CSCE 47503 Computer Networks – Wireshark TCP

Name: Blake Williams
ID: 010974718

35 points
11 questions

Instructions

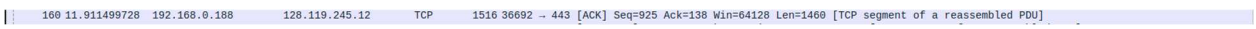
- Type your work, print it to a **single** PDF, and upload it to Blackboard before the due date and time. It is strongly suggested that you use the given document.
- Show all of your work. Correct answers alone may not carry full credit without proper justification and details of steps.
- -2 points if you do not insert your name and ID at the top of the document.
- -5 points if it is not typed or legible. For this homework, you may scan it with something like the CamScanner app, but just make sure it is a legible PDF.
- -5 points if it is not a PDF file.
- -5 points if it is not a single PDF file. Submit one PDF file. Do not submit zip files containing one or more files.
- -5 points if you present the worked problems out of order. In other words, please present the problems in the order assigned, 1, 2, 3, …

Please use the *tcp-ethereal-trace-1* pcap file provided to you to answer all the questions **except for question 3.**

1. [2 pts.] What is the IP address and TCP port number used by the client computer (source) that transfers the 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).
   IP ADDR = 192.168.1.102
   PORT = 1161

2. [2 pts.] What is the IP address of gaia.cs.umass.edu? What port number is it sending and receiving TCP segments for this connection?
   IP ADDR = 128.119.245.12
   PORT = 80

3. [3 pts.] For your own trace, what is the IP address and TCP port number used by your client computer (source) to transfer the file to gaia.cs.umass.edu? Please attach a screenshot or a picture, e.g., to print a packet, use File->Print, choose Selected packet only, choose Packet summary line, and select the minimum amount of packet detail you need to answer the question.

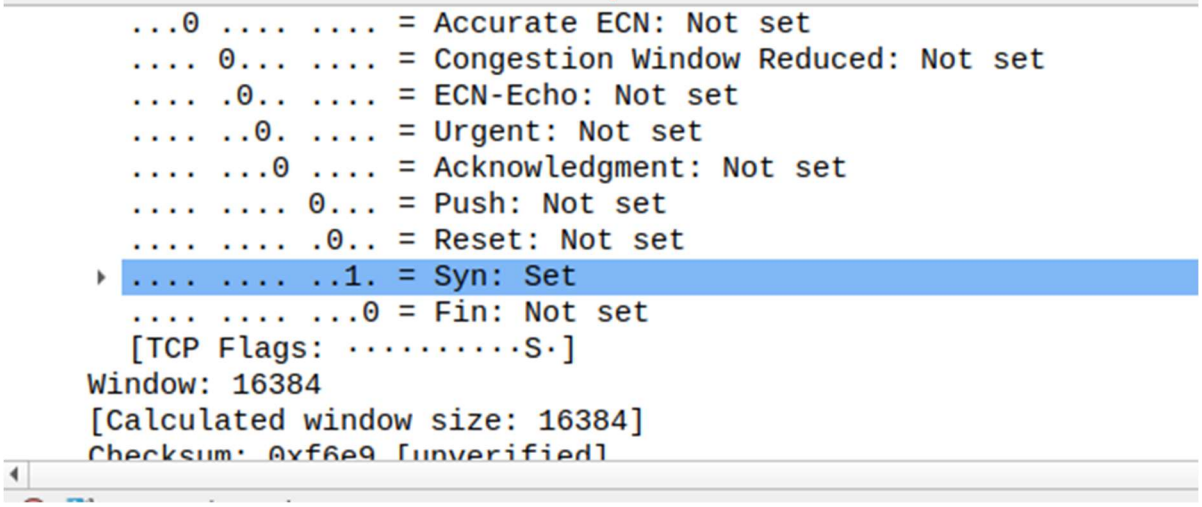


Client IP = 192.168.0.188

Port = 36692

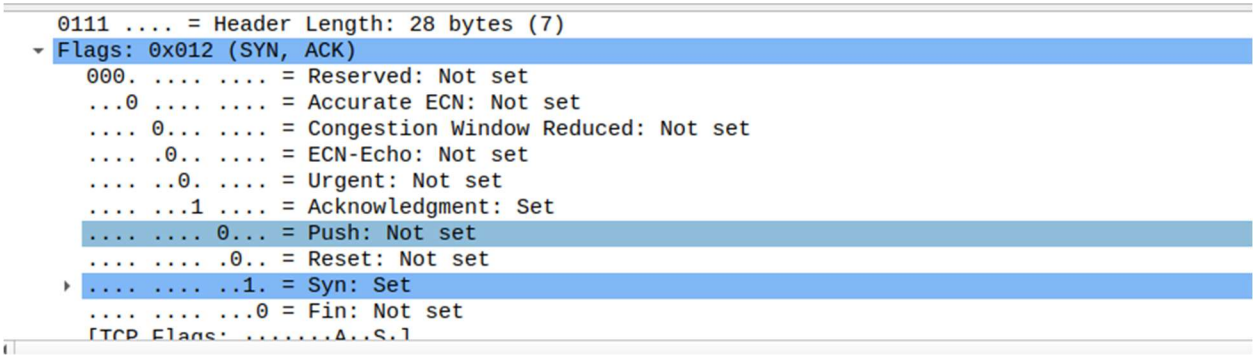
4. [3 pts.] What is the sequence number of the TCP SYN segment used to initiate the TCP connection between the client computer and gaia.cs.umass.edu? What is it in the segment that identifies it as an SYN segment? Please attach a screenshot with the corresponding packet(s) highlighted.

   a. Sequence = 0
   b. You know it is a SYN because the SYN flag is set



5. [3 pts.] 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? Please attach a screenshot with the corresponding packet(s) highlighted.

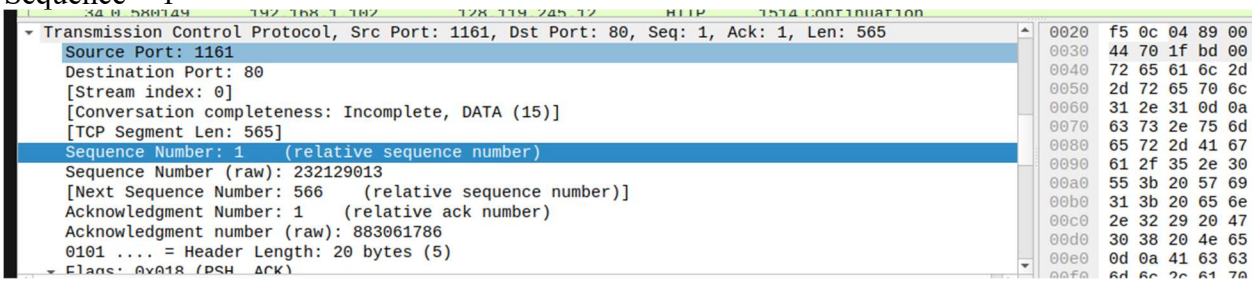
   a. Sequence = 0
   b. You know It is a SYNACK because the SYN and ACK flags are set



6. [2 pts.] What is the sequence number of the TCP segment containing the HTTP POST command? To find the POST command, dig into the packet content field at the bottom of the Wireshark window and look for a segment with a "POST" within its DATA field. Please attach a screenshot with the corresponding packet(s) highlighted.

a. Sequence = 1

b.



7. 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 TCP connection segments (including the HTTP POST segment)? 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 its acknowledgment 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 receiving 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 TCP segment sent. Select a TCP segment in the "listing of captured packets" window sent from the client to the gaia.cs.umass.edu server. Then select *Statistics->TCP Stream Graph->Round Trip Time Graph.*

Answer:

[2 pts.] The HTTP POST segment is considered as the first segment. Segments 1 – 6 are No. 4__, _5_, _7_, _8_, _10_, and _11_ in this trace respectively. The ACKs of segments 1 – 6 are No. _6_, _9_, _12_, _14_, _15_, and _16_ in this trace.

[2 pts.] Fill in the sequence number for Segments 1– 6 (use relative sequence# instead of raw sequence#).

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__

[3 pts.] The sending time and the received time of ACKs are tabulated in the following table.

	Sent time	**ACK received time**	**RTT (seconds)**
Segment 1	`08:44:20.596858000`	`08:44:20.624318000`	`0.027460000`

Segment 2	08:44:20.612118000	08:44:20.647675000	0.035557000
Segment 3	08:44:20.624407000	08:44:20.694466000	0.070059000
Segment 4	08:44:20.625071000	08:44:20.739499000	0.114428000
Segment 5	08:44:20.647786000	08:44:20.787680000	0.139894000
Segment 6	08:44:20.648538000	08:44:20.838183000	0.189645000

[3 pts.] `EstimatedRTT = 0.875 * EstimatedRTT + 0.125 * SampleRTT`

`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 =` _____0.28594625_____

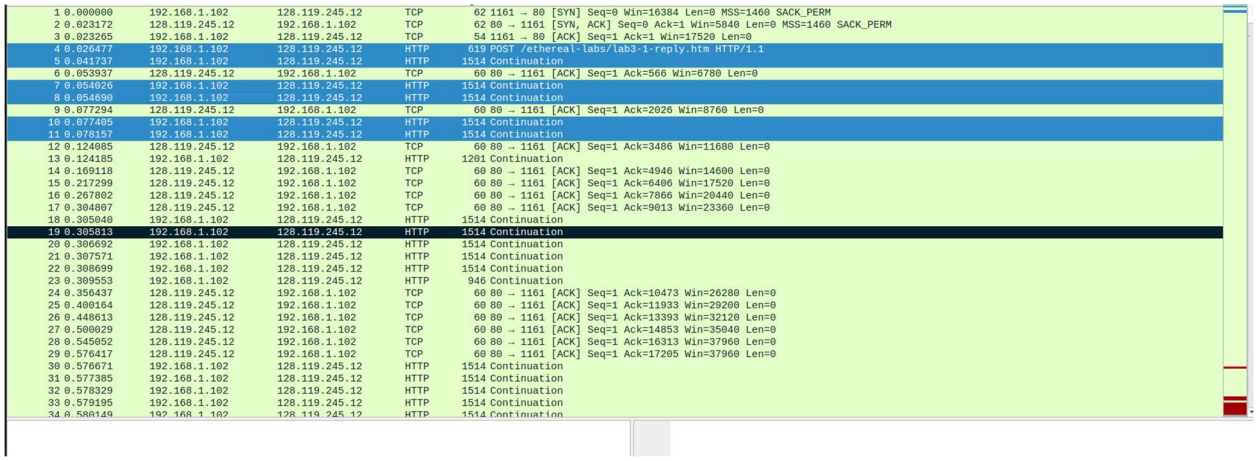
`EstimatedRTT` after the receipt of the ACK of segment 3:
`EstimatedRTT =` _____ 0.0294649219_____

`EstimatedRTT` after the receipt of the ACK of segment 4:
`EstimatedRTT =` _____ .0345391816_____

`EstimatedRTT` after the receipt of the ACK of segment 5:
`EstimatedRTT =` _____ 0.0445252839_____

`EstimatedRTT` after the receipt of the ACK of segment 6:
`EstimatedRTT =` _____ 0.0564463734_____

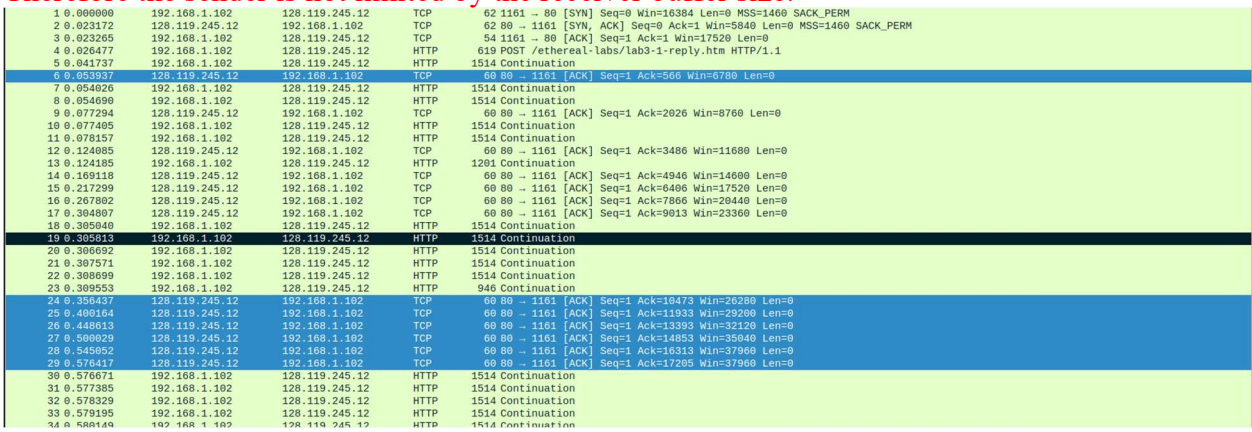
8. [3 pts.] What is the length of each of the first six TCP segments? Please attach a screenshot with the corresponding packet(s) highlighted.



1	0.000000	192.168.1.102	128.119.245.12	TCP	62 1161 → 80 [SYN] Seq=0 Win=16384 Len=0 MSS=1460 SACK_PERM
2	0.023172	128.119.245.12	192.168.1.102	TCP	62 80 → 1161 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 SACK_PERM
3	0.023265	192.168.1.102	128.119.245.12	TCP	54 1161 → 80 [ACK] Seq=1 Ack=1 Win=17520 Len=0
4	0.026477	192.168.1.102	128.119.245.12	HTTP	619 POST /ethereal-labs/lab3-1-reply.htm HTTP/1.1
5	0.041737	192.168.1.102	128.119.245.12	HTTP	1514 Continuation
6	0.053937	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=566 Win=6780 Len=0
7	0.054026	192.168.1.102	128.119.245.12	HTTP	1514 Continuation
8	0.054690	192.168.1.102	128.119.245.12	HTTP	1514 Continuation
9	0.077294	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=2026 Win=8760 Len=0
10	0.077405	192.168.1.102	128.119.245.12	HTTP	1514 Continuation
11	0.078157	192.168.1.102	128.119.245.12	HTTP	1514 Continuation
12	0.124085	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=3486 Win=11680 Len=0
13	0.124185	192.168.1.102	128.119.245.12	HTTP	1201 Continuation
14	0.169118	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=4946 Win=14600 Len=0
15	0.217299	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=6406 Win=17520 Len=0
16	0.267802	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=7866 Win=20440 Len=0
17	0.304807	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=9013 Win=23360 Len=0
18	0.305040	192.168.1.102	128.119.245.12	HTTP	1514 Continuation
19	0.305813	192.168.1.102	128.119.245.12	HTTP	1514 Continuation
20	0.306692	192.168.1.102	128.119.245.12	HTTP	1514 Continuation
21	0.307571	192.168.1.102	128.119.245.12	HTTP	1514 Continuation
22	0.308699	192.168.1.102	128.119.245.12	HTTP	1514 Continuation
23	0.309553	192.168.1.102	128.119.245.12	HTTP	946 Continuation
24	0.356437	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=10473 Win=26280 Len=0
25	0.400164	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=11933 Win=29200 Len=0
26	0.448613	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=13393 Win=32120 Len=0
27	0.500029	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=14853 Win=35040 Len=0
28	0.545052	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=16313 Win=37960 Len=0
29	0.576417	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=17205 Win=37960 Len=0
30	0.576671	192.168.1.102	128.119.245.12	HTTP	1514 Continuation
31	0.577385	192.168.1.102	128.119.245.12	HTTP	1514 Continuation
32	0.578329	192.168.1.102	128.119.245.12	HTTP	1514 Continuation
33	0.579195	192.168.1.102	128.119.245.12	HTTP	1514 Continuation
34	0.580149	192.168.1.102	128.119.245.12	HTTP	1514 Continuation

First TCP = 619
2-6 TCP = 1514

9. [3 pts.] What is the minimum available buffer space advertised at the receiver for the entire trace? Does the lack of receiver buffer space ever throttle the sender? Please attach a screenshot with the corresponding packet(s) highlighted.
    a. 5840 is the minimum available buffer space advertised.
    b. No, because the advertised window sizes are larger than the lower threshold. Therefore the sender is not limited by the receiver buffer size.



1	0.000000	192.168.1.102	128.119.245.12	TCP	62 1161 → 80 [SYN] Seq=0 Win=16384 Len=0 MSS=1460 SACK_PERM
2	0.023172	128.119.245.12	192.168.1.102	TCP	62 80 → 1161 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 SACK_PERM
3	0.023265	192.168.1.102	128.119.245.12	TCP	54 1161 → 80 [ACK] Seq=1 Ack=1 Win=17520 Len=0
4	0.026477	192.168.1.102	128.119.245.12	HTTP	619 POST /ethereal-labs/lab3-1-reply.htm HTTP/1.1
5	0.041737	192.168.1.102	128.119.245.12	HTTP	1514 Continuation
6	0.053937	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=566 Win=6780 Len=0
7	0.054026	192.168.1.102	128.119.245.12	HTTP	1514 Continuation
8	0.054690	192.168.1.102	128.119.245.12	HTTP	1514 Continuation
9	0.077294	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=2026 Win=8760 Len=0
10	0.077405	192.168.1.102	128.119.245.12	HTTP	1514 Continuation
11	0.078157	192.168.1.102	128.119.245.12	HTTP	1514 Continuation
12	0.124085	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=3486 Win=11680 Len=0
13	0.124185	192.168.1.102	128.119.245.12	HTTP	1201 Continuation
14	0.169118	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=4946 Win=14600 Len=0
15	0.217299	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=6406 Win=17520 Len=0
16	0.267802	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=7866 Win=20440 Len=0
17	0.304807	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=9013 Win=23360 Len=0
18	0.305040	192.168.1.102	128.119.245.12	HTTP	1514 Continuation
19	0.305813	192.168.1.102	128.119.245.12	HTTP	1514 Continuation
20	0.306692	192.168.1.102	128.119.245.12	HTTP	1514 Continuation
21	0.307571	192.168.1.102	128.119.245.12	HTTP	1514 Continuation
22	0.308699	192.168.1.102	128.119.245.12	HTTP	1514 Continuation
23	0.309553	192.168.1.102	128.119.245.12	HTTP	946 Continuation
24	0.356437	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=10473 Win=26280 Len=0
25	0.400164	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=11933 Win=29200 Len=0
26	0.448613	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=13393 Win=32120 Len=0
27	0.500029	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=14853 Win=35040 Len=0
28	0.545052	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=16313 Win=37960 Len=0
29	0.576417	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=17205 Win=37960 Len=0
30	0.576671	192.168.1.102	128.119.245.12	HTTP	1514 Continuation
31	0.577385	192.168.1.102	128.119.245.12	HTTP	1514 Continuation
32	0.578329	192.168.1.102	128.119.245.12	HTTP	1514 Continuation
33	0.579195	192.168.1.102	128.119.245.12	HTTP	1514 Continuation
34	0.580149	192.168.1.102	128.119.245.12	HTTP	1514 Continuation

10. [2 pts.] Are there any retransmitted segments in the trace file? What did you check for (in the trace) to answer this question?
    i. No, you can check ack values for duplicate values and there are not any.

11. [2 pts.] What is the TCP connection's throughput (bytes transferred per unit time)? Explain how you calculated this value.
    a. 164091/(26.026211 –20.596858)
    b. 30222.9381659