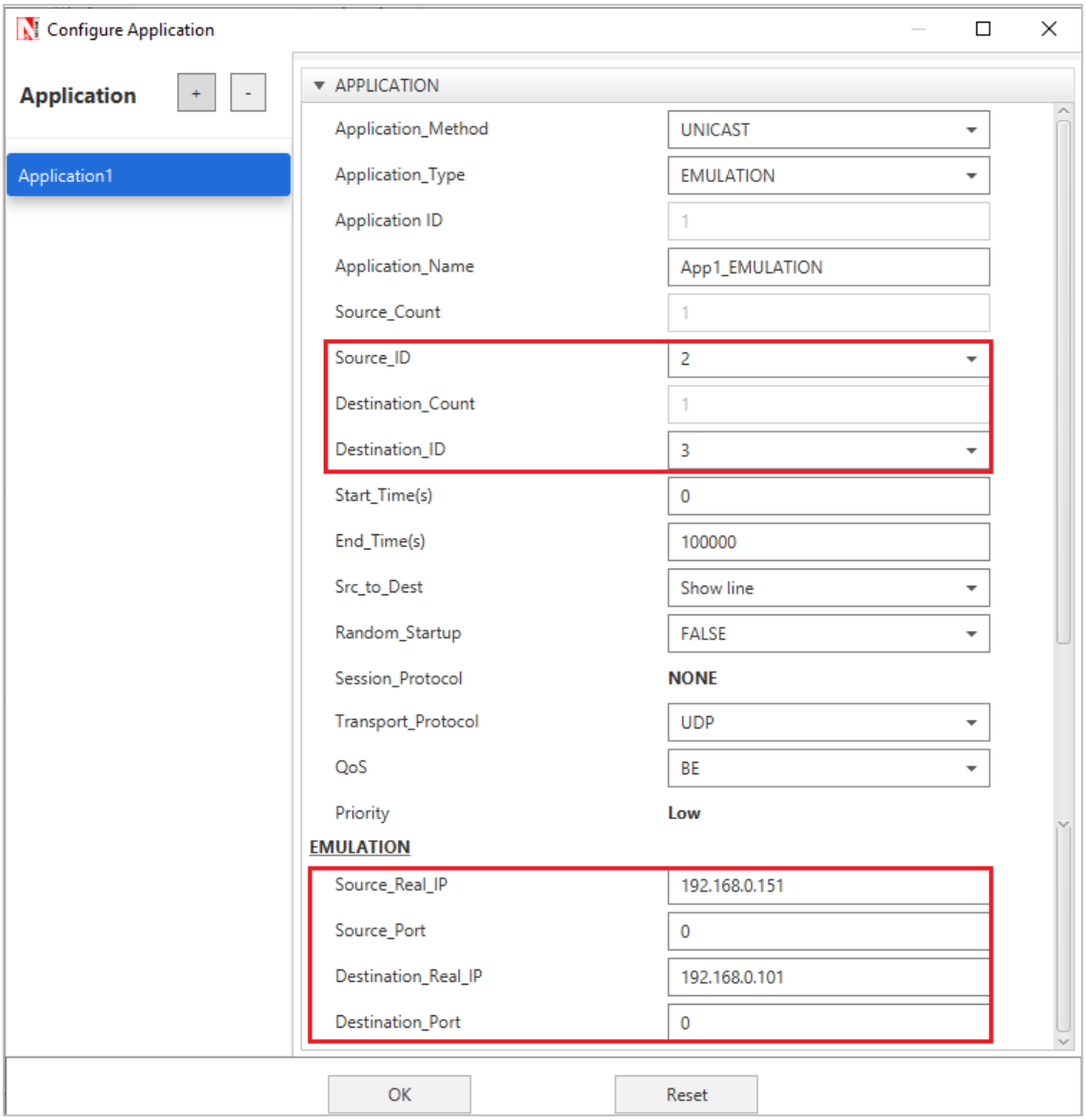


Figure 4-24: Application properties window

2. Provide the Simulation Time as how long you want the Emulation to be performed. Make sure client system(s) are ready and then click Run Simulation.

Results: Transfer speed from client without emulation:

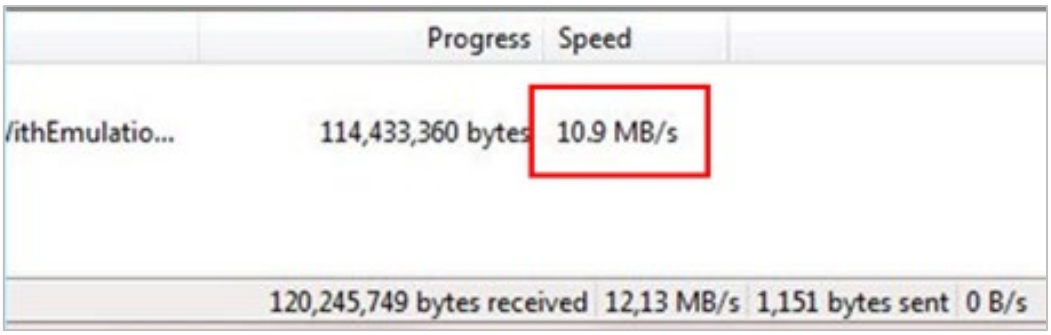


Figure 4-25: Transfer speed from client without emulation

Transfer speed from client with emulation:

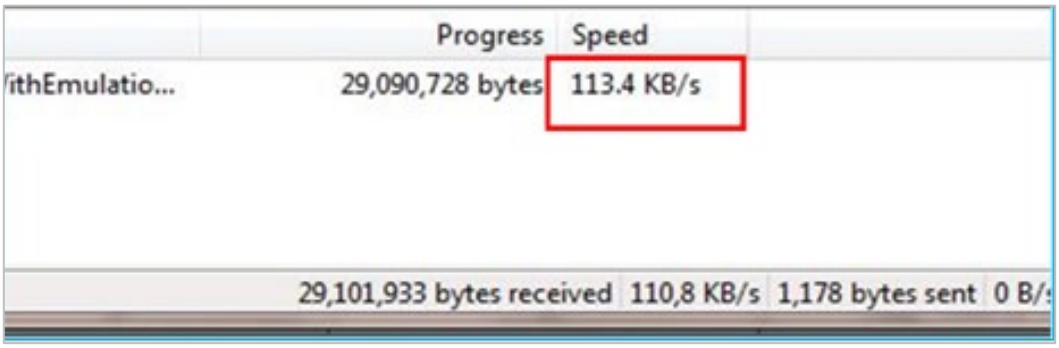


Figure 4-26: Transfer speed from client with emulation

4.5 Example 4: Skype (Two way Communication)

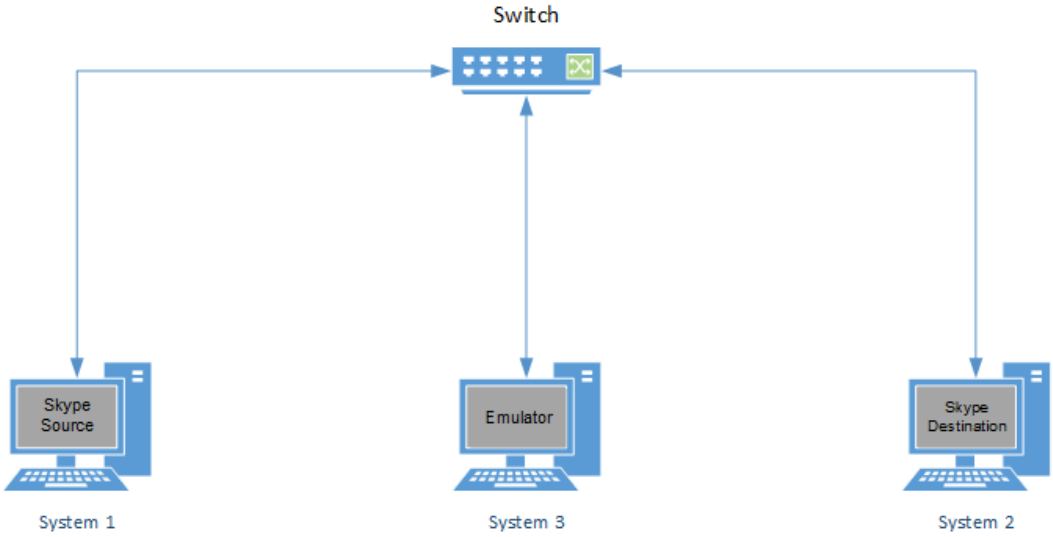


Figure 4-27: Skype Two-way Communication

Steps at NetSim Emulation Server

1. Run NetSim in Administrative Mode and create a basic network Scenario in any stack-based protocol (Any network except Legacy Networks, Wireless Sensor Network, and Cellular Network) in NetSim. Screenshot of a sample scenario in Internetworks is shown below Figure 4-28.

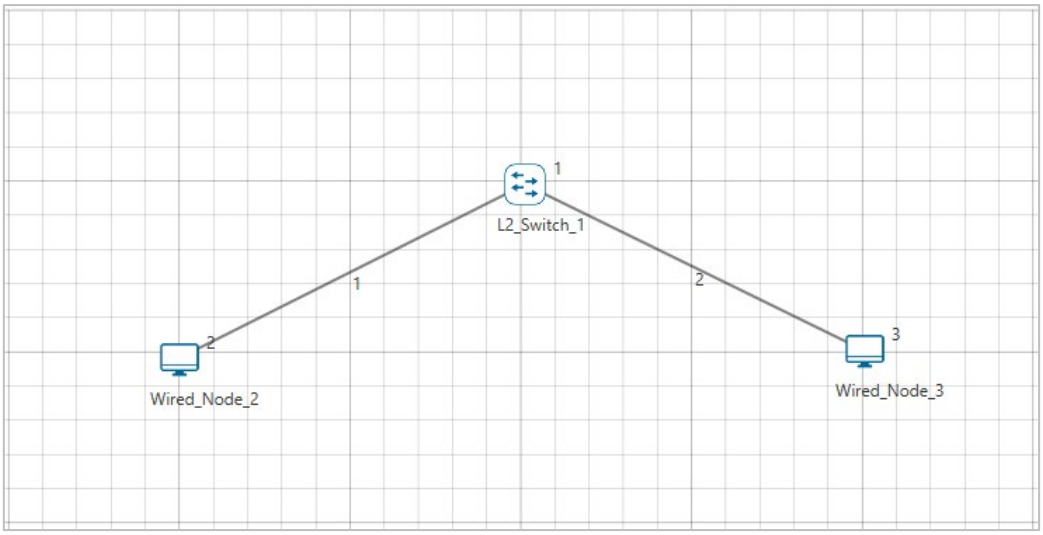


Figure 4-28: Network Topology in this experiment

2. Click on the Application icon present on the ribbon and set properties. As it is two-way communication, add and create two applications.
3. In both the Application Type select Emulation.
4. In one Application, select Source ID and Destination ID according to the network scenario and change the Source and Destination IP address according to the IP Address of the real system. In the second application, set the opposite of first application, i.e. Source ID and IP address will be exchanged with Destination ID and IP address. (Refer the IP settings in the screenshot to get a clear picture)

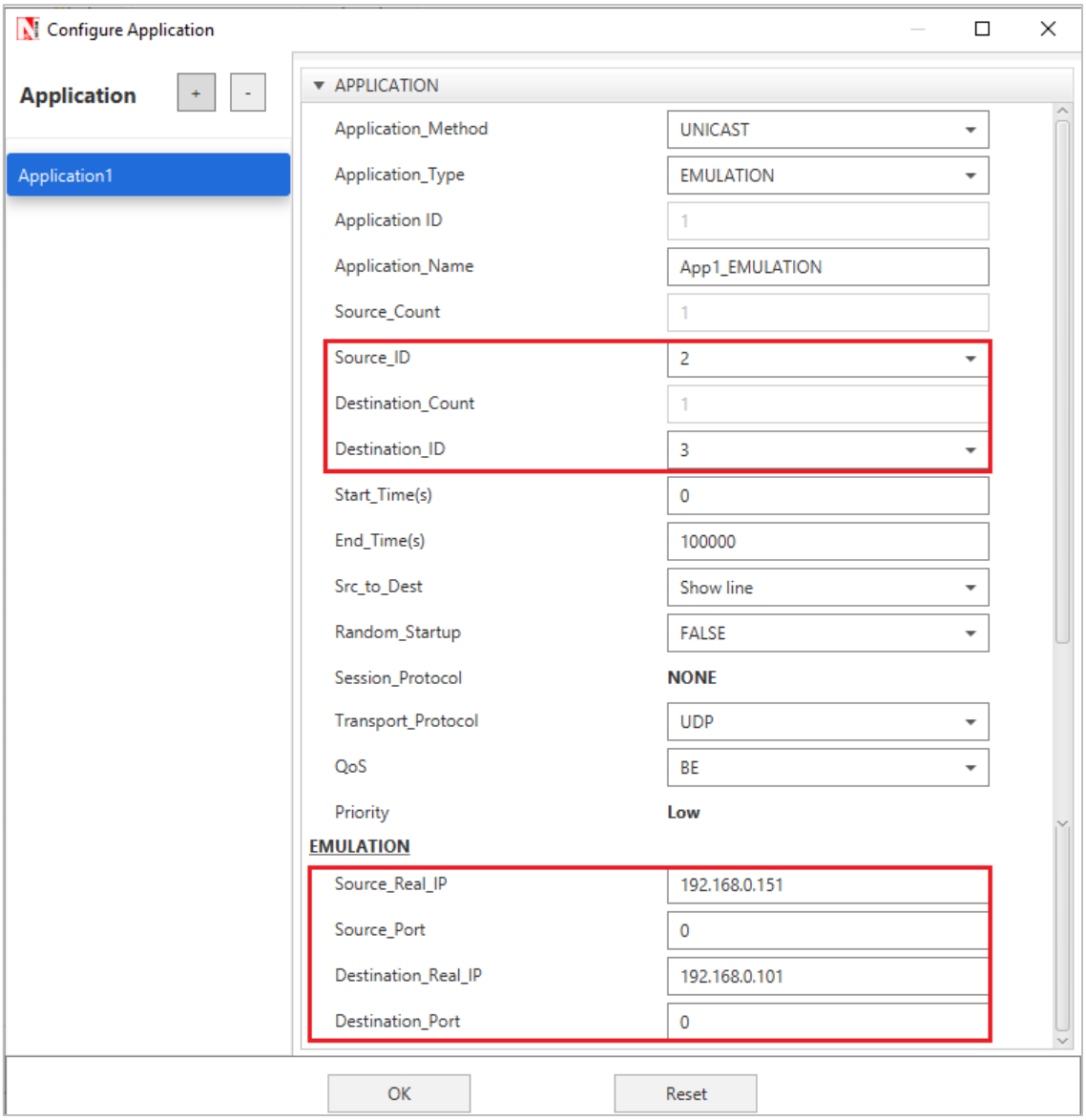


Figure 4-29: Application properties Window

5. Provide the Simulation Time as how long you want the Emulation to be performed. Make sure client system(s) are ready and then click Run Simulation.

Steps at Source PC

1. Follow steps as provided before in “Emulation Set-up: Setting up the NetSim Client”.
2. Run Skype and make a call to the destination system (Make sure that Skype is running in Destination PC).
3. Wireshark (if installed) will automatically start capturing the packets as soon as Emulation Server starts simulation.

Steps at Destination PC

- Follow steps as provided before in “**Emulation Set-up: Setting up the NetSim Client”**. After performing all the steps at Source PC and NetSim Emulation Server, open Skype.

- Wireshark (if installed) will automatically start capturing the packets as soon as Emulation Server starts simulation.

4.6 Example 5: Using JPerf

JPerf is a graphical front end for the popular network testing tool Iperf. Using JPerf you can quickly test a WAN or LAN connection to determine the maximum network throughput. The test results are automatically graphed and presented in a format that is easy to read. JPerf can also be used to detect packet loss, delay, jitter, and other common network problems.

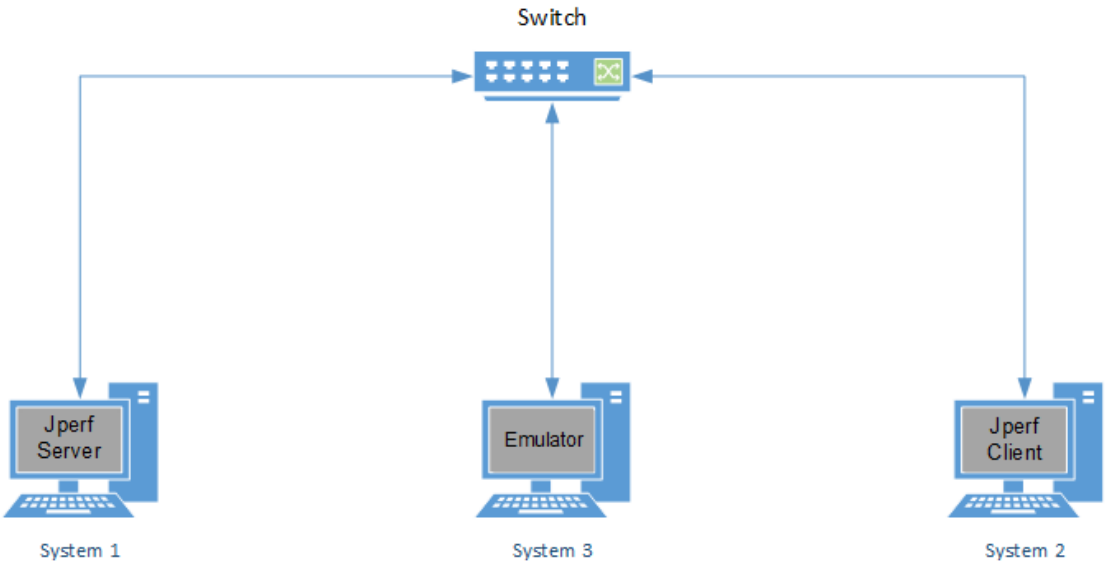


Figure 4-30: Jperf Client and server communication

Steps at NetSim Emulation Server

1. Run NetSim in Administrative Mode and create a basic network Scenario in any of the networks except Legacy Networks, Wireless Sensor Network, and Cellular Network. Screenshot of a sample scenario in Internetworks is shown below Figure 4-31.

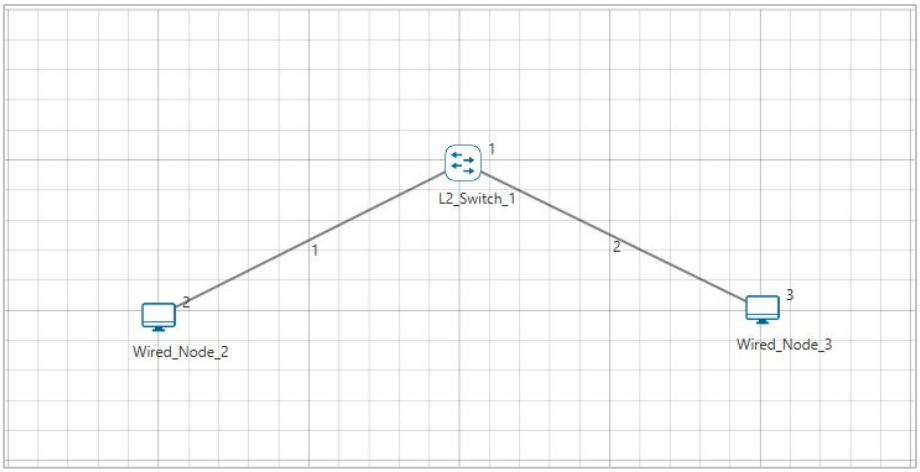


Figure 4-31: Network Topology in this experiment

- Click on the Application icon present on the ribbon and set properties.
- In the Application Type select Emulation.
- Select Source and Destination ID according to the network scenario and change the Source and Destination IP address according to the IP Address of the real system and click OK.

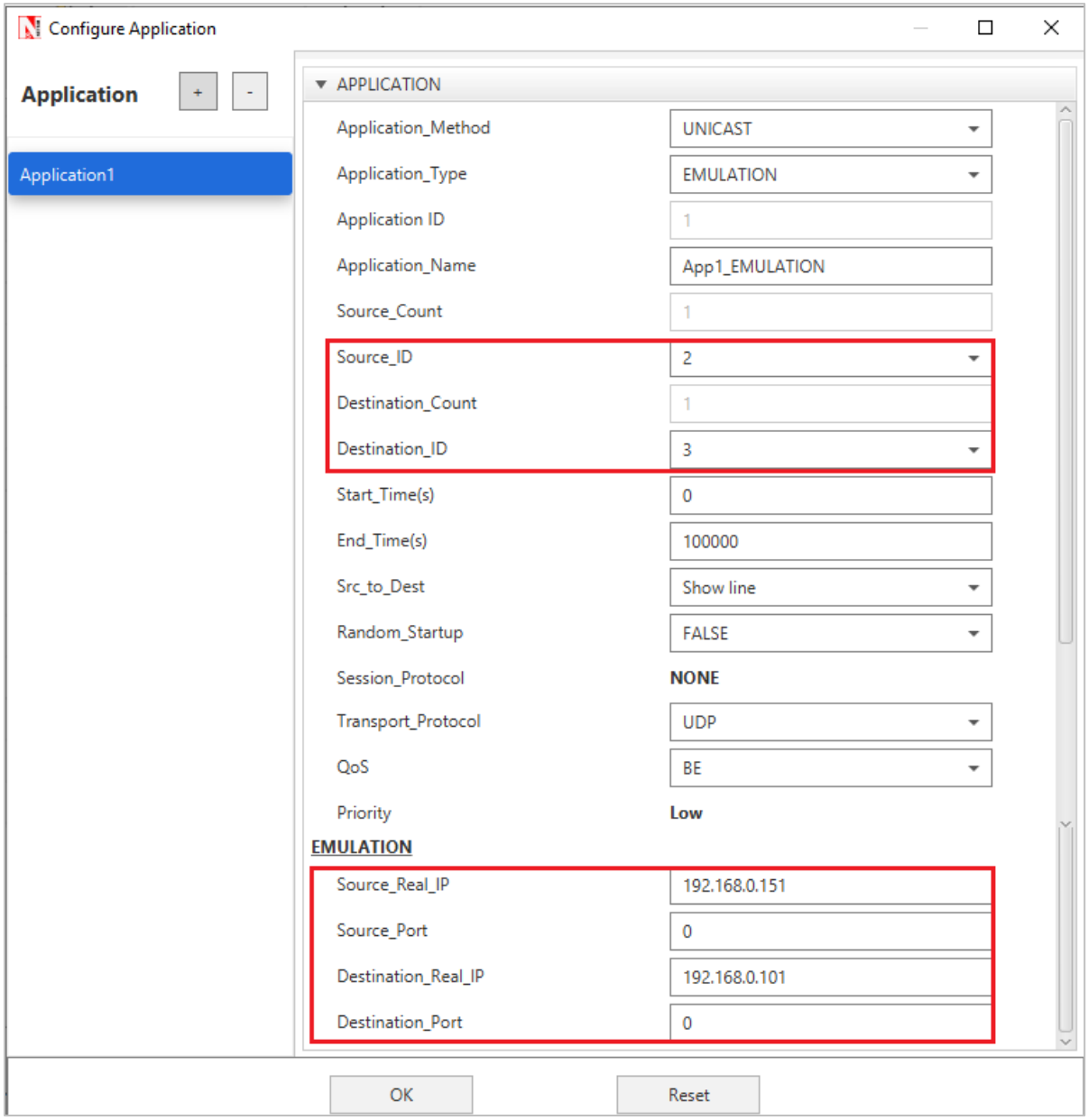


Figure 4-32: Application properties window

2. Provide the Simulation Time as per the time for which you want Emulation to be performed. Make sure client system(s) are ready and then click Run Simulation.

Steps at Source PC

Follow steps as provided before in “Emulation Set-up: Setting up the NetSim Client”. Run JPerf and select Client and set Server Address as <Server IP Address>. User can edit the

Application Layer options, Transport Layer options and IP Layer options depending on the type of data they want to transmit in the network.

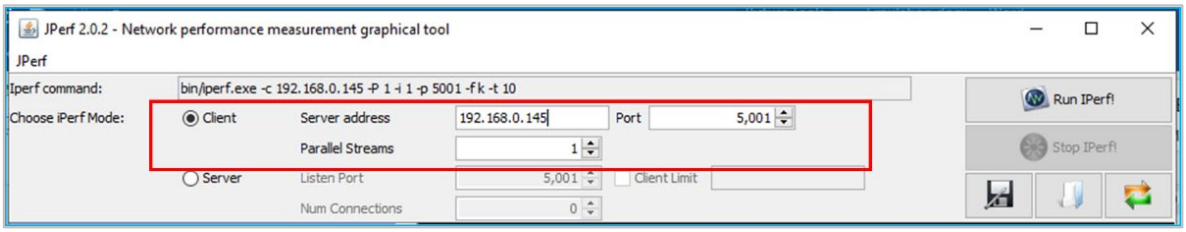


Figure 4-33: Select Client and setting done as per above screenshot in JPerf

Do not click “Run IPerf” until all the steps at NetSim Emulation Server are done. Also

Wireshark (if installed) will automatically start capturing the packets as soon as Emulation.

Server starts simulation.

Steps at Destination PC

Follow steps as provided before in “Emulation Set-up: Setting up the NetSim Client”. Run JPerf and select Server.

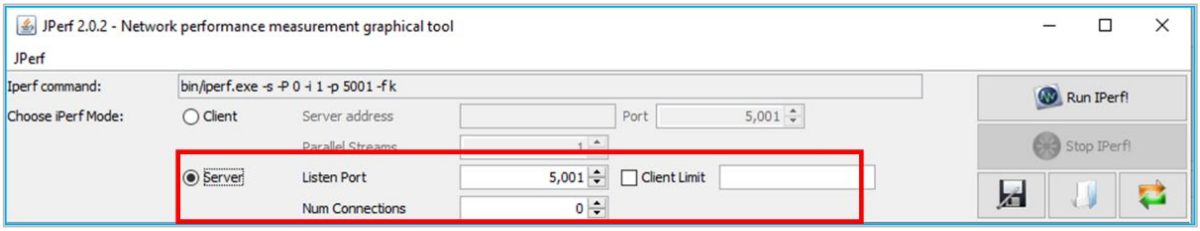


Figure 4-34: Select Server and setting done as per above screenshot in JPerf

Click on “**Run IPerf**” after the Source PC starts running JPerf.

4.7 Example 6: Simple Military (TDMA) Radio Use Case

Lab Setup: Before starting emulation, ensure that the real traffic that is generated at the NetSim Client (say Ping or Iperf traffic) is routed through NetSim Emulator System. This means the traffic just flows through the system (OS) and doesn't yet flow through NetSim.

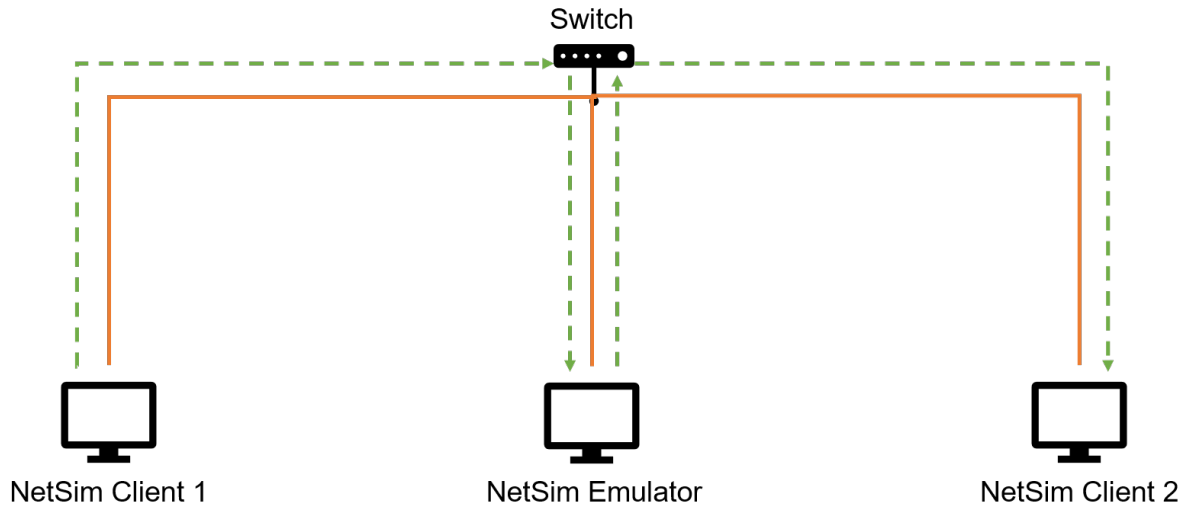


Figure 4-35: Lab set up where by two systems connect to the NetSim emulation server

- The Lab Setup where, Client 1 generates traffic, and Client 2 is the real destination to receive the traffic.
- Route the traffic generated by Client 1 via NetSim Emulation server. This is achieved by configuring static routes in the network. Please refer section 2.2.

Next, we create the scenario in NetSim, and this time run the emulator, whereby the traffic flows through NetSim. The steps to do this are:

- On NetSim Emulation Server, run NetSim in administrative mode and choose TDMA Radio Networks and create a scenario

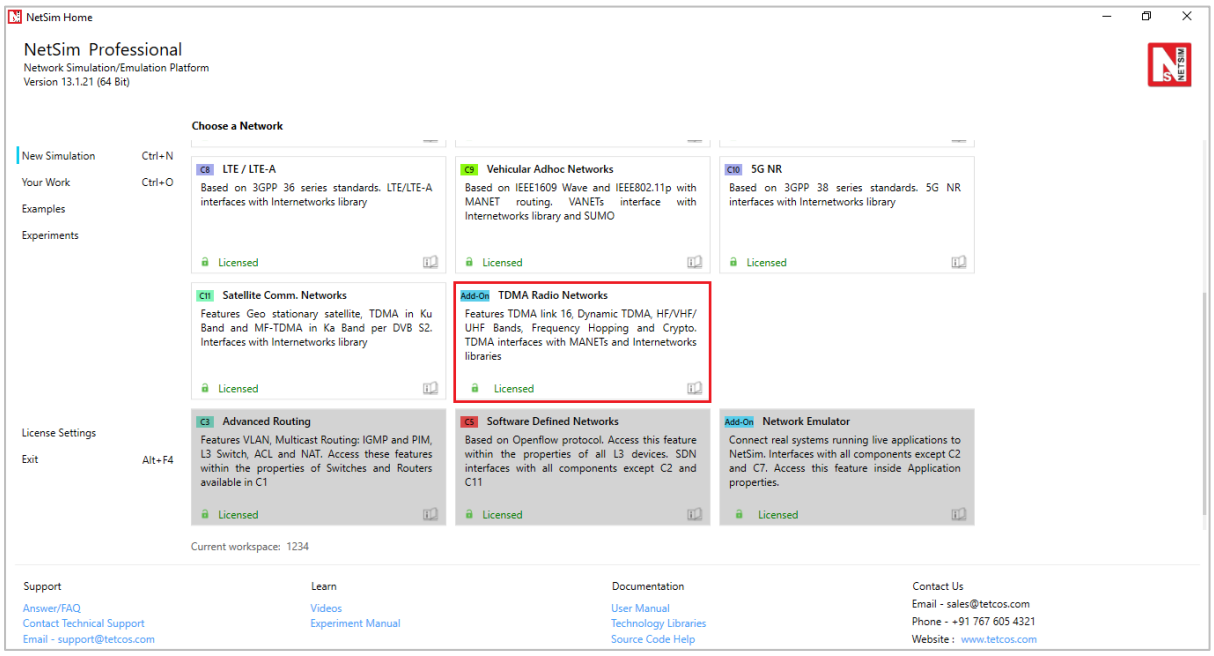


Figure 4-36: Select TDMA Radio Networks in NetSim home screen

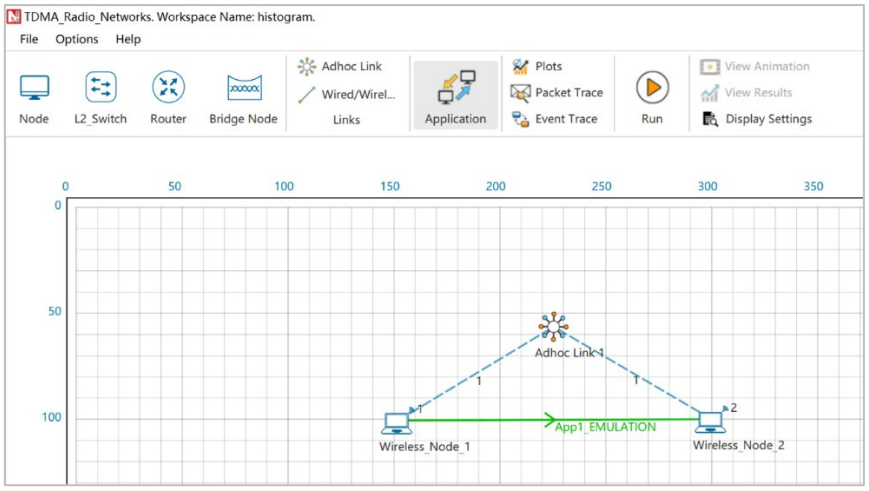


Figure 4-37: A simple TDMA radio network is created with two nodes

- Click on the **Application icon** present on the **ribbon** and set properties.
- In the **Application Type** select **Emulation**.
- Select Source and Destination ID according to the network scenario and change the Source and Destination IP address according to the IP address of the real system.

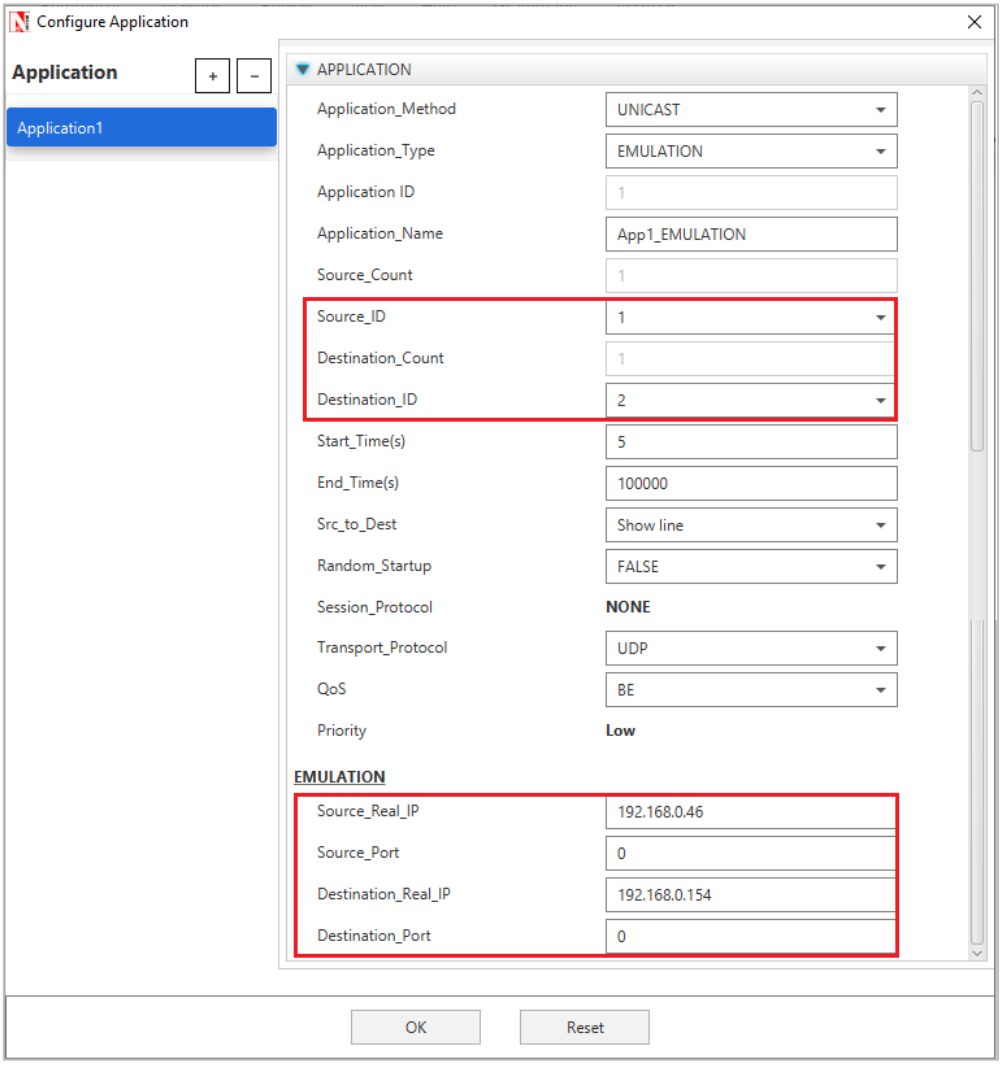


Figure 4-38: Application settings window and the parameters to be configured

Next, we change a property and see the effect. For example, change the PHY rate. The parameters that can be varied are:

- Frequency Band: L- Band
- Bandwidth:128KHz
- Data Symbol Rate: 128kBd
- Modulation Tech: QPSK
- Coding Rate: 1/2
- Min Slot Per Device: 0
- Static Route from Source to Destination

Right Click on Wireless node > Properties > Interface Layer > Physical Layer change parameters highlighted in the figure below. Ensure that you perform this in both the nodes since some of the parameter's scope is local in the Network.

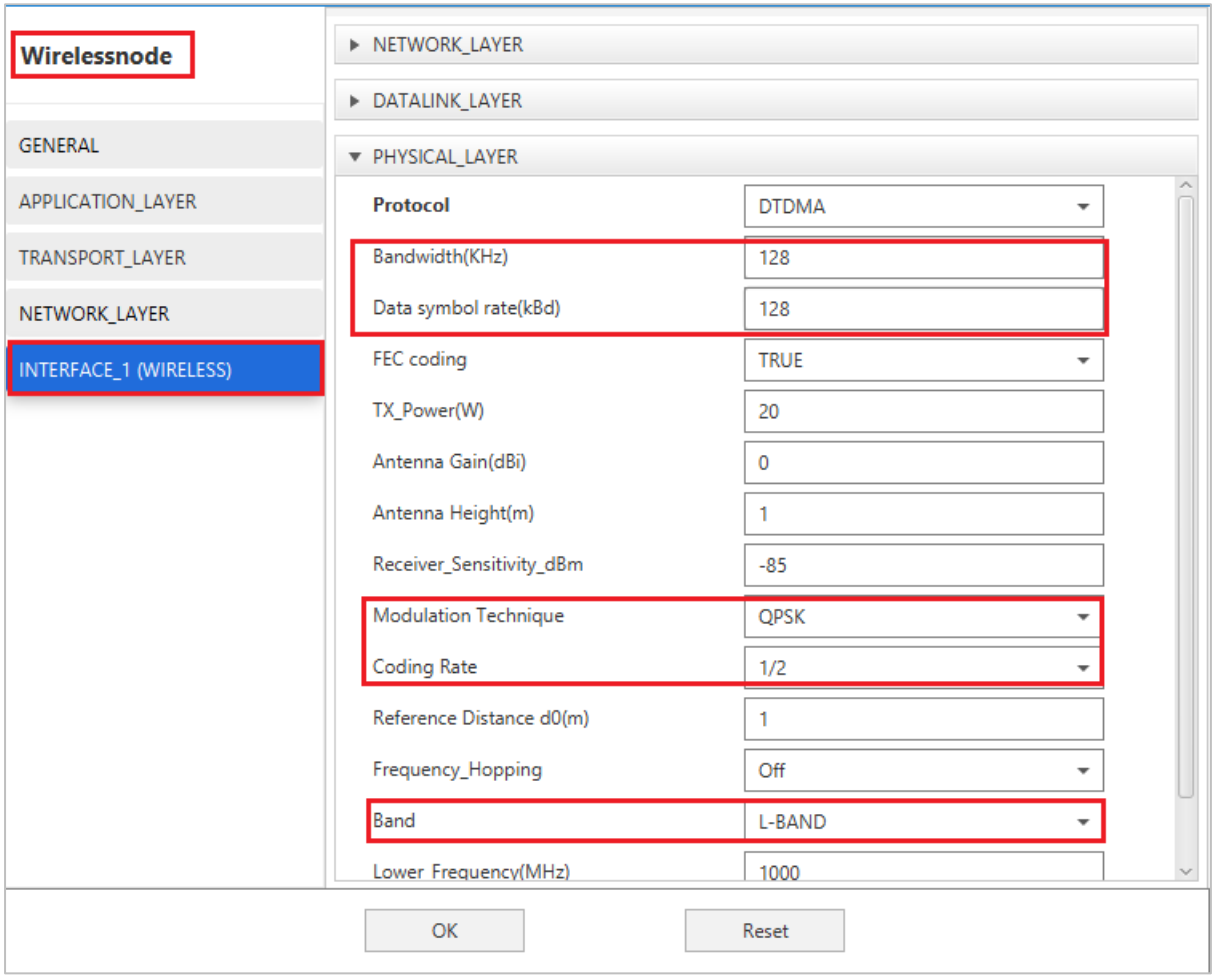


Figure 4-39: The parameters to be configured in the node's wireless interface

Steps to Set Static Route in NetSim:

1. Go to Display Settings > Check Device IP

2. Now go to Wireless Node properties->Network_Layer. Enable - Static IP Route ->Click on Configure Static Route IP and set the properties as per the screenshot below and click on Add and then click on OK.

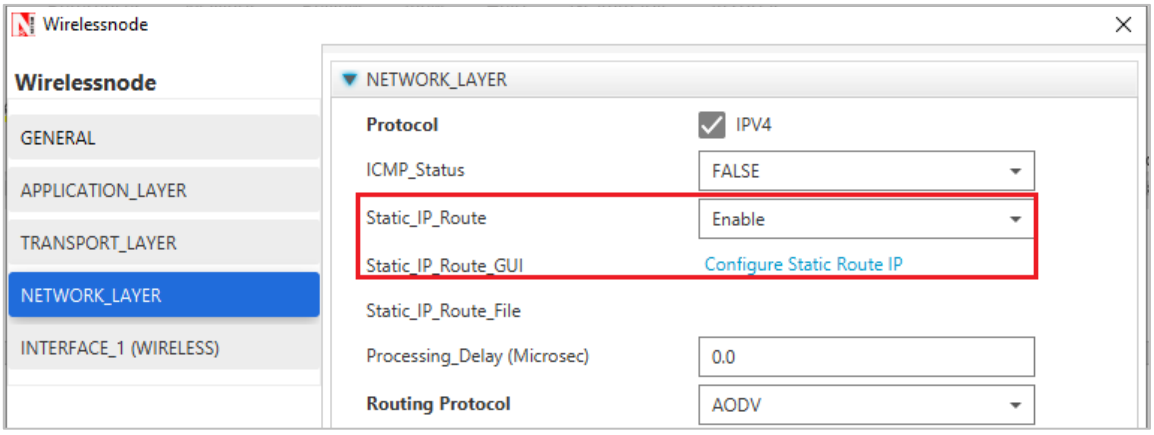


Figure 4-40: Wizard to configure static route

3. Add appropriate route from Source Node to Destination Node and vice versa

For Source Node:

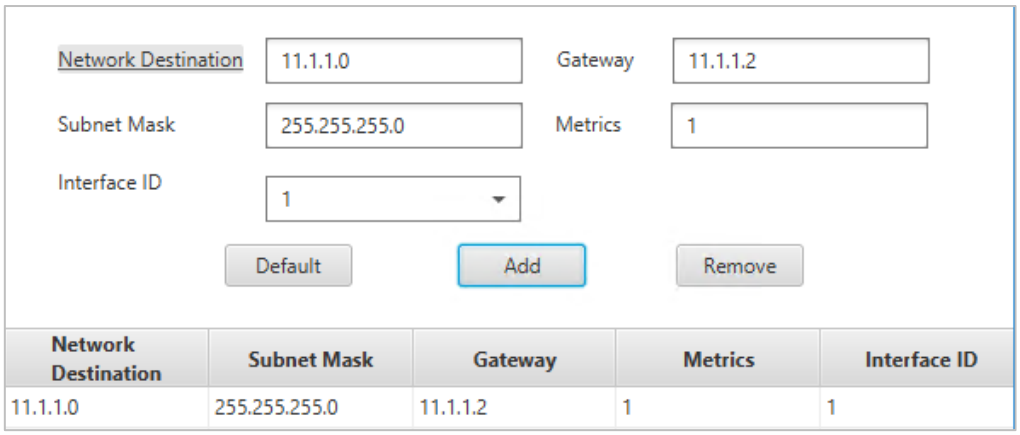


Figure 4-41: Static route setting in source node

For Destination Node:

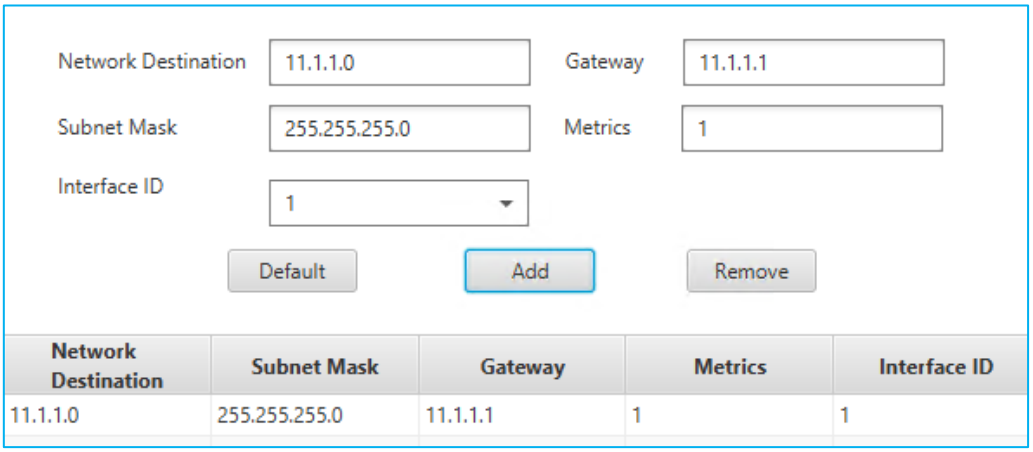


Figure 4-42: Static route setting in destination node

Provide the Simulation Time as how long you want the Emulation to be performed. Make sure client system(s) are ready and then click Run Simulation.

Now notice the following change in the iperf traffic. Let NetSim Client 1 be the Iperf Client and NetSim Client 2 be iperf server. Then commands at

Iperf Server: *iperf.exe -s -u -P 0 -i 1 -p 5001 -f k*

Iperf Client: *iperf.exe -c 192.168.0.154 -u -P 1 -i 1 -p 5001 -f m -b 1.0M -t 100000 -T 1*

Perform the steps at Emulation Server as provided and simulate. During simulation, generate 1 Mbps at iperf client (NetSim Client 1) to NetSim Client 2. You will notice that the even though the iperf client is generating 1Mbps iperf server will be receiving less than 120Kbps. This is because the network created in NetSim has PhyRate (wireless link capacity) set to 120Kbps.

Iperf Client Generating 1 Mbps traffic:

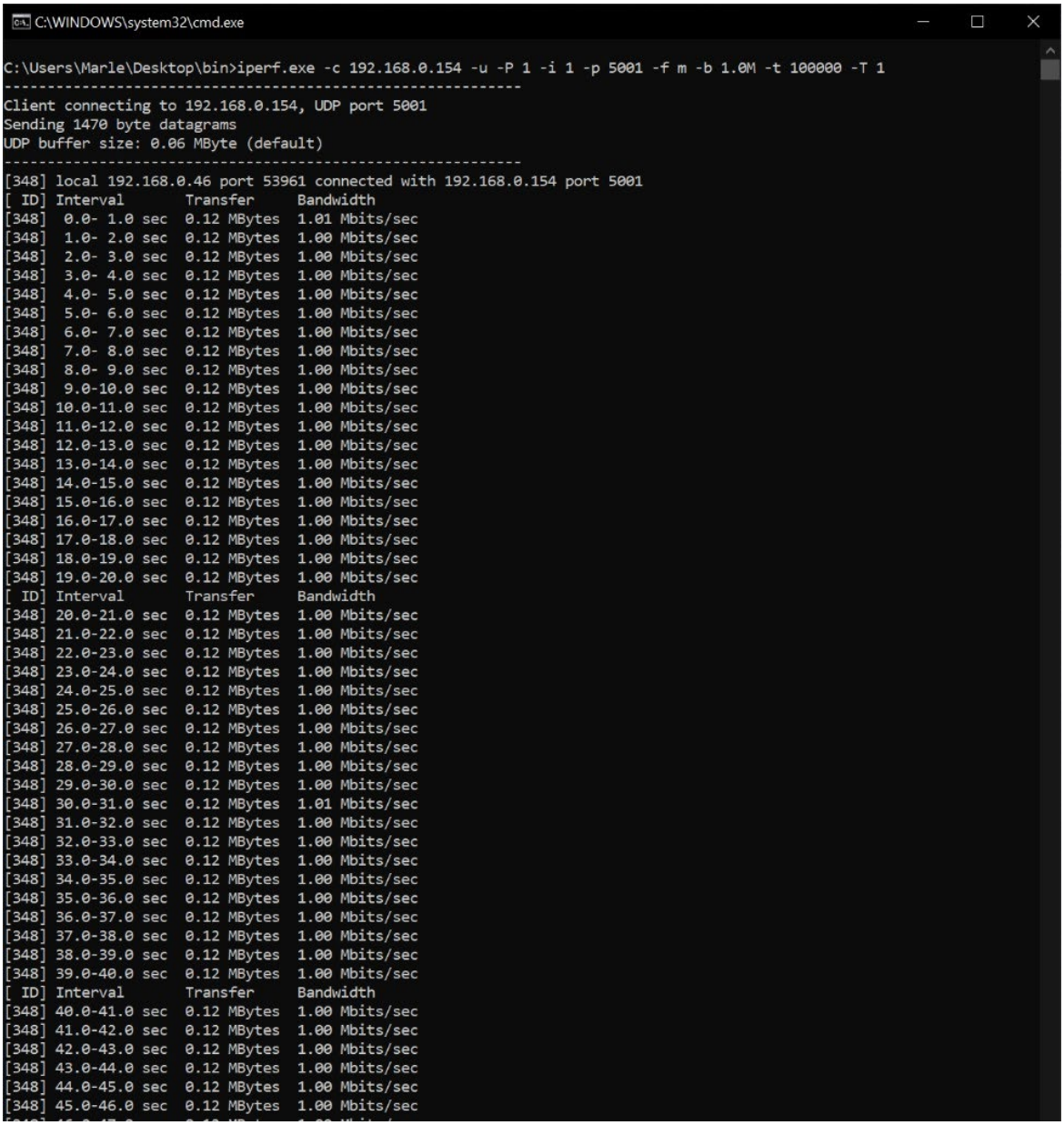


Figure 4-43: Iperf running in the client

Iperf Server receiving 120Kbps since Phyrate has been set to 120Kbps in NetSim

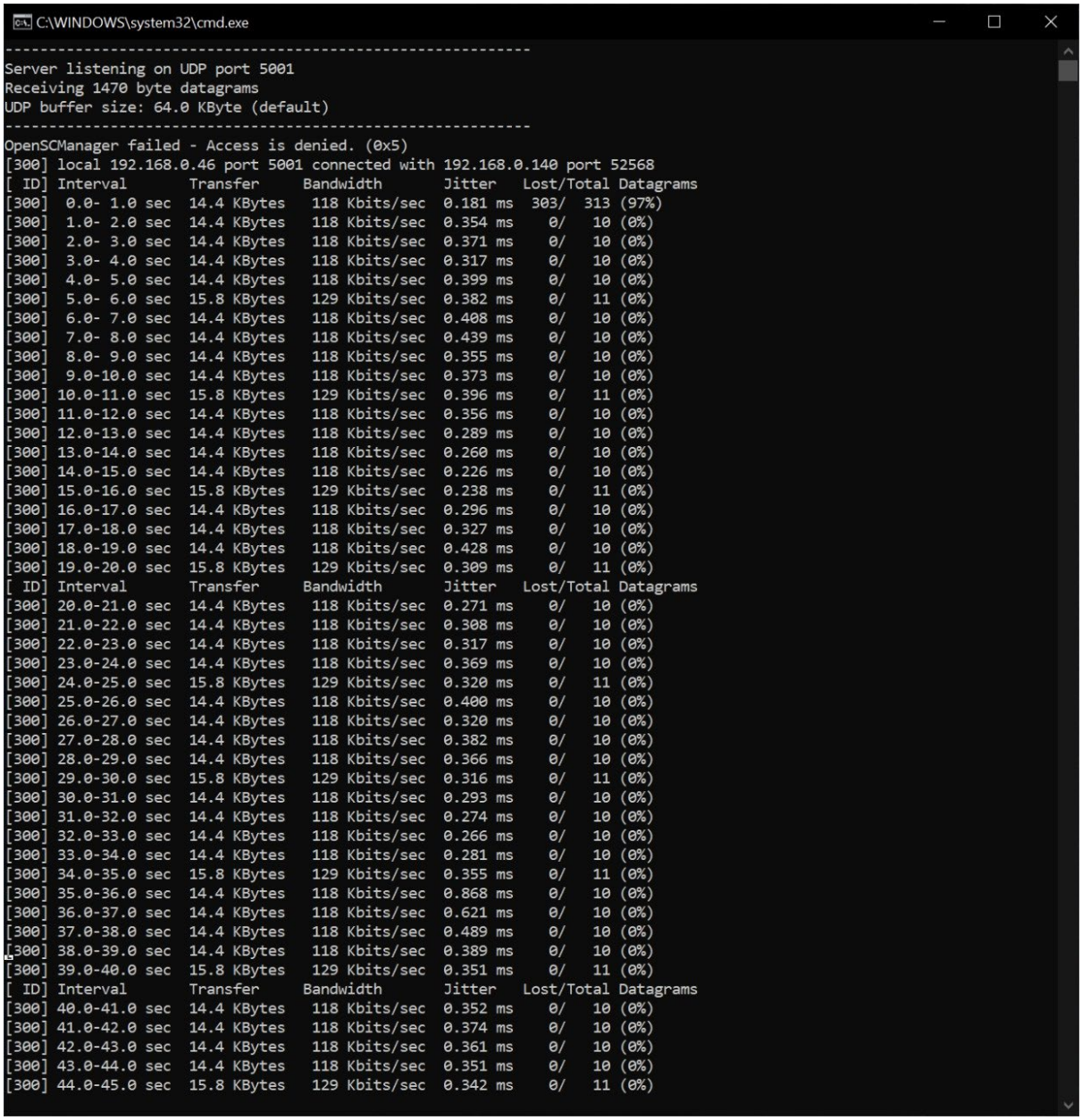


Figure 4-44: iperf running at the server showing throughput received

4.8 Providing pcap file as input to NetSim Emulator

NetSim has an inbuilt traffic generator (Application) which can be used to model unicast/multicast/broadcast traffic in the network with support for application types such as CBR, Custom, Voice, Video, FTP, Database, etc. NetSim also allows users to provide pcap file as input to NetSim traffic generator and configure applications using the real IP addresses and port numbers.

4.8.1 Generating a pcap file for NetSim

Any pcap file can be provided as input to NetSim by following the procedure explained here. In this example we are generating a pcap file by running wireshark and generating traffic using ping and saved it as Raw.pcap (while saving please make sure that save as file type must be

Wireshark/tcpdump/.. -pcap). Ping is initiated form the **Source IP:** 192.168.0.154 to the **Destination IP:** 192.168.0.192. The pcap file will contain all incoming and outgoing packets from the system in which the capture is being done.

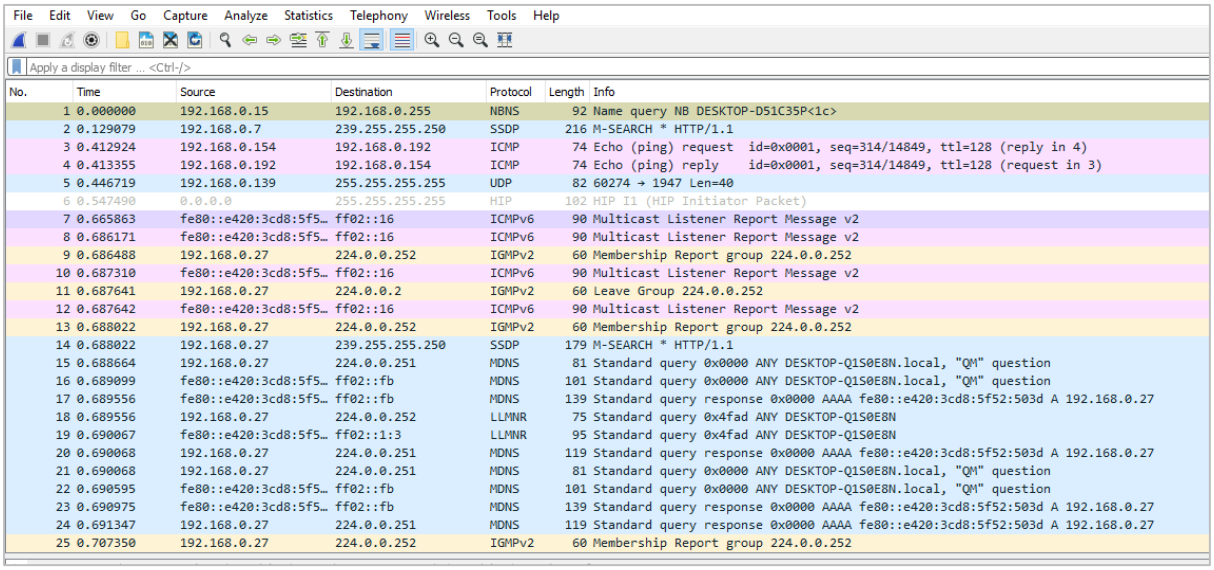


Figure 4-45: Packet captured in Wireshark

Wireshark capture can be stopped after capturing required packets and saved in desired location with a user defined name (*.pcap). E.g.: Input_to_NetSim.pcap

PCAP file needs to be edited suitably before providing it as input to NetSim. The "editcap" application in Wireshark installation directory can be used to edit any pcap file to be provided as input to NetSim.

Go to wireshark installation directory [C:\Program Files\Wireshark]

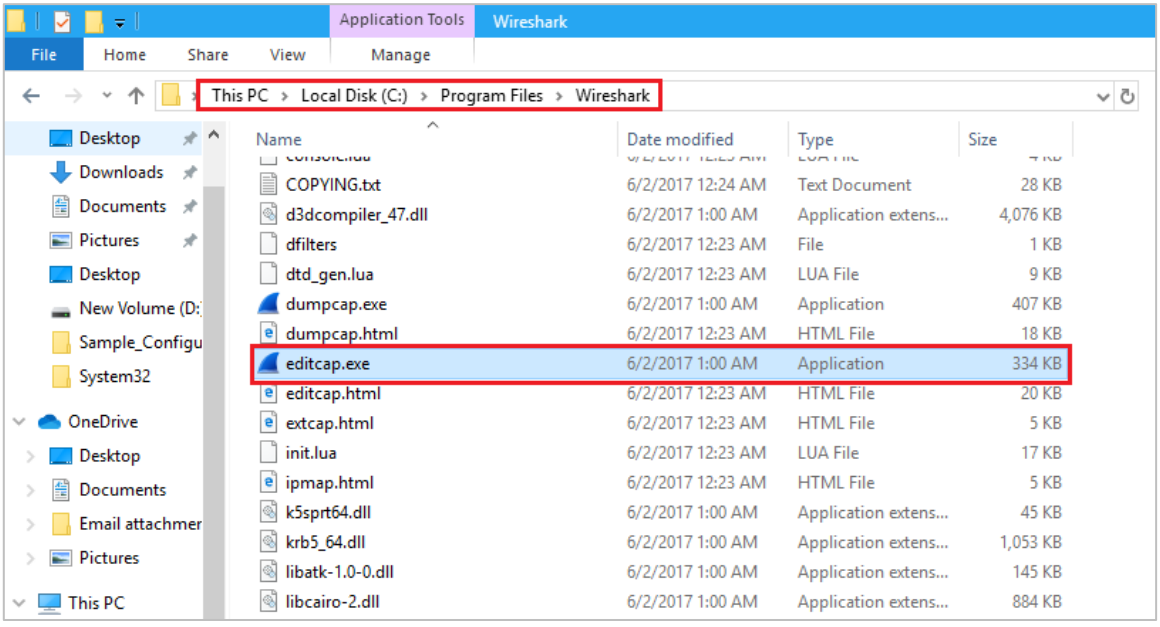


Figure 4-46: Installation directory of Wireshark

Open command prompt, and execute the following command:

editcap -C 14 -L -T rawip -F pcap "<File Location where the file is present>/Raw.pcap" "<File Location where the file needs to be saved>/Input_to_NetSim.pcap"

Where,

- **-C** [offset:]<choplen> chop each packet by <choplen> bytes. Positive values chop at the packet beginning, negative values at the packet end. If an optional offset precedes the length, then the bytes chopped will be offset from that value. Positive offsets are from the packet beginning, negative offsets are from the packet end. You can use this option more than once, allowing up to 2 chopping regions within a packet provided that at least 1 choplen is positive and at least 1 is negative**.**
- **-L** adjust the frame (i.e. reported) length when chopping and/or snapping
- **-T** <encap type> set the output file encapsulation type; default is the same as the input file. An empty "-T" option will list the encapsulation types.
- **-F** <capture type> set the output file type; default is pcapng. An empty "-F" option will list the file types.

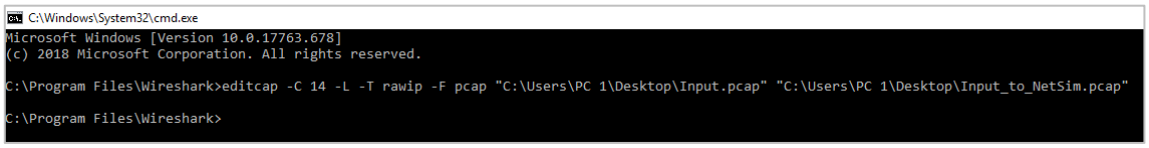


Figure 4-47: Enter the above command in command prompt

4.8.2 Providing pcap file as input to NetSim

1. Create a system environment Variable with Variable Name as “EMULATOR_INPUT” and value as the path of the pcap file along with the file name (Ex: C:\Users\bhatv\Desktop\NetSim\INPUT_TO_NETSIM.pcap).

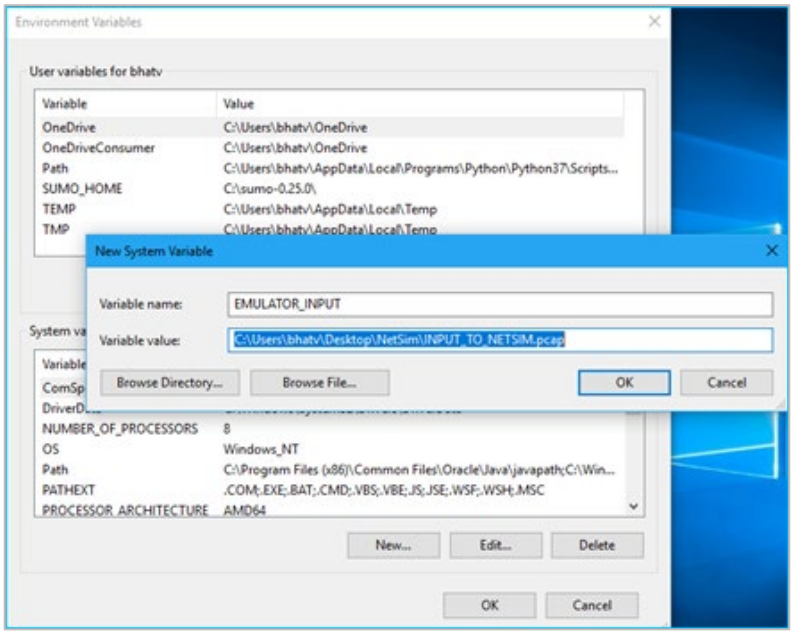


Figure 4-48: Create a system environment Variable name and providing input file path as input

2. Start NetSim as administrator. Now create a simple scenario in NetSim. For example, create a scenario in Internetworks with 1 router and 2 wired nodes as shown below. Create an Emulation application by giving the real source and destination IP's present in the pcap file. In this case we have used the IP addresses 192.168.0.154 and 192.168.0.192 which we have used for generating traffic using PING.

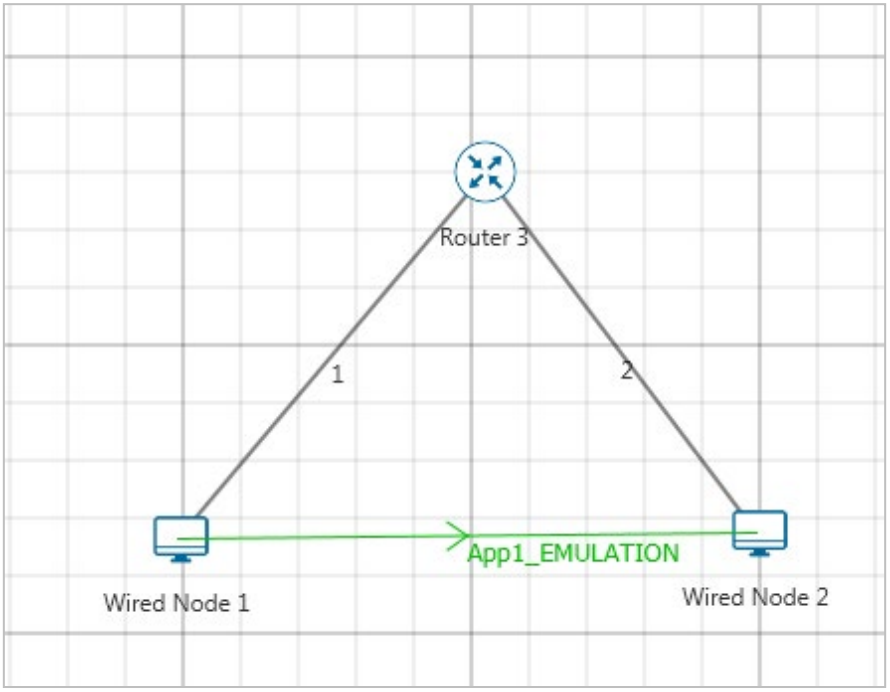


Figure 4-49: Network Topology in this experiment

3. Run NetSim in administrator mode for Emulator application.
4. Now create a simple scenario in NetSim. For example, create a scenario in Internetworks with 1 router and 2 wired nodes.
5. Create an Emulator application by giving the real source and destination IP's present in the pcap file.

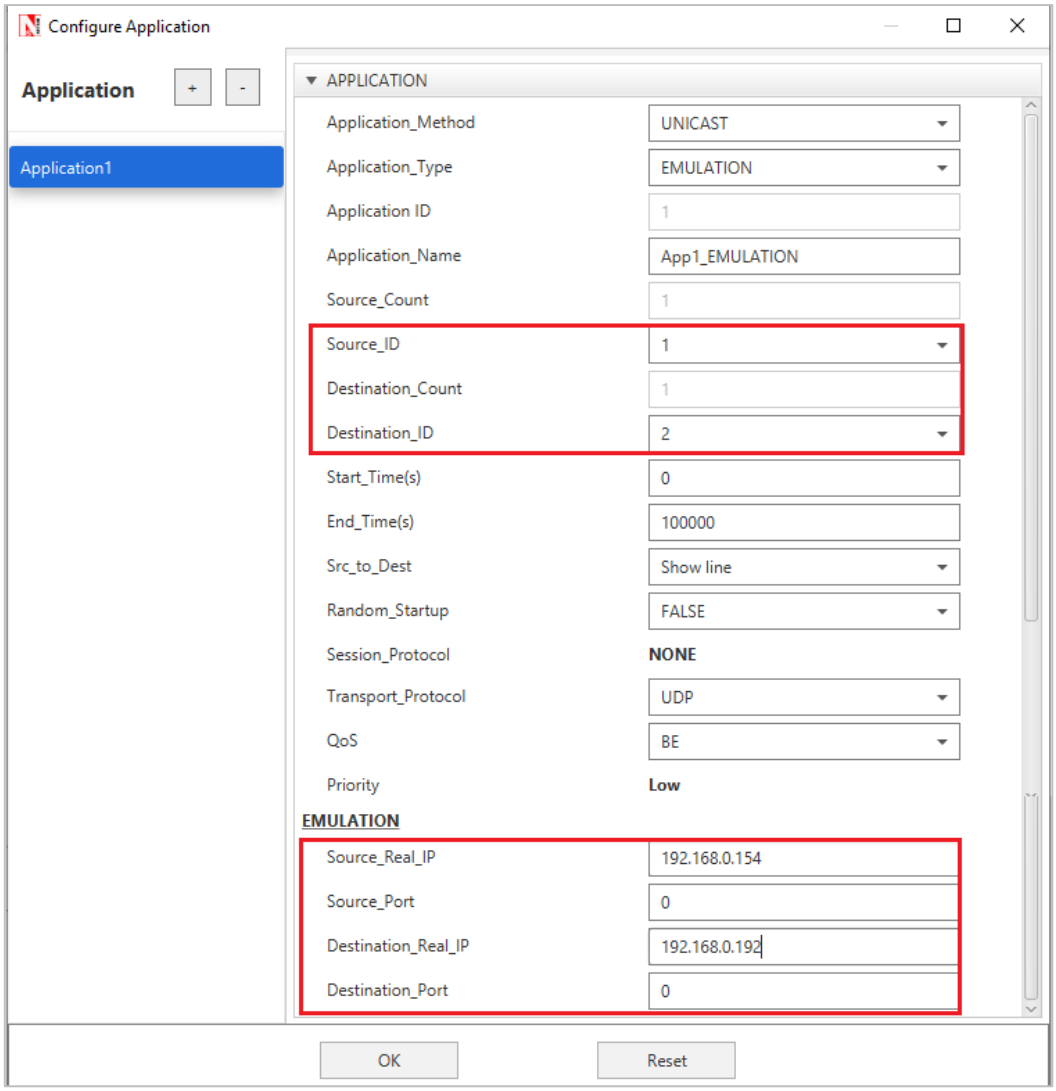


Figure 4-50: Application properties Window

- Multiple applications can be configured based on the packets captured. For instance, there can be one more application configured for packets from 192.168.0.192 to 192.168.0.154
- NetSim Emulator will read the packets from pcap file as per the source and destination that we are giving in the application properties.
- After simulation, NetSim results window provides Packet Capture Metrics. Here users can observe 4 different types of log files as explained in section 3.1
- Open Dispatch to Emulator packets, it contains only the packets whose source and destination IP addresses match with the source and destination IP addresses that we have configured.

5 Trouble shooting

5.1 "Ping: Request timed out" in DTDMA Radio Networks

This is specific to scenarios which used DEMAND BASED slot allocation in the MAC layer and having multiple emulation applications. When multiple systems trying to ping, the slot allocation methodology, causes delays that exceed the TTL of ping packet. A simple way to overcome this problem is to set the slot allocation technique as ROUNDROBIN.

It may happen that even after setting ROUNDROBIN as the slot allocation technique, a user could see the "Ping: Request timed out" error. This is due to high-rate traffic flows of non-emulation applications flowing through the emulator system. For example, consider a scenario with 4 systems A, B, C and D, such that

- A is the source system. The system sending Ping Request.
- B is the emulation server where NetSim is run.
- C is the destination System. The system sending Ping Reply.
- D is another system that is not a part of the Emulation set up under study

In this example, A pings C with traffic flowing through B. If simultaneously there is other traffic flowing from A to D (applications that are not part of Emulation set-up under study), then this traffic would also flow via emulator system due to static route setting. This will cause delays and the ping packets may then exceed the TTL. To avoid this set static route to a *particular Host system instead of adding static route based on Network*. In this same example, say the systems A, B, C and D are in the network 192.168.0.0/24, with IP addresses 192.168.0.2, 192.168.0.3, 192.168.0.4 and 192.168.0.5 respectively. Then user should not set

route add 192.168.0.0 MASK 255.255.255.0 192.168.0.3 METRIC 1

but instead, add the host specific static route

route add 192.168.0.4 MASK 255.255.255.255 192.168.0.3 METRIC 1

6 Latest FAQs

Up to date FAQs on NetSim's Emulation library is available at
https://tetcos.freshdesk.com/support/solutions/folders/14000115083