Computer Networks Lab, Assignment 5

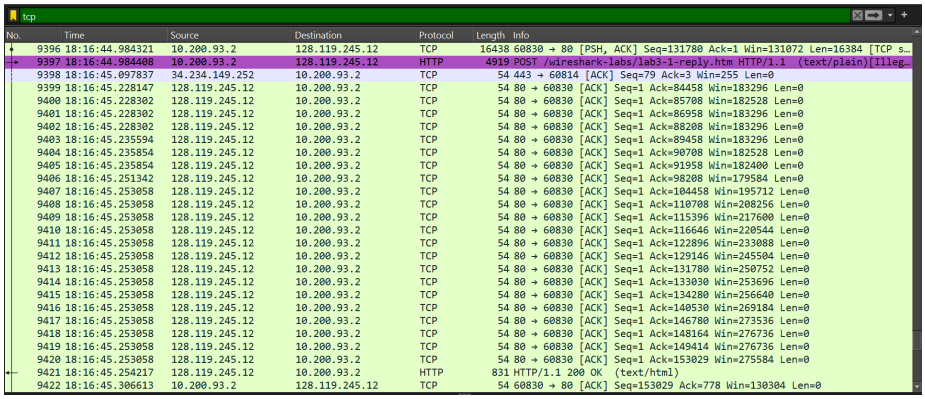
Suyash Gaurav

210010054

---

1 Part 1: Capturing a bulk TCP transfer from your computer to a remote server

---



tcp

No.	Time	Source	Destination	Protocol	Length	Info
9396	18:16:44.984321	10.200.93.2	128.119.245.12	TCP	16438	60830 → 80 [PSH, ACK] Seq=131780 Ack=1 Win=131072 Len=16384 [TCP s…
9397	18:16:44.984408	10.200.93.2	128.119.245.12	HTTP	4919	POST /wireshark-labs/lab3-1-reply.htm HTTP/1.1 (text/plain)[Illeg…
9398	18:16:45.097837	34.234.149.252	10.200.93.2	TCP	54	443 → 60814 [ACK] Seq=79 Ack=3 Win=255 Len=0
9399	18:16:45.228147	128.119.245.12	10.200.93.2	TCP	54	80 → 60830 [ACK] Seq=1 Ack=84458 Win=183296 Len=0
9400	18:16:45.228302	128.119.245.12	10.200.93.2	TCP	54	80 → 60830 [ACK] Seq=1 Ack=85708 Win=182528 Len=0
9401	18:16:45.228302	128.119.245.12	10.200.93.2	TCP	54	80 → 60830 [ACK] Seq=1 Ack=86958 Win=183296 Len=0
9402	18:16:45.228302	128.119.245.12	10.200.93.2	TCP	54	80 → 60830 [ACK] Seq=1 Ack=88208 Win=183296 Len=0
9403	18:16:45.235594	128.119.245.12	10.200.93.2	TCP	54	80 → 60830 [ACK] Seq=1 Ack=89458 Win=183296 Len=0
9404	18:16:45.235854	128.119.245.12	10.200.93.2	TCP	54	80 → 60830 [ACK] Seq=1 Ack=90708 Win=182528 Len=0
9405	18:16:45.235854	128.119.245.12	10.200.93.2	TCP	54	80 → 60830 [ACK] Seq=1 Ack=91958 Win=182400 Len=0
9406	18:16:45.251342	128.119.245.12	10.200.93.2	TCP	54	80 → 60830 [ACK] Seq=1 Ack=98208 Win=179584 Len=0
9407	18:16:45.253058	128.119.245.12	10.200.93.2	TCP	54	80 → 60830 [ACK] Seq=1 Ack=104458 Win=195712 Len=0
9408	18:16:45.253058	128.119.245.12	10.200.93.2	TCP	54	80 → 60830 [ACK] Seq=1 Ack=110708 Win=208256 Len=0
9409	18:16:45.253058	128.119.245.12	10.200.93.2	TCP	54	80 → 60830 [ACK] Seq=1 Ack=115396 Win=217600 Len=0
9410	18:16:45.253058	128.119.245.12	10.200.93.2	TCP	54	80 → 60830 [ACK] Seq=1 Ack=116646 Win=220544 Len=0
9411	18:16:45.253058	128.119.245.12	10.200.93.2	TCP	54	80 → 60830 [ACK] Seq=1 Ack=122896 Win=233088 Len=0
9412	18:16:45.253058	128.119.245.12	10.200.93.2	TCP	54	80 → 60830 [ACK] Seq=1 Ack=129146 Win=245504 Len=0
9413	18:16:45.253058	128.119.245.12	10.200.93.2	TCP	54	80 → 60830 [ACK] Seq=1 Ack=131780 Win=250752 Len=0
9414	18:16:45.253058	128.119.245.12	10.200.93.2	TCP	54	80 → 60830 [ACK] Seq=1 Ack=133030 Win=253696 Len=0
9415	18:16:45.253058	128.119.245.12	10.200.93.2	TCP	54	80 → 60830 [ACK] Seq=1 Ack=134280 Win=256640 Len=0
9416	18:16:45.253058	128.119.245.12	10.200.93.2	TCP	54	80 → 60830 [ACK] Seq=1 Ack=140530 Win=269184 Len=0
9417	18:16:45.253058	128.119.245.12	10.200.93.2	TCP	54	80 → 60830 [ACK] Seq=1 Ack=146780 Win=273536 Len=0
9418	18:16:45.253058	128.119.245.12	10.200.93.2	TCP	54	80 → 60830 [ACK] Seq=1 Ack=148164 Win=276736 Len=0
9419	18:16:45.253058	128.119.245.12	10.200.93.2	TCP	54	80 → 60830 [ACK] Seq=1 Ack=149414 Win=276736 Len=0
9420	18:16:45.253058	128.119.245.12	10.200.93.2	TCP	54	80 → 60830 [ACK] Seq=1 Ack=153029 Win=275584 Len=0
9421	18:16:45.254217	128.119.245.12	10.200.93.2	HTTP	831	HTTP/1.1 200 OK (text/html)
9422	18:16:45.306613	10.200.93.2	128.119.245.12	TCP	54	60830 → 80 [ACK] Seq=153029 Ack=778 Win=130304 Len=0

Figure 1: Part 1 wireshark screenshot

2 Part - 2: A first look at the captured trace

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"

The source IP addr is 10.200.93.2 and port number is 60830.

Figure 2: Q1. IP address

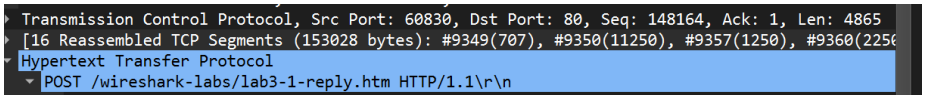


Figure 3: Q1. Port no

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 destination IP addr is 128.119.245.12 and receiving port number is 80.

3 Part - 3: TCP Basics

---

1. 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 this TCP segment that identifies the segment as a SYN segment?

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 is 0. The message contains a SYN flag indicates that it is a SYN segment. (screenshot attached)

Figure 4: Q1. sequence number

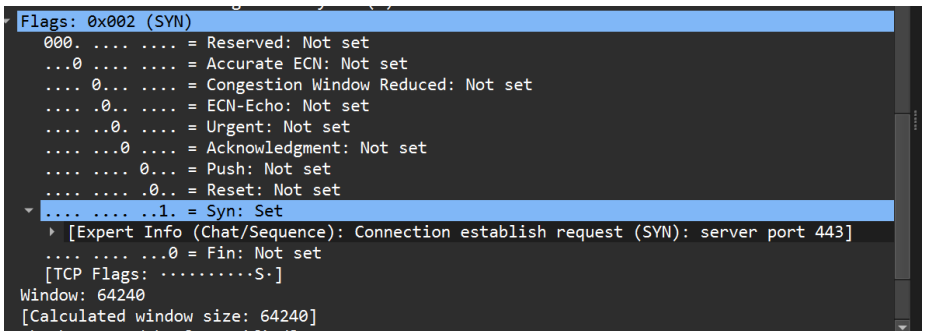


Figure 5: Q1. Flag SYN

2.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 it in the segment that identifies the segment as a SYNACK segment? What is the value of the Acknowledge-

ment field in the SYNACK segment? How did gaia.cs.umass.edu determine that value?

The sequence number of the SYNACK segment sent by gaia.cs.umass.edu to the client computer in reply to the SYN is 0. The value of the acknowledgment field is 1 as shown in screenshot.Initial sequence number is +1 which determines it i.e. ACK = SEQ + 1. Flags shows it to be a SYN ACK message which is carried in message.

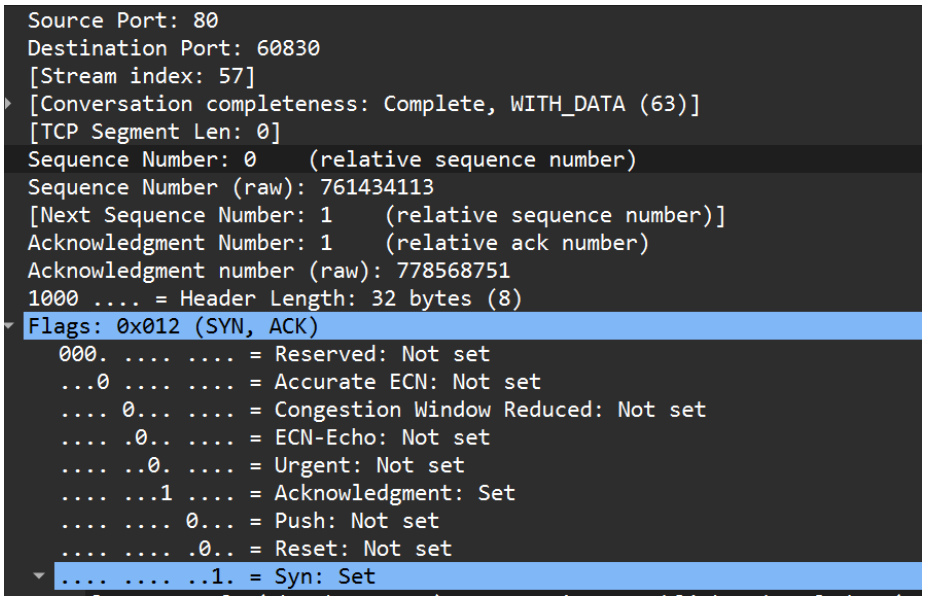
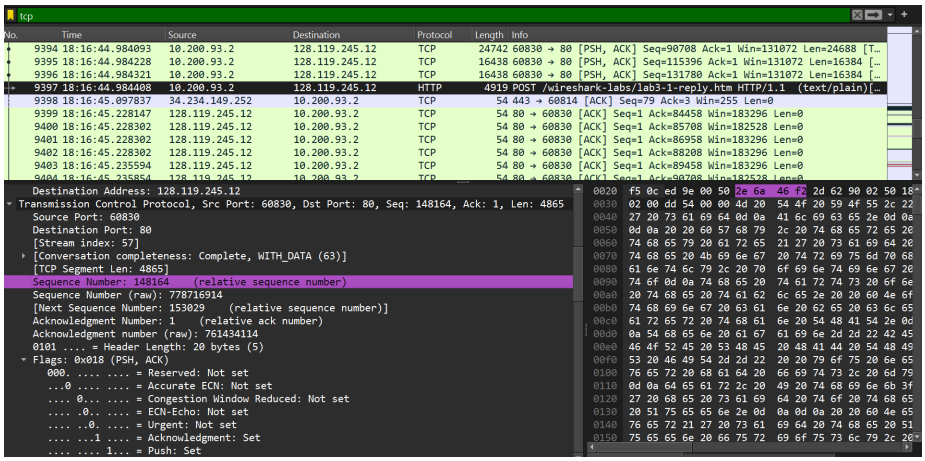


Figure 6: Q1. Flag SYN ACK

3. 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 field. 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?

Sequence Number: 148164 (relative sequence number) screenshot is attached. TCP payload (471 bytes). No, all of the data in the transferred file alice.txt does not fit into this single segment



4. Consider the TCP segment containing the HTTP "POST" as the first segment in the data transfer part of the TCP connection. (i) At what time was the first segment (the one containing the HTTP POST) in the data-transfer part of the TCP connection sent? (ii) At what time was the ACK for this first data-containing segment received? (iii) What is the RTT for this first data-containing segment? (iv) What is the RTT value of the second data-carrying TCP segment and its ACK?

(i) time: 19:47:10
(ii) time : 19: 47: 10
(iii) RTT: 0.000128 seconds
(iv) RTT: 0.3070 seconds

Figure 7:

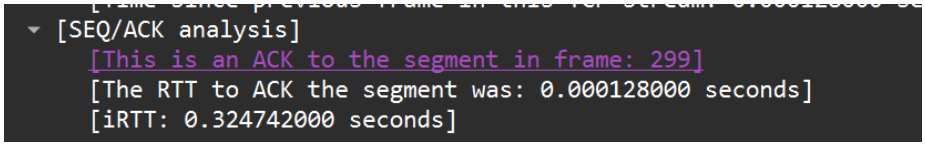


Figure 8: Q4. (iii)

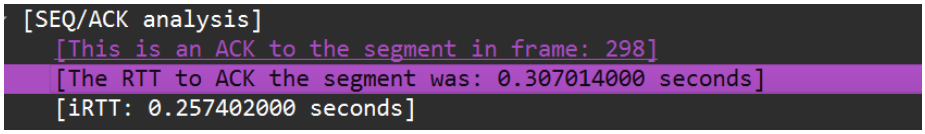


Figure 9: Q4. (iv)

5. What is the length (header plus payload) of each of the first four data-carrying TCP segments?

First four data-carrying TCP segments:
707+54 = 761 Bytes 11250 + 54=11304 Bytes 23750 + 54 = 23804 Bytes 47500 + 54 = 47554 Bytes

6. 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 receiver window, or buffer space, has a value of 65535. We do not approach this buffer size, so it does not throttle the sender.

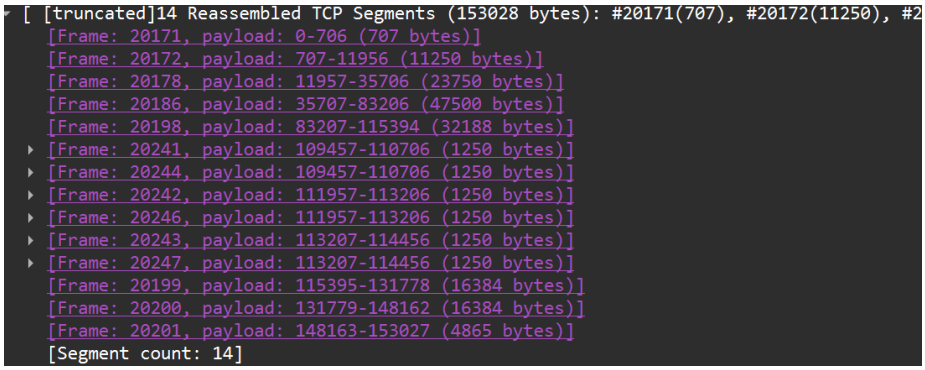


Figure 10: Q5. length (header plus payload)

7. Are there any retransmitted segments in the trace file? What did you check for (in the trace) in order to answer this question?

Yes, I checked for **tcp.analysis.retransmission**

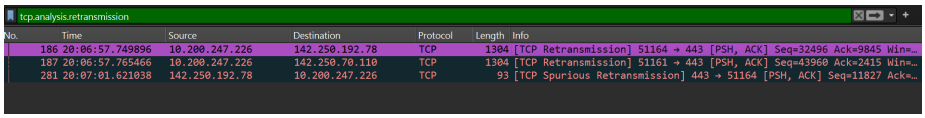


Figure 11: 7. tcp.analysis.retransmission

8. 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 ACK-ing every other received segment among these first ten data-carrying segments?

480 bits of data are normally acknowledged by the receiver. A receiver may occasionally ACK each and every other received segment. When there are two consecutive ACKs, this is evident.

9. What is the throughput (bytes transferred

per unit time) for the TCP connection? Explain how you calculated this value.

File Data: 152321 bytes
File Data//(0.4620033- 0.413820) = 152321/0.01187 = 1282196.78 bytes per sec

---

4 Part - 4 TCP congestion control in action

---

1. 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 slow start phase begins and ends, and where congestion avoidance takes over?

Observe the given diagram. Slow start time starts at 0.25 and ends at 0.56 seconds. And congestion time starts at 0.8 seconds.

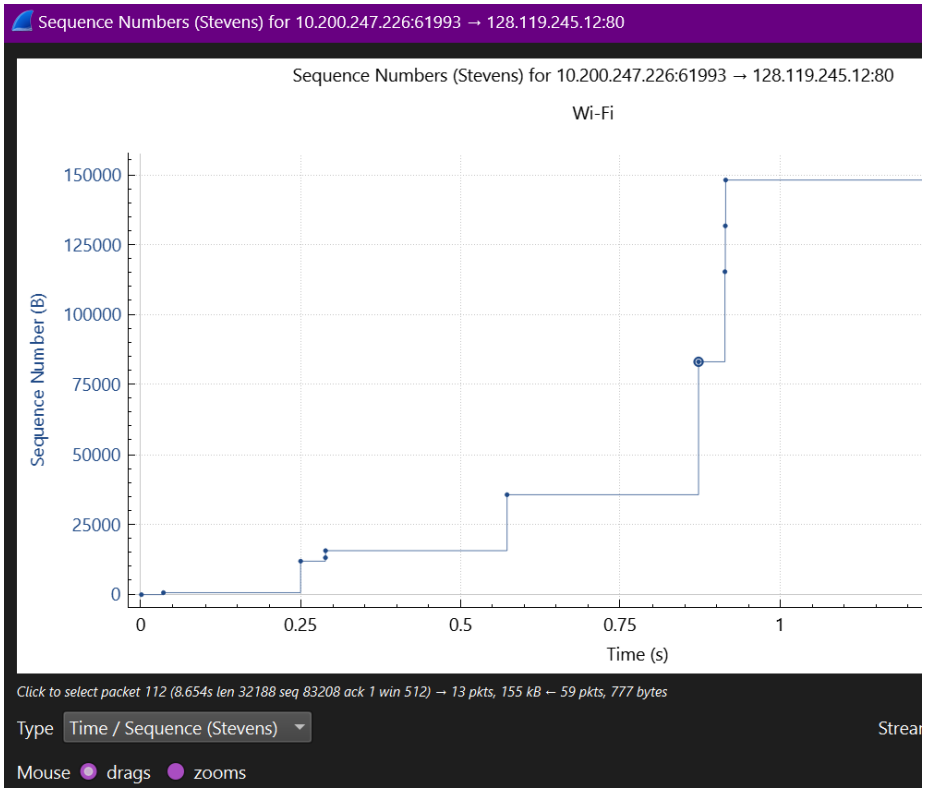


Figure 12: Q4. Time-Sequence-Graph (Stevens)