University at Albany

Department of Computer Science

ICSI 516- COMPUTER COMMUNICATION NETWORKS

REPORT FOR PROJECT-1

The cache: socket programming and reliable data transfer

Submitted by: CHARAN KASALA

Wire Shark Analysis:

(1) Methodology:

The methodology that I followed for Wireshark analysis is manual and is as follows.

First, I connected to the VM and deployed my server files. Then I started the server on IP 169.226.22.10 and port 21027. Then I started the cache and client locally on my local machine.

Then I issued get <FILE_NAME> command to client and made sure that these files are not present in the cache. Then I stared to capturing the traffic on Wireshark and stopped the capture once the files are downloaded at the client.

I captured pcap traces for 4 Files using TCP protocol and 4 files using SNW protocol.

(i) TCP Protocol:

For pcap traces generated using the TCP protocol, I calculated the delay as time difference between the time the last packet was received at the cache and timestamp at which get request was received at the server (After three-way handshake).

Delay

File1 – Trace sl no 4 to 25 - 0.012502 - 0.004368 = 0.008134

File 2- Trace sl no 22 to 54 - 23.398978 - 23.368666 = 0.030312

File 3- Trace sl no 11 to 61 - 2.195217- 2.163312 = 0.031905

File 4- Trace sl no 15 to 81 - 11.746734- 11.712115 = 0.034619

File Sizes (in bits)

File 1- 119040   File 2- 238080   File 3- 357152   File 4- 476192

Throughput = $\frac{\text{File size in bits}}{\text{delay}}$

(ii) SNW protocol:

For pcap traces generated using the SNW protocol, I calculated the delay as the time difference between the time at which ACK was received for the last packet and the time stamp at which first packet was sent.

Delay

File1 – Trace sl no 36 to 66 - 8.139200- 8.199279= 0.060079

File 2- Trace sl no 8 to 68 - 0.006339- 0.112298 = 0.105959

File 3- Trace sl no 9 to 100 – 1.018035 - 0.845804 = 0.172231

File 4- Trace sl no 5 to 81 - 0.22617- 0.004544= 0.221626

File Sizes (in bits)

File 1- 119040 File 2- 238080 File 3- 357152 File 4- 476192

Throughput = $\frac{\text{File size in bits}}{\text{delay}}$

(2) Results

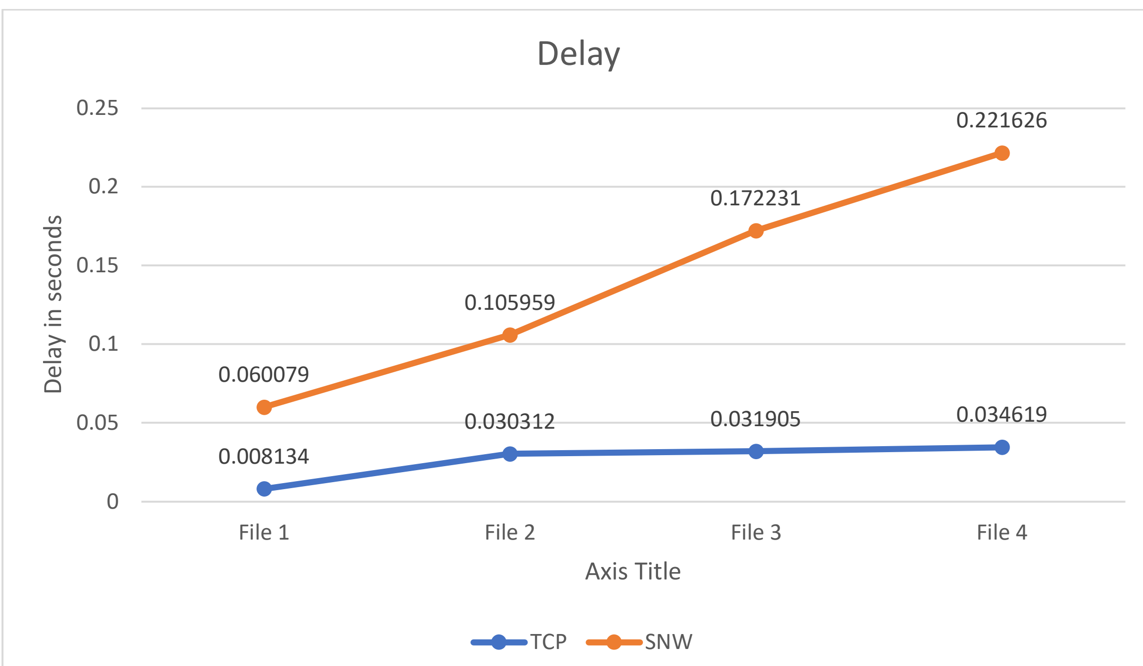
Using the above value generated using the Wireshark and using the above formulae, I have arrived at the following results.

Delay	File 1 (16KB)	File 2 (32KB)	File 3 (48KB)	File 4 (62KB)
TCP (sec)	0.008134	0.030312	0.031905	0.034619
SNW (sec)	0.060079	0.105959	0.172231	0.221626

Throughput	File 1 (16KB)	File 2 (32KB)	File 3 (48KB)	File 4 (62KB)
TCP (sec)	14634866	7854315.123	11194233	13755221.12
SNW (sec)	1981391	2246906.822	2073680	2148628.771

(3) Trends

From the above results the trends noticed can be plotted in graphs as follows:

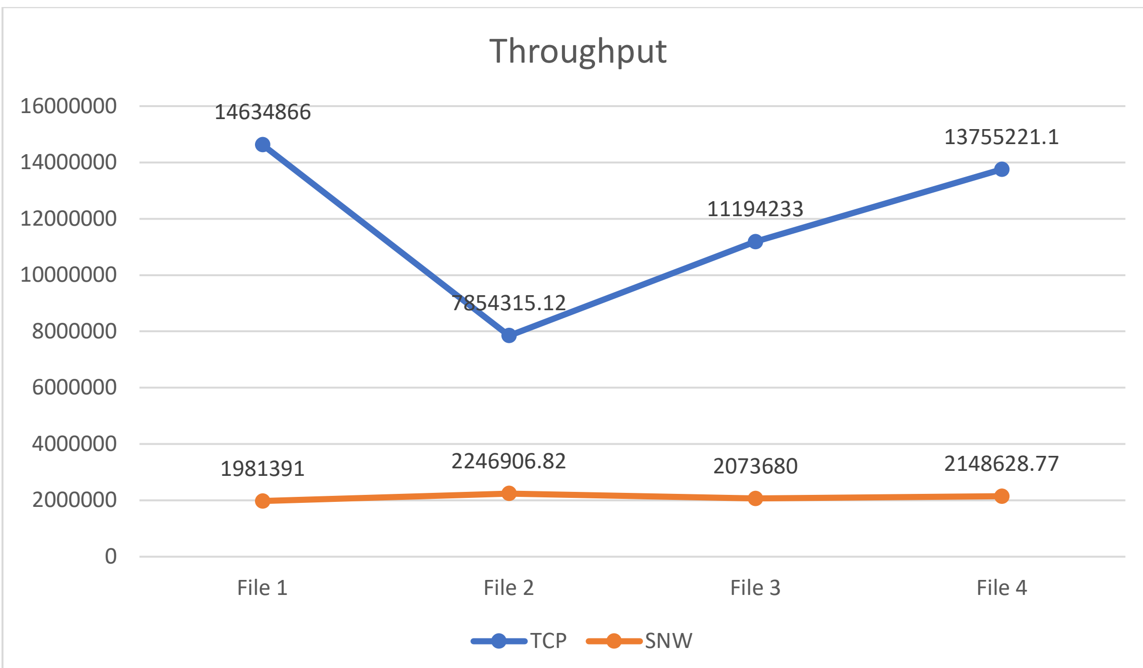


<u>Trend 1:</u> Delays are less in TCP than in UDP.

<u>Justification:</u>

TCP is reliable connection-oriented protocol that has flow control, error handling and retransmission of lost packets.

UPD is connectionless best effort delivery service and doesn't have any flow control or error handling and hence the delays are higher.



<u>Trend 2:</u> Throughput is higher in TCP than UDP.

<u>Justification:</u>
TCP offers reliable connection-oriented data transfer and error control. UDP on the other hand is connectionless and doesn't provide any error control. Therefore, the throughput is significantly higher in TCP when compared to UDP.

Increase in file size increases delay.

With the increase in the file size, the delay is increased in both TCP and SNW UDP protocols. However, the increase is consistent in case of TCP but in case of SNW UDP protocol, the delay is not proportional and consistent.

<u>Justification</u>:
TCP has flow control mechanisms to ensure data is transmitted at the rate the receiver can handle it. When the file size is large the data will be transmitted based on receiver's capability which can lead to increased delay but in a gradual manner.

In case of UDP, the delay is inconsistent with the increase in file size because it doesn't have flow control and it tries to send data as fast as possible leading to inconsistent trend.

Increase in file size increase throughput.

With the increase in file size, the throughput increases in TCP protocol. Even in SNW, the throughput increases with increase in the file size but the increase here is not consistent.

<u>Justification:</u>
There is rapid increase in TCP throughput when the file size increases because as the file size increases, the window size increase enabling the TCP to give increase throughput.

UDP doesn't have this capability and it just blasts off packets into the network. When all packets are transmitted correctly, the throughput increases and when there is packet loss the throughput decreases.