Remote Client-Server System with TCP/UDP

Implemented by

Raghavendra Vamshi Teja Chathurajupalli

Email Id: raghavendravamshi.chathu@gmail.com

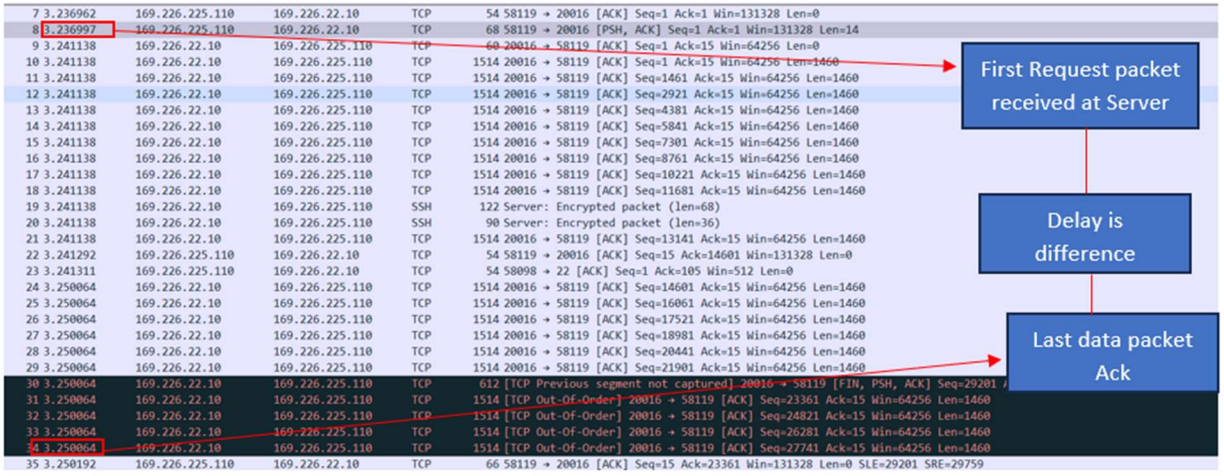
INDEX:

A. Methodology

TCP Delay Calculation:

For Calculating delay for TCP transfer of packets, the difference of timestamps between the request packet and the last data packet, that is the time between the request delivered to client and the last packet of data arrived at the cache.

Delay(to transfer N packets) = N^{th} Data Packet's Timestamp - 1^{st} Requested Data packet Timestamp.

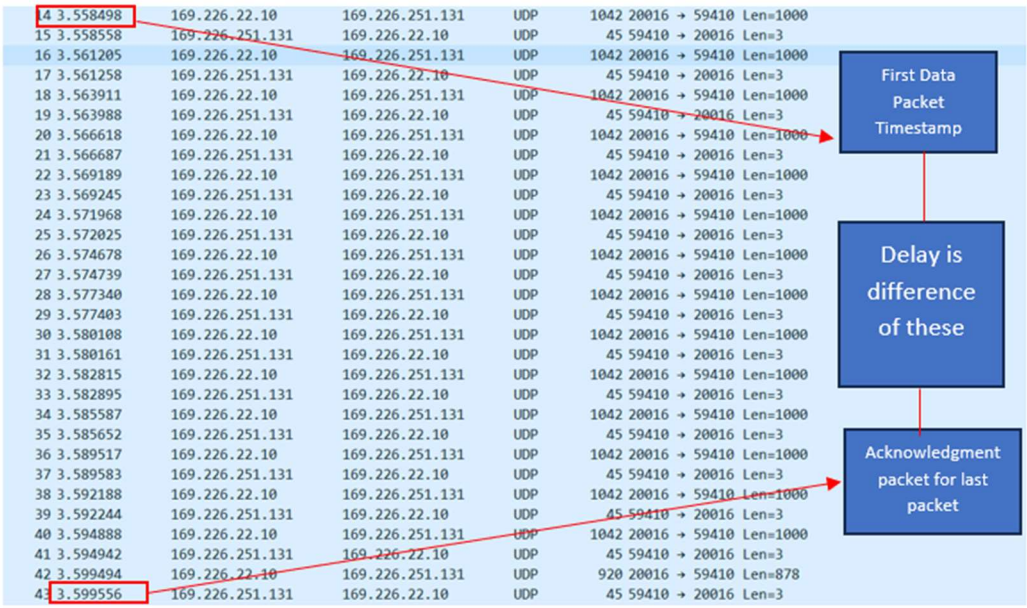


The reason for considering from get request is because, a stream of 10 packets are sent at the same instant and the acknowledgment is received for every 10 packets, and for first file the total packets is within 10 packets and these are sent at the same instant of time, thereby considering data packets alone will yield a delay of 0 which may be incorrect.

SNW Delay Calculation:

For calculating Delay for SNW data transfer, the difference of timestamps between 1st data packet and last acknowledgment packets are taken into consideration.

Delay(to transfer N packets) = N^{th} Data Packet's Acknowledgement Timestamp-1^{st} Data packet Timestamp.



Throughput: (File Size in bytes x 8 bits) / Delay

B. Results:

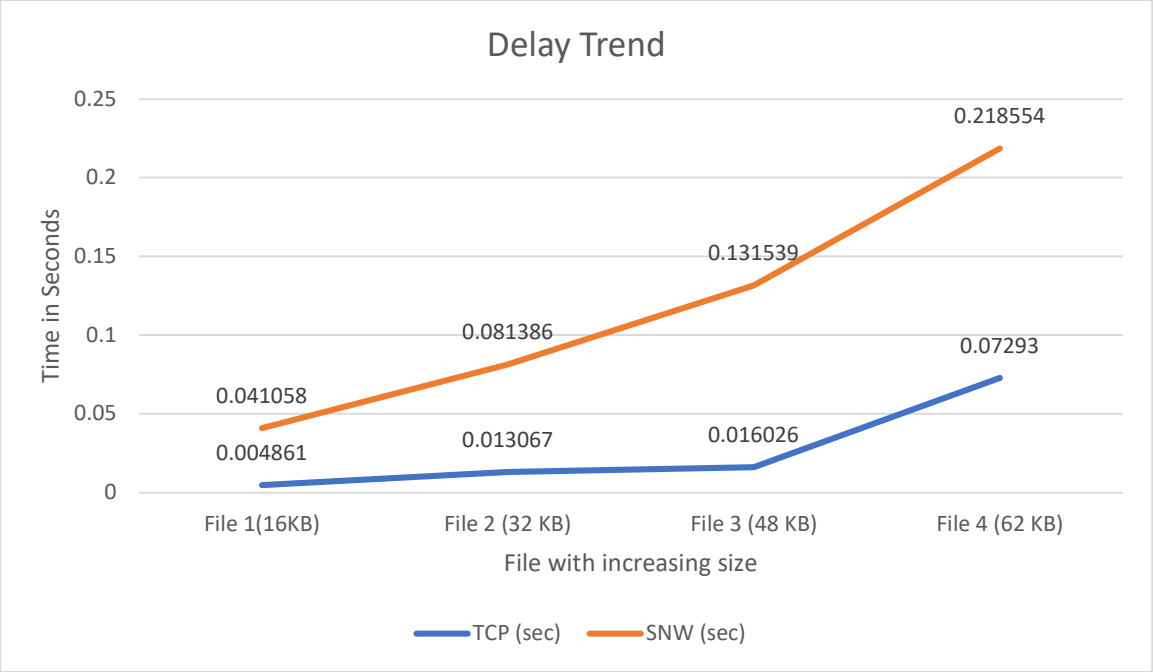
Delay	File 1 (16KB)	File 2 (32KB)	File 3 (48KB)	File 4 (62KB)
TCP (sec)	**0.004861**	**0.013067**	**0.016026**	**0.07293**
SNW (sec)	**0.041058**	**0.081386**	**0.131539**	**0.218554**

Throughput	File 1 (16KB)	File 2 (32KB)	File 3 (48KB)	File 4 (62KB)
TCP (bps)	2,44,85,496.81	1,81,86,719.31	2,22,84,787.22	65,29,219.76
SNW (bps)	28,98,923.47	29,25,122.26	27,13,057.89	21,78,756.72

Note: File sizes are based on lengths of data packets summed together**.**

C. Trends Analysis and Justification:

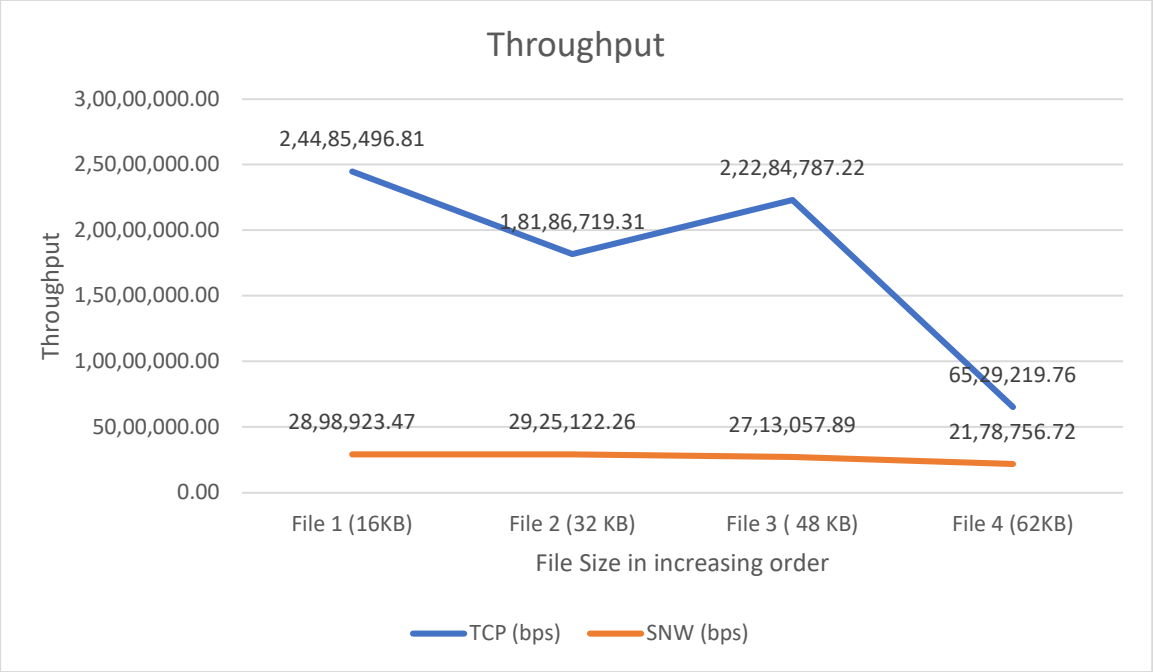
For Delay:



As per the results, the delays for TCP are quite low, which is expected for a well-functioning TCP connection. TCP sends out multiple packets of data at the same time and waits for single ack per multiple packets with reliable transfer and congestion handling mechanisms to adapt to different network conditions. Also, the factor to consider here is about the low congestion in network as proximity between Access point and VM is very less as there are located at same place.

The Delays for SNW (stop and Wait Protocol) are significantly higher in relative to TCP. This is expected as the sender waits for acknowledgment for each packet before sending the next one which results in increase of overall delay especially as file size increases.

For Throughput:



The throughput for TCP is higher for smaller files but as the file size increases the throughput decreases. This could be due to the influence of slow start and congestion avoidance mechanisms. The significant drop in throughput for 62 KB file maybe due to detection of congestion and the congestion control algorithm significantly reduced sending rate (multiplicative decrease).

Throughput for SNW is quite lower than TCP and remains consistent across file size. This is because, SNW is designed to send limited packet at a time and must wait for acknowledgment. this limits the bandwidth utilization efficiency and lowers the throughput.

References:

1. J. Postel, "Transmission Control Protocol,
2. Chui, Jain, “Analysis of the Increase and Decrease, Algorithms for Congestion Avoidance in Computer Networks”

3. Brief details of Code Execution

Brief Overview of Code working is given in client, cache and server codes as part of documentation along with comments.

Code Execution Steps:

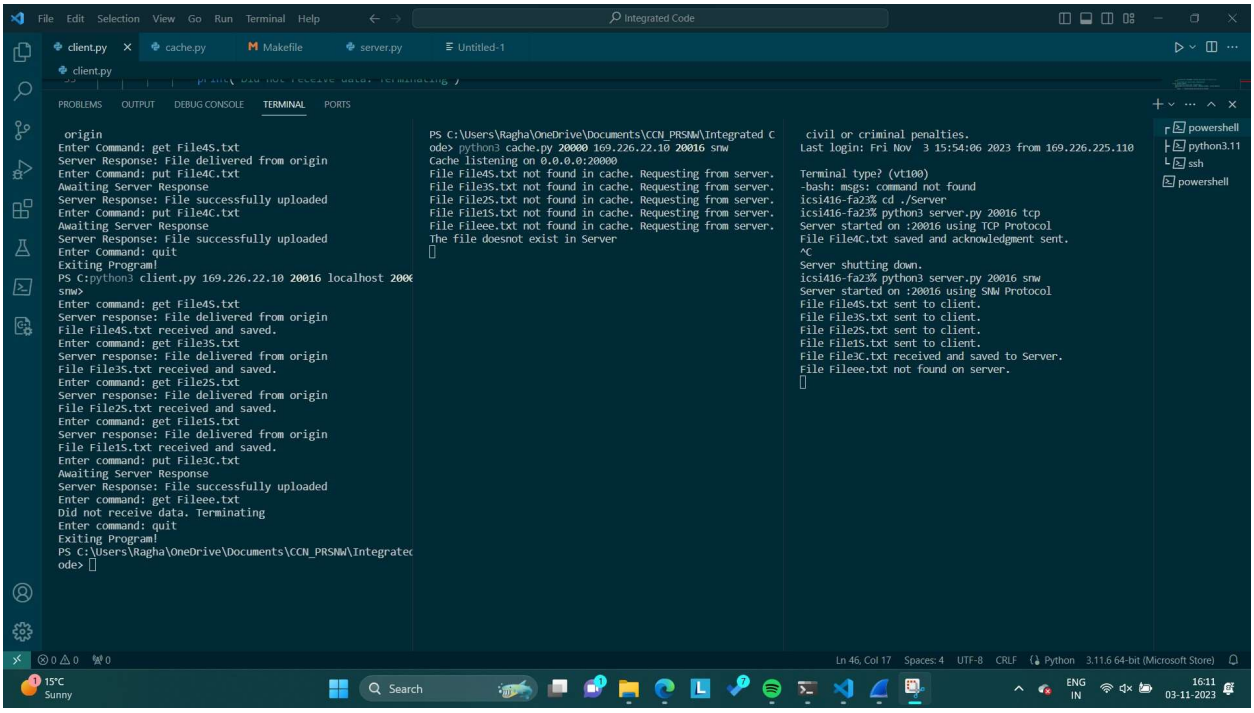
For Server:
python3 server.py <server_port> <protocol>

For Cache:
python3 cache.py <cache_port> <server_host> <server_port> <protocol>

For Client:
pyhton3 client.py <server_host> <server_port> <cache_port> <cache_host> <protocol>

Result Screenshots for Reference:

SNW Screenshots:



TCP Screenshots: