International Journal of Engineering Research and General ScienceVolume 2, Issue 5, August – September 2014
ISSN 2091-2730

Impact of Network Size & Link Bandwidth in Wired TCP & UDP Network Topologies

Mrs. Meenakshi.
Assistant Professor, Computer Science & Engineering Department,
Nitte Meenakshi Institute of Technology, Bangalore 560064
kmeenarao@gmail.com

Abstract—The transmission of information in a network relies on the performance of the traffic scenario (application traffic agent and data traffic) used in a network. The traffic scenario determines the reliability and capability of information transmission, which necessitates its performance analysis.

The objective of this paper is to calculate and compare the performance of TCP/FTP and UDP/CBR traffic in wired networks. Study has been done using NS-2 and AWK scripts. Exhaustive simulations have been done to analyze results, which are evaluated for performance metrics, such as link throughput, and packet delivery ratio. The effect of variations in link bandwidth, number of nodes on the network performance is analyzed over a wide range of their values. Results are shown in terms of graphs and tables.

Keywords—protocol stack, TCP, UDP, NS-2, agent, performance metrics, throughput, packet delivery ratio, bandwidth.

I. INTRODUCTION

Introduction section gives brief knowledge about TCP/IP protocol stack, feature, application, advantages and disadvantages of TCP and UDP protocols respectively.

1.1TCP/IP Protocol Stack

It is based on the two primary protocols, namely, TCP and IP, is used in the current Internet [1]. These protocols have proven very powerful, and as a result have experienced widespread use and implementation in the existing computer networks. Figure1 is TCP/IP Protocol Stack.

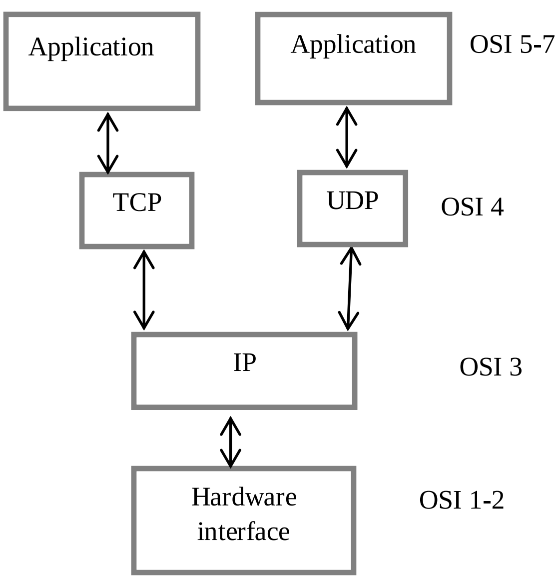
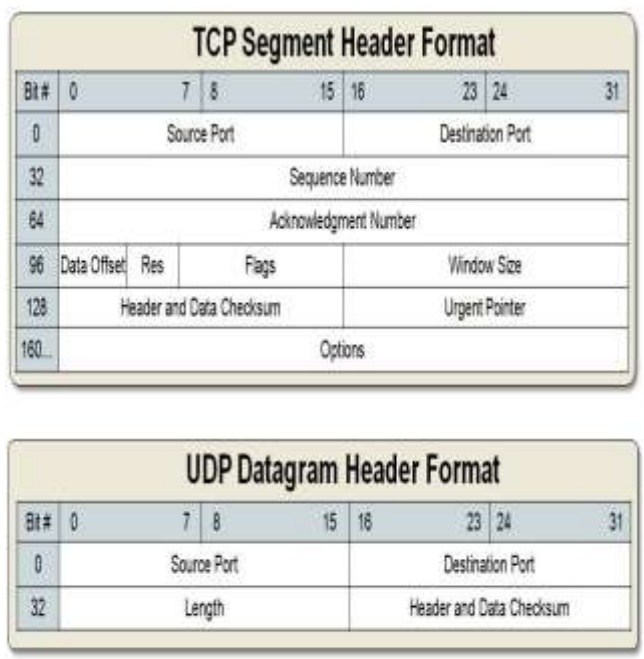


Figure1. TCP/IP Protocol Stack



TCP Segment Header Format

Bit #	0 – 7	8 – 15	16 – 23	24 – 31
0	Source Port		Destination Port	
32	Sequence Number			
64	Acknowledgment Number			
96	Data Offset / Res	Flags	Window Size	
128	Header and Data Checksum		Urgent Pointer	
160...	Options			

UDP Datagram Header Format

Bit #	0 – 7	8 – 15	16 – 23	24 – 31
0	Source Port		Destination Port	
32	Length		Header and Data Checksum	

Figure2. TCP and UDP headers

1.2. Transmission Control Protocol (TCP)

TCP is a connection-oriented protocol [2]. Function as a message makes its way across the internet from one computer to another. This is connection based. TCP is suited for applications that require high reliability, and transmission time is relatively less critical. TCP is usedby other protocols like HTTP, HTTPs, FTP, SMTP, and Telnet. TCP rearranges data packets in the order specified. The speed for TCP is slower than UDP.

TCP is reliable in the sense there is absolute guarantee that the data transferred remains intact and arrives in the same order in which it was sent. TCP header size is 20 bytes as shown in Figure2 namely TCP and UDP headers. Data is read as a byte stream, no distinguishing indications are transmitted to signal message (segment) boundaries. TCP is heavy-weight. TCP requires three packets to set up a socket connection, before any user data can be sent. TCP handles reliability and congestion control. TCP does error checking. SYN, SYN-ACK and ACK are three handshake related messages.

1.3. User Datagram Protocol or Universal Datagram Protocol (UDP)

UDP is a connectionless protocol. UDP is also a protocol used in message transport or transfer. This is not connection based which means that one program can send a load of packets to another and that would be the end of the relationship.

-

UDP is suitable for applications that need fast, efficient transmission, such as games. UDP's stateless nature is also useful for servers that answer small queries from huge numbers of clients. DNS, DHCP, TFTP, SNMP, RIP and VOIP protocols use UDP. UDP has no inherent order as all packets are independent of each other. If ordering is required, it has to be managed by the application layer. UDP is faster because there is no error-checking for packets. There is no guarantee that the messages or packets sent would reach at all.

UDP Header size is 8 bytes as in Figure 1.2: TCP and UDP headers [3]. Source port, Destination port and Check Sum are common in both TCP and UDP. Packets are sent individually and are checked for integrity only if they arrive. Packets have definite boundaries which are honored upon receipt, meaning a read operation at the receiver socket will yield an entire message as it was originally sent.

UDP is lightweight. There is no ordering of messages and no tracking connections. It is a small transport layer designed on top of IP. UDP does not have an option for flow control. UDP does error checking, but no recovery options. No Acknowledgment and also no handshake since it is connectionless protocol.

1.4. TCP/IP Application Protocols

FTP (File Transfer Protocol), HTTP (Hyper Text Transfer Protocol), NNTP (Network News Transfer Protocol), Remote Login (**rlogin**), Telnet, X Window System depends on TCP to guarantee the correct and orderly delivery of data across the network.

SNMP sends traffic through UDP because of its relative simplicity and low overhead.When NFS (Network File System) runs over UDP the RPC implementation must provide its own guarantees of correctness. When NFS runs over TCP, the RPC layer can depend on TCP to provide this kind of correctness.

DNS uses both UDP and TCP. It used UDP to carry simple queries and responses but depends on TCP to guarantee the correct and orderly delivery of large amounts of bulk data (e.g. transfers of entire zone configurations) across the network.

II. MATERIAL AND METHODOLOGY

The network performance can be measured with many metrics. Following sections give brief about few of those metrics and simulation setup of the experiment done in this paper.

2.1 Performance metrics

The performance of any system needs to be evaluated on certain criteria, these criteria then decide the basis of performance of any system. Such parameters are known as performance metrics [4], [5], [6]. The different types of performance metrics used to evaluate performance of any networks are described below:

2.1.1 Throughput

The throughput is the measure of how fast we can actually send data through the network. It is the measurement of number of packets that are transmitted through the network in a unit of time. It is desirable to have a network with high throughput.

$$\textbf{Throughput} = \sum P_R / (\sum t_{sp} - \sum t_{st})$$

P_R – Received Packet Size,
t_{st} – Start Time,
t_{sp} – Stop Time.
Unit – Kbps (Kilobits per second)

2.1.2 Link Throughput

In computer technology, throughput is the amount of work that a computer can do in a given time period. In communication networks, such as Ethernet or packet radio, network throughput is the average of successful message delivery over a communication channel.
Transmission Time = File Size / Bandwidth (sec).
Throughput = File Size / Transmission Time (bps).
Link throughput say from node S to D is given by the following formula:

$$\surd = \sum N_b / t$$

N_b- Number of bits transmitted from node S to D
t **-** Observation duration.

2.1.3 Packet Delivery Ratio (PDR)

It is the ratio of number of packets received at the destination to the number of packets generated at the source. A network should work to attain high PDR in order to have a better performance. PDR shows the amount of reliability offered by the network. The greater value of packet delivery ratio means the better performance of the protocol.

$$PDR = (\sum N_R / \sum N_G) * 100$$

N_R – Number of Received Packets,
N_G – Number of Generated Packets,
Unit – Percentage ratio (%).

2.1.4 Average End – to – End Delay (AED)

This is the average time delay consumed by data packets to propagate from source to destination. This delay includes the total time of transmission i.e. propagation time, queuing time, route establishment time etc. A network with minimum AED offers better speed of communication.

$$AED = \sum t_{PR} - \sum t_{PS}$$

t_{PR} – Packet Receive Time,
t_{PS}– Packet Send Time,
Unit – milliseconds (ms).

2.2 Simulation

Simulation of wired as well as wireless network functions and protocols (e.g., routing algorithms, TCP, UDP) can be done using NS2 [7], [8], [9]. Network Simulator, Version-2, widely known as NS2, is simply an event-driven simulation tool that has proved useful in studying the dynamic nature of communication networks. Figure 3.1 and 3.2 show simple network topologies used for experiments carried out in this paper.

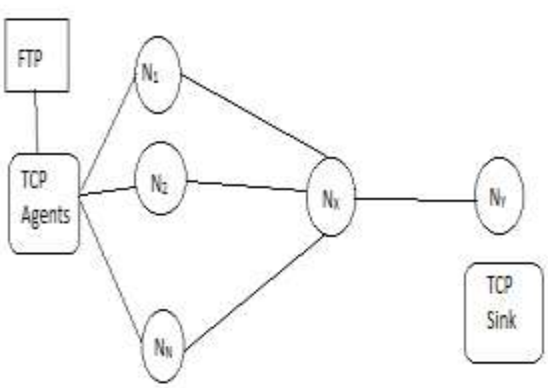


Figure3. A sample network topology: TCP

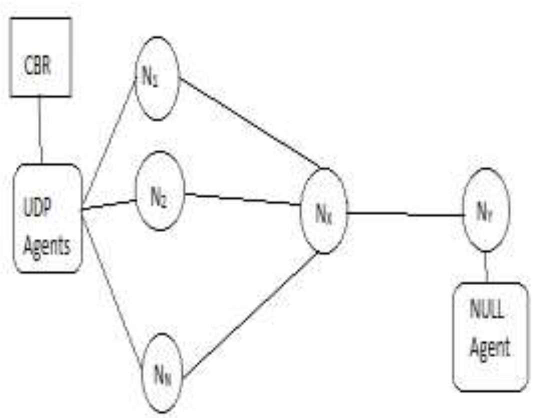


Figure4. A sample network topology: UDP

In figure 3.1 and 3.2 $N_1, N_2 ... N_N$ are nodes. FTP and CBR are applications of TCP and UDP respectively. $N_x$and $N_y$are nodes of bottleneck link whereas $N_y$is final destination of packets generated from all sources. Corresponding sender-agents have to be attached to all sending nodes. TCP/Sink is agent to be attached to TCP-destination and Null agent is for UDP-receiver.

III. RESULTS AND CHARTS

A systematic study and analysis of all the aspects of wired networks is carried out by executing ns2 and AWK scripts. Comparison is made for link throughput and packet delivery ratio. Followings are tables and graphs obtained from executing AWK scripts [10] for which ns2 trace files were input. In first scenario, number of nodes was varying and corresponding changes in throughput had been observed. In second scenario again number of nodes was varying and corresponding PDR had been noted down. In last scenario bandwidth is the varying factor and different throughput had been tabled.

Nodes	tcp/tp(kbps)	udp/tp(kbps)
5	4640.95	2188.05
25	4613.26	4882.98
50	4733.14	4882.98
100	4745.9	4882.98
200	4750.14	4882.98
300	4744.03	4882.98

Table1. Node Vs Bottleneck-link throughput

Nodes	tcp/PDR	udp/PDR
5	99.81	100
10	99.64	100
20	99.21	77.92
30	98.66	68.61
40	98.06	63.96
50	97.46	61.17
100	95.62	55.58

Table2. Nodes Vs packet delivery ratio

Bandwidth	tcp/tp(kbps)	udp/tp(kbps)
0.5	487.809	488.314
1	954.95	976.596
1.5	1440.2	1464.88
2	1926.27	1953.16
2.5	2411.64	2187.55
3	2897.37	2187.57
3.5	3396.37	2187.58
4	3880.92	2187.59

Table3. Bandwidth Vs Bottleneck-link throughput

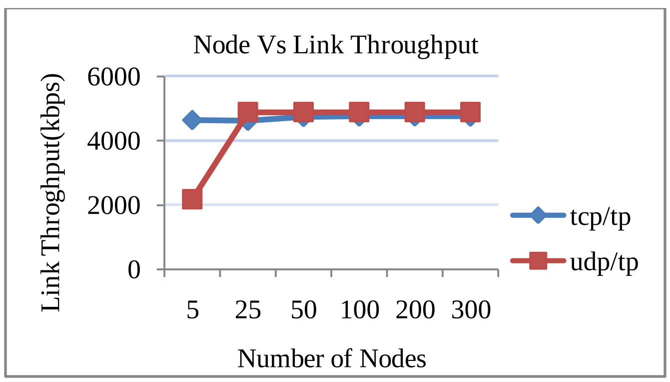


Figure5. Node Vs Bottleneck link-throughput

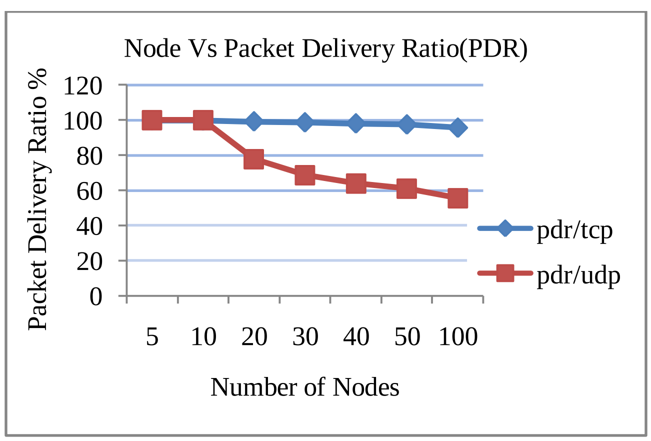


Figure6. Node Vs Packet Delivery Ratio

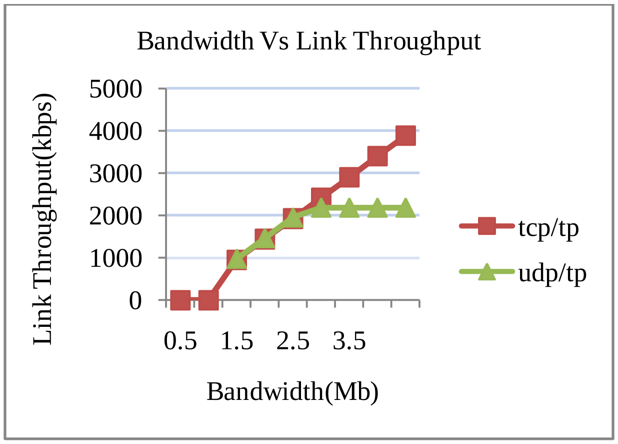


Figure7. Bandwidth Vs Link Throughput

IV.CONCLUSION

Bottleneck link throughput and packet delivery ratio have been calculated using ns2 and AWK scripts. Link bandwidth and nodes are varying factors respectively. Packet delivery ratio is much better in TCP than of UDP. In case of link band width, TCP shows better link throughput than that of UDP.

Depending on application requirement one has to decide the suitable protocols. This study can be extended for other traffic generators namely exponential On/Off, Pareto On/Off and Traffic Trace. More over experiment can be carried out for wireless networks as a future work.

ACKNOWLEDGEMENT

I would like to express my gratitude and appreciation toInternational Journal of Engineering Research and General Science team who gave me the opportunity to publish this report. Special thanks to Nitte Meenakshi Institute of Technology Bangalore, management, Computer Science HOD and all staffs whose stimulating suggestions and mainly encouragement helped me to write this report.I would also like to acknowledge with much appreciation my family members especially my husband Mr.Ajith and my sons Aadithya and Abhirama for their cooperation and support.

REFERENCES:

[1] Miss. SoniSamprati, "Next Generation of Internet Protocol for TCP/IP Protocol Suite", International Journal of Scientific and Research Publications, Volume 2, Issue 6, June 2012.
[2] Santosh Kumar and SonamRai, "Survey on Transport Layer Protocols TCP & UDP", International Journal of Computer Applications 46(7):20-25, May 2012.
[3] Fahim A. Ahmed Ghanem, Vilas M. Thakare, "Optimization Of IPv4 Packet's Headers", IJCSI International Journal of Computer Science Issues, Vol. 10, Issue 1, No 2, January 2013.
[4] YogeshGolhar ,R.K.Krishna and Mahendra A. Gaikwad, "Implementation & Throughput Analysis of Perfect Difference Network (PDN) in Wired Environment", IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 1, No 1, January 2012.
[5] Performance Measurements and Metrics, http://webstaff.itn.liu.se/~davgu/tnk087/Fo_8.pdf.
[6] Computer Networks, Text Book byAndrew S. Tanenbaum.
[7] TeerawatIssariyakul and Ekram Hossain, Text Book "Introduction to Network Simulator NS2", Second Edition.
[8] The ns manual, Kevin Fall and KannanVaradhan.

[9] Ns simulator, Wikipedia.org.
[10] AWK scripts, http: //wing.nitk.ac.in/ resources/ Awk.pdf