Computer Networks 1

Lab 3a

Wireshark Lab: TCP v8.0

Student Name: **Nguyễn Quý Hải**

Student No.: **2052974**

1. **Capturing a bulk TCP transfer from your computer to a remote server**

   Follow the instructions in lab

2. **A first look at the captured trace**

- Q1: What is the IP address and TCP port number used by the client computer (source) that is transferring the file to gaia.cs.umass.edu?

  Answer: client computer (src)

  +IP address: 192.168.1.102

  +TCP Port number: 1161

- Q2: What is the IP address of gaia.cs.umass.edu? On what port number is it sending and receiving TCP segments for this connection?

  Answer: gaia.cs.umass.edu

  +IP address: 128.119.245.12

  +TCP port number: 80

3. **TCP Basics**

- Q4: 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? What is it in the segment that identifies the segment as a SYN segment?

  Answer: SEQ = 0. Syn Flag = 1 that identifies the segment as a SYN segment

  Flags: 0x002 (SYN)

  000. .... = Reserved: Not set

  ...0 .... = Nonce: Not set

  .... 0... .... = Congestion Window Reduced (CWR): Not set

  .... .0.. .... = ECN-Echo: Not set

  .... ..0. .... = Urgent: Not set

  ....0 .... = Acknowledgment: Not set

  .... 0... = Push: Not set

  .... .0.. = Reset: Not set

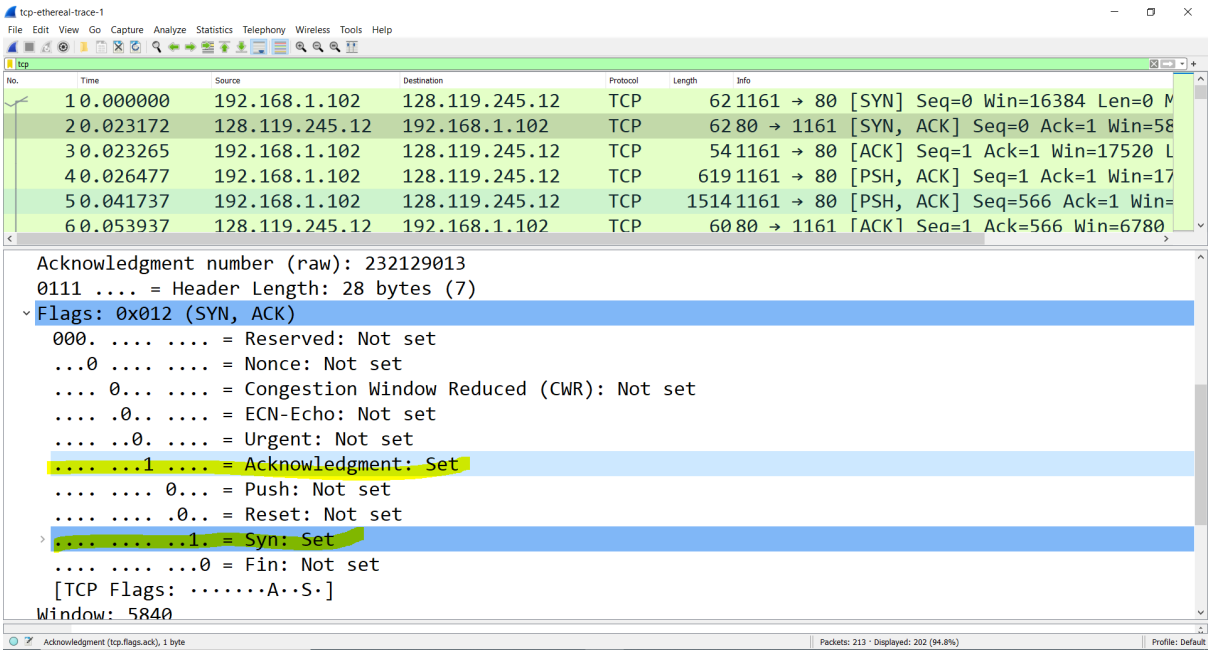
....1. = Syn: Set

....0 = Fin: Not set

[TCP Flags: ·····S·]

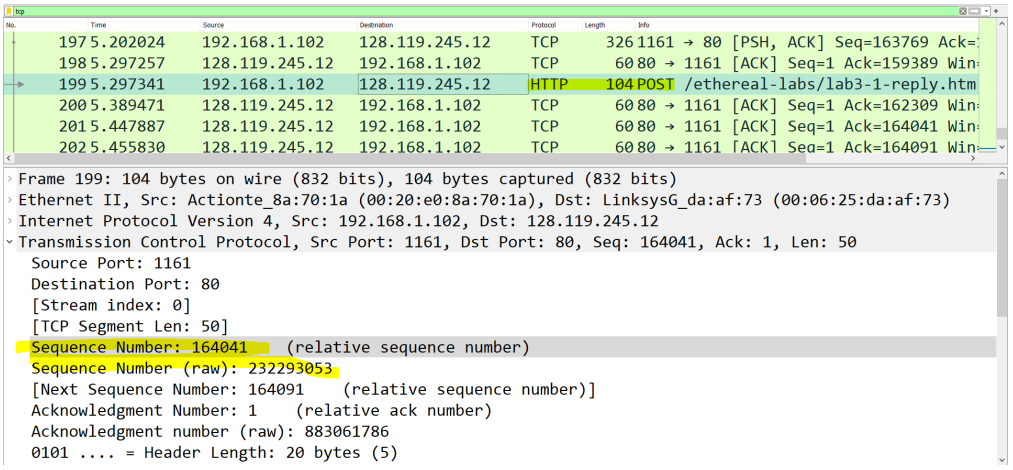
- Q5. What is the sequence number of the SYNACK segment sent by gaia.cs.umass.edu to the client computer in reply to the SYN? What is the value of the Acknowledgement field in the SYNACK segment? How did gaia.cs.umass.edu determine that value? What is it in the segment that identifies the segment as a SYNACK segment?

  Answer: SEQ = 0, Acknowledgement = 1, gaia.cs.umass.edu determined that value by using bitwise operation. The Syn flag and ACK flag are both 1 that identifies the segment as a SYNACK segment.



- Q6. What is the sequence number of the TCP segment containing the HTTP POST command? Note that in order to find the POST command, you'll need to dig into the packet content field at the bottom of the Wireshark window, looking for a segment with a "POST" within its DATA field.

  Answer: Sequence Number: 164041 (relative sequence number)



- Q7. Consider the TCP segment containing the HTTP POST as the first segment in the TCP connection. What are the sequence numbers of the first six segments in the TCP connection (including the segment containing the HTTP POST)? At what time was each segment sent? When was the ACK for each segment received? Given the difference between when each TCP segment was sent, and when its acknowledgement was received, what is the RTT value for each of the six segments? What is the EstimatedRTT value (see Section 3.5.3, page 242 in text) after the receipt of each ACK? Assume that the value of the EstimatedRTT is equal to the measured RTT for the first segment, and then is computed using the EstimatedRTT equation on page 242 for all subsequent segments.

  Note: Wireshark has a nice feature that allows you to plot the RTT for each of the TCP segments sent. Select a TCP segment in the “listing of captured packets” window that is being sent from the client to the gaia.cs.umass.edu server. Then select: Statistics->TCP Stream Graph- >Round Trip Time Graph.

  Answer:

  - Segment 1 sequence number: 1
  - Segment 2 sequence number: 566
  - Segment 3 sequence number: 2026
  - Segment 4 sequence number: 3486
  - Segment 5 sequence number: 4946
  - Segment 6 sequence number: 6406

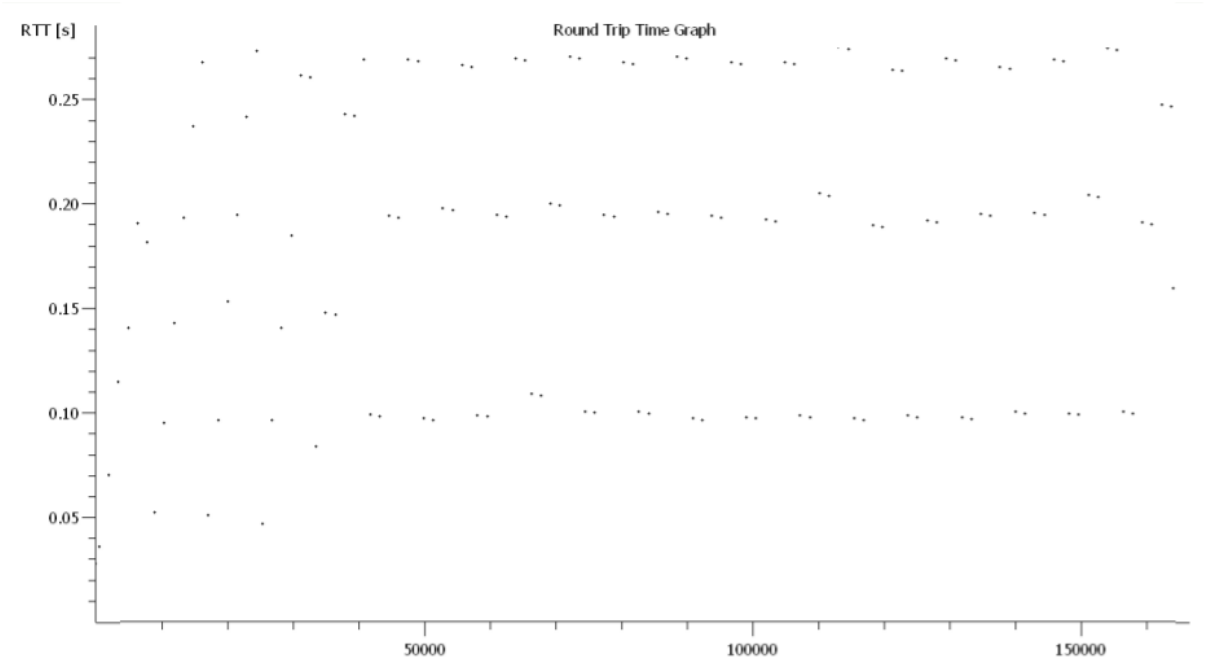
	Sent time	ACK received time	RTT
Segment 1	0.026477	0.053937	0.02746
Segment 2	0.041737	0.077294	0.035557

Segment 3	0.054026	0.124085	0.070059
Segment 4	0.054690	0.169118	0.11443
Segment 5	0.077405	0.217299	0.13989
Segment 6	0.078157	0.267802	0.18964

EstimatedRTT = (1 – α) * EstimatedRTT + α * SampleRTT

In this case, I choose α = 0.125, then we have:

- EstimatedRTT after the receipt of the ACK of segment 1:
  - EstimatedRTT = RTT for Segment 1 = **0.02746 second**
- EstimatedRTT after the receipt of the ACK of segment 2:
  - EstimatedRTT = 0.875 * 0.02746 + 0.125 * 0.035557 = **0.0285**
- EstimatedRTT after the receipt of the ACK of segment 3:
  - EstimatedRTT = 0.875 * 0.0285 + 0.125 * 0.070059 = **0.0337**
- EstimatedRTT after the receipt of the ACK of segment 4:
  - EstimatedRTT = 0.875 * 0.0337+ 0.125 * 0.11443 = **0.0438**
- EstimatedRTT after the receipt of the ACK of segment 5:
  - EstimatedRTT = 0.875 * 0.0438 + 0.125 * 0.13989 = **0.0558**
- EstimatedRTT after the receipt of the ACK of segment 6:
  - EstimatedRTT = 0.875 * 0.0558 + 0.125 * 0.18964 = **0.0725 second**



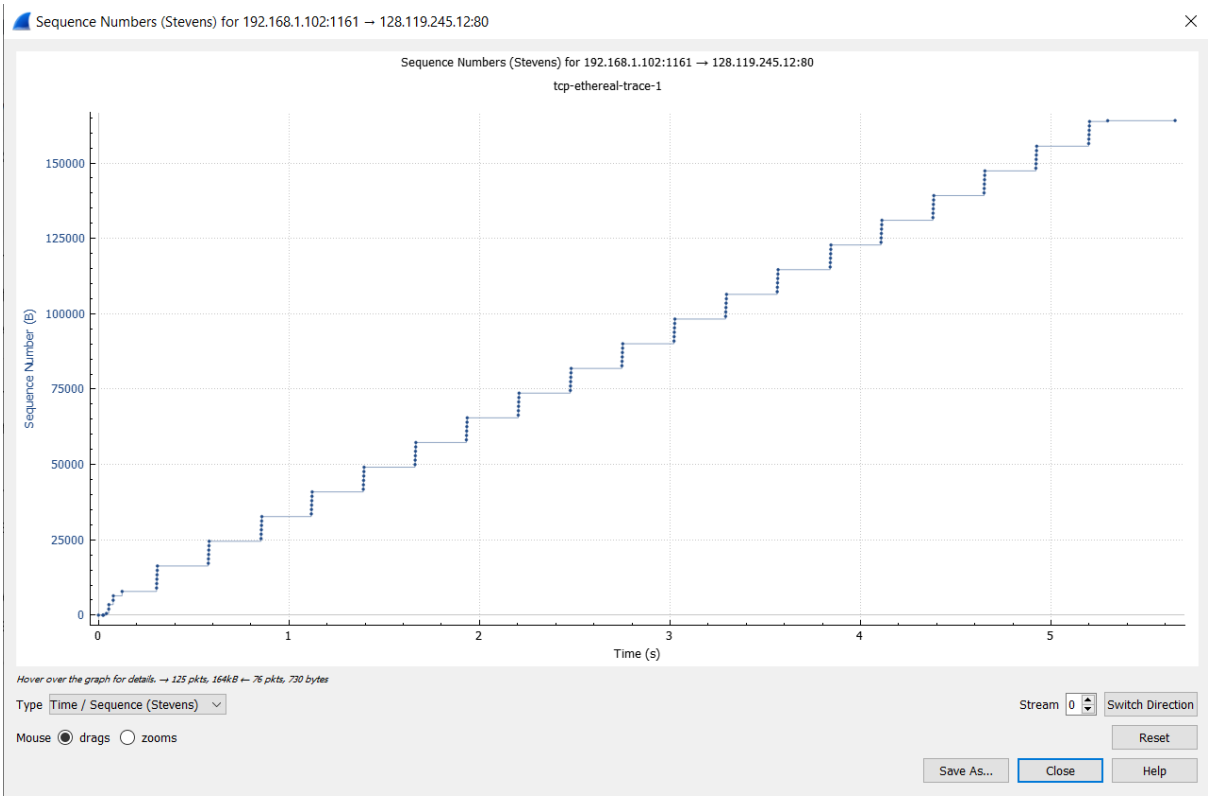
- Q8. What is the length of each of the first six TCP segments?

  Answer:

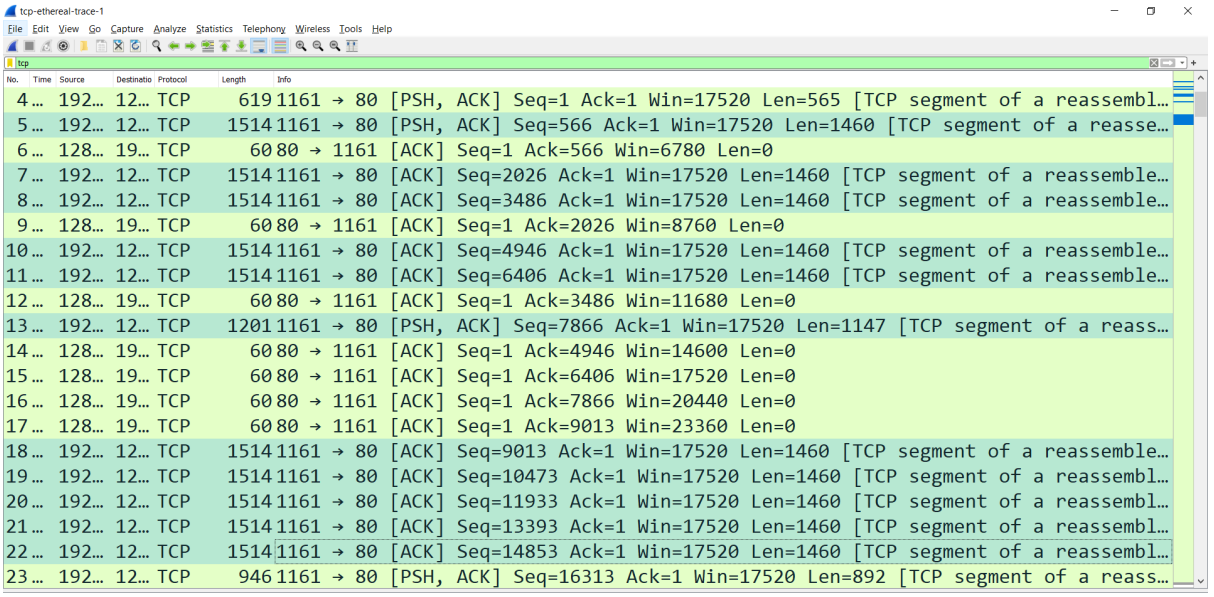
  - Length of the first TCP segment: 565 bytes

- Length of each of the other five TCP segments: 1460 bytes

- Q9. What is the minimum amount of available buffer space advertised at the received for the entire trace? Does the lack of receiver buffer space ever throttle the sender?
  Answer: The minimum amount of buffer space advertised at gaia.cs.umass.edu for the entire trace is 5840 bytes. The sender is never throttled due to the lack of receiver buffer space by inspecting this trace.
- Q10. Are there any retransmitted segments in the trace file? What did you check for (in the trace) in order to answer this question?
  Answer: There are no retransmitted segments in the trace file. I checked the sequence numbers of TCP segments in the trace file in Stevens graph



- Q11. How much data does the receiver typically acknowledge in an ACK? Can you identify cases where the receiver is ACKing every other received segment (see Table 3.2 on page 250 in the text).
  Answer:



- Q12. What is the throughput (bytes transferred per unit time) for the TCP connection? Explain how you calculated this value.

  Answer: **58.985** KByte/sec. I choose from Seq = 566 -> Seq = 14853

  The total data: 1460*10+1147=15747 bytes, time instant: 1093095860.879080000 - 1093095860.612118000 = 0.26696205139. And then, throughput = total data / time.

4. TCP congestion control in action

- Q13. 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. Can you identify where TCP's slowstart phase begins and ends, and where congestion avoidance takes over? Comment on ways in which the measured data differs from the idealized behavior of TCP that we've studied in the text.

  Answer: The slow start phase begins around zero and ends around .15 seconds according to the graph; after that congestion takes over. The measured data uses only a fraction of the window size instead of the 1/3 to a half.