IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Volume-3, Issue-12 E-ISSN: 2347-2693

Advanced Open Source Simulator: NS-3

Rakesh Kumar Jha[1*] and Pooja Kharga [2]

[1*,2] Dept. of Electronics & Communication Engineering,
Shri Mata Vaishno Devi University, Katra, Jammu & Kashmir, INDIA

www.ijcseonline.org

Received: 24/Nov/2015 Revised:07/Dec/2015 Accepted:19/Dec/2015 Published: 31/Dec/21015

***Abstract*—** in the current scenario various network simulators are available like OPNET, QualNet etc but all these software are commercial that means for using mentioned software, first one has to get licence. Therefore we focus on open source simulator (NS2 and NS3) that is open for everyone. This paper presents guideline for understanding the NS3 (Network Simulation Tool). NS3 focuses on improvement of software integrations, core architecture, models and educational components of NS2. This paper deals with basic idea about the simulator tool, installation procedure and how to design a network using it. In this paper Solution for common error occurred during installation of simulator and running a network is also provided. The purpose of this paper is to make network's user familiar with the NS3 and its GUI (NetAnim).

***Keywords*—** NS3, NS2, NetAnim, wired, wireless

I. INTRODUCTION

Network simulator is used to test new protocols or change the existing protocols in a controlled and reproducible environment. We can design networks using various types of nodes, bridges, hubs, Routers and switches with the help of network simulator. It enables user to specify the nodes on the network, the links and the traffic between the nodes. There are number of network simulators [9] available and NS3 [8] is one of them. NS3 [1] [3] is an open source discrete- event network simulator and improves simulation integrity. NS3 [4] [5] is not backward compatible with NS2 but built from the scratch to replace NS2. Both NS2 and NS3 are written with the help of C++ but NS3 does not support NS2 APIs. Some models have been ported from NS2 to NS3. NS3 is written in C++, with optional python bindings. GUI based network simulator named NetAnim is used for NS3 [6] [7]. It is based on Qt4 GUI toolkit. To display the simulation graphically, it uses XML trace files.

Section II describes the overview of NS3 and its installation procedure. Section III discusses NetAnim. Section IV and V presents the simulation of wired and Wireless Network. Conclusion of the whole paper is provided in Section VI. Section VII presents the future scope of NS3.

II. NS3

NS3 is modular design network simulator can be connected to a real network. It has logging facility for debugging and tracing for getting output. Most of the users focus on wireless simulation includes models for Wi-Fi, LTE or WiMAX and routing protocols such as AODV and OLSR. NS3 is split over number of modules having one or more models for real network device and protocols.

Installation:

We need to install following packages before installing NS3.

$ sudo apt-get install gcc g++ python

$ sudo apt-get install gcc g++ python python-dev

$ sudo apt-get install mercurial

$ sudo apt-get install bzr

$ sudo apt-get install gdb valgrind

$ sudo apt-get install gsl-bin libgsl0-dev libgsl0ldbl

$ sudo apt-get install flex bison libfl-dev

$ sudo apt-get install g++-3.4 gcc-3.4

$ sudo apt-get install tcpdump

$ sudo apt-get install sqlite sqlite3 libsqlite3-dev

$ sudo apt-get install libxml2 libxml2-dev

$ sudo apt-get install libgtk2.0-0 libgtk2.0-dev

$ sudo apt-get install vtun lxc

$ sudo apt-get install uncrustify

$ sudo apt-get install doxygen graphviz imagemagick

$ sudo apt-get install texlive texlive-extra-utils texlive-latex-extra

$ sudo apt-get install python-sphinx dia

$ sudo apt-get install python-pygraphviz python-kiwi python-pygoocanvas libgoocanvas-dev

$ sudo apt-get install libboost-signals-dev libboost-filesystem-dev

$ sudo apt-get install openmpi*

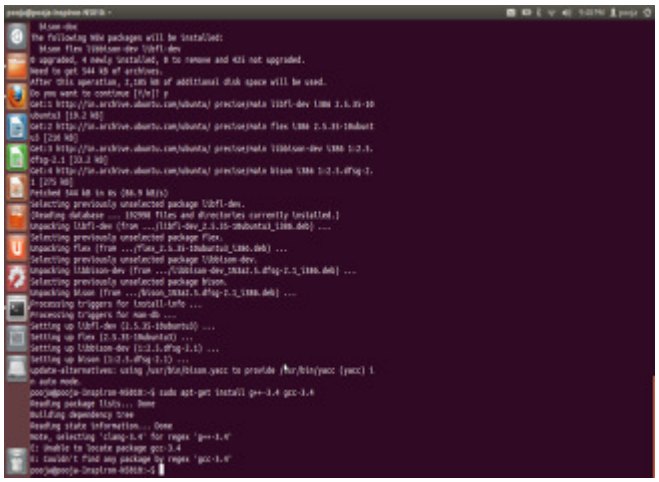


Figure 1: Error occurred during installation of required packages

While getting an error as shown in Figure 1 during execution of command "sudo apt-get install g++-3.4 gcc-3.4", just run following command.

$ sudo apt-get install gcc g++

$ sudo apt-get update

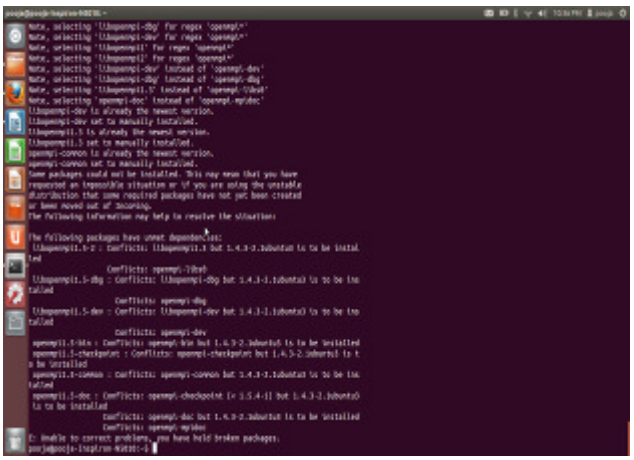


Figure 2: Error occurred during installation of required packages

While getting an error as shown in Figure 2 during executing the command "sudo apt-get install openmpi*", just run this command

$ sudo apt-get install openmpi-bin openmpi-doc libopenmpi-dev

To install NS3, follow these steps:

Clone repositories and download code:

$ hg clone http://code.nsnam.org/ns-3-allinone

$ cd ns-3-allinone

$ ./download.py -n ns-3.16

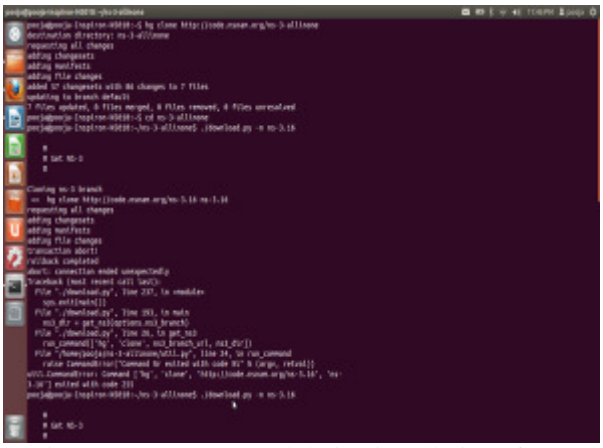


Figure 3: Error occurred during installation of NS3

If internet connection is too slow, we could get an error like that so for installing this internet connection must be too good.

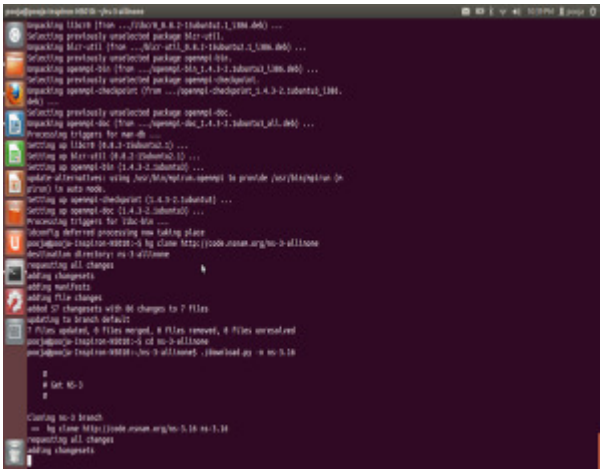


Figure 4: Installation of NS3

Build the code:

$ ./build.py --enable-examples --enable-tests

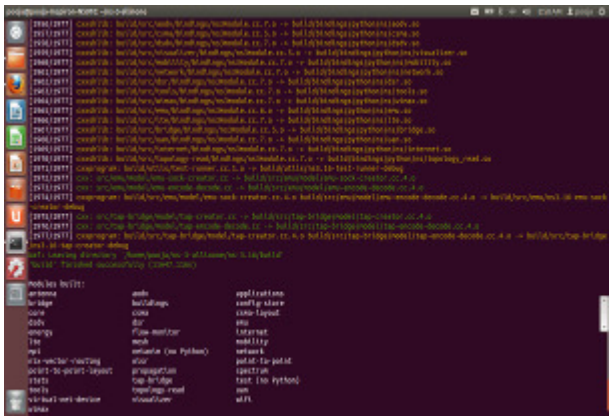


Figure 5: Building the code

tests for build:
$ cd ns-3.16
$ ./test.py -c core

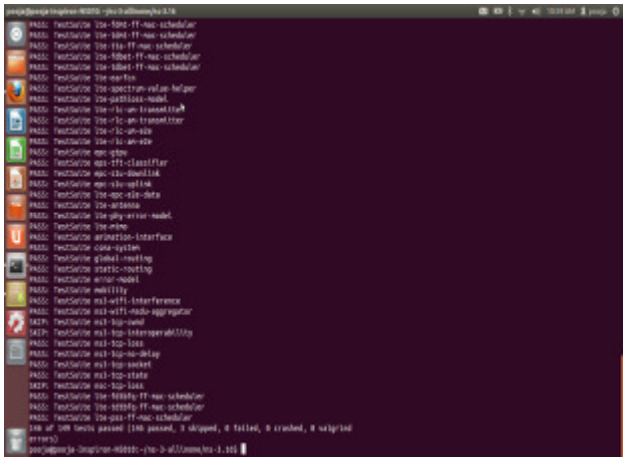


Figure 6: Testing for Build

Running a script
$ cd ns-3-allinone
$ cd ns-3.16
$ ./waf --run examples/tutorial/first.cc

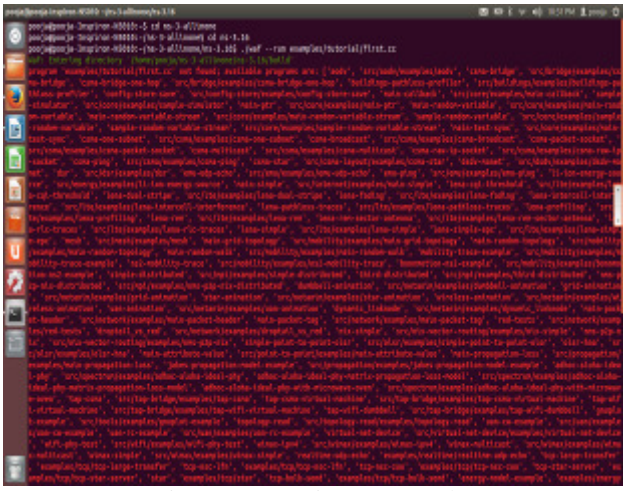


Figure 7: Running a network

All scripts should be run under scratch folder. If we run a script directly then we get a result as shown in figure 7. So first we need to copy the script into the scratch folder and then run the script.

$ cd ns-3-allinone
$ cd ns-3.16
$ cp examples/tutorial/first.cc scratch/tut1.cc
$ ./waf
$ ./waf --run scratch/tut1

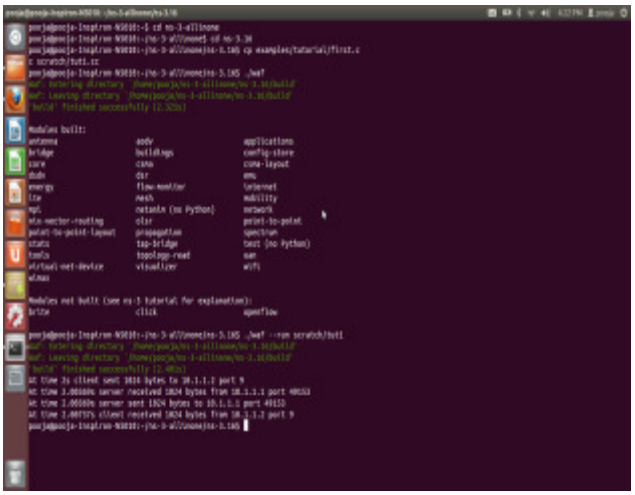


Figure 8: Running a Network

As shown in figure 8; build system checks whether file has been build and then runs it or not perfectly. Client sent 1024 bytes to 10.1.1.2 at port 9 and server acknowledges the client at port 49153. The echo server silently echoes the packet and the echo client log that it has received its packet back from the server.

III. NETANIM

NetAnim [2] is Graphical user interface (GUI) based network simulator used for NS3. IP address and Mac address of nodes can also be checked in it.

We need to install this package before configuring NetAnim.
$ sudo apt-get install qt4-dev-tools

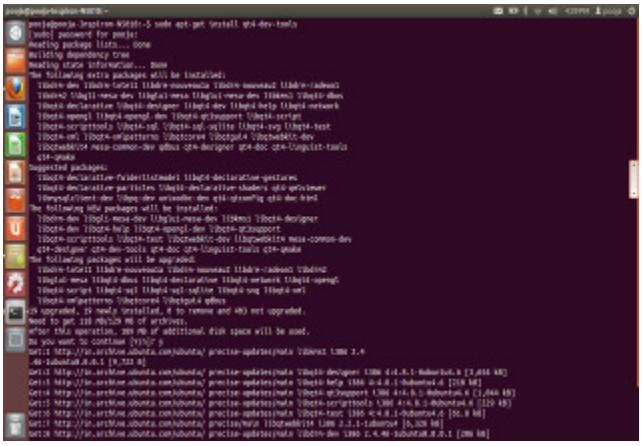


Figure 9: Installation of required package

Configure NetAnim
$ cd ns-3-allinone
$ cd netanim
$ make clean
$ qmake NetAnim.pro
$ make

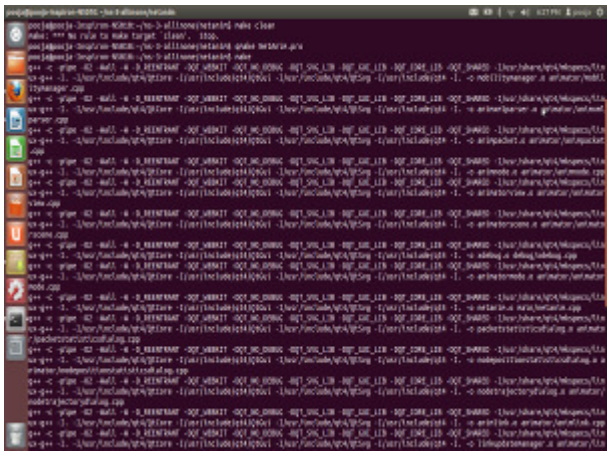


Figure 10: Configuring NetAnim

$ geany scratch/tut1.cc

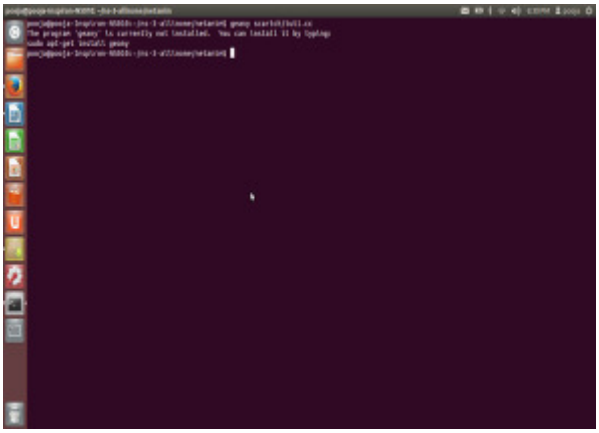


Figure 11: Open the script with text editor

If the program geany is not already installed, just type the following command for installing it.
$ sudo apt-get install geany

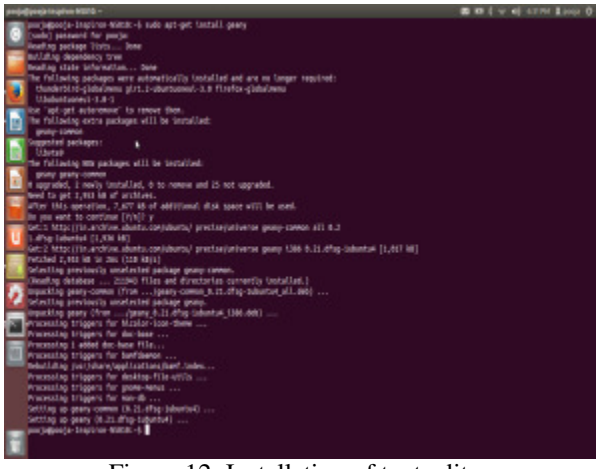


Figure 12: Installation of text editor

Geany is a text editor used to provide a small and fast integrated development environment. Syntax highlighting, auto closing of XML and HTML, and build system to compile and execute code are main features of it.

IV. SIMULATED WIRED NETWORK

Let's discuss the wired network with example first.cc
$ cd ns-3-allinone
$ cd ns-3.16
$ geany scratch/tut1.cc
Add the header file to the script tut1.cc
#include "ns3/netanim-module.h"

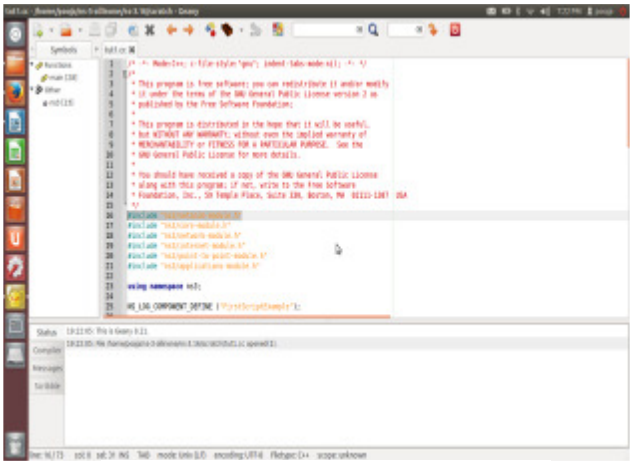
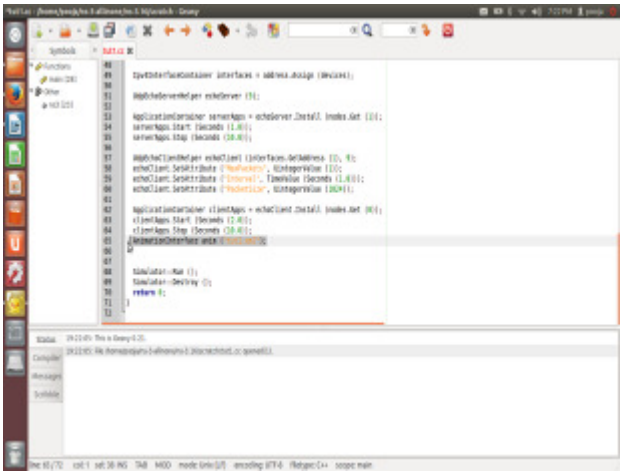


Figure 13: Adding header file to the script

AnimationInterface anim ("animation.xml");
AnimationInterface is responsible for creation of XML file and uses the tracing infrastructure to track packets flow between nodes.
Add the statement before Simulation::Run()
AnimationInterface anim ("tut1.xml");

Figure 14: Adding required statement to the script



"SetConstantPosition" set the x-y coordinates of a node

which is stationary.
anim.SetConstantPosition (Ptr< Node > n, double x, double y);
Set the positions into tut1.cc
anim.SetConstantPosition (nodes.Get(0), 1.0, 2.0);
anim.SetConstantPosition (nodes.Get(1), 2.0, 3.0);

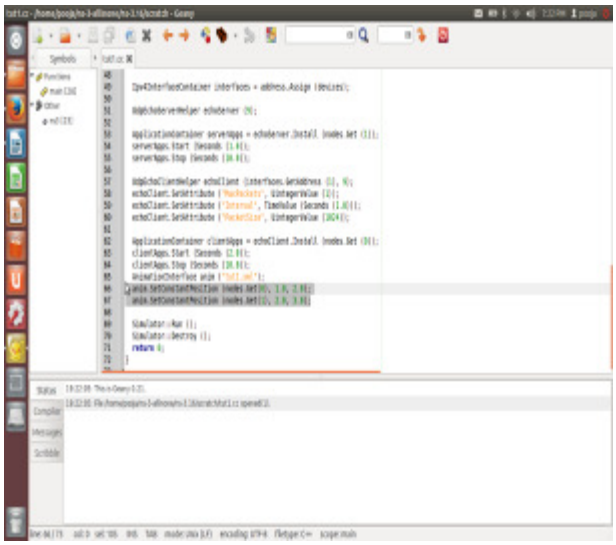


Figure15: Adding required statement for setting the position of Network

NetAnim uses Metadata to provide better statistics and filter, and some brief information about the packets such as TCP sequence number or source and destination IP address during packet animation.

Add this statement into tut1.cc
anim.EnablePacketMetadata (true);
But we shouldn't enable this feature when using Wimax links.

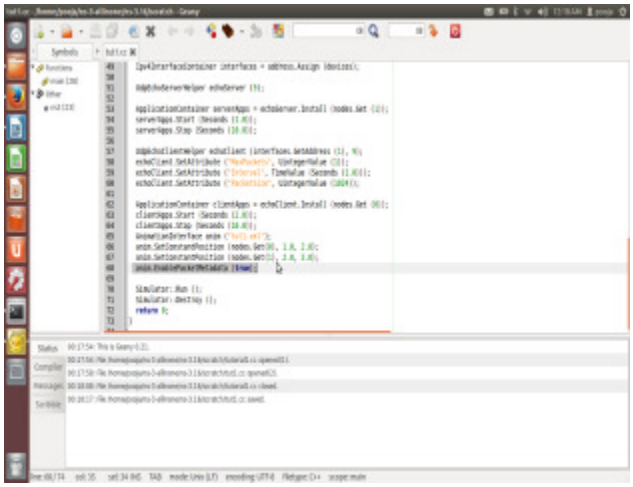


Figure 16: Adding statement to enable metadata

Save and run the script

$ cd ns-3-allinone
$ cd ns-3.16
$ ./waf --run tut1

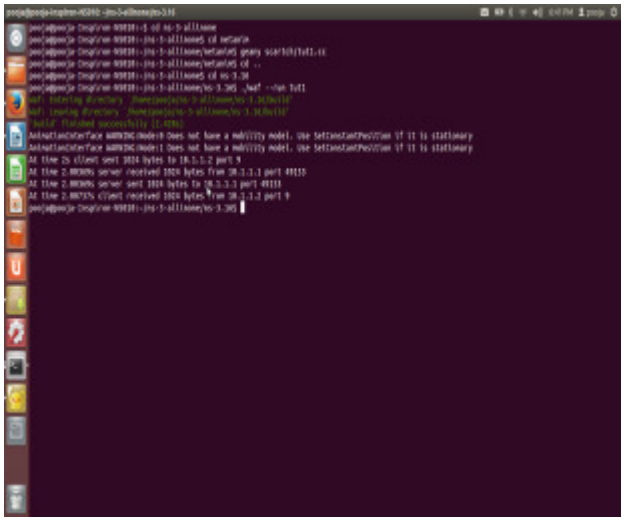


Figure 17: Running a Wired Network

$ cd ..
$ cd netanim
$ ./NetAnim
Open XML trace file, go to /ns-3-allinone/ns-3.16 and then open tut1.xml.

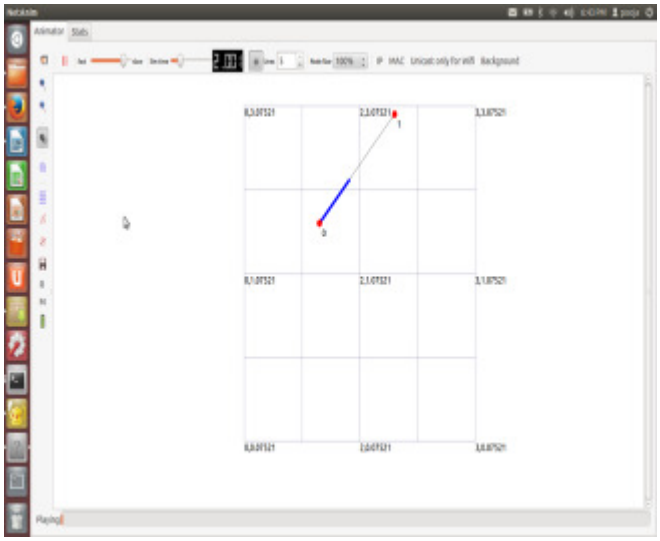


Figure 18: Running a wired Network using GUI

Figure 18 depicts that client (node 0) make request to a server (node 1) by sending 1024 bytes and server responds to client by acting on request and returning result. Here Metadata is not enabled.

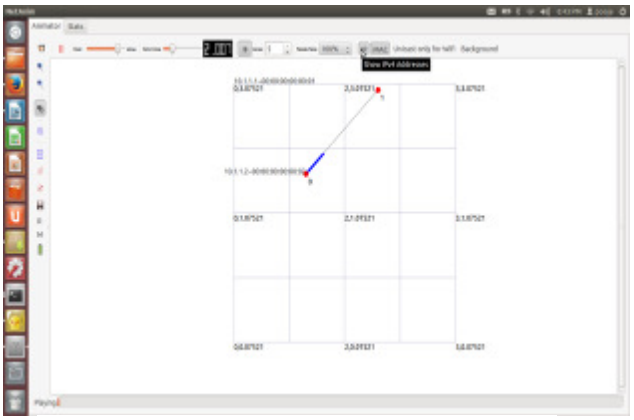


Figure 19: Running a Wired Network using GUI

As shown in figure 19; IP address and MAC address of each node can also be shown in GUI based network simulator.

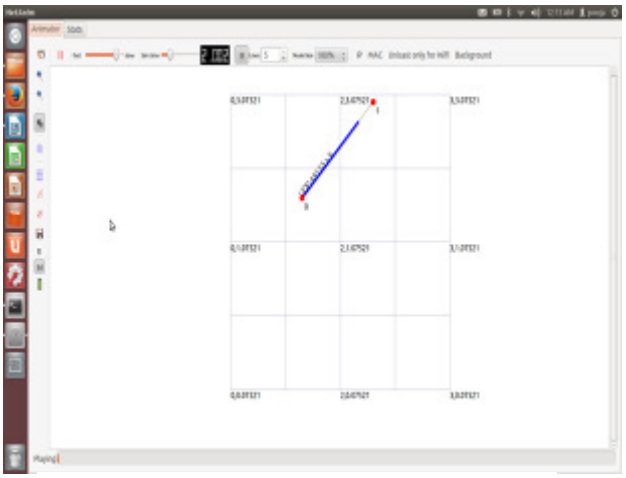


Figure 20: Running a Wired Network using GUI

We have enabled packet meta data as shown in figure.

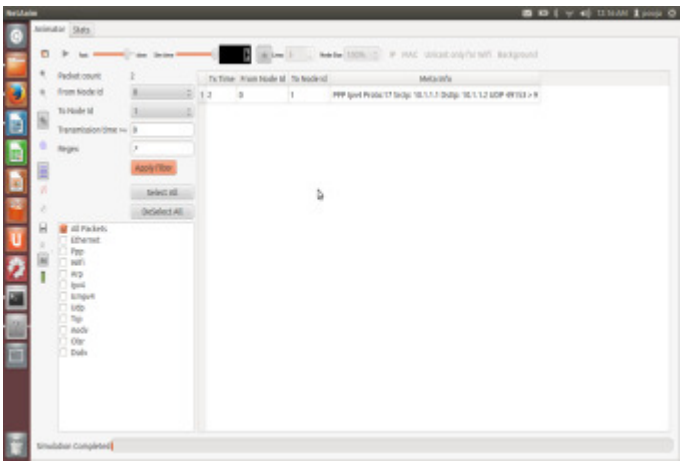


Figure 21: Meta information during simulation

Here we use regex filter to get packet and statistic data.

V. SIMULATED WIRELESS NETWORK

Now let's discuss the wireless Network with third.cc.

$ cd ns-3-allinone
$ cd ns-3.16
$ cp examples/tutorial/third.cc scratch/tutorial3.cc
$ ./waf
$ ./waf --run scratch/tutorial3
$ geany scratch/tutorial3.cc

In order to generate .xml file in NetAnim we have to add following codes into the script:

Add the header file to the script tutorial3.cc

#include "ns3/netanim-module.h"

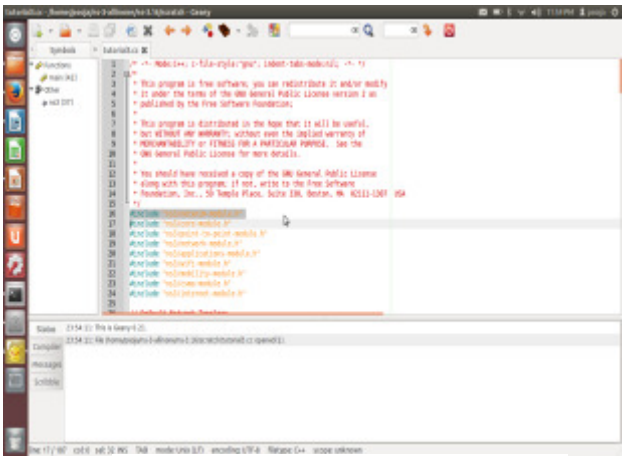


Figure 22: Adding header file to the script

Add these statement before Simulation::Run()

AnimationInterface anim ("tutorial3.xml");

set the positions into tutorial3.cc

anim.SetConstantPosition (wifiStaNodes.Get (0), 1.0, 6.0);
anim.SetConstantPosition (csmaNodes.Get (0), 6.0, 10.0);
anim.SetConstantPosition (csmaNodes.Get (1), 4.0, 12.0);
anim.SetConstantPosition (csmaNodes.Get (2), 6.0, 12.0);
anim.SetConstantPosition (csmaNodes.Get (3), 8.0, 12.0);

Add this statement into tutorial3.cc

anim.EnablePacketMetadata (true);

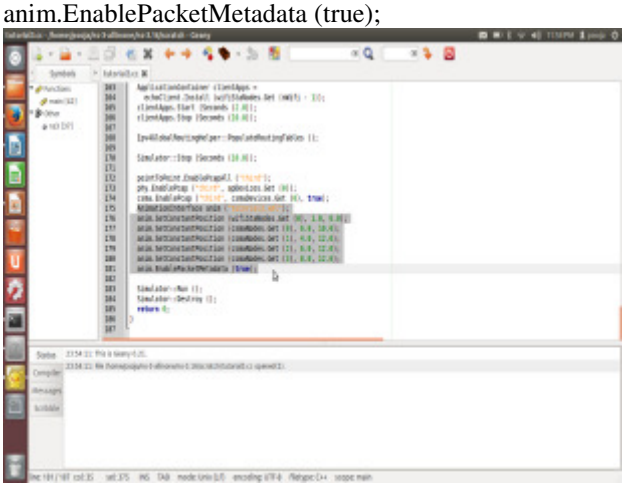


Figure 23: Adding required statement to the script

Save the file and run the script.
$ ./waf --run scratch/tutorial3

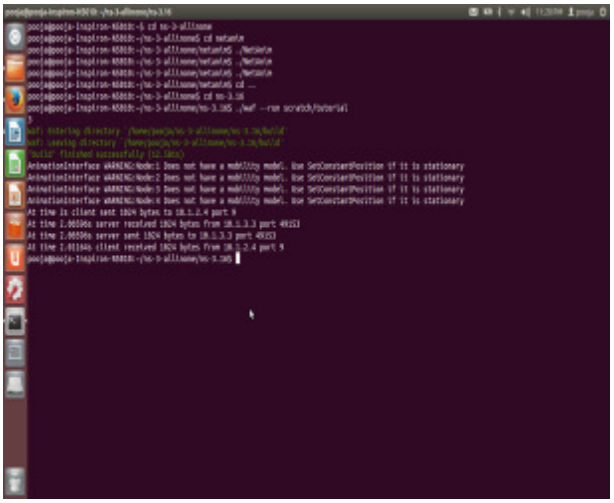


Figure 24: Running a Wireless Network

Here UDP echo client sent 1024 bytes packet to server (10.1.2.4). In this case client is on the wireless network (10.1.3.0). When server receives the echo packets, it acknowledges. The final message indicates that client received the echo packet back.

$ cd ..
$ cd netanim
$ ./NetAnim

Open XML trace file, go to /ns-3-allinone/ns-3.16 and then open tutorial3.xml.

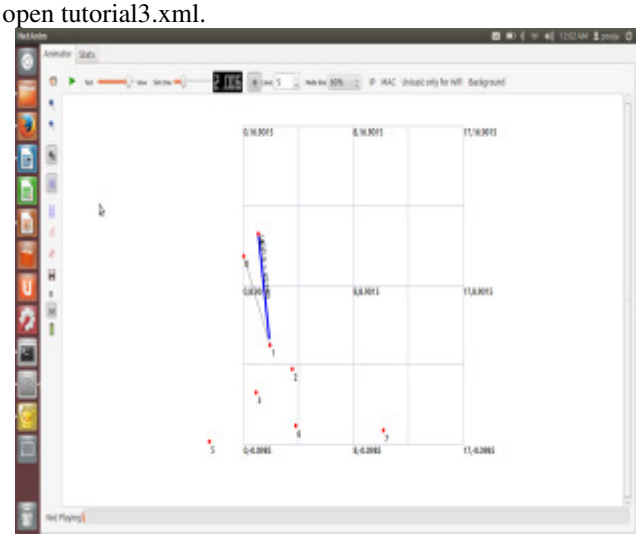


Figure 25: Running a Wireless Network using GUI

Figure 25 depicts Network Topology of thid.cc

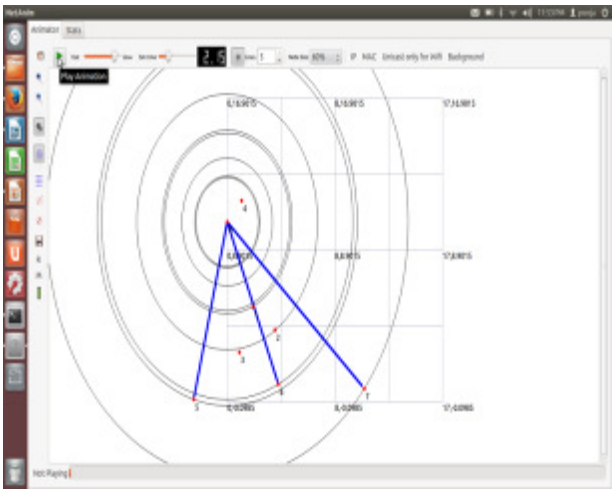


Figure 26: Running a Wireless Network using GUI

We have disabled packet Metadata as shown in figure 26.

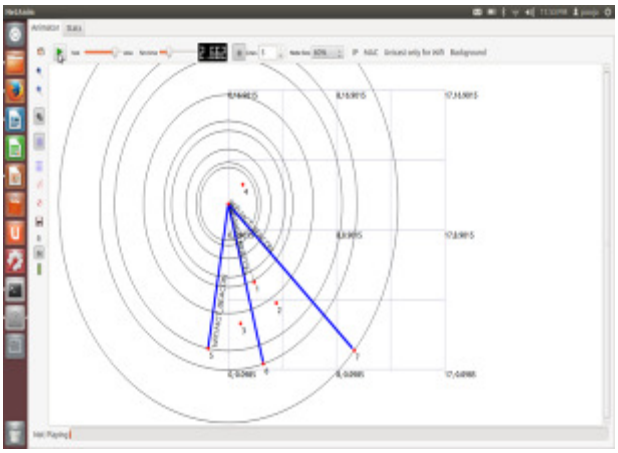


Figure 27: Running a Wireless Network using GUI

We have enabled packet Metadata as shown in figure 27.

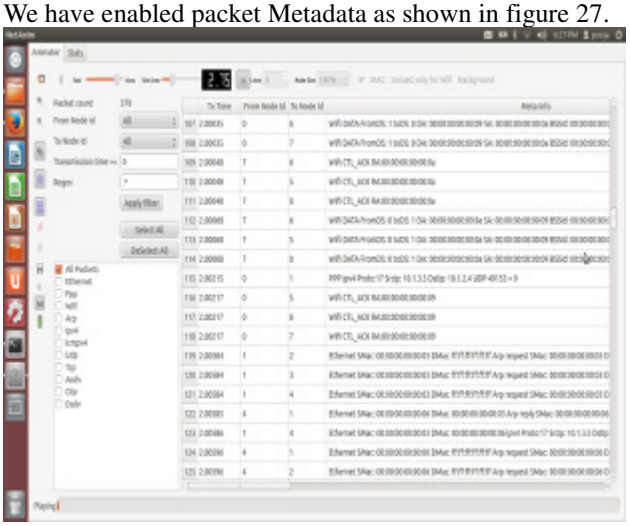


Figure 28: Meta information during simulation

In order to get packet and statistic data, we use regex filter as shown in figure 28. It provides better statistics and filter, along with providing some brief information about the packet during packet animation.

VI. CONCLUSION

This paper is to make a new user aware of network simulation tool (NS3). NS3 provides the solution for model validation, implementation verification and robust emulation that are main issues of NS2. In this paper NetAnim (GUI based network simulator) is also studied. This paper provides the solution for appeared error during installation procedure and also discusses how to run the network with the help of GUI.

VII. FUTURE SCOPE

This technology can help to deploy NGN network such as 5G SDN interface network because it is an open source and platform independent tool. With the help of NS3 we can propose different types of algorithm and can compare them.

REFERENCES

[1] (2011-2012) ns-3. [Online]. Available: https://www.nsnam.org/

[2] NetAnim. [Online]. Available: http://www.nsnam.org/wiki/NetAnim_1.0

[3] ns simulator. [Online]. Available: http://en.wikipedia.org/wiki/Ns_(simulator)

[4] ns-3 tutorial. [Online]. Available: http://www.nsnam.org/docs/release/3.14/tutorial/singlehtml/index.html

[5] Rakesh Kumar Jha, Pooja Kharga,"A Comparative Performance Analysis of Routing Protocols in MANET using NS3 Simulator", IJCNIS, vol.7, no.4, pp. 62-68, 2015.DOI: 10.5815/ijcnis.2015.04.08

[6] Rachna Choudhary, Shweta Sethi, Rita Keshari, Sakshi Goel, "A study of comparison of Network Simulator -3 and Network Simulator -2", (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 3 (1) , 2012.

[7] Atta ur Rehman Khan · Sardar M. Bilal · Mazliza Othman, " A Performance Comparison of Open Source Network Simulators for Wireless Networks", IEEE International Conference on Control System, Computing and Engineering; 11/2012

[8] Rakesh Kumar Jha, Pooja Kharga, " A Journey Starts from Basic Understanding of NS2 to NS3", LAP Lambert Academic Publishing,, ISBN-13: 9783659637582

[9] Rakesh k. Jha, Pooja Kharga, Idris Z. Bholebawa, Sangeet Satyarthi , Anuradha, Shashi Kumari,"OpenFlow Technology: A Journey of Simulation Tools", IJCNIS, vol.6, no.11, pp.49-55, 2014. DOI: 10.5815/ijcnis.2014.11.07

AUTHORS PROFILE

Rakesh K Jha: currently an assistant professor in school of electronics and communication department, SMVD University Katra (J&K). He is carrying out his research in WiMaX and Security issues in the laboratory ECED Lab, SMVDU. Involved research topics include WiMaX performance analysis, LBRRA, power optimization and security analysis. He has done B.tech in Electronics & Communication from Bhopal and M.tech from NIT Jalandhar, INDIA. Received his PhD degree from NIT Surat in 2013. Published more than 50 International Conference and Journal papers. His area of interest is Wireless communication, Communication System and computer network, and Security issues (Opti System). One concept releated to router of Wireless Communication has been accepted by ITU (International Telecommunication Union) in 2010. Received young scientist author award by ITU in Dec 2010 and APAN fellowship in 2011. Also received student travel grant from COMSNET 2012. Dr. Rakesh K Jha is a member of IEEE, GISFI and SIAM, International Association of Engineers (IAENG) and ACCS (Advance Computing and Communication Society).

Pooja Kharga: Received her B.Tech degree (Electronics & Communication) from SMVDU, Katra (J&K) in May, 2014. She has completed her final year project on OpenFlow Technology. Her research interest's area is designing a Network with the help of simulators.