EE450

TCP Lab Report

Yuhang Xiao

Abstract

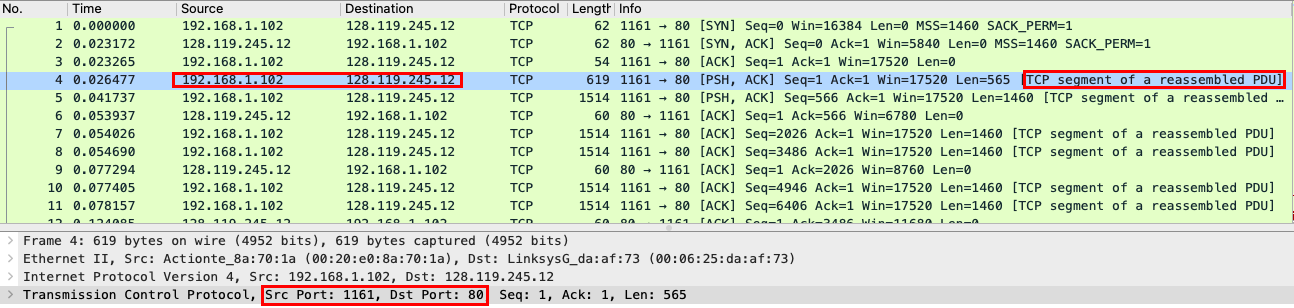
This report investigates the behavior of the celebrated TCP protocol in detail. We analyze a trace of the TCP segments sent and received in transferring a 150KB file from client computer to gaia.cs.umass.edu. Through the first 12 questions, the sequence and acknowledgement numbers for providing reliable data transfer in TCP protocol is discussed. TCP connection setup and the performance (throughput and round-trip time) of the TCP connection between client computer and the remote server are also investigated. The last two question discuss the congestion control algorithm – slow start and congestion avoidance, and TCP’s receiver-advertised flow control mechanism.

1.

Source:

IP address: 192.168.1.102

TCP Port number: 1161



2.

According to the above figure,

Destination: gaia.cs.umass.edu

IP address: 128.119.245.12

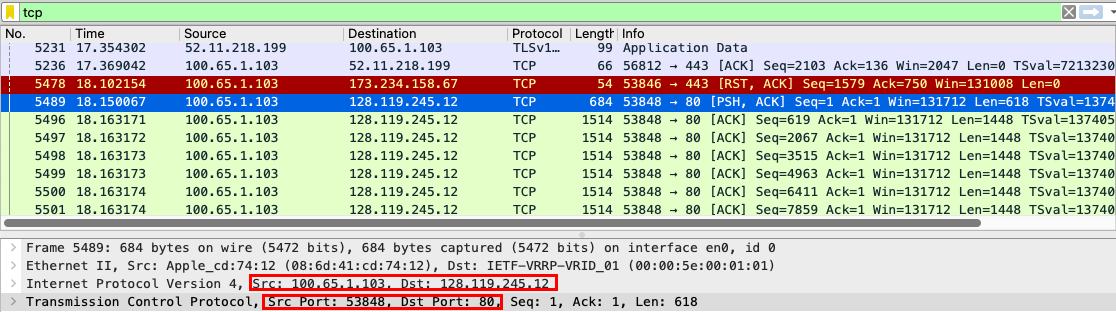
TCP Port number: 80

3.

Source:

IP address: 100.65.1.103

TCP Port number: 53848

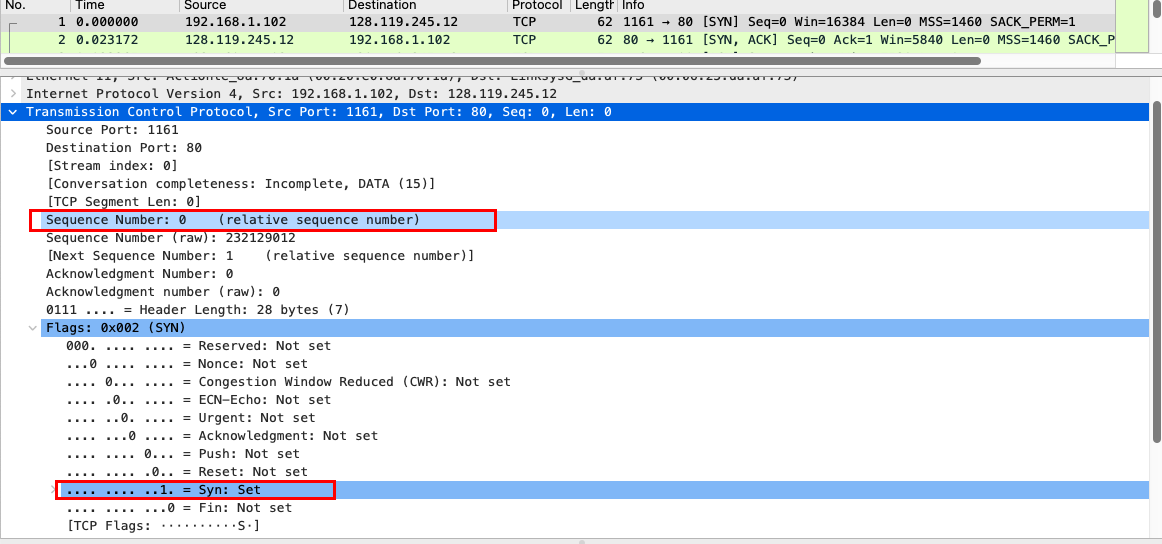


4.

According to the figure below,

TCP SYN segment uses sequence number 0 to initiate the TCP connection.

The SYN flag is set to 1 to indicate it’s a SYN segment.



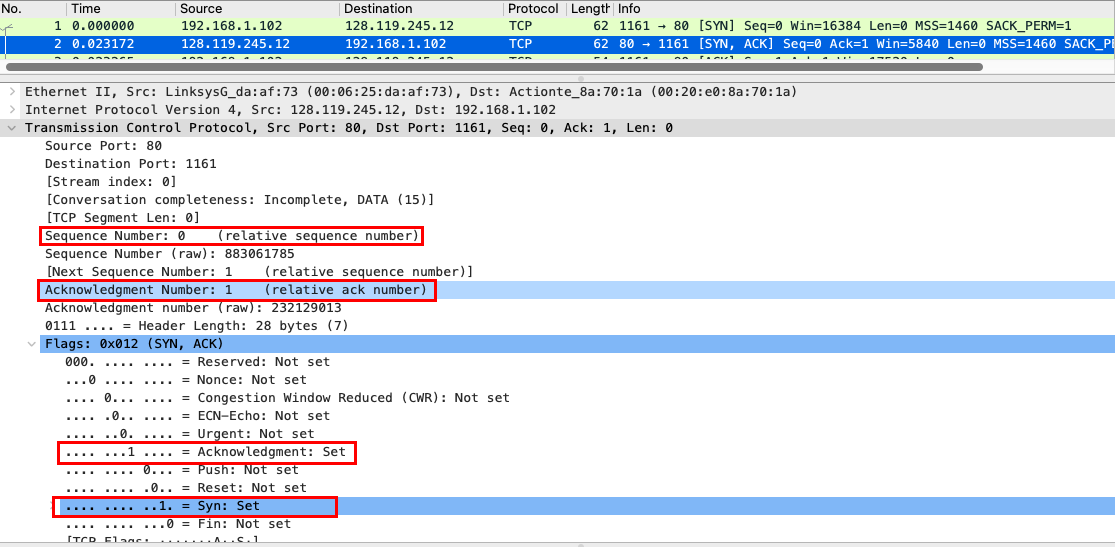
5.

The sequence number of SYNACK segment is 0.

The value of Acknowledgement field is 1.

The value is determined by adding 1 to the sequence number 0 in the initial SYN segment.

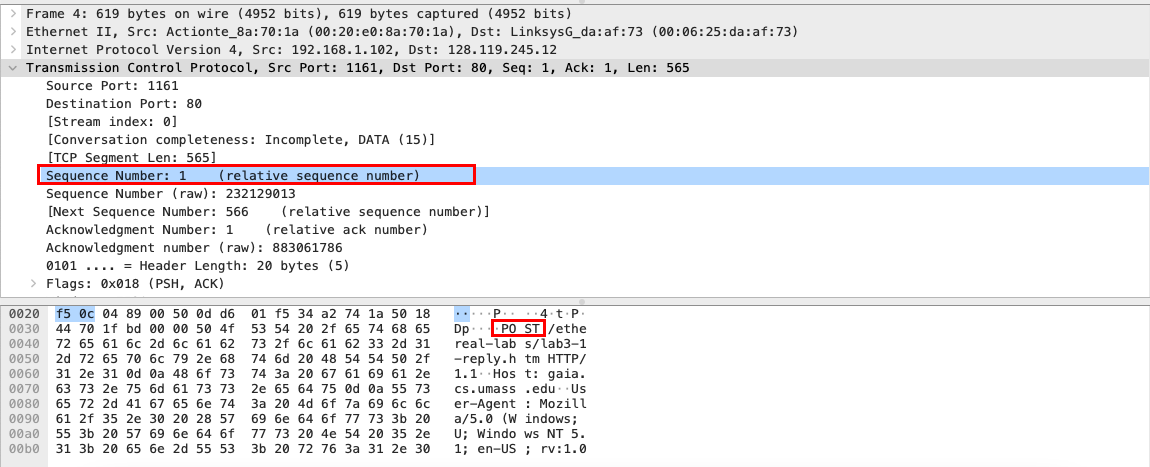
The Syn flag and Acknowledgement flag is set to 1 to indicate it’s a SYNACK segment.



6.

According to the figure below,

The TCP segment with sequence number 1 containing the HTTP POST command.



7.

The sequence numbers of the first six segments in the TCP connection are 1, 566, 2026, 3486, 4946, 6406.

Each segment was sent at 0.026477, 0.041737, 0.054026, 0.054690, 0.077405 and 0.078157 respectively.

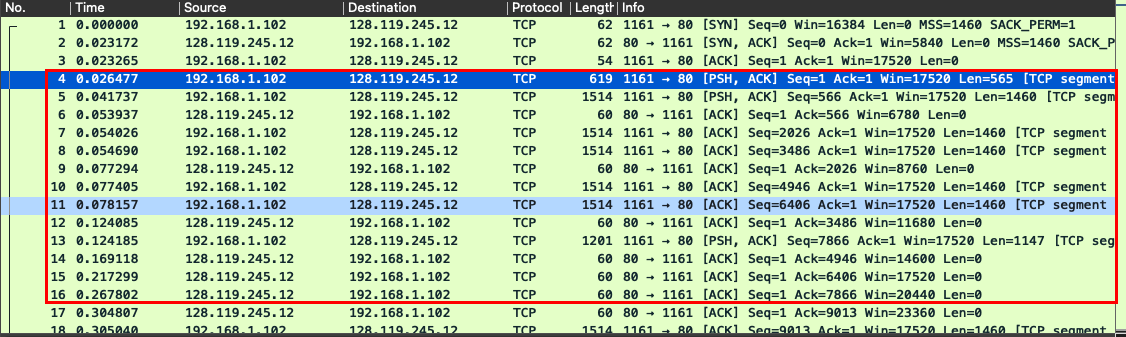
Each ACK was received at 0.053937, 0.077294, 0.124085, 0.169118, 0.217299 and 0.267802 respectively.

Thus, the RTT value of each segment is 0.027460, 0.035557, 0.070059, 0.114430, 0.139890 and 0.189640 respectively.

Based on the equation,

EstimatedRTT = 0.875 EstimatedRTT + 0.125 SampleRTT

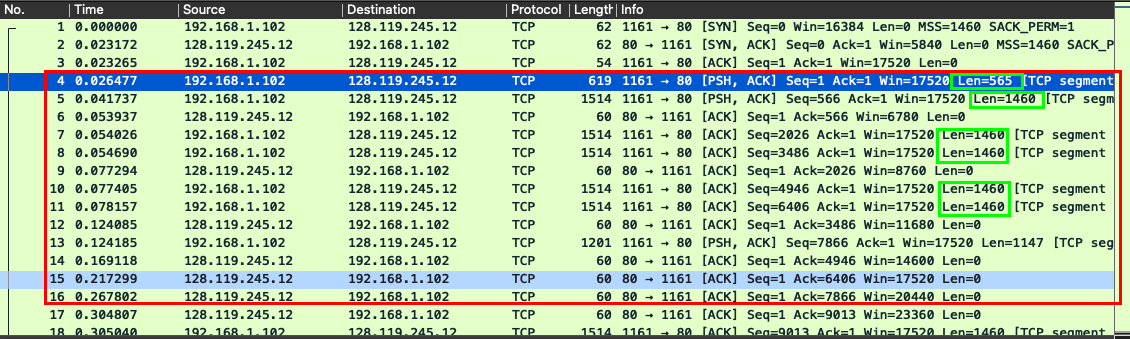
we can get the EstimatedRTT for each segment is 0.02746, 0.0285, 0.0337, 0.0438, 0.0558 and 0.0725 respectively.



8.

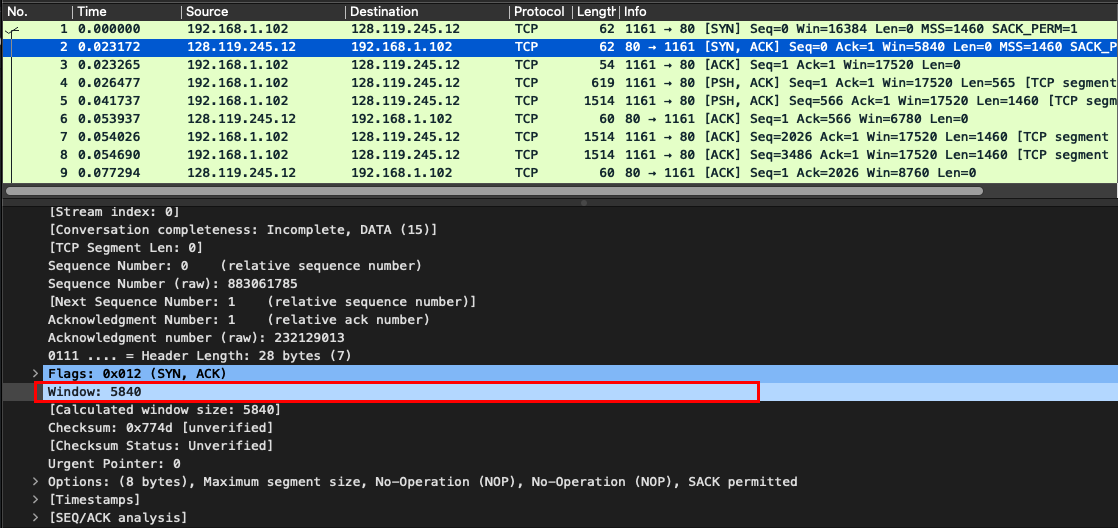
According to the figure below,

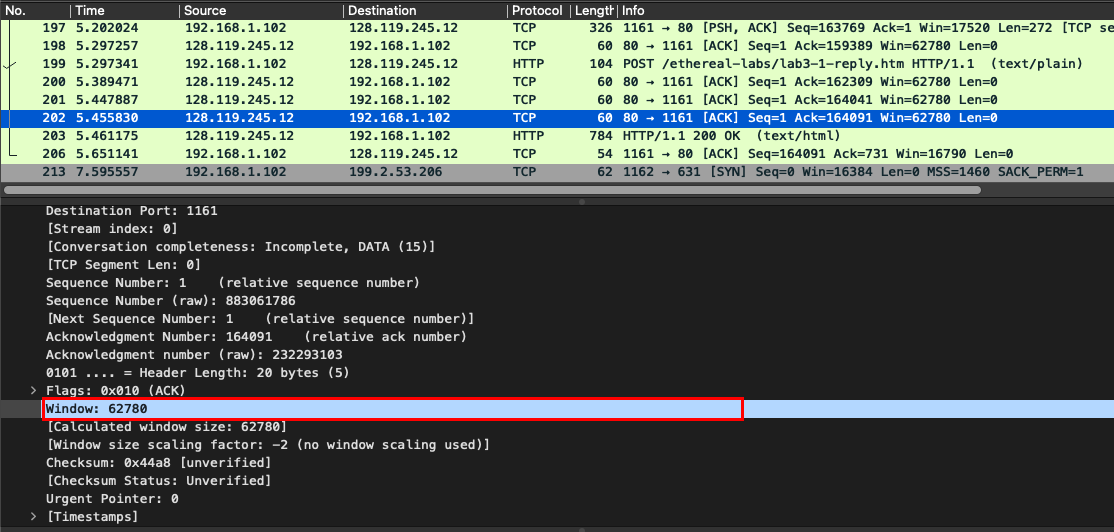
the lengths of the first six TCP segments are 565 B, 1460 B, 1460 B, 1460 B, 1460 B and 1460 B respectively.



9.

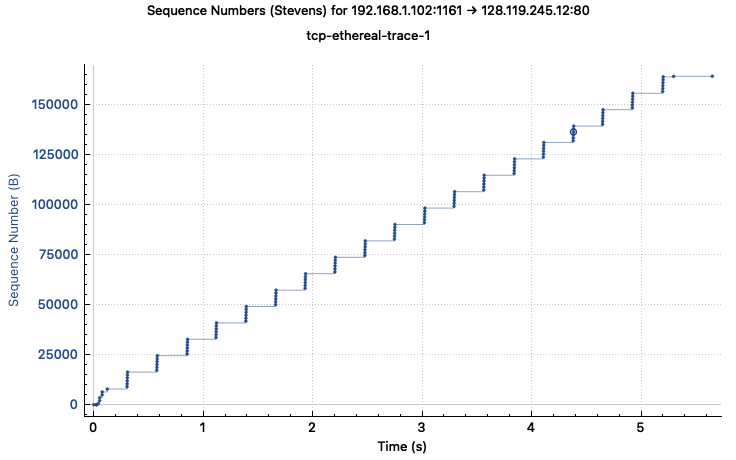
The minimum amount of buffer space is 5840 B. The sender is never throttled because the buffer space grows gradually and reaches 62780 B.





10.

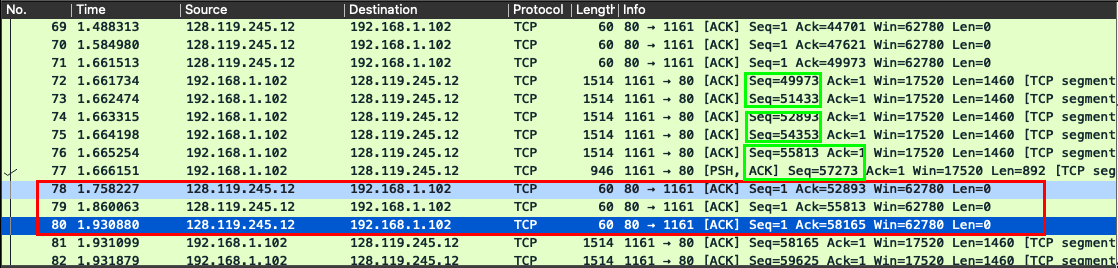
According to the Time-Sequence-Graph, we can find that the sequence number sent from source to destination increases monotonically. Thus, there is no retransmitted segments in this trace.



11.

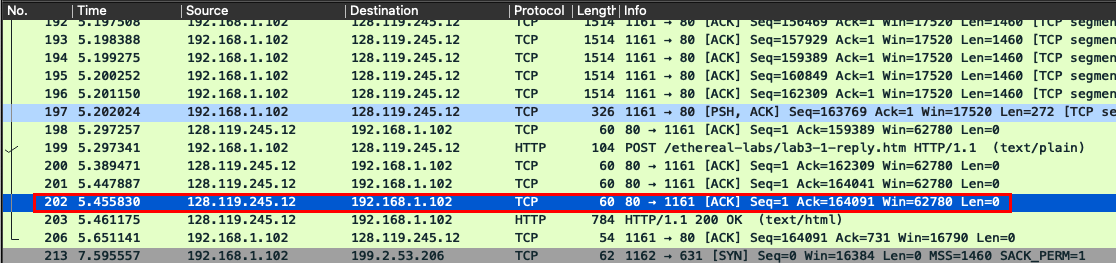
The receiver typically acknowledges 1460 B data in one ACK, which is exactly one TCP segment.

There are cases where the receiver is ACKing other received segments. For example, No. 78, 79, 80 frame acknowledges 2 segments, which is 2920 B.



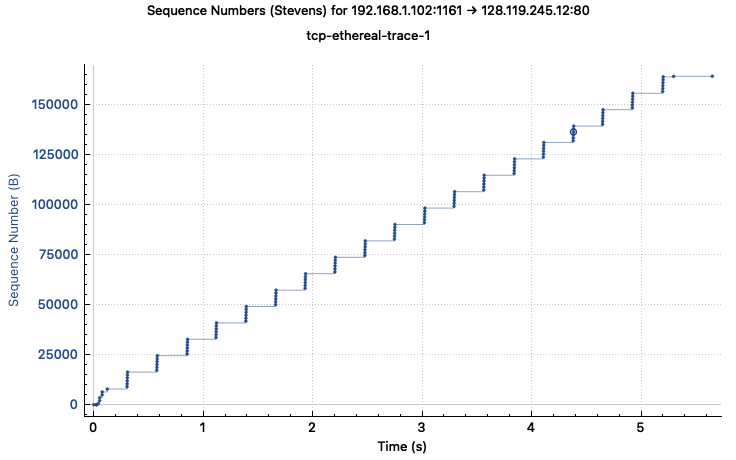
12.

We can compute the throughput by dividing the total data transferred by the total time of the TCP connection. Notice that the ACK number is equal to the number of bytes reliably transferred plus one. Thus, we can get the total data transferred via the last ACK segment, which is 164091 – 1 = 164090 B. The total time can also be derived, which is the last ACK time minus the time when the first segment starts transmitting. That is 5.455830 – 0.026477 = 5.429353. Hence, the throughput is 164090 / 5.429353 = 30222.8 B/s.



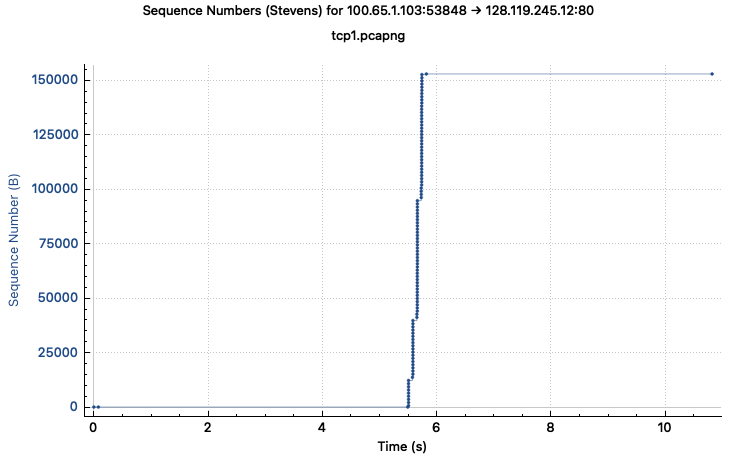
13.

The TCP’s slowstart begins at about 0 and ends at about 0.2s. After that, the congestion avoidance takes over. Observe that the sender always transmits the frames in a batch of 6. However, the receiver’s advertised window size grows during connection and bigger than 6 frames, which means the limitation is not caused by flow control. It may be caused by some rate limit enforced by the HTTP server. Thus, compared to idealized TCP, the slowstart phase ends very quickly and the window size doesn’t grow linearly during the congestion avoidance phase.



14.

The TCP’s slowstart time begins at about 5.5s and before the end of slowstart phase, the transmission is over. Compared to idealized TCP, in this case, the slowstart phase follow the idealized behavior, during which the window size grows exponentially. However, the TCP behavior also depends on the application. When the web object is of very small size, the transmission will terminate very quickly even before the end of slowstart phase.



Conclusion

This report investigates the behavior of the celebrated TCP protocol in detail. A trace of the TCP segments sent and received in transferring a 150KB file from client computer to gaia.cs.umass.edu is analyzed. The first 12 questions explore TCP connection setup (SYN and SYNACK), the sequence numbers and ACK numbers for providing reliable data transfer and the performance (averaging throughput and EstimatedRTT) of the TCP connection between client computer and the remote server. The congestion control algorithm – slow start and congestion avoidance, and TCP’s receiver-advertised flow control mechanism are investigated in the last two questions. The behavior of TCP protocol is well discussed in this report.