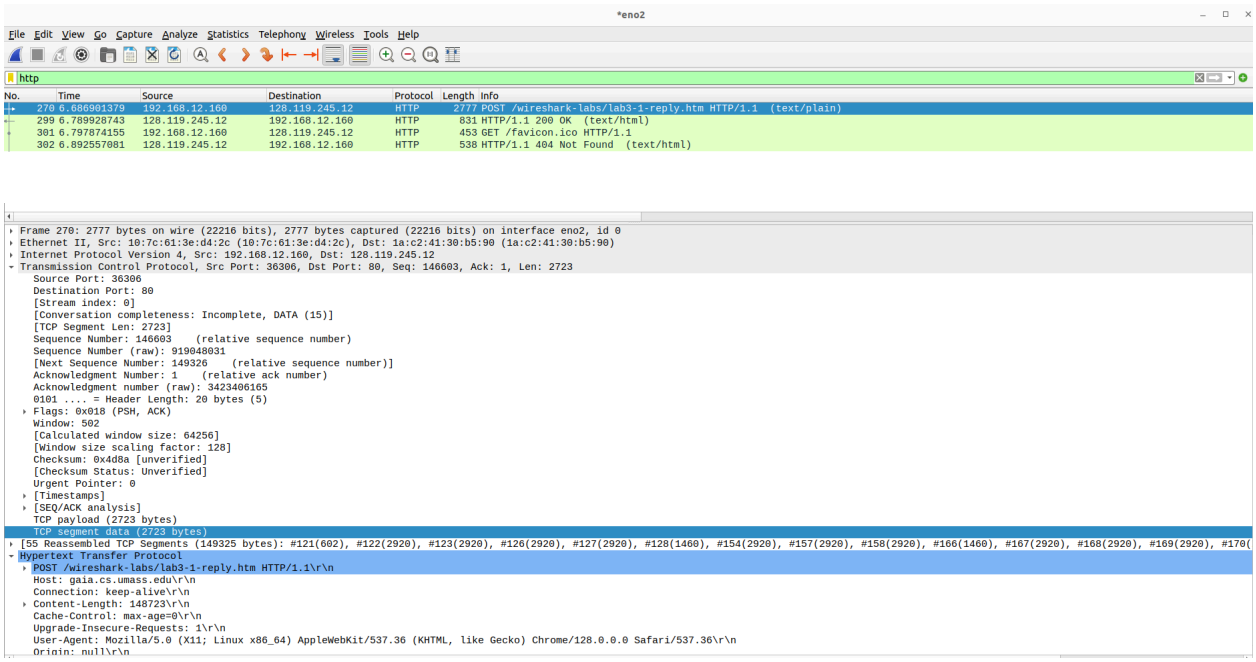
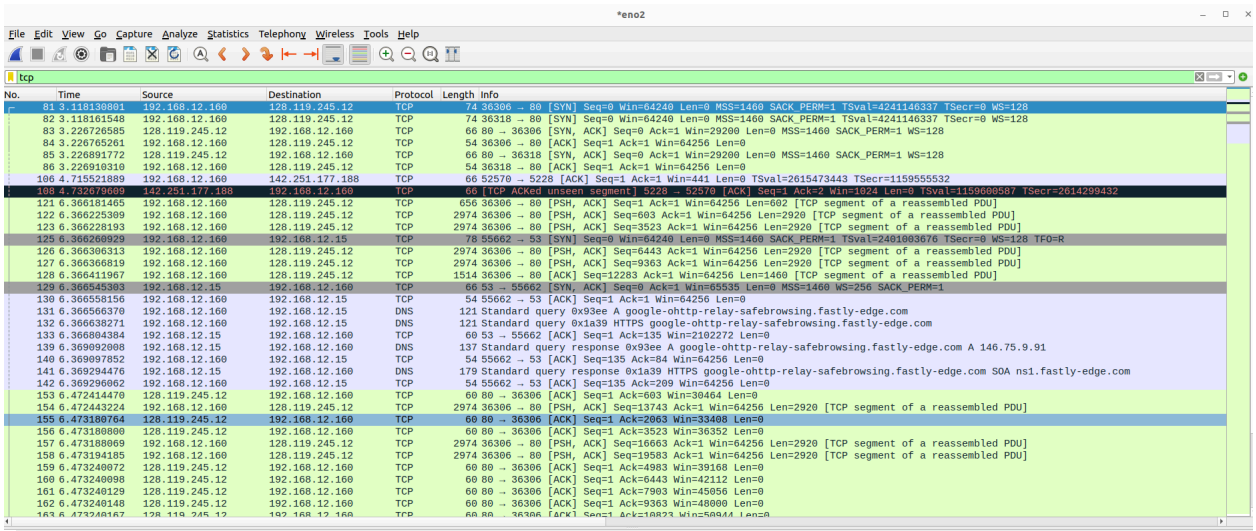
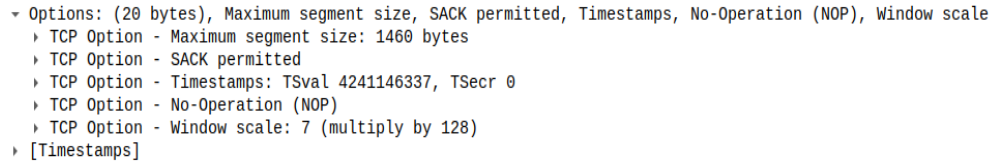
Sarwar, Andre and Mohammed
October 8, 2024
Lab 5 - TCP



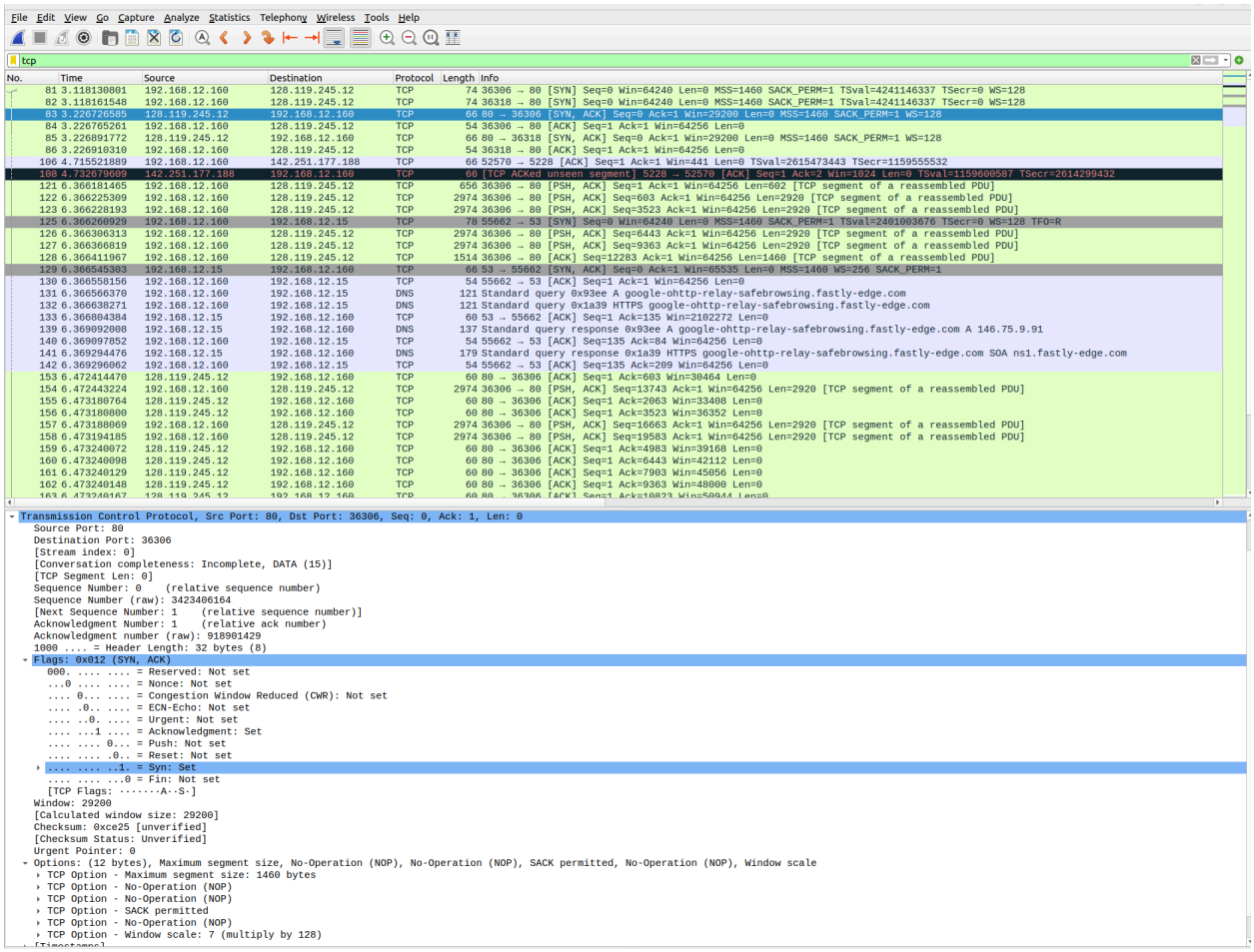

1. What is the IP address and TCP port number used by the client computer (source) that is transferring the alice.txt file to gaia.cs.umass.edu? To answer this question, it's probably easiest to select an HTTP message and explore the details of the TCP packet used to carry this HTTP message, using the "details of the selected packet header window" (refer to Figure 2 in the "Getting Started with Wireshark" Lab if you're uncertain about the Wireshark windows).
    - The IP of the client computer is 192.168.12.160 and the TCP port number is 36306.
2. What is the IP address of gaia.cs.umass.edu? On what port number is it sending and receiving TCP segments for this connection?
    - The IP address of gaia.cs.umass.edu is 128.119.245.12 and the port number it is sending and receiving TCP segment is 80.
3. What is the sequence number of the TCP SYN segment that is used to initiate the TCP connection between the client computer and gaia.cs.umass.edu? (Note: this is the "raw" sequence number carried in the TCP segment itself; it is NOT the packet # in the "No." column in the Wireshark window. Remember there is no such thing as a "packet number" in TCP or UDP; as you know, there are sequence numbers in TCP and that's what we're after here. Also note that this is not the relative sequence number with respect to the starting sequence number of this TCP session.). What is it in this TCP segment that

identifies the segment as a SYN segment? Will the TCP receiver in this session be able to use Selective Acknowledgments (allowing TCP to function a bit more like a "selective repeat" receiver, see section 3.4.5 in the text)?



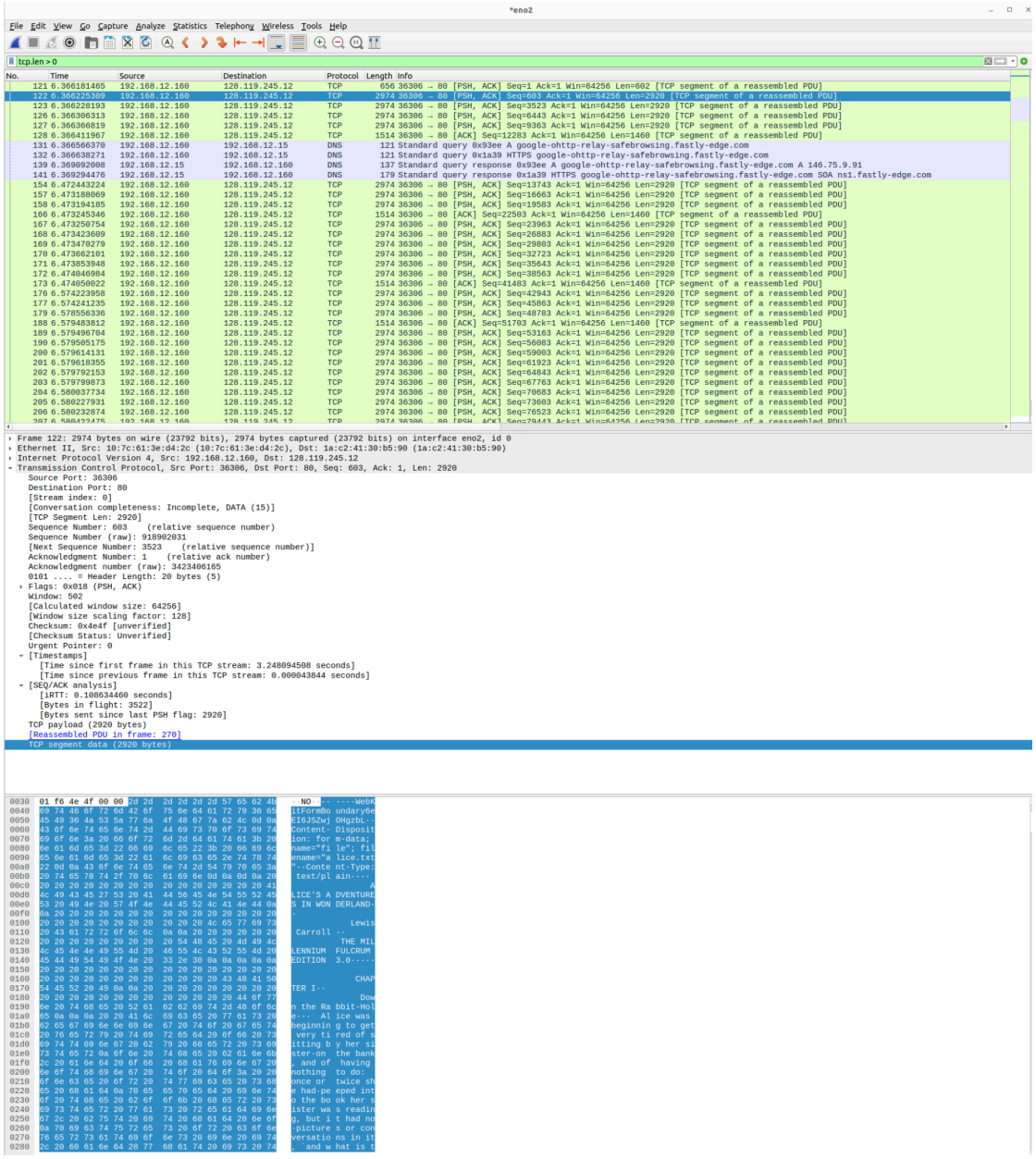

- The raw sequence number of the TCP SYN segment is **918901428** and the relative sequence number of the TCP SYN segment is **0**. The TCP segment identifies the segment as a SYN segment because it has the SYN flag at the header and it is set to 1. The TCP receiver in this session will be able to use Selective Acknowledgments because it says SACK is permitted in the TCP Option.




4. What is the sequence number of the SYN ACK segment sent by gaia.cs.umass.edu to the client computer in reply to the SYN? What is it in the segment that identifies the segment as a SYNACK segment? What is the value of the Acknowledgement field in the SYNACK segment? How did gaia.cs.umass.edu determine that value?
    - The raw sequence number of the SYN ACK segment sent by gaia.cs.umass.edu to the client computer in reply to the SYN is **3423406164** and the relative sequence number is **0**. The TCP segment identifies the segment as a SYN ACK segment because it has the SYN and ACK flag at the header and both are set to 1. The value of the Acknowledgement field is 1. gaia.cs.umass.edu determined that value by adding 1 to the initial sequence value.
5. What is the sequence number of the TCP segment containing the header of the HTTP POST command? Note that in order to find the POST message header, you'll need to dig into the packet content field at the bottom of the Wireshark window, looking for a segment with the ASCII text "POST" within its DATA field4,5. How many bytes of data are contained in the payload (data) field of this TCP segment? Did all of the data in the transferred file alice.txt fit into this single segment?


- The sequence number of the TCP segment containing the header of the HTTP POST command is **1.** The amount of bytes of data contained in the payload field of this TCP segment are **602** bytes. No, the data in the transferred file did not fit into this single segment.

6. Consider the TCP segment containing the HTTP "POST" as the first segment in the data transfer part of the TCP connection.




- At what time was the first segment (the one containing the HTTP POST) in the data-transfer part of the TCP connection sent?
  - The time for the first segment is **6.366181465.**
- At what time was the ACK for this first data-containing segment received?
  - The ACK for the first data-containing segment was received at **6.366225309**.
- What is the RTT for this first data-containing segment?
  - The round trip time (RTT) for the first data-containing segment is **0.108634460** seconds.

- What is the RTT value the second data-carrying TCP segment and its ACK?
  - The round trip time (RTT) for the second data-containing segment is **0.108634460** seconds and the ACK value is **1**.
- What is the EstimatedRTT value (see Section 3.5.3, in the text) after the ACK for the second data-carrying segment is received? Assume that in making this calculation after the received of the ACK for the second segment, that the initial value of EstimatedRTT is equal to the measured RTT for the first segment, and then is computed using the EstimatedRTT equation on page 242, and a value of α = 0.125.
  - $EstimatedRTT=(1-\alpha)\cdot EstimatedRTT+\alpha\cdot SampleRTT$
  - $\alpha=0.125$
  - The initial EstimatedRTT is equal to the measured RTT for the first segment, which is 0.108634460 seconds.
  - The SampleRTT for the second segment is also 0.108634460 seconds.
  - *New estimatedRTT = (.875)*.108634460 + .125*.108634460*
  - ***New estimatedRTT = .10863446***

7. What is the length (header plus payload) of each of the first four data-carrying TCP segments?
   - The length of each of the first four data-carrying TCP segments are **2920**.

8. What is the minimum amount of available buffer space advertised to the client by gaia.cs.umass.edu among these first four data-carrying TCP segments? Does the lack of receiver buffer space ever throttle the sender for these first four data carrying segments?
   - The minim amount of available buffer space advertised to the client by the gaia.cs.umass.edu is **Window:30464.** No, it did not because the the window size increased over time.

where did you find that? the screen capture doesn't show it.

9. Are there any retransmitted segments in the trace file? What did you check for (in the trace) in order to answer this question?
   - There are no retransmitted segment in the trace file. The fact that an old Acknowledgement number was never resent in order to re-request previous packets demonstrates this.

you should also check duplicate sequence numbers from the sender in case of duplicate packets. Not only based on duplicate acks.

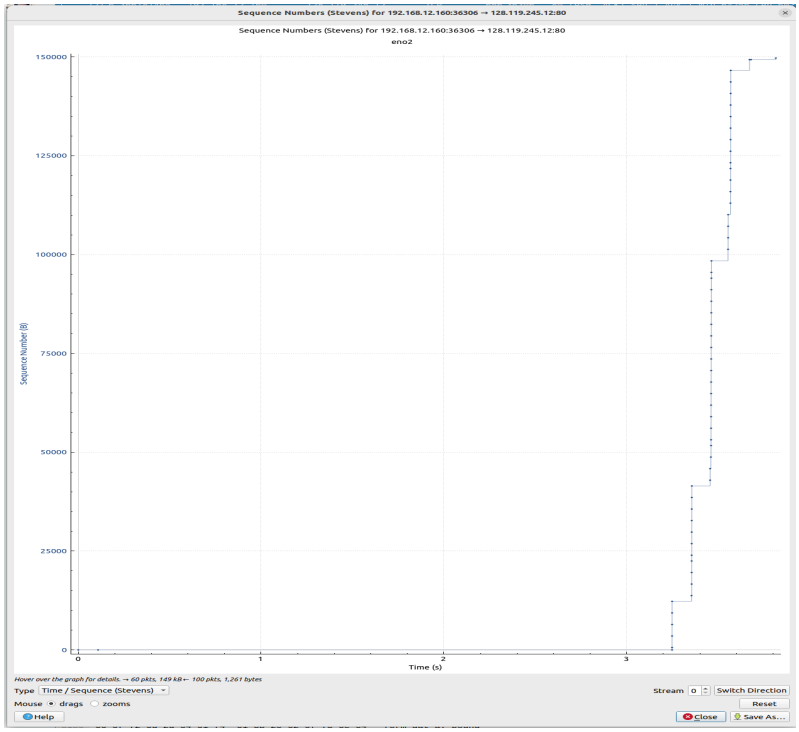
10. How much data does the receiver typically acknowledge in an ACK among the first ten data-carrying segments sent from the client to gaia.cs.umass.edu? Can you identify cases where the receiver is ACKing every other received segment (see Table 3.2 in the text) among these first ten data-carrying segments?
    - The number of data the receiver acknowledges is 2920 bytes of payload data for each acknowledgement. No, each of the first 10 ACKs acknowledgements exactly 2920 bytes of payload data.

11. What is the throughput (bytes transferred per unit time) for the TCP connection? Explain how you calculated this value.
    - The start time is 6.366411967
    - The finished time is 6.789928743
    - Difference: 6.789928743 − 6.366411967 = 0.423516776
    - Calculated window size: 256256
    - Throughput: 256256÷0.42356776 = 604994.110033304 bits/sec

12. Use the Time-Sequence-Graph(Stevens) plotting tool to view the sequence number versus time plot of segments being sent from the client to the gaia.cs.umass.edu server. Consider the "fleets" of packets sent around t = 0.025, t = 0.053, t = 0.082 and t = 0.1. Comment on whether this looks as if TCP is in its slow start phase, congestion avoidance phase or some other phase. Figure 6 shows a slightly different view of this data.
    - TCP is in its slow start phase. We start off with 3 packets, then 6, and then even more after that.

show couple numbers (doubled numb rs) fir every RTT

13. These "fleets" of segments appear to have some periodicity. What can you say about the period?
    - We can say that the period corresponds to the Round Trip Time between the sender and the receiver.
14. Answer each of two questions above for the trace that you have gathered when you transferred a file from your computer to gaia.cs.umass.edu




- TCP is in its slow start phase. We start off with 6, then we get 11, and this continues. Just like number 13, the period corresponds to the Round Trip Time between the sender and the receiver.