Volume 5, Issue 4, 2015 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Simulation Based Performance Analysis of TCP Transport Layer Protocol based on Wired Network Using NS-2

Preeti , Mtech CSE, Renu Singla
Dept. of Computer Science & Engineering,
Shri Ram College of Engineering and Management,
Palwal, Affiliated to M.D.U Rohtak, Harayana, India

Abstract- TCP was mainly developed considering assumption of wired network, ignoring the properties of wireless transmission. Wireless transmission links are highly unreliable causing loss of packets all the time. The proper approach to dealing with lost packets is to send them again, and as quickly as possible. This paper aims at studying the effects of unidirectional and bidirectional networks on various TCP variants. Our research aim is to appreciate the key terminologies of UDP and TCP, to recognize the key difference between them. This work use to simulation for designing and studying wired network by using NS2 simulator. In the wired scenario, we consider first analyzed packet transmission into the UDP and TCP, second calculate bandwidth in UDP and TCP transport layer protocol and finally compare UDP and TCP performance based on their bandwidth. The reason of using wired network because it is easy to understand for all new users. The simulator used for implementation in Network Simulator-2 (NS2).. NS2 is an open source and freely distributed simulator for the new researcher.

Keywords— Transmission Control Protocol (TCP), Snoop Protocol, NS-2, TCP,UDP,Perl.

I. INTRODUCTION

Mobile Ad-hoc Networks (MANETs) are self -configuring networks consisting of mobile nodes that are communicating through wireless links. There is a cooperative engagement of a collection of mobile nodes without the required intervention of any centralized access point or existing infrastructure. The nodes move arbitrarily; therefore, the network may experience unpredictable topology changes. It means that a formed network can be deformed on the fly due to mobility of nodes. Hence, it is said that an Ad-hoc wireless network is self -organizing and adaptive. Due to infrastructure less and self-organizing nature of Ad-hoc networks, it has several applications in the area of commercial sector for emergency rescue operations and disaster relief efforts. MANETs also provides a solution in the field of military battlefield to detect movement of enemies as well as for information exchange among military headquarters and so on [1]. Also, MANET provides an enhancement to cellular based mobile network infrastructure. Nowadays, it is an inexpensive alternative for data exchange among cooperative mobile nodes

II. RELATED WORK

Network Simulator (version-2) commonly known as NS-2.It is widely used in network research. It is an open source software and combination of many protocols and inbuilt routing algorithms such as AODV, DSDV, and DSR. It also has routing queue mechanism such as RED, DropTail. With the help of NS-2 we can design both wired and wireless network and can compare many protocols and algorithms for better output before designing a network in the real world. In our project we design a network of 4 nodes with duplex link in between all the nodes. All the nodes are using DropTail queue mechanism. Node 0 follow the TCP agent and node 1 follow the UDP agent. Mobile Ad-hoc Network are highly dynamic in nature and no fixed infrastructure in these type of network.In this I did the comparative study of UDP and TCP based on NS-2 by calculating their individual bandwidth and analysis of their packet flow.

III. CLASSIFICATION OF NETWORKS

A. Wired Network :- A network is called wired networks which are connected through physical wires with each other. It is also called an Ethernet network which is the part LAN (Local Area Network) technology. Wired network is just a collection of two or more devices, these devices could be a combination of computers, printers, and any other devices linked together by Ethernet cables An Ethernet cable is required if anyone want to connect a computer to the network and computer must also have an Ethernet adapter also called NIC (Network Interface Card). NIC can be installed internal or external both. Internal means that installed in a computer, some computer include built-in NIC which eliminates the need of extra Ethernet Adapter.

B. Wireless Network:- A network is called wireless if it works without wire. In A wireless network, devices communicate with each other via some frequency wave rather than using the wires to communicate with each other, now a day's use of Wireless network is in trend [3] and it has become frequently adopted as an option for home or business networking. Even Individuals are using wireless network technology as another option of wired technology because this is more reliable than wired network and doesn't require any cable to connect with it and it allows a device to share its

data with any other device without any networking cable and network topology. Wireless network is of mainly two types: Ad-Hoc Wireless Network and Infrastructure Wireless Network.

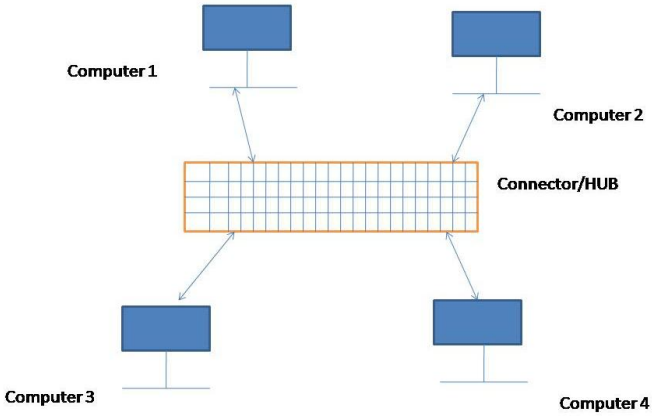

Figure 3.1: Wired Network

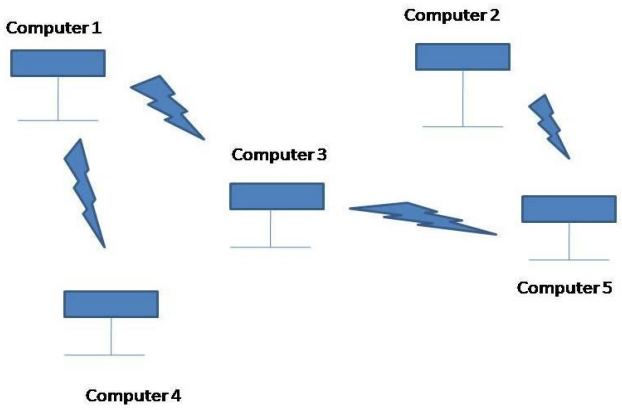

Figure 3.2: wireless Network

IV. SIMULATION STUDY OF WIRED SCENARIO WITH TCP

In this paper total seven nodes are taken from which two nodes, node no 0 and node no 1 act as source node and node 6 and node 7 act as destination node for node no 0 and node no 7 respectively. And node no 2, node no 3, node no 4 and node no 5 behave as a router nodes to forward packet source node to destination node. Source node represented by Green Color and destination nodes represented by red color and router nodes represented by black color. In TCP transport layer protocol first receiver send acknowledgement to the sender then sender send packets to the receiver, so that packets loss of in the path reduces to a very much extent as you can see in the following figure.

For writing the simulation script we use TCL (Tool Command Language) which is the part of OTCL (object Oriented Tool Command Language).In TCL script a NS Simulator object is created first which is used for many purposes such as creating nodes, for providing shapes and color to nodes, creating agents and links between them. This script consist of 4 nodes which all have different color and shapes such as node no 0 and node no 1 acts as a source node and represented by circle but node no 0 represent by blue color and node 1 represent by red color, node no 2 assumes as a router which forward the packets coming from the source node 0 and 1 to the node no 3 which assumes the destination node. Node no 2 represent by hexagon and black in color and node 3 represent by square and green in color.

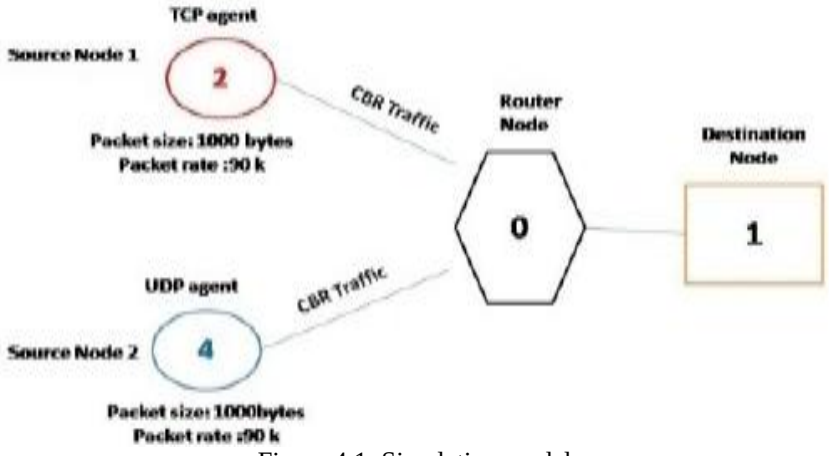

Figure 4.1: Simulation model

Performance Analysis of TCP Transport Layer Protocol based on NS2:

A duplex link is established between node no1 and node no 2, node no 0 and node no 2, node no 2 and node no 3 with bandwidth 1Mb and time delay is 20ms. All the nodes follow the Drop Tail routing queue mechanism of which maximum size is 20.A TCP agent is attached to node no 0 and connection is established to node no 3 which is a destination node by attaching "sink" agent to it. Same as A TCP agent is attach to node no 1 and "null" agent attach to node no 3 for connection establishment.CBR traffic is generated on both the TCP agent. For this two agents of CBR is made, one is cbr0 which is attach to node no 0 and 2nd is cbr1 which is attach to node no1. The size of CBR traffic for both TCP agent is set to 512 bytes and packet rate is set to 1mb.cbr0 starts at 0.1 and stop at 4.0, same as cbr1 starts at 0.5 and ends at 4.5.

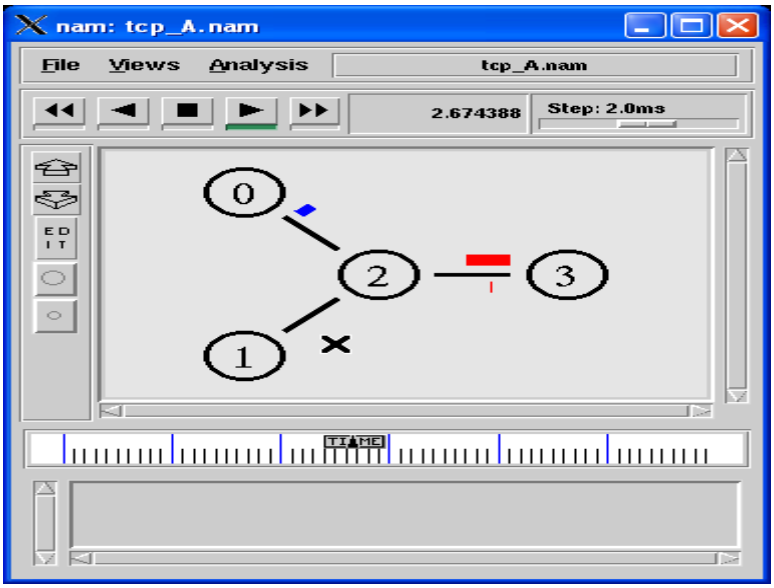

Fig 4.2 NAM Window

When we run any TCL simulation script then it generates two type of file: first is trace analysis file with tr extension and 2nd is NAM file with nam extension. Nam trace file contain all the information of TCL script like source node, destination node, type of protocol used, topology used, packet number, type of traffic which we are using in our script.

.In this paper we generated five Trace files to evaluate the performance to transport layer protocol TCP. First trace file calculate the total number of acknowledgments send by TCP, 2nd trace file calculate Total CWind of TCP, 3rd trace file calculate the Packet drop rate, 4th trace file calculate the sequence of TCp and Fifth trace file calculate the total throughput of TCP protocol.

(i) Simulation Parameters

In this simulation script we used many parameters which are shown in the following table:

Table 4.1: Parameter table of simulation script

NS version	Network Simulator 2 (NS-2)
Network Interface	Wired
No. of Nodes	4
Total No. of Source Node	2
Destination Node	1
Router Node	1
Simulation Area Size	800*600
Transport Layer Protocol	TCP
Traffic generator	CBR
Packet Size	512 bytes
Packet Rate	90k
Start Time	0.5 ms
Stop Time	5.0 ms
Interface Queue	Drop Tail

Simulation script generate the packets based on the packet size given in simulation parameters, Performance of any transport layer protocol depend on its traffic generator used and flow of packets according to the packets rate given to them.

(ii) Trace File of TCP Packet Acknowledgement

In transport layer protocol, before sending a packet to the receiver, it send an acknowlegment to the receiver and if sender gets an acknowgement as a reply then it transmit packets to receiver. With this acknowlegment there is less chances of lossing any packet. With the help of simulation script we generate a trace file for sending acknowlegemt between sender and receiver. In this trace file column one shows the event time at whice h acknowlegement is send and column two shows for which packet acknowledgement is send. So that TCP shows the reliable recovery of packet loss with the help of acknowlegemnts.

```
0.12    -1
0.13    -1
0.14000000000000001         -1
0.15000000000000002         0
0.16000000000000003         0
0.17000000000000004         0
0.18000000000000005         0
0.19000000000000006         0
0.20000000000000007         1
0.21000000000000008         2
0.22000000000000008         2
0.23000000000000009         2
0.2400000000000001          2
0.25000000000000011         3
0.26000000000000012         4
0.27000000000000013         5
0.28000000000000014         6
0.29000000000000015         6
0.30000000000000016         7
0.31000000000000016         8
0.32000000000000017         9
0.33000000000000018         10
0.34000000000000019         11
0.3500000000000002          13
0.36000000000000021         13
0.37000000000000022         14
0.38000000000000023         15
0.39000000000000024         15
0.40000000000000024         16
0.41000000000000025         17
0.42000000000000026         17
0.43000000000000027         18
0.44000000000000028         18
0.45000000000000029         19
0.4600000000000003          20
0.47000000000000031         20
0.48000000000000032         21
0.49000000000000032         22
0.50000000000000033         22
0.51000000000000034         23
0.52000000000000035         23
0.53000000000000036         24
```

Figure 4.3: Trace File of TCP Packet Acknowledgement

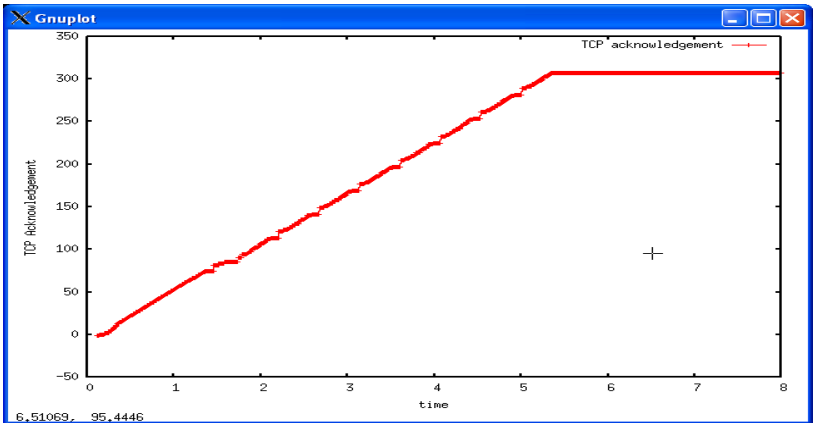

Figure 4.4 TCP Packet Acknowledgements

(iii) Trace file for CWind of TCP :

```
0.12    1
0.13    1
0.14000000000000001         1
0.15000000000000002         2
0.16000000000000003         2
0.17000000000000004         2
0.18000000000000005         2
0.19000000000000006         2
0.20000000000000007         3
0.21000000000000008         4
0.22000000000000008         4
0.23000000000000009         4
0.2400000000000001          4
0.25000000000000011         5
0.26000000000000012         6
0.27000000000000013         7
0.28000000000000014         8
0.29000000000000015         8
0.30000000000000016         9
0.31000000000000016         10
0.32000000000000017         11
0.33000000000000018         12
0.34000000000000019         13
0.3500000000000002          15
0.36000000000000021         15
0.37000000000000022         16
0.38000000000000023         17
0.39000000000000024         17
0.40000000000000024         18
0.41000000000000025         19
0.42000000000000026         19
0.43000000000000027         20
0.44000000000000028         20
0.45000000000000029         21
0.4600000000000003          22
0.47000000000000031         22
0.48000000000000032         23
0.49000000000000032         24
0.50000000000000033         24
0.51000000000000034         25
0.52000000000000035         25
0.53000000000000036         26
```

Figure 4.5 trace file of TCP CWind

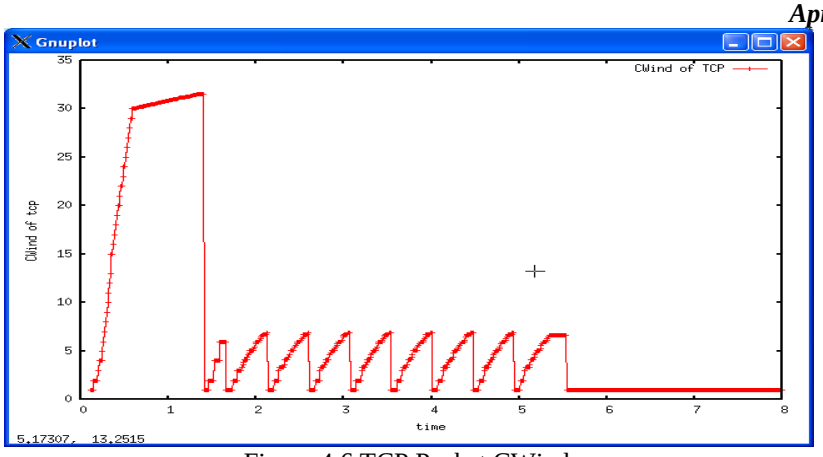

Figure 4.6 TCP Packet CWind

(iv) trace file for Packet Drop rate of TCP

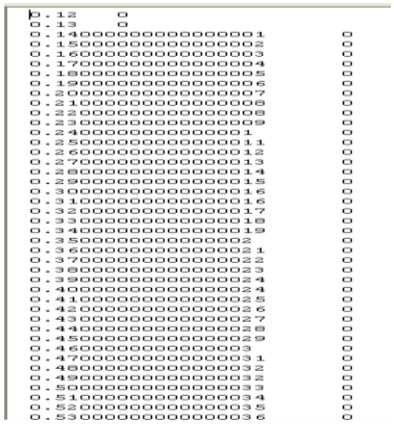

Figure 4.7: Trace File of Packet Drop Rate of TCP

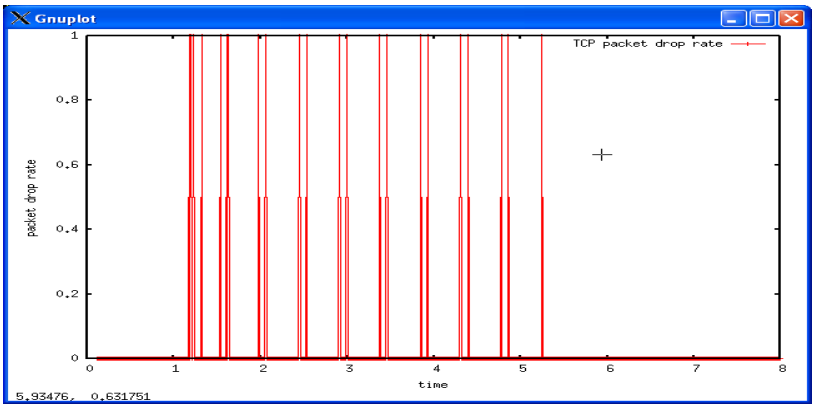

Figure 4.8: Packet Drop Rate of TCP

(v) Trace File of Packet Sequence Of TCP

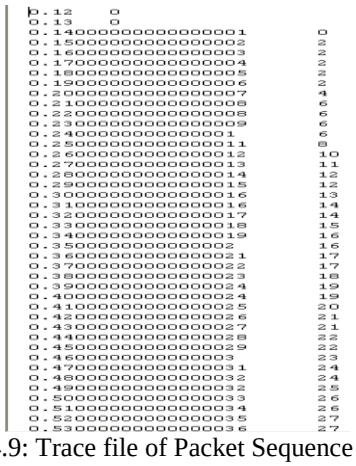

Figure 4.9: Trace file of Packet Sequence of TCP

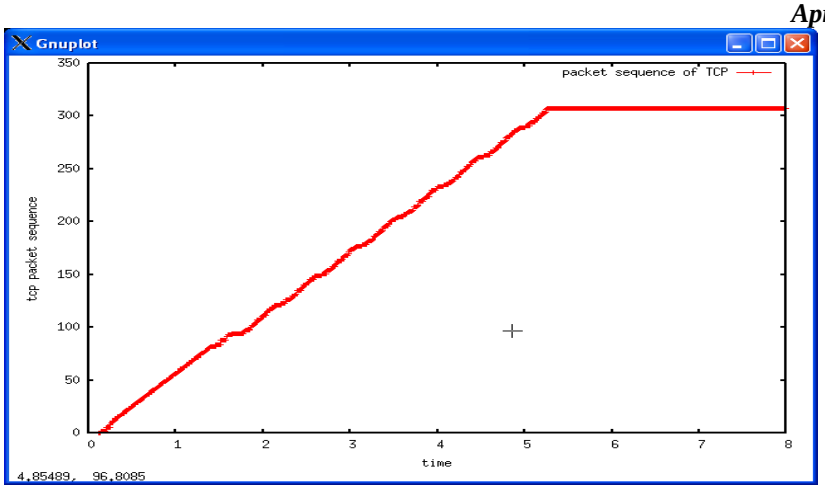

Figure 4.10: Packet Sequence of TCP

(v) Trace file of total throughput of TCP

```
0.12    32000.0
0.13    0.0
0.14000000000000001         0.0
0.15000000000000002         0.0
0.16000000000000003         832000.0
0.17000000000000004         832000.0
0.18000000000000005         0.0
0.19000000000000006         0.0
0.20000000000000007         0.0
0.21000000000000008         832000.0
0.22000000000000008         832000.0
0.23000000000000009         832000.0
0.2400000000000001          832000.0
0.25000000000000011         0.0
0.26000000000000012         0.0
0.27000000000000013         1664000.0
0.28000000000000014         832000.0
0.29000000000000015         832000.0
0.30000000000000016         832000.0
0.31000000000000016         832000.0
0.32000000000000017         832000.0
0.33000000000000018         832000.0
0.34000000000000019         832000.0
0.3500000000000002          0.0
0.36000000000000021         832000.0
0.37000000000000022         0.0
0.38000000000000023         832000.0
0.39000000000000024         832000.0
0.40000000000000024         0.0
0.41000000000000025         832000.0
0.42000000000000026         832000.0
0.43000000000000027         0.0
0.44000000000000028         832000.0
0.45000000000000029         0.0
0.4600000000000003          832000.0
0.47000000000000031         832000.0
0.48000000000000032         0.0
0.49000000000000032         832000.0
0.50000000000000033         832000.0
0.51000000000000034         0.0
0.52000000000000035         832000.0
0.53000000000000036         0.0
```

Figure 4.11: Trace file of total throughput of TCP

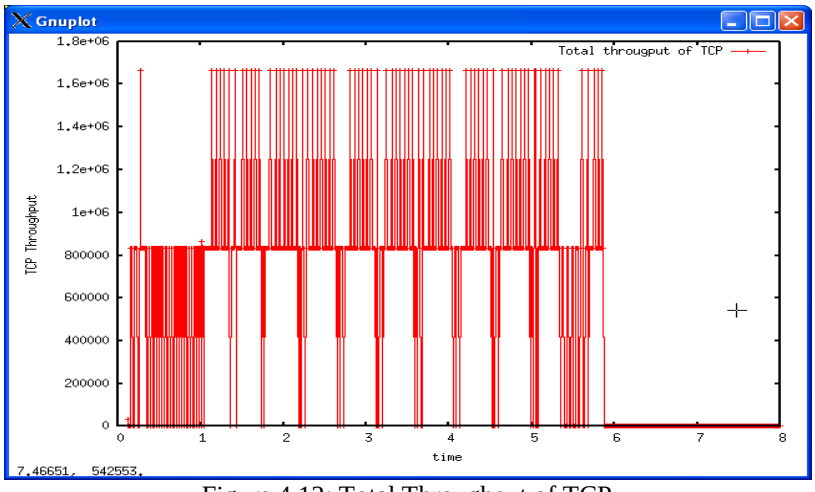

Figure 4.12: Total Throughput of TCP

V. CONCLUSION

As we said from the starting NS-2 used to design and implement both wired and wireless network before building it in the real world which will help us to obtain better output and less drop rate of packets. In our project we implement a network with four nodes, node 0, node 1, node 2, and node 4. Here node 0, node 1 acts as source node, node 2 acts as router and node 3 acts as destination node and try to count the drop rate of packets. A visualization tool (Network Animator) is used in the NS-2 which commonly called as NAM. This visualization tool provides a GUI interface for running NS scripts.

ACKNOWLEDGMENT

It is a great pleasure for me to express my sincere gratitude to my supervisor, **Dr. Renu Singla,** Professor /Assistant Professor, Department of Computer Science & engineering of Shri Ram College of Engineering & Management palwal, for his valuable guidance, timely advice and constant encouragement during the project work.

REFERENCES

[1] Ajay singh, Dr. Pankaj Dashore, "A Comparative Study of UDP and TCP by Using NS2", International Journal of Computer Science and Information Security (IJCSIS) Volume (1): Issue (1), June 2013

[2] Ritika Sharma, Kamlesh Gupta, Comparison based Performance Analysis of UDP/CBR and TCP/FTP Traffic under AODV Routing Protocol in MANET, International Journal of Computer Applications (0975 – 8887) Volume 56– No.15, October 2012

[3] Sirwan A. Mohammed, Prof. Dr. Sattar B. Sadkhan, "DESIGN OF WIRELESS NETWORK BASED ON NS2", Journal of Global Research in Computer Science, Volume 3, No. 12, December 2012

[4] Velibor Markovski, Fei Xue, and Ljiljana Trajkovi´c, "Simulation and analysis of packet loss in video transfers using User Datagram Protocol", paper.tex; 2/05/2001

[5] Er. Pragati1, Dr. Rajender Nath, "Performance Evaluation of AODV, LEACH & TORA Protocols through Simulation", Volume 2, Issue 7, July 2012

[6] Dr. Neeraj Bhargava, Dr. Ritu Bhargava, Anchal Kumawat, Bharat Kumar, "Performance of TCP-Throughput on NS2 by Using Different Simulation Parameters", International Journal of Advanced Computer Research Volume-2 Number-4 Issue-6 December-2012

[7] Assist Prof. Dr. Saad Talib Hasson, 2Ahmed Jawad Kadhim, 3Zainab Saad Talib, "Enhancing the NS-2 Traffic Generator for the MANET", IOSR Journal of Computer Engineering (IOSRJCE) Volume 4, Issue 2 (Sep.-Oct. 2012), PP 12-16

[8] Tarik ANOUARI, Abdelkrim HAQIQ, "Performance Analysis of VoIP Traffic in WiMAX using various Service Classes", International Journal of Computer Applications (0975 – 8887) Volume 52– No.20, August 2012

[9] Ajay Singh, Dr. Pankaj Dashore, "Packet Flow Analysis in Wireless Scenario", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 3, Issue 6, June 2013.

[10] Johanna Antila, "TCP Performance Simulations Using Ns2", 51189d TLT

[11] Sirwan A. Mohammed, Prof. Dr. Sattar B.Sadkhan, "Design of wireless network based on NS-2", *Journal of Global Research in Computer Science,* Volume 3, No. 12, December 2012

[12] Yun Zhou and Yuguang Fang, "Network security and attack Defense", Institue of electrical and Electronics Engineers, 2009.

[13] FIPS PUB 46-2, "Data Encryption Standard (DES)", Dec.1993

[14] IETF RFC 2040, "The rc5,rc5-cbc, rc-5-cbc-pad, abd rc-5-cts algorithms", Oct.1996

[15] FIPS PUB 197, "Advanced Encryption Standards (AES)", Nov.2001

[16] W.Diffie and M.E. Hellman, "New directions in Cryptography", *IEEE Transaction on Information Theory*, Vol. IT-22, no.6, Nov 1976, pp-644-654

[17] R.L. Rivest, A. Shamir, and L.Adleman, "A method for obtaining digital signatures and public-key cryptosystems", *Communications of the ACM*, vol. 21, no. 2, Feb. 1978, pp-120-126

[18] Hamza Rahmani, Nabil Sahli, Farouk Kamoun, "DDoS flooding attack detection scheme based on F-divergence", Computer Communications 35 (2012) 1380–1391, Elsevier

[19] Panayiotis kotzanikolaou, Christos douligeris "computer network security: basic background and current issues", institute of electrical and electronics engineers, Inc. 2007.

[20] Y. Xiang, Y. Lin, W.L. Lei ,S.J. Huang, "Detecting DDOS Attack on network Similarity", IEEE Proc.-Commun., Vol. 151, No. 3, June 2004.

[21] Prinima, Dr.R.K.Tuteja, "Comparative Performance Analysis Of Routing Protocols Using Ns2 Simulator". "International Journal Of Computer Engineering & Technology (Ijcet)".

[22] Y. Xie, S. Tang, X. Huang, C. Tang, X. Liu, "Detecting latent attack behavior from aggregated Web traffic", Computer Communications 36 (2013) 895–907, Elsevier

[23] Silvia Farraposo, Laurent Gallon, Philippe Owezarski "Network security and DOS attack", Documtery history.

[24] S.M. bellovin "Security problem in TCP/IP protocol suite", Computer Communication Review, Vol. 19, No. 2, pp. 32-48, April 1989.

[25] Gaston Ormazabal, Sarvesh Nagpal, Eilon Yardeni, and Henning Schulzrinn "Secure SIP: A Scalable Prevention Mechanism for DoS Attacks on SIP Based VoIP Systems", Springer-Verlag Berlin Heidelberg 2008

