

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

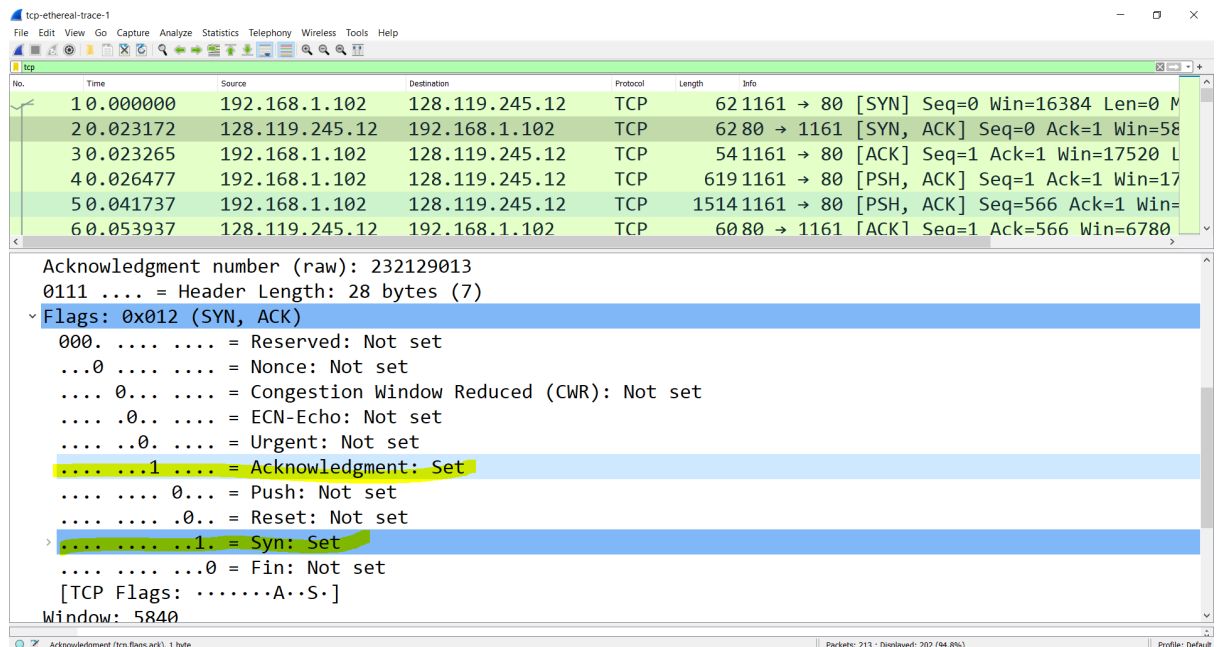
.... ..1. = Syn: Set

.... ..0 = Fin: Not set

[TCP Flags:S.]

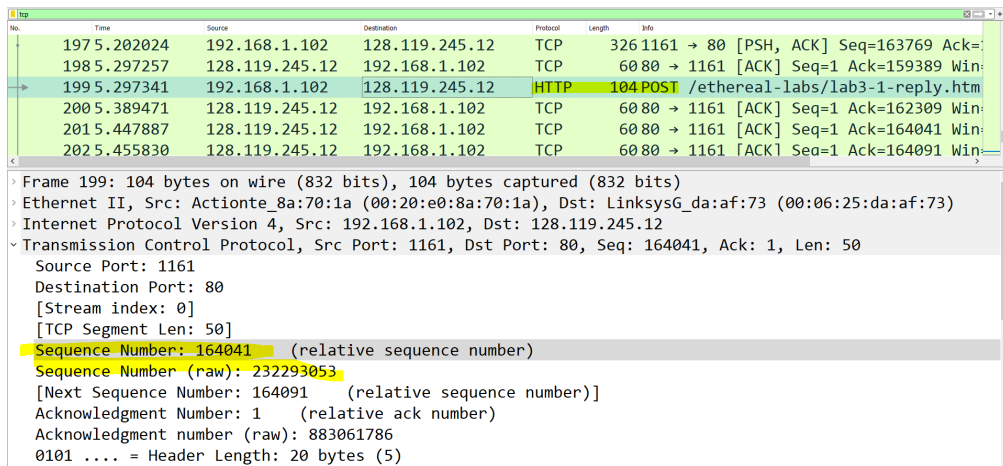
- Q5. What is the sequence number of the SYNACK segment sent by gaia.cs.umass.edu to the client computer in reply to the SYN? What is the value of the Acknowledgement field in the SYNACK segment? How did gaia.cs.umass.edu determine that value? What is it in the segment that identifies the segment as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



No.	Time	Source	Destination	Protocol	Length	Info
197	5.202024	192.168.1.102	128.119.245.12	TCP	326	1161 → 80 [PSH, ACK] Seq=163769 Ack=
198	5.297257	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=159389 Win=
199	5.297341	192.168.1.102	128.119.245.12	HTTP	104	POST /ethereal-labs/lab3-1-reply.htm
200	5.389471	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=162309 Win=
201	5.447887	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=164041 Win=
202	5.455830	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=164091 Win=

Frame 199: 104 bytes on wire (832 bits), 104 bytes captured (832 bits)
 Ethernet II, Src: Actionte_8a:70:1a (00:20:e0:8a:70:1a), Dst: LinksysG_da:af:73 (00:06:25:da:af:73)
 Internet Protocol Version 4, Src: 192.168.1.102, Dst: 128.119.245.12
 Transmission Control Protocol, Src Port: 1161, Dst Port: 80, Seq: 164041, Ack: 1, Len: 50
 Source Port: 1161
 Destination Port: 80
 [Stream index: 0]
 [TCP Segment Len: 50]
 Sequence Number: 164041 (relative sequence number)
 Sequence Number (raw): 232293053
 [Next Sequence Number: 164091 (relative sequence number)]
 Acknowledgment Number: 1 (relative ack number)
 Acknowledgment number (raw): 883061786
 0101 = Header Length: 20 bytes (5)

- 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

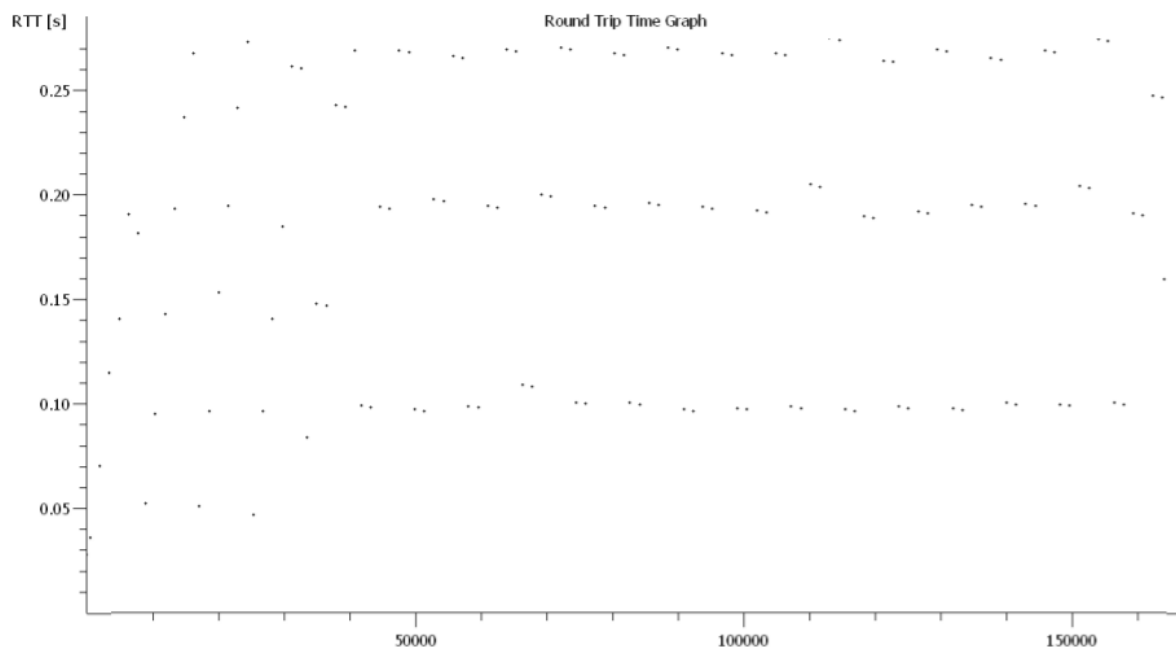
	Sent time	ACK received time	RTT
Segment 1	0.026477	0.053937	0.02746
Segment 2	0.041737	0.077294	0.035557

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 $\alpha = 0.125$, 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 = $0.875 * 0.02746 + 0.125 * 0.035557 = \mathbf{0.0285}$
- EstimatedRTT after the receipt of the ACK of segment 3:
 - EstimatedRTT = $0.875 * 0.0285 + 0.125 * 0.070059 = \mathbf{0.0337}$
- EstimatedRTT after the receipt of the ACK of segment 4:
 - EstimatedRTT = $0.875 * 0.0337 + 0.125 * 0.11443 = \mathbf{0.0438}$
- EstimatedRTT after the receipt of the ACK of segment 5:
 - EstimatedRTT = $0.875 * 0.0438 + 0.125 * 0.13989 = \mathbf{0.0558}$
- EstimatedRTT after the receipt of the ACK of segment 6:
 - EstimatedRTT = $0.875 * 0.0558 + 0.125 * 0.18964 = \mathbf{0.0725 \text{ second}}$



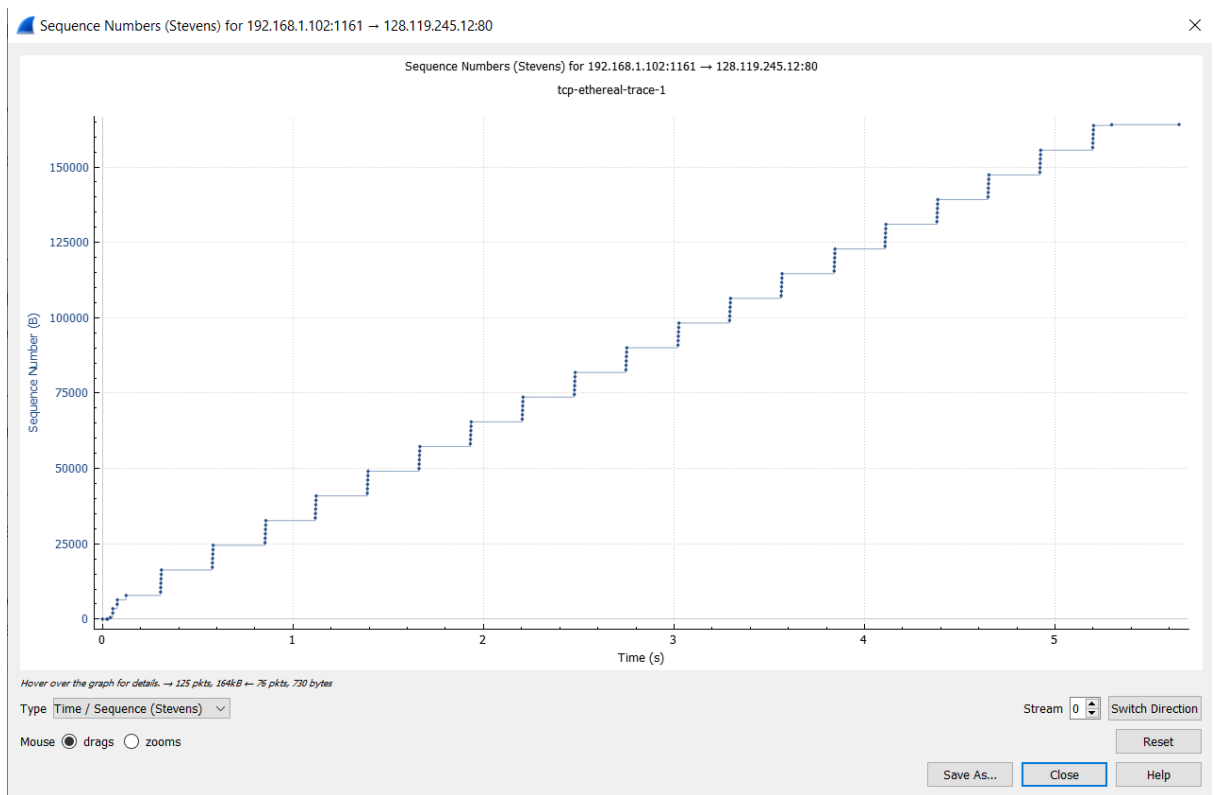
- Q8. What is the length of each of the first six TCP segments?

Answer:

- Length of the first TCP segment: 565 bytes

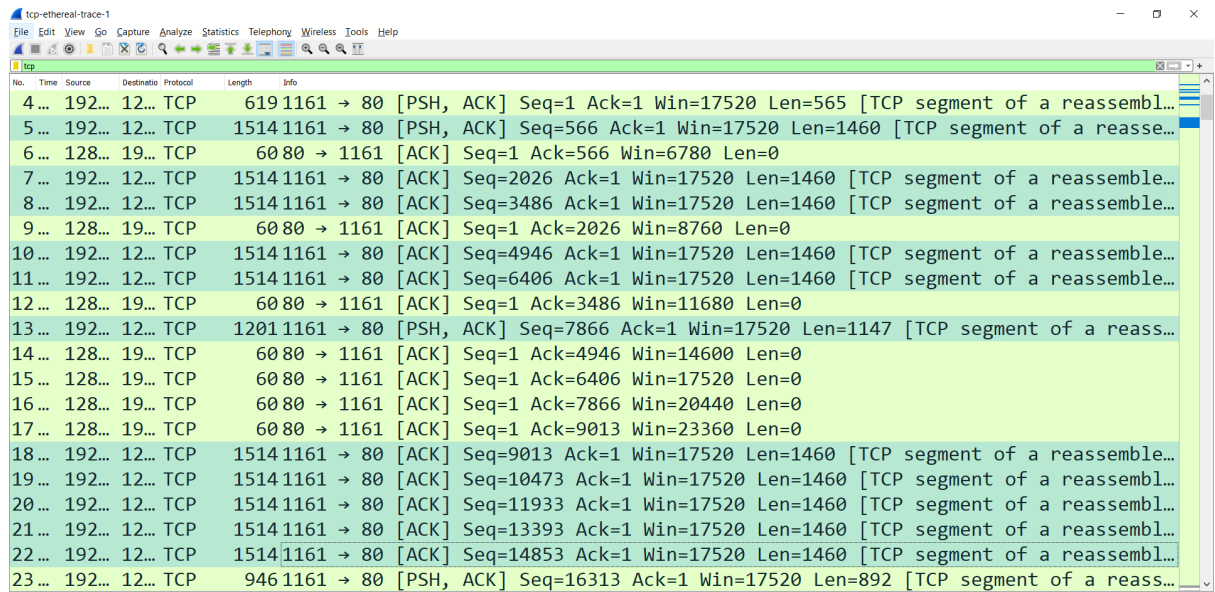
- Length of each of the other five TCP segments: 1460 bytes
- Q9. What is the minimum amount of available buffer space advertised at the received for the entire trace? Does the lack of receiver buffer space ever throttle the sender?
Answer: The minimum amount of buffer space advertised at gaia.cs.umass.edu 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:



No.	Time	Source	Destination	Protocol	Length	Info
4	...	192...	12...	TCP	619	1161 → 80 [PSH, ACK] Seq=1 Ack=1 Win=17520 Len=565 [TCP segment of a reassembl...
5	...	192...	12...	TCP	1514	1161 → 80 [PSH, ACK] Seq=566 Ack=1 Win=17520 Len=1460 [TCP segment of a reasse...
6	...	128...	19...	TCP	60	80 → 1161 [ACK] Seq=1 Ack=566 Win=6780 Len=0
7	...	192...	12...	TCP	1514	1161 → 80 [ACK] Seq=2026 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembl...
8	...	192...	12...	TCP	1514	1161 → 80 [ACK] Seq=3486 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembl...
9	...	128...	19...	TCP	60	80 → 1161 [ACK] Seq=1 Ack=2026 Win=8760 Len=0
10	...	192...	12...	TCP	1514	1161 → 80 [ACK] Seq=4946 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembl...
11	...	192...	12...	TCP	1514	1161 → 80 [ACK] Seq=6406 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembl...
12	...	128...	19...	TCP	60	80 → 1161 [ACK] Seq=1 Ack=3486 Win=11680 Len=0
13	...	192...	12...	TCP	1201	1161 → 80 [PSH, ACK] Seq=7866 Ack=1 Win=17520 Len=1147 [TCP segment of a reass...
14	...	128...	19...	TCP	60	80 → 1161 [ACK] Seq=1 Ack=4946 Win=14600 Len=0
15	...	128...	19...	TCP	60	80 → 1161 [ACK] Seq=1 Ack=6406 Win=17520 Len=0
16	...	128...	19...	TCP	60	80 → 1161 [ACK] Seq=1 Ack=7866 Win=20440 Len=0
17	...	128...	19...	TCP	60	80 → 1161 [ACK] Seq=1 Ack=9013 Win=23360 Len=0
18	...	192...	12...	TCP	1514	1161 → 80 [ACK] Seq=9013 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembl...
19	...	192...	12...	TCP	1514	1161 → 80 [ACK] Seq=10473 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembl...
20	...	192...	12...	TCP	1514	1161 → 80 [ACK] Seq=11933 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembl...
21	...	192...	12...	TCP	1514	1161 → 80 [ACK] Seq=13393 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembl...
22	...	192...	12...	TCP	1514	1161 → 80 [ACK] Seq=14853 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembl...
23	...	192...	12...	TCP	946	1161 → 80 [PSH, ACK] Seq=16313 Ack=1 Win=17520 Len=892 [TCP segment of a reass...

- 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: $1460 \times 10 + 1147 = 15747$ 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.