Eth.J.Sci & Technol. 5(2): 136-141, 2008 ISSN 1816-3378

Performance Analysis of Throughput at Bahir Dar University LAN

Prathap M[1] and E Chandra[2]

[1]*Department of Computer Science and Engineering, Bahir Dar University, Bahir Dar Ethiopia: prathaparthi@yahoo.co.in*

[2]*Asst Prof and Head, D.J .Academy for Managerial Excellence, India: crcspeech@gmail.com*

Abstract

It is obvious that the network performance cannot exceed the hardware capacity. Frequent up gradation of hardware is not possible in order to achieve high performance. Hence changes are needed in the technical modification, which helps economically. Computer scientists and network users have discovered that standard TCP does not perform well in high bandwidth delay environments. As a model, the Local Area Network of Bahir Dar University Engineering Faculty was tested and reported. . In this paper, we explore the challenges of achieving high throughput over real environments. Moreover, various performance factors like Delay, Throughput, Network capacity, Available and achievable performance were studied and discussed. Comments were made on decreasing delay factors and better solutions were recommended for buffer management.

Keywords: TCP; delay; throughput; available and achievable performance

1. Introduction

The network performance characteristics like delay, throughput were measured and studied. Due to congestion, the delay increases The Local Area Network of Bahir Dar University was chosen for this case study and data readings were noted by varying the number of users. The users vary in using the network from time to time. It was concluded that when the users are more the traffic is more and the delay increases resulting in slower transmission of bits. Though TCP/IP had gained much importance, the standard TCP does not perform well in high bandwidth delay environments (Kelly T, 2003). Trans-port Control Protocol (TCP) is a reliable, end-to-end, transport, protocol that is widely used to support applications like telnet, ftp, and http (Stevens W. R, 1994). Kelly's Scalable TCP on real networks with a set of systematic tests using different network were already tested (Li Yee-Ting et al, 2004). This paper analyzes these factors and reports the necessary improvements needed for controlling congestion thereby improving performance.

2. Delay

Delay is the time taken for a bit to travel form one end to another which could be measured in seconds or fraction of seconds. Delay may differ slightly depending on the specific pair of computers. The first important property of networks that can be measured quantitatively is delay.

2.1 Throughput

The second fundamental property of networks that can be measured quantitatively is throughput. Throughput is a measure of the rate at which data can be sent though the Network, and is usually specified in bits per second (bps) or measure of the network capacity. When the network is idle, the number of users is kept as zero and when the network is functioning with maximum numbers of users, it is taken as one. If the numbers of users are half, then the network usage is also half. The Delay and throughput factors are analyzed and the effective delay is given by $D = D_o / (1-U)$ where D_o is the network is idle and U is the Utilization.

When U = 0,

$$D = D_o / (1-U), D= D_o$$

When U = 1/2,

$D = D_o / (1-U)$, $D= 2D$
When $U = 1$,
$D = D_o / (1-U)$, $D= \infty$

Table 1. Effect of delay

Users in Percentage	Status of delay
0	Minimum
50	Twice
100	Maximum

If the network is idle then the effective delay D will be D_0. When the network is utilized to half of its capacity then the effective delay will be twice and when the all the users are utilizing the effective delay becomes infinity. As the number of users are close to the maximum then delay increases rapidly to maximum as shown in Table 1. This shows clearly that as the delay increases the traffic increases, resulting the bit in slower transmission.
As the users increases the delay increases which shows that they are directly proportional. The volume V is given by $V = T \times D$

Hence, D=V/T
Hence in order to make the delay D to minimum, either the volume of data has to be decreased or increase the throughput. We conclude that if the congestion is less then the throughput increases.

Table 2. Data showing the flow of bits

Time	Received	Sent
06:06	546	2101
06:07	866	2757
06:08	1038	3136

Table 3. Throughput analysis

T	Minimum		Maximum	
	Received	Sent	Received	Sent
Bits/sec	6.31	2.86	10.93	5.33

Congestion Control algorithm can be used so that the data traveling in the less congested medium will transmit faster. To measure the delay, the client machine of the faculty is connected to the server through two switches and the readings were noted as shown in Table 2. Based on the delay of bits transmission throughput was calculated per second.

The configuration of the network server used is
Pentium IV 1.76 GHz

128 MB of RAM

20 GB of Hard disk

The Client 1 configurations used are as follows:

Pentium IV 2.0 GHz

256 MB of RAM

Intel PRO/100 VE Network Connection

The server having an Internet Connection with proxy is allowed to enter the common sites of www Viz., www.google.com and the timing is noted on the client machine and the data transfer rate is given in table 2.

Based on the readings from Table 2 and the time difference, the Minimum and Maximum Received and sent bits per second were calculated and the results are shown in Table 3. The percentage of change in received and sent bytes are 54 and 57 respectively.

In theory, the delay and throughput of a network are independent but they may be related.

As shown in table 3, the bits per second varies while they are sent and received. If a bit slows down in a network, it leads to cause congestion. Hence data entering a congested network will undergo longer delay than data entering an idle network. For this, as shown in Fig. 1 the time interval between each bit is monitored. If for any reasons, the bit slows down we propose an algorithm based on congestion control. In order to control the congestion in the medium, the data has to be accelerated with free movement and less delay. This algorithm could be used for traffic management and algorithm in Fig. 2 is used for buffer management. In order to manage the buffer efficiently and reduce the time for reading the bits, this buffering concept could be implemented. Fig. 3 shows the situation after the buffers are created. These buffers are named as b1 and b2. After the second block has been read, the pointer P is set to the first record in buffer b1. The pointed bit positions are sent and the pointer is incremented till the end of the buffer. After sending the last bit then the pointer is moved to the first bit of buffer b2. This process is repeated until the last bit is sent. During sending process of the buffer b2 the buffer b1 is loaded. Hence, while the buffer b2 is sending, b1 is receiving bits both at the same time. Also after b2 is loaded b1 becomes empty and they interchange their functions. When the queue is empty, all the buffers and pointers are deleted. This helps in efficient time and memory management

3. Measuring Available Bandwidth

Available bandwidth is network capacity minus utilization (C-U). After performing TCP tuning techniques, a user or developer might think that they have fully utilized available bandwidth (Tierney, 2001).

```
get time interval
monitor bit transmission
if previous bit slows down then
   place previous bit in buffer
time slice of each bit is set by the receiving entity
if time slice expires and the bit not reached
request retransmission
check buffer and select the bits for retransmission
send only missing bits which ensures faster transfer of data.
```

Fig. 1. Algorithm used for shorter delay

```
name previous buffer as b1, next incoming buffer as b2
Repeat
   load b1 and b2 with octet of bit
   set pointer to b1.
     Repeat
          send pointed bit of b1
          move pointer to next bit
     until last bit of b1.
 If last bit sent is equal to 8 set pointer to bit 9 of b2
     Repeat
          send pointed bit of b2
          move pointer to next bit
     until last bit of b2.
 If last bit sent is equal to 2 times of 8 set pointer to bit 1 of b1
Until the buffer queue is empty.
```

Fig. 2. Buffer Management

However it is obvious that it is impossible that they are only utilizing the bandwidth that their application is capable of achieving, and not all the available bandwidth. Moreover functional, scalable network, could be upgraded in future according to the necessity of the bandwidth required results in Quality network, (Prathap and Chakravarthi, 2005). When the available bandwidth is unknown, TCP can be used to achieve the best performance.

TCP often does not perform well in high bandwidth long delay paths, due to the fact that it recovers very slowly from packet loss. Hence if we know the Maximum Burst Size (MBS — the maximum number of bytes that can travel through a network path without causing packet loss), both of those mechanisms potentially violate the fairly sharing policy of network resources. The knowledge of the MBS can aid in optimal and fair use of the network. We recommend for the use of tools like netest which is designed to provide information about each element on a path between two end hosts (Guojun and Tierney, 2003).

This information includes the available bandwidth of the path and/or the maximum achievable throughput. Netest can help to identify the source of poor network performance such as a problematic router, sending host, receiving host, lack of TCP buffers, etc. Netest also provides advice on what one can do to improve application throughput.

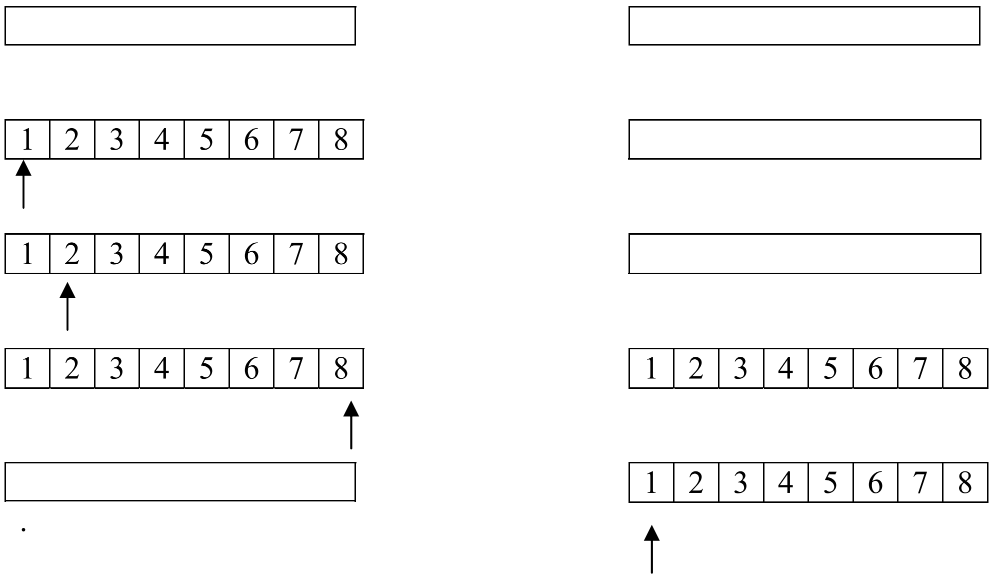

Fig. 3. Buffer b1 and b2 with pointer

Table 4. Data showing the flow of bits

Time	Received
06:06	946
06:07	1340
06:08	1561

4. Conclusion

The network performance factors delay and throughput were discussed and measured. The data collected showed that the congestion is more and hence the delay. This resulted in affecting the performance of the network. Using the above algorithms, the network performance could be tuned which results in shorter delay and faster transmission of bits. Also the network should include tools like Netest which helps in monitoring the bad performing hardware and provides way for better performance. The results obtained are shown in Tables 4 and 5 considering only the receiving bits.

The time and the number of users are kept as that of Table 2 and the data transmission readings were noted. Received bytes increased by 23% as compared to Table 2.

Table 5 clearly shows that the Throughput increased as compared to Table 3. Hence we conclude that the algorithm above can be used in order to control the congestion and accelerate the transmission.

5. Acknowledgements

We thank the Engineering Faculty of Bahir Dar University for providing the server resource and hence for data collection. Our sincere thanks to Ato Zenebe, the network administrator for his timely support

Table 5. Throughput analysis

T	Minimum		Maximum	
	Received	Percentage improved	Received	Percentage Improved
Bits/sec	3.68	28	6.56	23

6. References

Kelly T (2003). Scalable TCP: Improving Performance in High Speed Wide Area Networks. *First International Workshop on Protocols for Fast Long-Distance Networks. Geneva, Switzerland*

Stevens WR (1994). TCP/IP Illustrated. Volume 1, Addison Wesley. The Protocols. Addison-Wesley.

Li Yee-Ting, Dallison S, Hughes-Jones R, Clarke P (2004). A Systematic Analysis of High Throughput TCP in Real Network Environments. *Paper Presented* at *the Second International Workshop on Protocols for Fast Long-Distance Networks, Argonne National Laboratory, Argonne, Illinois USA-DataTAG-Final-Dissemination-D5.5-1.0.doc*

Tierney B (2001). TCP tuning guide for distributed application on wide area networks. *USENIX & SAGE Login*. 26(1): 33

Prathap M and Chakravarthi R (2005). A Systematic Methodology for Effective Implementation of Local Area Network (LAN*). Proceedings of the 23rd Annual May Seminar of Bahir Dar University Bahir Dar, Ethiopia.*

Guojun J and Tierney B (2003). Netest: A Tool to Measure the Maximum Burst Size, Available Bandwidth and Achievable Throughput. *Proceedings of the 2003 International Conference on Information Technology August 10-13, 2003, Newark, New Jersey, LBNL-48350*