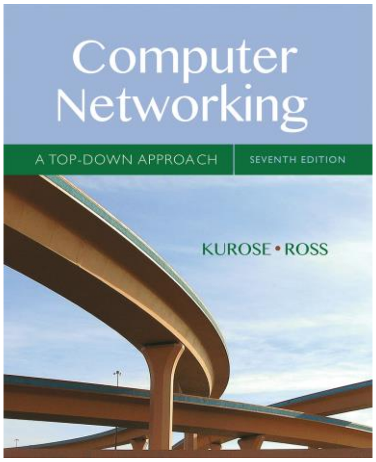
Name: ________Timur Guner___________

Wireshark Lab: TCP v7.0

Supplement to *Computer Networking: A Top-Down Approach, 7th ed.,* J.F. Kurose and K.W. Ross

"Tell me and I forget. Show me and I remember. Involve me and I understand." Chinese proverb



In this lab, we'll investigate the behavior of the celebrated TCP protocol in detail. We'll do so by analyzing a trace of the TCP segments sent and received in transferring a 150KB file (containing the text of Lewis Carrol's *Alice's Adventures in Wonderland*) from your computer to a remote server. We'll study TCP's use of sequence and acknowledgement numbers for providing reliable data transfer; we'll see TCP's congestion control algorithm – slow start and congestion avoidance – in action; and we'll look at TCP's receiver-advertised flow control mechanism. We'll also briefly consider TCP connection setup and we'll investigate the performance (throughput and round-trip time) of the TCP connection between your computer and the server.

Before beginning this lab, you'll probably want to review sections 3.5 and 3.7 in the text[1].

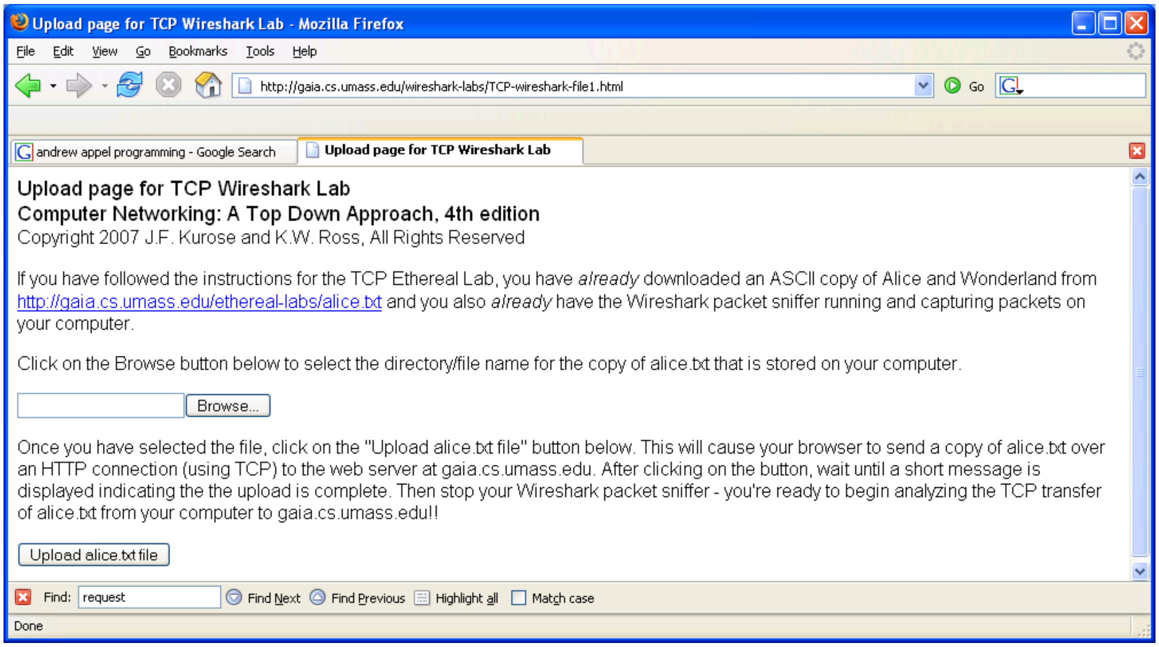
1. Capturing a bulk TCP transfer from your computer to a remote server

Before beginning our exploration of TCP, we'll need to use Wireshark to obtain a packet trace of the TCP transfer of a file from your computer to a remote server. You'll do so by accessing a Web page that will allow you to enter the name of a file stored on your computer (which contains the ASCII text of *Alice in Wonderland*), and then transfer the file to a Web server using the HTTP POST method (see section 2.2.3 in the text). We're using the POST method rather than the GET method as we'd like to transfer a large amount of data *from* your computer to another computer. Of course, we'll be running Wireshark during this time to obtain the trace of the TCP segments sent and received from your computer.

[1] References to figures and sections are for the 7th edition of our text, *Computer Networks, A Top-down Approach, 7th ed., J.F. Kurose and K.W. Ross, Addison-Wesley/Pearson, 2016.*

Do the following:

- Start up your web browser. Go the http://gaia.cs.umass.edu/wireshark-labs/alice.txt and retrieve an ASCII copy of *Alice in Wonderland.* Store this file somewhere on your computer.
- Next go to http://gaia.cs.umass.edu/wireshark-labs/TCP-wireshark-file1.html.
- You should see a screen that looks like:


Upload page for TCP Wireshark Lab - Mozilla Firefox
File
Edit
View
Go
Bookmarks
Tools
Help
Go
andrew appel programming - Google Search
Upload page for TCP Wireshark Lab
Browse...
Upload alice.txt file
Find Next
Find Previous
Highlight all
Match case
Done

http://gaia.cs.umass.edu/wireshark-labs/TCP-wireshark-file1.html

Upload page for TCP Wireshark Lab
Computer Networking: A Top Down Approach, 4th edition
Copyright 2007 J.F. Kurose and K.W. Ross, All Rights Reserved

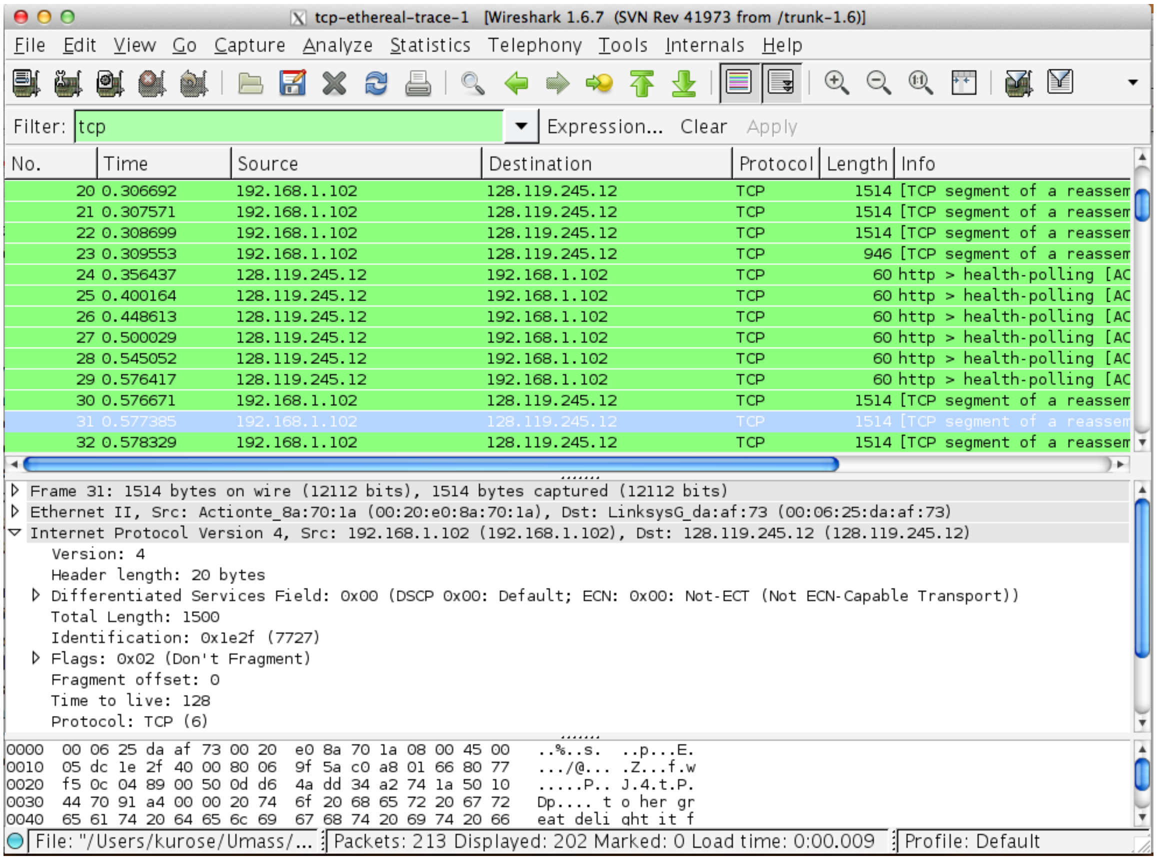
If you have followed the instructions for the TCP Ethereal Lab, you have *already* downloaded an ASCII copy of Alice and Wonderland from http://gaia.cs.umass.edu/ethereal-labs/alice.txt and you also *already* have the Wireshark packet sniffer running and capturing packets on your computer.

Click on the Browse button below to select the directory/file name for the copy of alice.txt that is stored on your computer.

Once you have selected the file, click on the "Upload alice.txt file" button below. This will cause your browser to send a copy of alice.txt over an HTTP connection (using TCP) to the web server at gaia.cs.umass.edu. After clicking on the button, wait until a short message is displayed indicating the the upload is complete. Then stop your Wireshark packet sniffer - you're ready to begin analyzing the TCP transfer of alice.txt from your computer to gaia.cs.umass.edu!!

Find: request

- Use the *Browse* button in this form to enter the name of the file (full path name) on your computer containing *Alice in Wonderland* (or do so manually). Don't yet press the "*Upload alice.txt file*" button.
- Now start up Wireshark and begin packet capture *(Capture->Start)* and then press *OK* on the Wireshark Packet Capture Options screen (we'll not need to select any options here).
- Returning to your browser, press the "*Upload alice.txt file*" button to upload the file to the gaia.cs.umass.edu server. Once the file has been uploaded, a short congratulations message will be displayed in your browser window.
- Stop Wireshark packet capture. Your Wireshark window should look similar to the window shown below.


tcp-ethereal-trace-1 [Wireshark 1.6.7 (SVN Rev 41973 from /trunk-1.6)]
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Expression... Clear Apply

Filter: tcp

```
No.   Time       Source          Destination     Protocol Length Info
   20 0.306692   192.168.1.102   128.119.245.12  TCP        1514 [TCP segment of a reassem
   21 0.307571   192.168.1.102   128.119.245.12  TCP        1514 [TCP segment of a reassem
   22 0.308699   192.168.1.102   128.119.245.12  TCP        1514 [TCP segment of a reassem
   23 0.309553   192.168.1.102   128.119.245.12  TCP         946 [TCP segment of a reassem
   24 0.356437   128.119.245.12  192.168.1.102   TCP          60 http > health-polling [AC
   25 0.400164   128.119.245.12  192.168.1.102   TCP          60 http > health-polling [AC
   26 0.448613   128.119.245.12  192.168.1.102   TCP          60 http > health-polling [AC
   27 0.500029   128.119.245.12  192.168.1.102   TCP          60 http > health-polling [AC
   28 0.545052   128.119.245.12  192.168.1.102   TCP          60 http > health-polling [AC
   29 0.576417   128.119.245.12  192.168.1.102   TCP          60 http > health-polling [AC
   30 0.576671   192.168.1.102   128.119.245.12  TCP        1514 [TCP segment of a reassem
   31 0.577385   192.168.1.102   128.119.245.12  TCP        1514 [TCP segment of a reassem
   32 0.578329   192.168.1.102   128.119.245.12  TCP        1514 [TCP segment of a reassem
```

```
▷ Frame 31: 1514 bytes on wire (12112 bits), 1514 bytes captured (12112 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 (192.168.1.102), Dst: 128.119.245.12 (128.119.245.12)
    Version: 4
    Header length: 20 bytes
  ▷ Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00: Not-ECT (Not ECN-Capable Transport))
    Total Length: 1500
    Identification: 0x1e2f (7727)
  ▷ Flags: 0x02 (Don't Fragment)
    Fragment offset: 0
    Time to live: 128
    Protocol: TCP (6)
```

```
0000  00 06 25 da af 73 00 20  e0 8a 70 1a 08 00 45 00   ..%..s.  ..p...E.
0010  05 dc 1e 2f 40 00 80 06  9f 5a c0 a8 01 66 80 77   .../@... .Z...f.w
0020  f5 0c 04 89 00 50 0d d6  4a dd 34 a2 74 1a 50 10   .....P.. J.4.t.P.
0030  44 70 91 a4 00 00 20 74  6f 20 68 65 72 20 67 72   Dp.... t o her gr
0040  65 61 74 20 64 65 6c 69  67 68 74 20 69 74 20 66   eat deli ght it f
```

File: "/Users/kurose/Umass/... Packets: 213 Displayed: 202 Marked: 0 Load time: 0:00.009 Profile: Default

If you are unable to run Wireshark on a live network connection, you can download a packet trace file that was captured while following the steps above on one of the author's computers[2]. You may well find it valuable to download this trace even if you've captured your own trace and use it, as well as your own trace, when you explore the questions below.

2. A first look at the captured trace

Before analyzing the behavior of the TCP connection in detail, let's take a high level view of the trace.

- First, filter the packets displayed in the Wireshark window by entering "tcp" (lowercase, no quotes, and don't forget to press return after entering!) into the display filter specification window towards the top of the Wireshark window.

What you should see is series of TCP and HTTP messages between your computer and gaia.cs.umass.edu. You should see the initial three-way handshake containing a SYN message. You should see an HTTP POST message. Depending on the version of

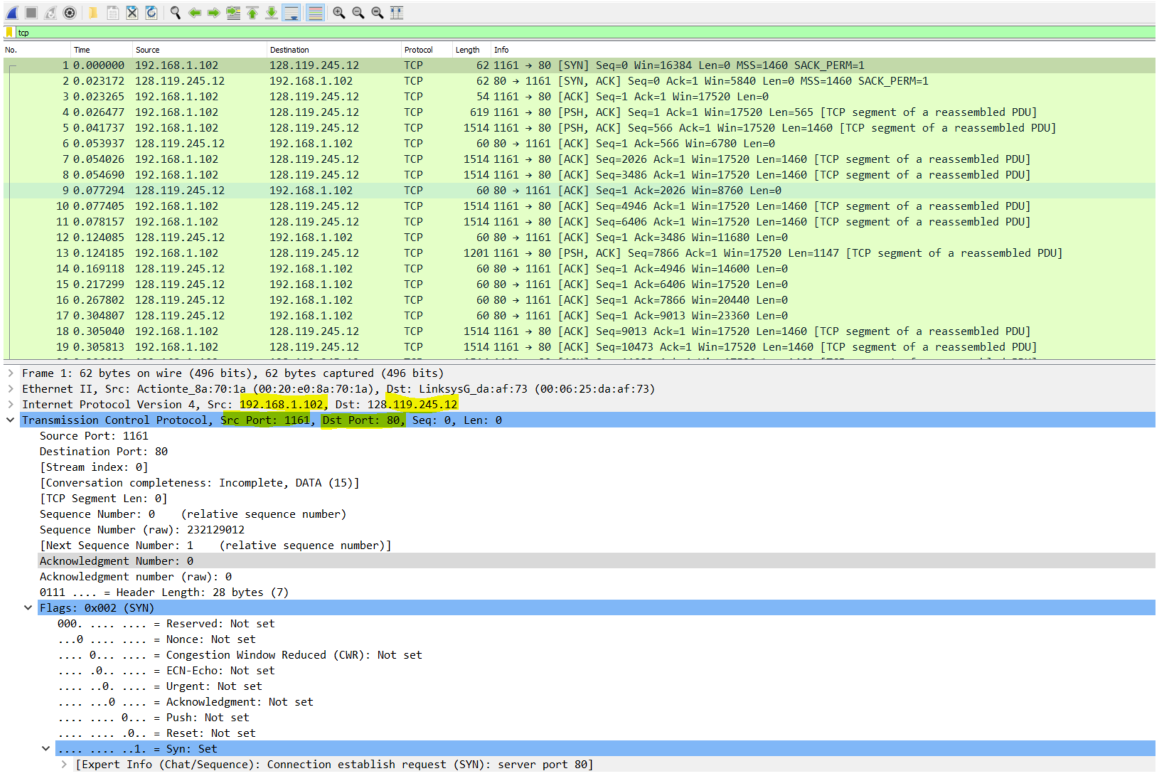
[2] Download the zip file http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip and extract the file tcp-ethereal-trace-1. The traces in this zip file were collected by Wireshark running on one of the author's computers, while performing the steps indicated in the Wireshark lab. Once you have downloaded the trace, you can load it into Wireshark and view the trace using the *File* pull down menu, choosing *Open*, and then selecting the tcp-ethereal-trace-1 trace file.

Wireshark you are using, you might see a series of "HTTP Continuation" messages being sent from your computer to gaia.cs.umass.edu. Recall from our discussion in the earlier HTTP Wireshark lab, that is no such thing as an HTTP Continuation message – this is Wireshark's way of indicating that there are multiple TCP segments being used to carry a single HTTP message. In more recent versions of Wireshark, you'll see "[TCP segment of a reassembled PDU]" in the Info column of the Wireshark display to indicate that this TCP segment contained data that belonged to an upper layer protocol message (in our case here, HTTP). You should also see TCP ACK segments being returned from gaia.cs.umass.edu to your computer.

Answer the following questions, by opening the Wireshark captured packet file *tcp-ethereal-trace-1* in http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip (that is download the trace and open that trace in Wireshark; see footnote 2). Whenever possible, when answering a question you should include a screenshot of the packet(s) within the trace that you used to answer the question asked. Make sure to include in the screenshot ALL and ONLY the minimum amount of packet detail that you need to answer the question.

1. 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? 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 / PORT: 192.168.1.102:1161
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?
   IP / PORT: 128.119.245.12:80

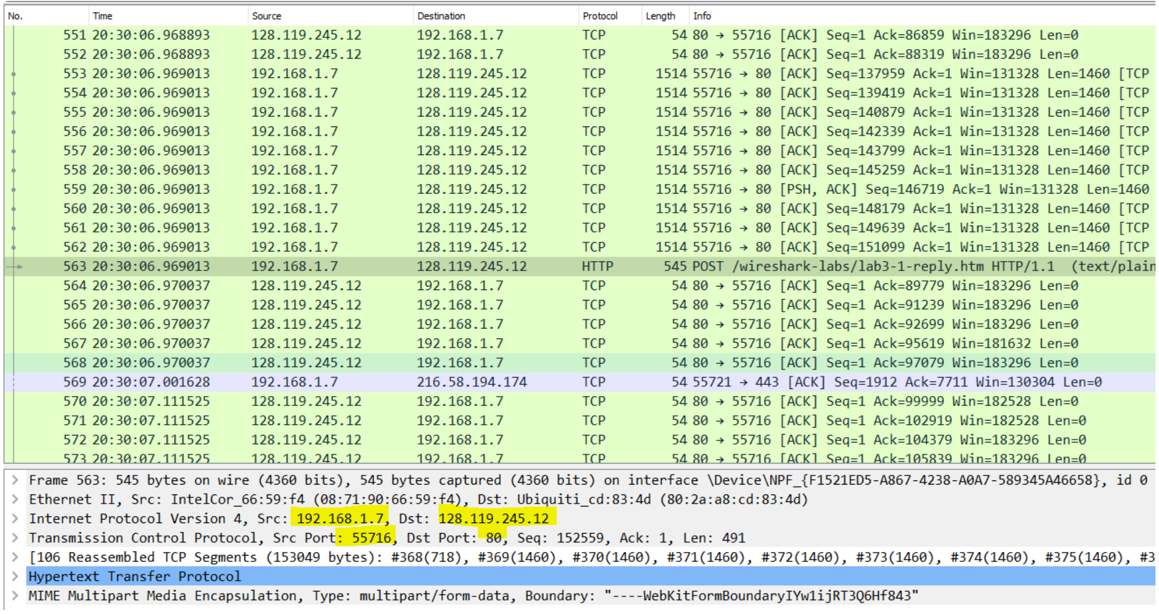
   Screenshot for first two questions



```
tcp
No.  Time      Source          Destination     Protocol Length Info
   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=1
   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=1
   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  TCP       619 1161 → 80 [PSH, ACK] Seq=1 Ack=1 Win=17520 Len=565 [TCP segment of a reassembled PDU]
   5 0.041737  192.168.1.102   128.119.245.12  TCP      1514 1161 → 80 [PSH, ACK] Seq=566 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
   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  TCP      1514 1161 → 80 [ACK] Seq=2026 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
   8 0.054690  192.168.1.102   128.119.245.12  TCP      1514 1161 → 80 [ACK] Seq=3486 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
   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  TCP      1514 1161 → 80 [ACK] Seq=4946 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  11 0.078157  192.168.1.102   128.119.245.12  TCP      1514 1161 → 80 [ACK] Seq=6406 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  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  TCP      1201 1161 → 80 [PSH, ACK] Seq=7866 Ack=1 Win=17520 Len=1147 [TCP segment of a reassembled PDU]
  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  TCP      1514 1161 → 80 [ACK] Seq=9013 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  19 0.305813  192.168.1.102   128.119.245.12  TCP      1514 1161 → 80 [ACK] Seq=10473 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]

> Frame 1: 62 bytes on wire (496 bits), 62 bytes captured (496 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
v Transmission Control Protocol, Src Port: 1161, Dst Port: 80, Seq: 0, Len: 0
    Source Port: 1161
    Destination Port: 80
    [Stream index: 0]
    [Conversation completeness: Incomplete, DATA (15)]
    [TCP Segment Len: 0]
    Sequence Number: 0    (relative sequence number)
    Sequence Number (raw): 232129012
    [Next Sequence Number: 1    (relative sequence number)]
    Acknowledgment Number: 0
    Acknowledgment number (raw): 0
    0111 .... = Header Length: 28 bytes (7)
  v 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
    v .... .... ..1. = Syn: Set
      > [Expert Info (Chat/Sequence): Connection establish request (SYN): server port 80]
```

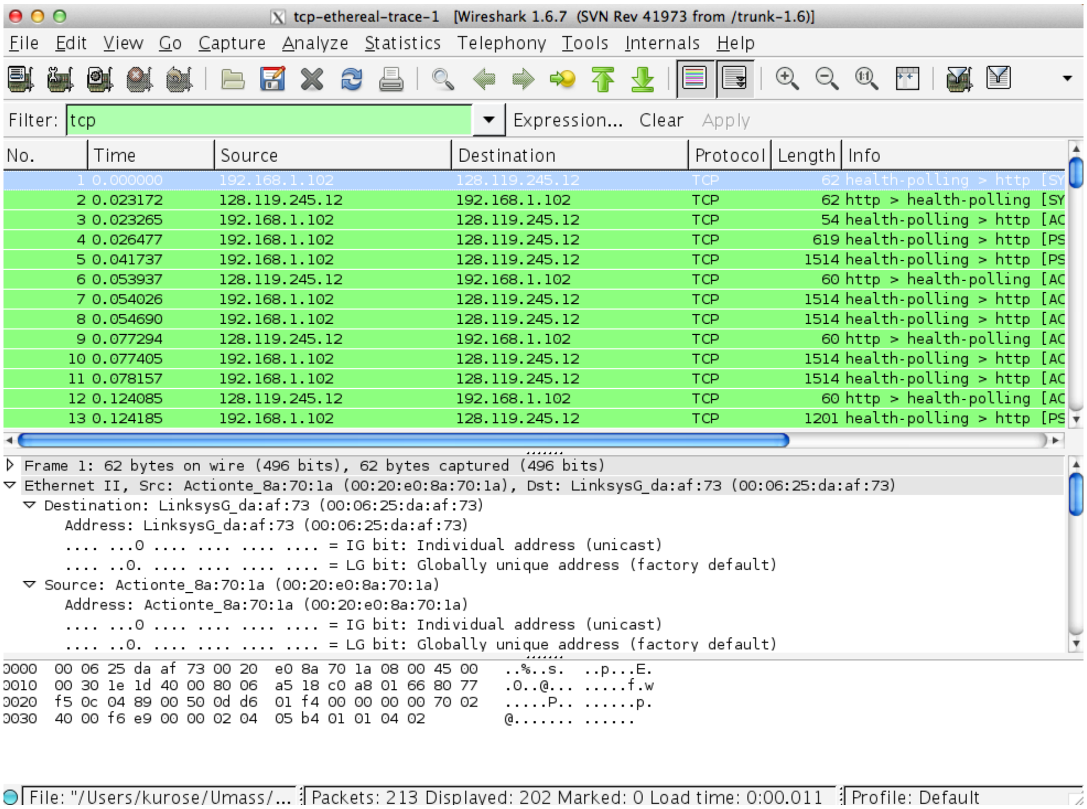
If you have been able to create your own trace, answer the following question:

3. What is the IP address and TCP port number used by your client computer (source) to transfer the file to gaia.cs.umass.edu?
   IP / PORT: 192.168.1.7:55716



```
No.  Time             Source          Destination     Protocol Length Info
 551 20:30:06.968893  128.119.245.12  192.168.1.7     TCP        54 80 → 55716 [ACK] Seq=1 Ack=86859 Win=183296 Len=0
 552 20:30:06.968893  128.119.245.12  192.168.1.7     TCP        54 80 → 55716 [ACK] Seq=1 Ack=88319 Win=183296 Len=0
 553 20:30:06.969013  192.168.1.7     128.119.245.12  TCP      1514 55716 → 80 [ACK] Seq=137959 Ack=1 Win=131328 Len=1460 [TCP
 554 20:30:06.969013  192.168.1.7     128.119.245.12  TCP      1514 55716 → 80 [ACK] Seq=139419 Ack=1 Win=131328 Len=1460 [TCP
 555 20:30:06.969013  192.168.1.7     128.119.245.12  TCP      1514 55716 → 80 [ACK] Seq=140879 Ack=1 Win=131328 Len=1460 [TCP
 556 20:30:06.969013  192.168.1.7     128.119.245.12  TCP      1514 55716 → 80 [ACK] Seq=142339 Ack=1 Win=131328 Len=1460 [TCP
 557 20:30:06.969013  192.168.1.7     128.119.245.12  TCP      1514 55716 → 80 [ACK] Seq=143799 Ack=1 Win=131328 Len=1460 [TCP
 558 20:30:06.969013  192.168.1.7     128.119.245.12  TCP      1514 55716 → 80 [ACK] Seq=145259 Ack=1 Win=131328 Len=1460 [TCP
 559 20:30:06.969013  192.168.1.7     128.119.245.12  TCP      1514 55716 → 80 [PSH, ACK] Seq=146719 Ack=1 Win=131328 Len=1460
 560 20:30:06.969013  192.168.1.7     128.119.245.12  TCP      1514 55716 → 80 [ACK] Seq=148179 Ack=1 Win=131328 Len=1460 [TCP
 561 20:30:06.969013  192.168.1.7     128.119.245.12  TCP      1514 55716 → 80 [ACK] Seq=149639 Ack=1 Win=131328 Len=1460 [TCP
 562 20:30:06.969013  192.168.1.7     128.119.245.12  TCP      1514 55716 → 80 [ACK] Seq=151099 Ack=1 Win=131328 Len=1460 [TCP
 563 20:30:06.969013  192.168.1.7     128.119.245.12  HTTP      545 POST /wireshark-labs/lab3-1-reply.htm HTTP/1.1  (text/plain
 564 20:30:06.970037  128.119.245.12  192.168.1.7     TCP        54 80 → 55716 [ACK] Seq=1 Ack=89779 Win=183296 Len=0
 565 20:30:06.970037  128.119.245.12  192.168.1.7     TCP        54 80 → 55716 [ACK] Seq=1 Ack=91239 Win=183296 Len=0
 566 20:30:06.970037  128.119.245.12  192.168.1.7     TCP        54 80 → 55716 [ACK] Seq=1 Ack=92699 Win=183296 Len=0
 567 20:30:06.970037  128.119.245.12  192.168.1.7     TCP        54 80 → 55716 [ACK] Seq=1 Ack=95619 Win=181632 Len=0
 568 20:30:06.970037  128.119.245.12  192.168.1.7     TCP        54 80 → 55716 [ACK] Seq=1 Ack=97079 Win=183296 Len=0
 569 20:30:07.001628  192.168.1.7     216.58.194.174  TCP        54 55721 → 443 [ACK] Seq=1912 Ack=7711 Win=130304 Len=0
 570 20:30:07.111525  128.119.245.12  192.168.1.7     TCP        54 80 → 55716 [ACK] Seq=1 Ack=99999 Win=182528 Len=0
 571 20:30:07.111525  128.119.245.12  192.168.1.7     TCP        54 80 → 55716 [ACK] Seq=1 Ack=102919 Win=182528 Len=0
 572 20:30:07.111525  128.119.245.12  192.168.1.7     TCP        54 80 → 55716 [ACK] Seq=1 Ack=104379 Win=183296 Len=0
 573 20:30:07.111525  128.119.245.12  192.168.1.7     TCP        54 80 → 55716 [ACK] Seq=1 Ack=105839 Win=183296 Len=0

> Frame 563: 545 bytes on wire (4360 bits), 545 bytes captured (4360 bits) on interface \Device\NPF_{F1521ED5-A867-4238-A0A7-589345A46658}, id 0
> Ethernet II, Src: IntelCor_66:59:f4 (08:71:90:66:59:f4), Dst: Ubiquiti_cd:83:4d (80:2a:a8:cd:83:4d)
> Internet Protocol Version 4, Src: 192.168.1.7, Dst: 128.119.245.12
> Transmission Control Protocol, Src Port: 55716, Dst Port: 80, Seq: 152559, Ack: 1, Len: 491
> [106 Reassembled TCP Segments (153049 bytes): #368(718), #369(1460), #370(1460), #371(1460), #372(1460), #373(1460), #374(1460), #375(1460), #3
> Hypertext Transfer Protocol
> MIME Multipart Media Encapsulation, Type: multipart/form-data, Boundary: "----WebKitFormBoundaryIYw1ijRT3Q6Hf843"
```

Since this lab is about TCP rather than HTTP, let's change Wireshark's "listing of captured packets" window so that it shows information about the TCP segments containing the HTTP messages, rather than about the HTTP messages. To have Wireshark do this, select *Analyze->Enabled Protocols*. Then uncheck the HTTP box and select *OK*. You should now see a Wireshark window that looks like:


tcp-ethereal-trace-1 [Wireshark 1.6.7 (SVN Rev 41973 from /trunk-1.6)]
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Filter: tcp
Expression... Clear Apply

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.1.102	128.119.245.12	TCP	62	health-polling > http [SY
2	0.023172	128.119.245.12	192.168.1.102	TCP	62	http > health-polling [SY
3	0.023265	192.168.1.102	128.119.245.12	TCP	54	health-polling > http [AC
4	0.026477	192.168.1.102	128.119.245.12	TCP	619	health-polling > http [PS
5	0.041737	192.168.1.102	128.119.245.12	TCP	1514	health-polling > http [PS
6	0.053937	128.119.245.12	192.168.1.102	TCP	60	http > health-polling [AC
7	0.054026	192.168.1.102	128.119.245.12	TCP	1514	health-polling > http [AC
8	0.054690	192.168.1.102	128.119.245.12	TCP	1514	health-polling > http [AC
9	0.077294	128.119.245.12	192.168.1.102	TCP	60	http > health-polling [AC
10	0.077405	192.168.1.102	128.119.245.12	TCP	1514	health-polling > http [AC
11	0.078157	192.168.1.102	128.119.245.12	TCP	1514	health-polling > http [AC
12	0.124085	128.119.245.12	192.168.1.102	TCP	60	http > health-polling [AC
13	0.124185	192.168.1.102	128.119.245.12	TCP	1201	health-polling > http [PS

```
▷ Frame 1: 62 bytes on wire (496 bits), 62 bytes captured (496 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)
  ▽ Destination: LinksysG_da:af:73 (00:06:25:da:af:73)
      Address: LinksysG_da:af:73 (00:06:25:da:af:73)
      .... ...0 .... .... .... .... = IG bit: Individual address (unicast)
      .... ..0. .... .... .... .... = LG bit: Globally unique address (factory default)
  ▽ Source: Actionte_8a:70:1a (00:20:e0:8a:70:1a)
      Address: Actionte_8a:70:1a (00:20:e0:8a:70:1a)
      .... ...0 .... .... .... .... = IG bit: Individual address (unicast)
      .... ..0. .... .... .... .... = LG bit: Globally unique address (factory default)
```

```
0000  00 06 25 da af 73 00 20  e0 8a 70 1a 08 00 45 00   ..%..s.  ..p...E.
0010  00 30 1e 1d 40 00 80 06  a5 18 c0 a8 01 66 80 77   .0..@... .....f.w
0020  f5 0c 04 89 00 50 0d d6  01 f4 00 00 00 00 70 02   .....P.. ......p.
0030  40 00 f6 e9 00 00 02 04  05 b4 01 01 04 02         @....... ......
```

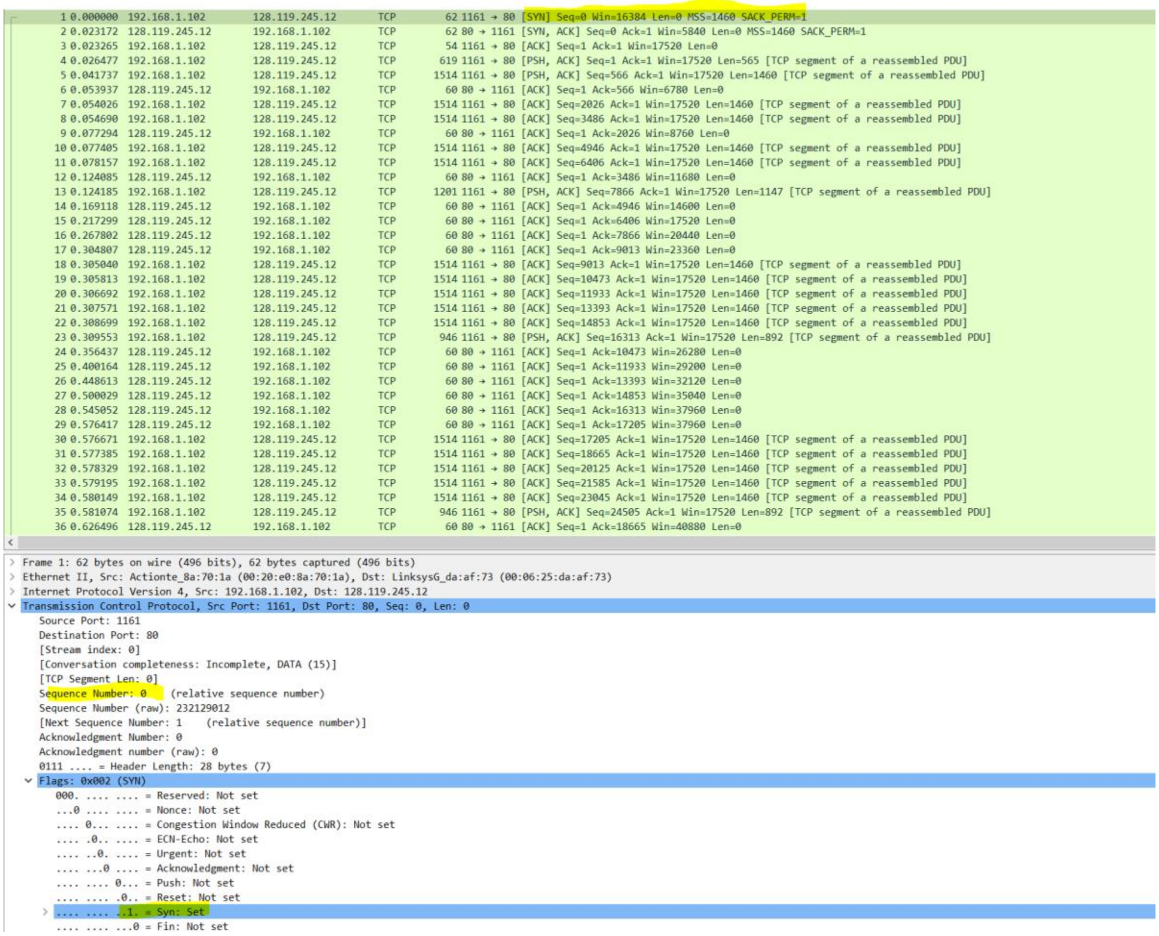
File: "/Users/kurose/Umass/... Packets: 213 Displayed: 202 Marked: 0 Load time: 0:00.011 Profile: Default

This is what we’re looking for - a series of TCP segments sent between your computer and gaia.cs.umass.edu. We will use the packet trace that you have captured (and/or the packet trace *tcp-ethereal-trace-1* in http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip; see earlier footnote) to study TCP behavior in the rest of this lab.

3. TCP Basics

Answer the following questions for the TCP segments:

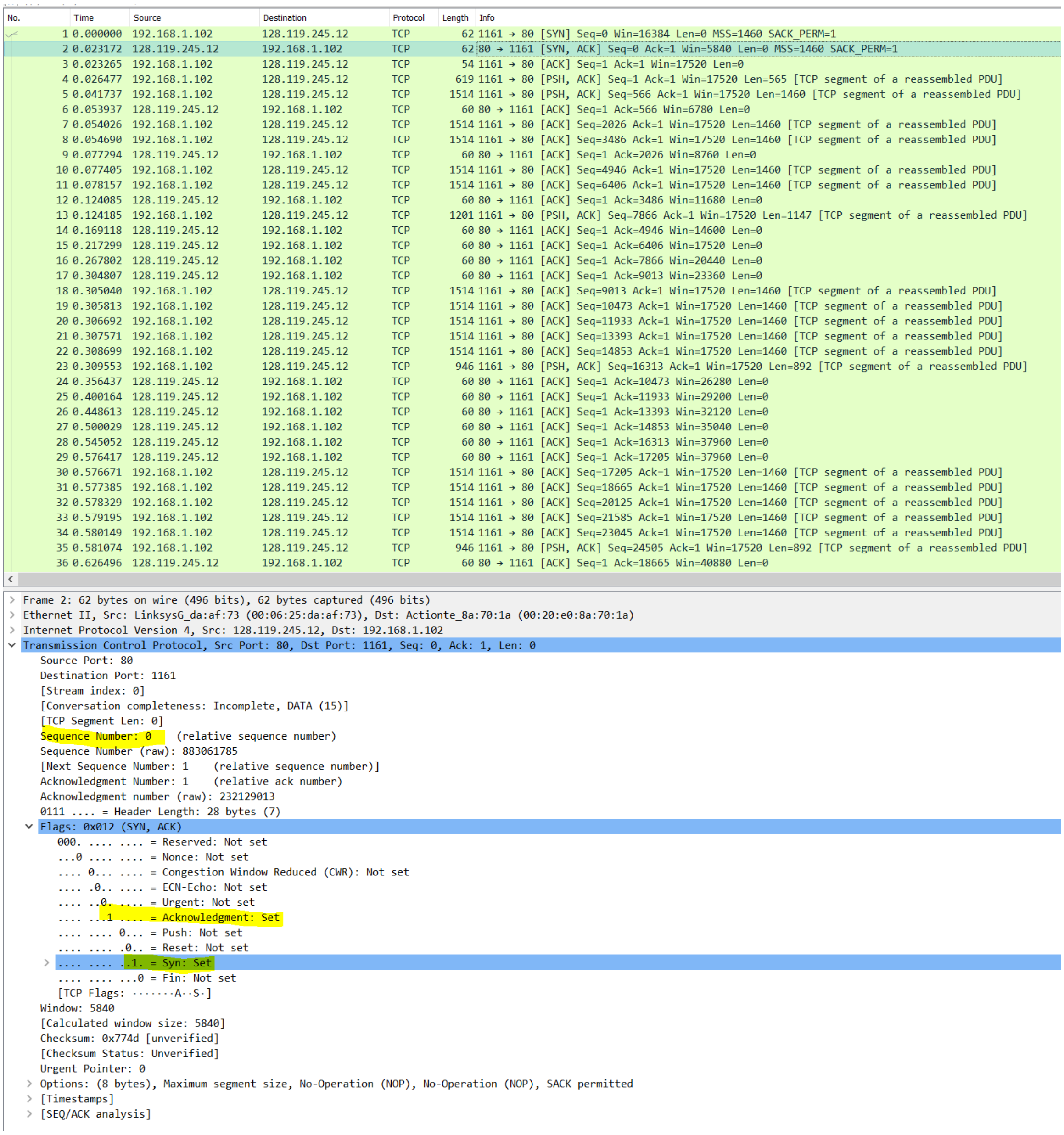
4. What is the sequence number of the TCP SYN segment that is used to initiate the TCP connection between the client computer and gaia.cs.umass.edu? What is it in the segment that identifies the segment as a SYN segment?
The sequence number is 0. The SYN flag is set to 1 to identify it as a SYN



```
    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=1
    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=1
    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    TCP     619 1161 → 80 [PSH, ACK] Seq=1 Ack=1 Win=17520 Len=565 [TCP segment of a reassembled PDU]
    5 0.041737  192.168.1.102     128.119.245.12    TCP    1514 1161 → 80 [PSH, ACK] Seq=566 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
    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    TCP    1514 1161 → 80 [ACK] Seq=2026 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
    8 0.054690  192.168.1.102     128.119.245.12    TCP    1514 1161 → 80 [ACK] Seq=3486 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
    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    TCP    1514 1161 → 80 [ACK] Seq=4946 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
   11 0.078157  192.168.1.102     128.119.245.12    TCP    1514 1161 → 80 [ACK] Seq=6406 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
   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    TCP    1201 1161 → 80 [PSH, ACK] Seq=7866 Ack=1 Win=17520 Len=1147 [TCP segment of a reassembled PDU]
   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    TCP    1514 1161 → 80 [ACK] Seq=9013 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
   19 0.305813  192.168.1.102     128.119.245.12    TCP    1514 1161 → 80 [ACK] Seq=10473 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
   20 0.306692  192.168.1.102     128.119.245.12    TCP    1514 1161 → 80 [ACK] Seq=11933 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
   21 0.307571  192.168.1.102     128.119.245.12    TCP    1514 1161 → 80 [ACK] Seq=13393 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
   22 0.308699  192.168.1.102     128.119.245.12    TCP    1514 1161 → 80 [ACK] Seq=14853 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
   23 0.309553  192.168.1.102     128.119.245.12    TCP     946 1161 → 80 [PSH, ACK] Seq=16313 Ack=1 Win=17520 Len=892 [TCP segment of a reassembled PDU]
   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    TCP    1514 1161 → 80 [ACK] Seq=17205 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
   31 0.577385  192.168.1.102     128.119.245.12    TCP    1514 1161 → 80 [ACK] Seq=18665 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
   32 0.578329  192.168.1.102     128.119.245.12    TCP    1514 1161 → 80 [ACK] Seq=20125 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
   33 0.579195  192.168.1.102     128.119.245.12    TCP    1514 1161 → 80 [ACK] Seq=21585 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
   34 0.580149  192.168.1.102     128.119.245.12    TCP    1514 1161 → 80 [ACK] Seq=23045 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
   35 0.581074  192.168.1.102     128.119.245.12    TCP     946 1161 → 80 [PSH, ACK] Seq=24505 Ack=1 Win=17520 Len=892 [TCP segment of a reassembled PDU]
   36 0.626496  128.119.245.12    192.168.1.102     TCP      60 80 → 1161 [ACK] Seq=1 Ack=18665 Win=40880 Len=0
```

```
> Frame 1: 62 bytes on wire (496 bits), 62 bytes captured (496 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
v Transmission Control Protocol, Src Port: 1161, Dst Port: 80, Seq: 0, Len: 0
    Source Port: 1161
    Destination Port: 80
    [Stream index: 0]
    [Conversation completeness: Incomplete, DATA (15)]
    [TCP Segment Len: 0]
    Sequence Number: 0    (relative sequence number)
    Sequence Number (raw): 232129012
    [Next Sequence Number: 1    (relative sequence number)]
    Acknowledgment Number: 0
    Acknowledgment number (raw): 0
    0111 .... = Header Length: 28 bytes (7)
  v 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
```

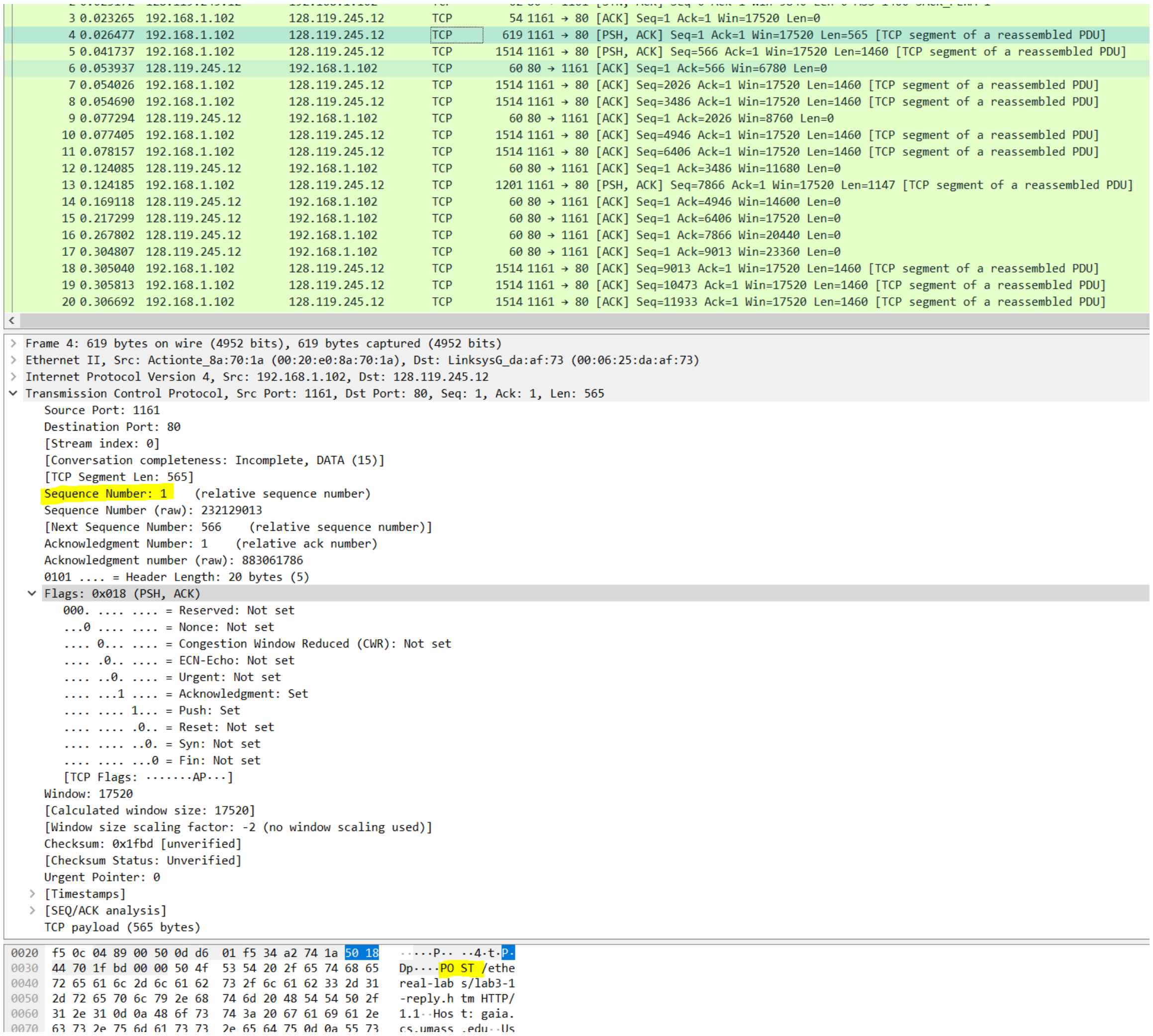
5. What is the sequence number of the SYNACK segment sent by gaia.cs.umass.edu to the client computer in reply to the SYN? What is the value of the Acknowledgement field in the SYNACK segment? How did gaia.cs.umass.edu determine that value? What is it in the segment that identifies the segment as a SYNACK segment?
   The sequence number is 0. The acknowledgement number is 1. It was determined by adding 1 to the sequence number. It is identified as SYNACK based on the SYN and ACK flags being set to 1.



```
No.  Time       Source          Destination     Protocol Length Info
   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=1
   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=1
   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  TCP       619 1161 → 80 [PSH, ACK] Seq=1 Ack=1 Win=17520 Len=565 [TCP segment of a reassembled PDU]
   5 0.041737   192.168.1.102   128.119.245.12  TCP      1514 1161 → 80 [PSH, ACK] Seq=566 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
   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  TCP      1514 1161 → 80 [ACK] Seq=2026 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
   8 0.054690   192.168.1.102   128.119.245.12  TCP      1514 1161 → 80 [ACK] Seq=3486 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
   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  TCP      1514 1161 → 80 [ACK] Seq=4946 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  11 0.078157   192.168.1.102   128.119.245.12  TCP      1514 1161 → 80 [ACK] Seq=6406 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  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  TCP      1201 1161 → 80 [PSH, ACK] Seq=7866 Ack=1 Win=17520 Len=1147 [TCP segment of a reassembled PDU]
  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  TCP      1514 1161 → 80 [ACK] Seq=9013 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  19 0.305813   192.168.1.102   128.119.245.12  TCP      1514 1161 → 80 [ACK] Seq=10473 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  20 0.306692   192.168.1.102   128.119.245.12  TCP      1514 1161 → 80 [ACK] Seq=11933 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  21 0.307571   192.168.1.102   128.119.245.12  TCP      1514 1161 → 80 [ACK] Seq=13393 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  22 0.308699   192.168.1.102   128.119.245.12  TCP      1514 1161 → 80 [ACK] Seq=14853 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  23 0.309553   192.168.1.102   128.119.245.12  TCP       946 1161 → 80 [PSH, ACK] Seq=16313 Ack=1 Win=17520 Len=892 [TCP segment of a reassembled PDU]
  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  TCP      1514 1161 → 80 [ACK] Seq=17205 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  31 0.577385   192.168.1.102   128.119.245.12  TCP      1514 1161 → 80 [ACK] Seq=18665 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  32 0.578329   192.168.1.102   128.119.245.12  TCP      1514 1161 → 80 [ACK] Seq=20125 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  33 0.579195   192.168.1.102   128.119.245.12  TCP      1514 1161 → 80 [ACK] Seq=21585 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  34 0.580149   192.168.1.102   128.119.245.12  TCP      1514 1161 → 80 [ACK] Seq=23045 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  35 0.581074   192.168.1.102   128.119.245.12  TCP       946 1161 → 80 [PSH, ACK] Seq=24505 Ack=1 Win=17520 Len=892 [TCP segment of a reassembled PDU]
  36 0.626496   128.119.245.12  192.168.1.102   TCP        60 80 → 1161 [ACK] Seq=1 Ack=18665 Win=40880 Len=0
```

```
> Frame 2: 62 bytes on wire (496 bits), 62 bytes captured (496 bits)
> Ethernet II, Src: LinksysG_da:af:73 (00:06:25:da:af:73), Dst: Actionte_8a:70:1a (00:20:e0:8a:70:1a)
> Internet Protocol Version 4, Src: 128.119.245.12, Dst: 192.168.1.102
v Transmission Control Protocol, Src Port: 80, Dst Port: 1161, Seq: 0, Ack: 1, Len: 0
    Source Port: 80
    Destination Port: 1161
    [Stream index: 0]
    [Conversation completeness: Incomplete, DATA (15)]
    [TCP Segment Len: 0]
    Sequence Number: 0    (relative sequence number)
    Sequence Number (raw): 883061785
    [Next Sequence Number: 1    (relative sequence number)]
    Acknowledgment Number: 1    (relative ack number)
    Acknowledgment number (raw): 232129013
    0111 .... = Header Length: 28 bytes (7)
  v Flags: 0x012 (SYN, ACK)
      000. .... .... = Reserved: Not set
      ...0 .... .... = Nonce: Not set
      .... 0... .... = Congestion Window Reduced (CWR): Not set
      .... .0.. .... = ECN-Echo: Not set
      .... ..0. .... = Urgent: Not set
      .... ...1 .... = Acknowledgment: Set
      .... .... 0... = Push: Not set
      .... .... .0.. = Reset: Not set
    > .... .... ..1. = Syn: Set
      .... .... ...0 = Fin: Not set
      [TCP Flags: ·······A··S·]
    Window: 5840
    [Calculated window size: 5840]
    Checksum: 0x774d [unverified]
    [Checksum Status: Unverified]
    Urgent Pointer: 0
  > Options: (8 bytes), Maximum segment size, No-Operation (NOP), No-Operation (NOP), SACK permitted
  > [Timestamps]
  > [SEQ/ACK analysis]
```

6. 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.
   The sequence number is 1



```
  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   TCP  619 1161 → 80 [PSH, ACK] Seq=1 Ack=1 Win=17520 Len=565 [TCP segment of a reassembled PDU]
  5 0.041737 192.168.1.102    128.119.245.12   TCP 1514 1161 → 80 [PSH, ACK] Seq=566 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  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   TCP 1514 1161 → 80 [ACK] Seq=2026 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  8 0.054690 192.168.1.102    128.119.245.12   TCP 1514 1161 → 80 [ACK] Seq=3486 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  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   TCP 1514 1161 → 80 [ACK] Seq=4946 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 11 0.078157 192.168.1.102    128.119.245.12   TCP 1514 1161 → 80 [ACK] Seq=6406 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 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   TCP 1201 1161 → 80 [PSH, ACK] Seq=7866 Ack=1 Win=17520 Len=1147 [TCP segment of a reassembled PDU]
 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   TCP 1514 1161 → 80 [ACK] Seq=9013 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 19 0.305813 192.168.1.102    128.119.245.12   TCP 1514 1161 → 80 [ACK] Seq=10473 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 20 0.306692 192.168.1.102    128.119.245.12   TCP 1514 1161 → 80 [ACK] Seq=11933 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
```

```
> Frame 4: 619 bytes on wire (4952 bits), 619 bytes captured (4952 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
v Transmission Control Protocol, Src Port: 1161, Dst Port: 80, Seq: 1, Ack: 1, Len: 565
    Source Port: 1161
    Destination Port: 80
    [Stream index: 0]
    [Conversation completeness: Incomplete, DATA (15)]
    [TCP Segment Len: 565]
    Sequence Number: 1    (relative sequence number)
    Sequence Number (raw): 232129013
    [Next Sequence Number: 566    (relative sequence number)]
    Acknowledgment Number: 1    (relative ack number)
    Acknowledgment number (raw): 883061786
    0101 .... = Header Length: 20 bytes (5)
  v Flags: 0x018 (PSH, ACK)
      000. .... .... = Reserved: Not set
      ...0 .... .... = Nonce: Not set
      .... 0... .... = Congestion Window Reduced (CWR): Not set
      .... .0.. .... = ECN-Echo: Not set
      .... ..0. .... = Urgent: Not set
      .... ...1 .... = Acknowledgment: Set
      .... .... 1... = Push: Set
      .... .... .0.. = Reset: Not set
      .... .... ..0. = Syn: Not set
      .... .... ...0 = Fin: Not set
      [TCP Flags: ·······AP···]
    Window: 17520
    [Calculated window size: 17520]
    [Window size scaling factor: -2 (no window scaling used)]
    Checksum: 0x1fbd [unverified]
    [Checksum Status: Unverified]
    Urgent Pointer: 0
  > [Timestamps]
  > [SEQ/ACK analysis]
    TCP payload (565 bytes)
```

```
0020  f5 0c 04 89 00 50 0d d6  01 f5 34 a2 74 1a 50 18   ·····P·· ··4·t·P·
0030  44 70 1f bd 00 00 50 4f  53 54 20 2f 65 74 68 65   Dp····PO ST /ethe
0040  72 65 61 6c 2d 6c 61 62  73 2f 6c 61 62 33 2d 31   real-lab s/lab3-1
0050  2d 72 65 70 6c 79 2e 68  74 6d 20 48 54 54 50 2f   -reply.h tm HTTP/
0060  31 2e 31 0d 0a 48 6f 73  74 3a 20 67 61 69 61 2e   1.1··Hos t: gaia.
0070  63 73 2e 75 6d 61 73 73  2e 65 64 75 0d 0a 55 73   cs.umass .edu··Us
```

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 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.*

Sequence	Sequence #	Send Time	Receive Time	RTT
1	1	.026477	.053937	.02746
2	566	.041737	.077294	.035557
3	2026	.054026	.124085	.070059
4	3486	.054690	.169118	.114428
5	4946	.077405	.217299	.139894
6	6406	.078157	.267802	.189645

EstimatedRTT = .875 * EstimatedRTT + .125 * SampleRTT

EstimatedRTT after the receipt of the ACK of segment 1:
EstimatedRTT = RTT for Segment 1 = .02746 secs

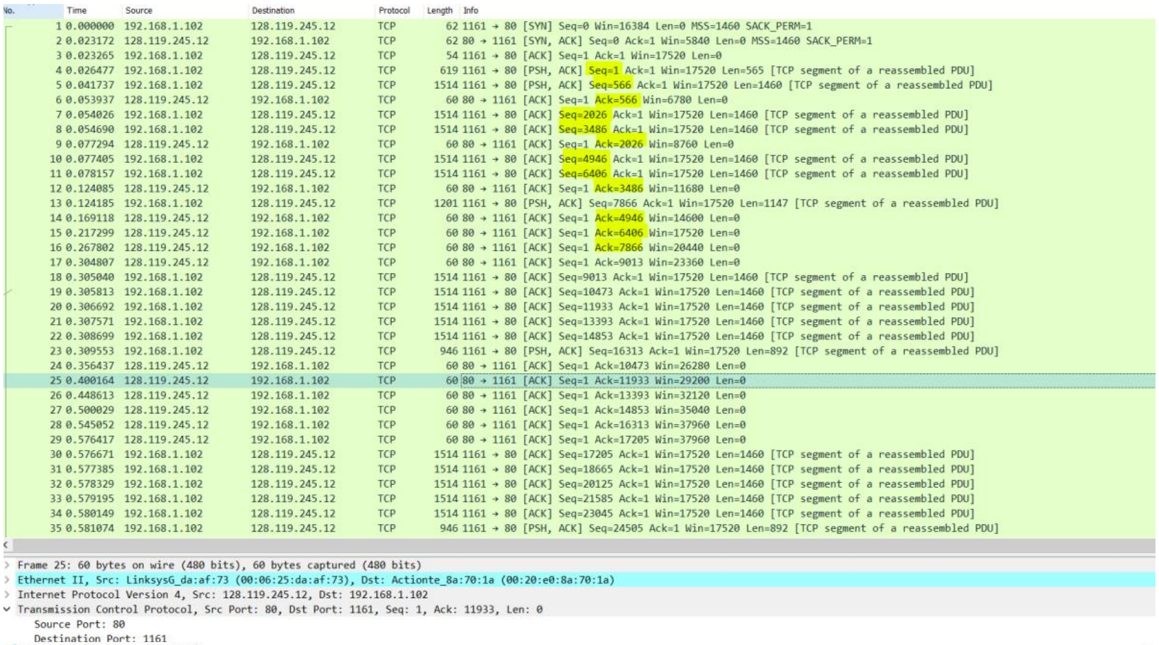
EstimatedRTT after the receipt of the ACK of segment 2:
EstimatedRTT = 0.875 * .02746 + 0.125 * .035557 = .028472125 secs

EstimatedRTT after the receipt of the ACK of segment 3:
EstimatedRTT = 0.875 * .028472125 + 0.125 * .070059 = .033670484375 secs

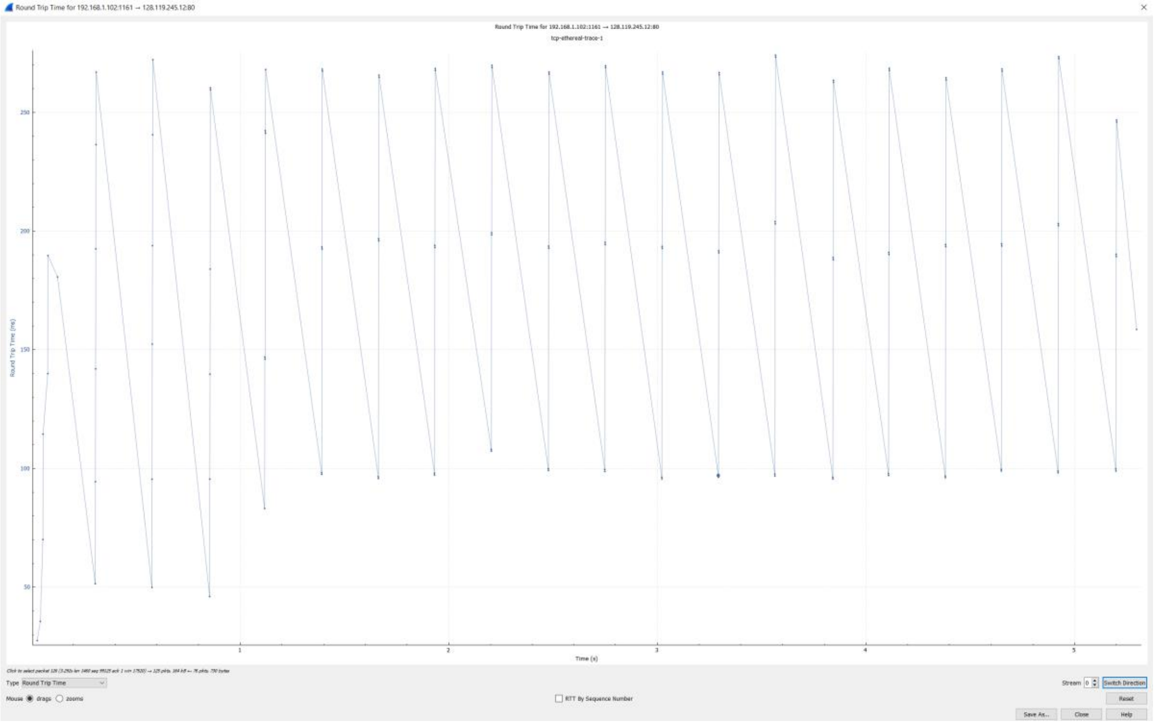
EstimatedRTT after the receipt of the ACK of segment 4:
EstimatedRTT = 0.875 * .033670484375 + 0.125 * .114428 = .043765173828125 secs

EstimatedRTT after the receipt of the ACK of segment 5:
EstimatedRTT = 0.875 * .043765173828125 + 0.125 * .139894 = .055781277099609375 secs

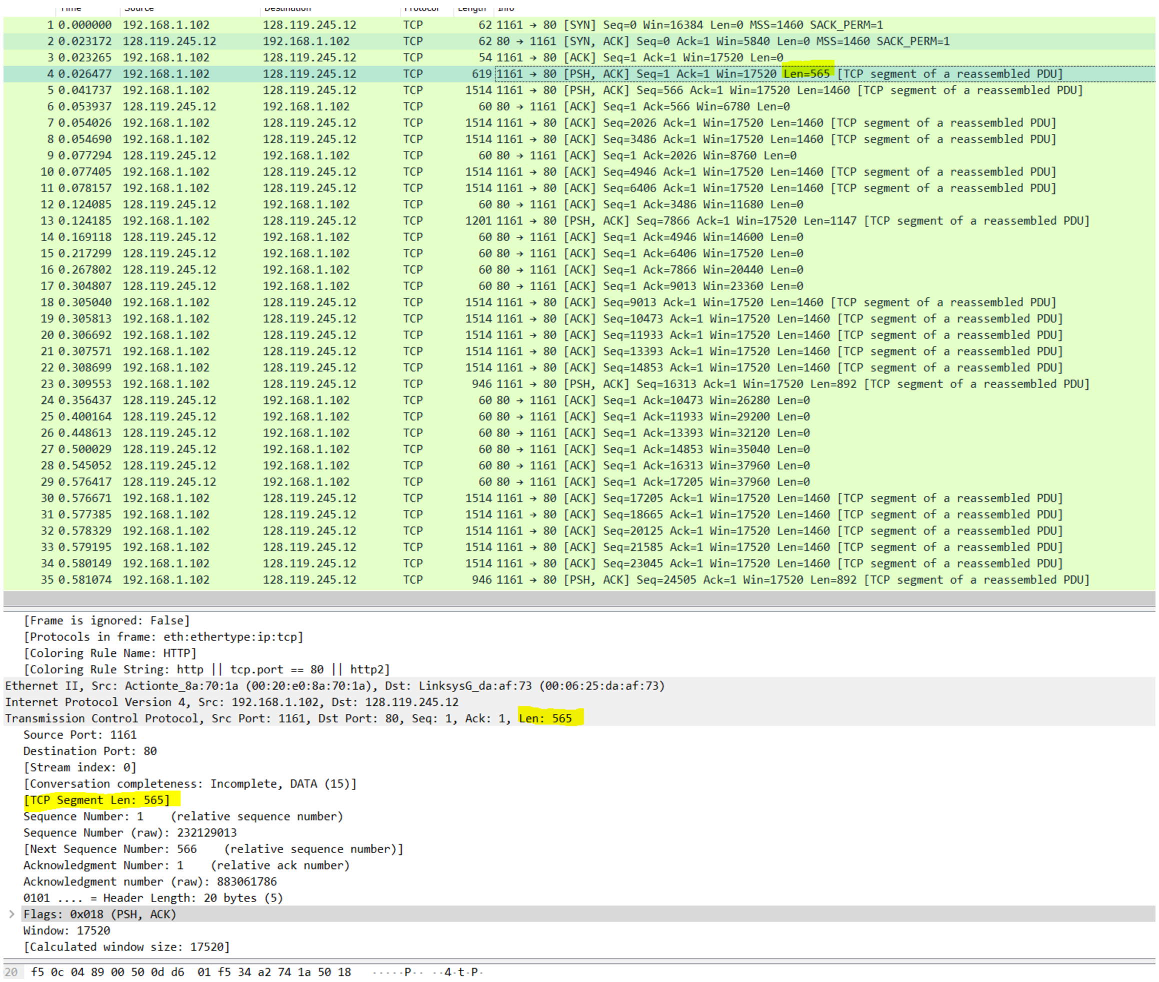
EstimatedRTT after the receipt of the ACK of segment 6:
EstimatedRTT = 0.875 * .055781277099609375 + 0.125 * .189645 = 0.072514242462158203125 secs



```
No. Time     Source          Destination     Protocol Length Info
  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=1
  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=1
  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  TCP       619 1161 → 80 [PSH, ACK] Seq=1 Ack=1 Win=17520 Len=565 [TCP segment of a reassembled PDU]
  5 0.041737 192.168.1.102   128.119.245.12  TCP      1514 1161 → 80 [PSH, ACK] Seq=566 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  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  TCP      1514 1161 → 80 [ACK] Seq=2026 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  8 0.054690 192.168.1.102   128.119.245.12  TCP      1514 1161 → 80 [ACK] Seq=3486 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  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  TCP      1514 1161 → 80 [ACK] Seq=4946 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 11 0.078157 192.168.1.102   128.119.245.12  TCP      1514 1161 → 80 [ACK] Seq=6406 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 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  TCP      1201 1161 → 80 [PSH, ACK] Seq=7866 Ack=1 Win=17520 Len=1147 [TCP segment of a reassembled PDU]
 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  TCP      1514 1161 → 80 [ACK] Seq=9013 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 19 0.305813 192.168.1.102   128.119.245.12  TCP      1514 1161 → 80 [ACK] Seq=10473 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 20 0.306692 192.168.1.102   128.119.245.12  TCP      1514 1161 → 80 [ACK] Seq=11933 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 21 0.307571 192.168.1.102   128.119.245.12  TCP      1514 1161 → 80 [ACK] Seq=13393 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 22 0.308699 192.168.1.102   128.119.245.12  TCP      1514 1161 → 80 [ACK] Seq=14853 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 23 0.309553 192.168.1.102   128.119.245.12  TCP       946 1161 → 80 [PSH, ACK] Seq=16313 Ack=1 Win=17520 Len=892 [TCP segment of a reassembled PDU]
 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  TCP      1514 1161 → 80 [ACK] Seq=17205 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 31 0.577385 192.168.1.102   128.119.245.12  TCP      1514 1161 → 80 [ACK] Seq=18665 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 32 0.578329 192.168.1.102   128.119.245.12  TCP      1514 1161 → 80 [ACK] Seq=20125 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 33 0.579195 192.168.1.102   128.119.245.12  TCP      1514 1161 → 80 [ACK] Seq=21585 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 34 0.580149 192.168.1.102   128.119.245.12  TCP      1514 1161 → 80 [ACK] Seq=23045 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 35 0.581074 192.168.1.102   128.119.245.12  TCP       946 1161 → 80 [PSH, ACK] Seq=24505 Ack=1 Win=17520 Len=892 [TCP segment of a reassembled PDU]

> Frame 25: 60 bytes on wire (480 bits), 60 bytes captured (480 bits)
> Ethernet II, Src: LinksysG_da:af:73 (00:06:25:da:af:73), Dst: Actionte_8a:70:1a (00:20:e0:8a:70:1a)
> Internet Protocol Version 4, Src: 128.119.245.12, Dst: 192.168.1.102
v Transmission Control Protocol, Src Port: 80, Dst Port: 1161, Seq: 1, Ack: 11933, Len: 0
    Source Port: 80
    Destination Port: 1161
```



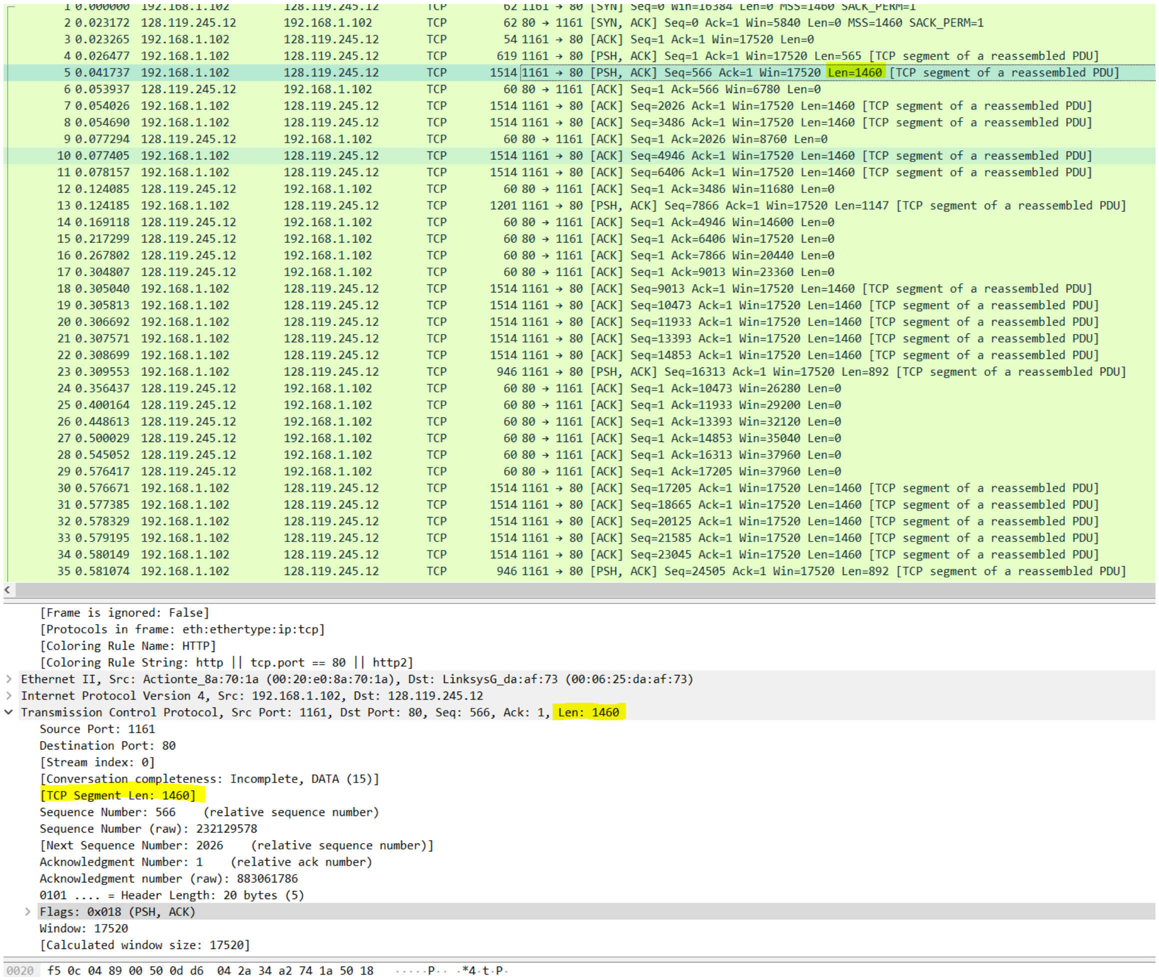
8. What is the length of each of the first six TCP segments?
   The first segment is 565 Bytes



```
     Time     Source           Destination      Protocol  Length Info
   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=1
   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=1
   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   TCP        619 1161 → 80 [PSH, ACK] Seq=1 Ack=1 Win=17520 Len=565 [TCP segment of a reassembled PDU]
   5 0.041737 192.168.1.102    128.119.245.12   TCP       1514 1161 → 80 [PSH, ACK] Seq=566 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
   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   TCP       1514 1161 → 80 [ACK] Seq=2026 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
   8 0.054690 192.168.1.102    128.119.245.12   TCP       1514 1161 → 80 [ACK] Seq=3486 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
   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   TCP       1514 1161 → 80 [ACK] Seq=4946 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  11 0.078157 192.168.1.102    128.119.245.12   TCP       1514 1161 → 80 [ACK] Seq=6406 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  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   TCP       1201 1161 → 80 [PSH, ACK] Seq=7866 Ack=1 Win=17520 Len=1147 [TCP segment of a reassembled PDU]
  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   TCP       1514 1161 → 80 [ACK] Seq=9013 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  19 0.305813 192.168.1.102    128.119.245.12   TCP       1514 1161 → 80 [ACK] Seq=10473 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  20 0.306692 192.168.1.102    128.119.245.12   TCP       1514 1161 → 80 [ACK] Seq=11933 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  21 0.307571 192.168.1.102    128.119.245.12   TCP       1514 1161 → 80 [ACK] Seq=13393 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  22 0.308699 192.168.1.102    128.119.245.12   TCP       1514 1161 → 80 [ACK] Seq=14853 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  23 0.309553 192.168.1.102    128.119.245.12   TCP        946 1161 → 80 [PSH, ACK] Seq=16313 Ack=1 Win=17520 Len=892 [TCP segment of a reassembled PDU]
  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   TCP       1514 1161 → 80 [ACK] Seq=17205 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  31 0.577385 192.168.1.102    128.119.245.12   TCP       1514 1161 → 80 [ACK] Seq=18665 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  32 0.578329 192.168.1.102    128.119.245.12   TCP       1514 1161 → 80 [ACK] Seq=20125 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  33 0.579195 192.168.1.102    128.119.245.12   TCP       1514 1161 → 80 [ACK] Seq=21585 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  34 0.580149 192.168.1.102    128.119.245.12   TCP       1514 1161 → 80 [ACK] Seq=23045 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  35 0.581074 192.168.1.102    128.119.245.12   TCP        946 1161 → 80 [PSH, ACK] Seq=24505 Ack=1 Win=17520 Len=892 [TCP segment of a reassembled PDU]
```

```
   [Frame is ignored: False]
   [Protocols in frame: eth:ethertype:ip:tcp]
   [Coloring Rule Name: HTTP]
   [Coloring Rule String: http || tcp.port == 80 || http2]
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: 1, Ack: 1, Len: 565
   Source Port: 1161
   Destination Port: 80
   [Stream index: 0]
   [Conversation completeness: Incomplete, DATA (15)]
   [TCP Segment Len: 565]
   Sequence Number: 1    (relative sequence number)
   Sequence Number (raw): 232129013
   [Next Sequence Number: 566    (relative sequence number)]
   Acknowledgment Number: 1    (relative ack number)
   Acknowledgment number (raw): 883061786
   0101 .... = Header Length: 20 bytes (5)
 > Flags: 0x018 (PSH, ACK)
   Window: 17520
   [Calculated window size: 17520]

20  f5 0c 04 89 00 50 0d d6  01 f5 34 a2 74 1a 50 18   ·····P·· ··4·t·P·
```

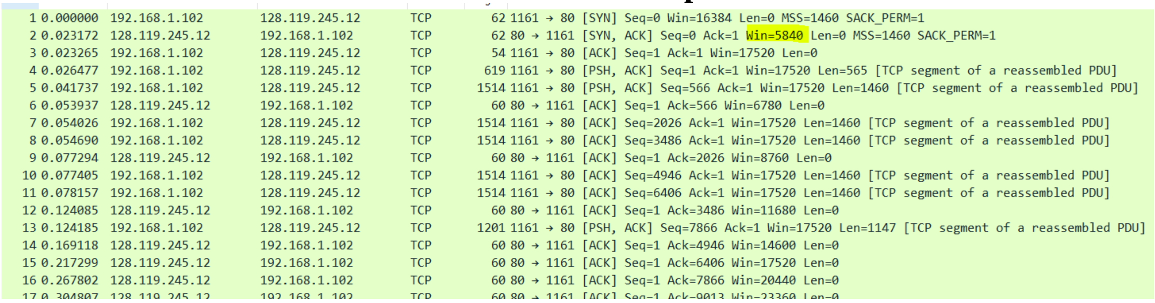
The segments after that are 1460 bytes.



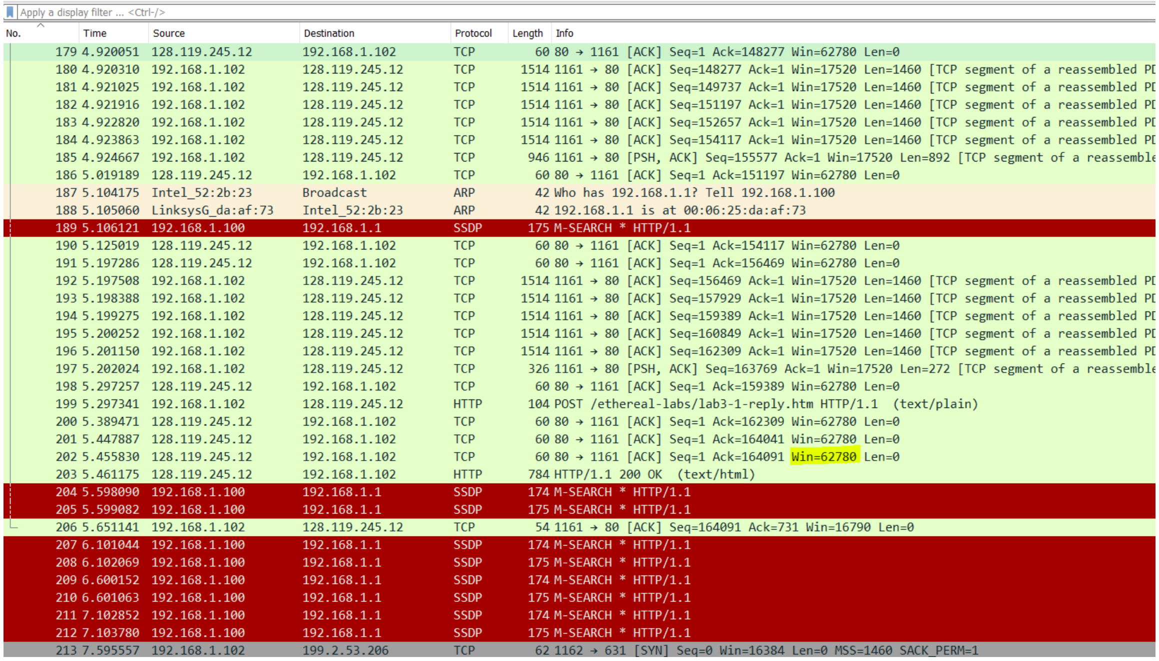
```
 1 0.000000  192.168.1.102   128.119.245.12  TCP   62 1161 → 80 [SYN] Seq=0 Win=16384 Len=0 MSS=1460 SACK_PERM=1
 2 0.023172  128.119.245.12  192.168.1.102   TCP   62 80 → 1161 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 SACK_PERM=1
 3 0.023265  192.168.1.102   128.119.245.12  TCP   54 1161 → 80 [ACK] Seq=1 Ack=1 Win=17520 Len=0
 4 0.026477  192.168.1.102   128.119.245.12  TCP  619 1161 → 80 [PSH, ACK] Seq=1 Ack=1 Win=17520 Len=565 [TCP segment of a reassembled PDU]
 5 0.041737  192.168.1.102   128.119.245.12  TCP 1514 1161 → 80 [PSH, ACK] Seq=566 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 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  TCP 1514 1161 → 80 [ACK] Seq=2026 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 8 0.054690  192.168.1.102   128.119.245.12  TCP 1514 1161 → 80 [ACK] Seq=3486 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 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  TCP 1514 1161 → 80 [ACK] Seq=4946 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
11 0.078157  192.168.1.102   128.119.245.12  TCP 1514 1161 → 80 [ACK] Seq=6406 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
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  TCP 1201 1161 → 80 [PSH, ACK] Seq=7866 Ack=1 Win=17520 Len=1147 [TCP segment of a reassembled PDU]
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  TCP 1514 1161 → 80 [ACK] Seq=9013 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
19 0.305813  192.168.1.102   128.119.245.12  TCP 1514 1161 → 80 [ACK] Seq=10473 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
20 0.306692  192.168.1.102   128.119.245.12  TCP 1514 1161 → 80 [ACK] Seq=11933 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
21 0.307571  192.168.1.102   128.119.245.12  TCP 1514 1161 → 80 [ACK] Seq=13393 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
22 0.308699  192.168.1.102   128.119.245.12  TCP 1514 1161 → 80 [ACK] Seq=14853 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
23 0.309553  192.168.1.102   128.119.245.12  TCP  946 1161 → 80 [PSH, ACK] Seq=16313 Ack=1 Win=17520 Len=892 [TCP segment of a reassembled PDU]
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  TCP 1514 1161 → 80 [ACK] Seq=17205 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
31 0.577385  192.168.1.102   128.119.245.12  TCP 1514 1161 → 80 [ACK] Seq=18665 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
32 0.578329  192.168.1.102   128.119.245.12  TCP 1514 1161 → 80 [ACK] Seq=20125 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
33 0.579195  192.168.1.102   128.119.245.12  TCP 1514 1161 → 80 [ACK] Seq=21585 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
34 0.580149  192.168.1.102   128.119.245.12  TCP 1514 1161 → 80 [ACK] Seq=23045 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
35 0.581074  192.168.1.102   128.119.245.12  TCP  946 1161 → 80 [PSH, ACK] Seq=24505 Ack=1 Win=17520 Len=892 [TCP segment of a reassembled PDU]
```

```
    [Frame is ignored: False]
    [Protocols in frame: eth:ethertype:ip:tcp]
    [Coloring Rule Name: HTTP]
    [Coloring Rule String: http || tcp.port == 80 || http2]
> 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
v Transmission Control Protocol, Src Port: 1161, Dst Port: 80, Seq: 566, Ack: 1, Len: 1460
    Source Port: 1161
    Destination Port: 80
    [Stream index: 0]
    [Conversation completeness: Incomplete, DATA (15)]
    [TCP Segment Len: 1460]
    Sequence Number: 566    (relative sequence number)
    Sequence Number (raw): 232129578
    [Next Sequence Number: 2026    (relative sequence number)]
    Acknowledgment Number: 1    (relative ack number)
    Acknowledgment number (raw): 883061786
    0101 .... = Header Length: 20 bytes (5)
  > Flags: 0x018 (PSH, ACK)
    Window: 17520
    [Calculated window size: 17520]

0020  f5 0c 04 89 00 50 0d d6  04 2a 34 a2 74 1a 50 18   ·····P·· ·*4·t·P·
```

9. 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?
The initial buffer size is 5840 bytes. It goes all the way up to 62780 bytes. There has not been a throttle for lack of receiver space.

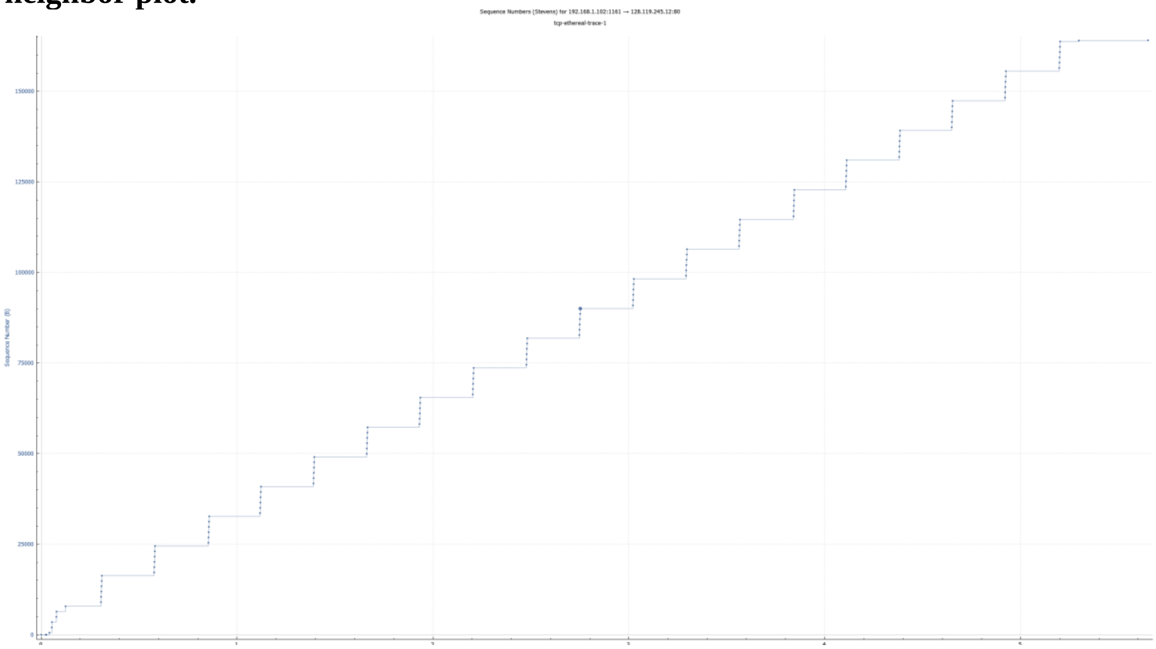


```
 1 0.000000  192.168.1.102   128.119.245.12  TCP   62 1161 → 80 [SYN] Seq=0 Win=16384 Len=0 MSS=1460 SACK_PERM=1
 2 0.023172  128.119.245.12  192.168.1.102   TCP   62 80 → 1161 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 SACK_PERM=1
 3 0.023265  192.168.1.102   128.119.245.12  TCP   54 1161 → 80 [ACK] Seq=1 Ack=1 Win=17520 Len=0
 4 0.026477  192.168.1.102   128.119.245.12  TCP  619 1161 → 80 [PSH, ACK] Seq=1 Ack=1 Win=17520 Len=565 [TCP segment of a reassembled PDU]
 5 0.041737  192.168.1.102   128.119.245.12  TCP 1514 1161 → 80 [PSH, ACK] Seq=566 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 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  TCP 1514 1161 → 80 [ACK] Seq=2026 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 8 0.054690  192.168.1.102   128.119.245.12  TCP 1514 1161 → 80 [ACK] Seq=3486 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 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  TCP 1514 1161 → 80 [ACK] Seq=4946 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
11 0.078157  192.168.1.102   128.119.245.12  TCP 1514 1161 → 80 [ACK] Seq=6406 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
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  TCP 1201 1161 → 80 [PSH, ACK] Seq=7866 Ack=1 Win=17520 Len=1147 [TCP segment of a reassembled PDU]
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
```



```
Apply a display filter ... <Ctrl-/>
No.     Time      Source             Destination       Protocol  Length  Info
    179 4.920051  128.119.245.12     192.168.1.102     TCP         60 80 → 1161 [ACK] Seq=1 Ack=148277 Win=62780 Len=0
    180 4.920310  192.168.1.102      128.119.245.12    TCP       1514 1161 → 80 [ACK] Seq=148277 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PD
    181 4.921025  192.168.1.102      128.119.245.12    TCP       1514 1161 → 80 [ACK] Seq=149737 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PD
    182 4.921916  192.168.1.102      128.119.245.12    TCP       1514 1161 → 80 [ACK] Seq=151197 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PD
    183 4.922820  192.168.1.102      128.119.245.12    TCP       1514 1161 → 80 [ACK] Seq=152657 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PD
    184 4.923863  192.168.1.102      128.119.245.12    TCP       1514 1161 → 80 [ACK] Seq=154117 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PD
    185 4.924667  192.168.1.102      128.119.245.12    TCP        946 1161 → 80 [PSH, ACK] Seq=155577 Ack=1 Win=17520 Len=892 [TCP segment of a reassemble
    186 5.019189  128.119.245.12     192.168.1.102     TCP         60 80 → 1161 [ACK] Seq=1 Ack=151197 Win=62780 Len=0
    187 5.104175  Intel_52:2b:23     Broadcast         ARP         42 Who has 192.168.1.1? Tell 192.168.1.100
    188 5.105060  LinksysG_da:af:73  Intel_52:2b:23    ARP         42 192.168.1.1 is at 00:06:25:da:af:73
    189 5.106121  192.168.1.100      192.168.1.1       SSDP       175 M-SEARCH * HTTP/1.1
    190 5.125019  128.119.245.12     192.168.1.102     TCP         60 80 → 1161 [ACK] Seq=1 Ack=154117 Win=62780 Len=0
    191 5.197286  128.119.245.12     192.168.1.102     TCP         60 80 → 1161 [ACK] Seq=1 Ack=156469 Win=62780 Len=0
    192 5.197508  192.168.1.102      128.119.245.12    TCP       1514 1161 → 80 [ACK] Seq=156469 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PD
    193 5.198388  192.168.1.102      128.119.245.12    TCP       1514 1161 → 80 [ACK] Seq=157929 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PD
    194 5.199275  192.168.1.102      128.119.245.12    TCP       1514 1161 → 80 [ACK] Seq=159389 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PD
    195 5.200252  192.168.1.102      128.119.245.12    TCP       1514 1161 → 80 [ACK] Seq=160849 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PD
    196 5.201150  192.168.1.102      128.119.245.12    TCP       1514 1161 → 80 [ACK] Seq=162309 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PD
    197 5.202024  192.168.1.102      128.119.245.12    TCP        326 1161 → 80 [PSH, ACK] Seq=163769 Ack=1 Win=17520 Len=272 [TCP segment of a reassemble
    198 5.297257  128.119.245.12     192.168.1.102     TCP         60 80 → 1161 [ACK] Seq=1 Ack=159389 Win=62780 Len=0
    199 5.297341  192.168.1.102      128.119.245.12    HTTP       104 POST /ethereal-labs/lab3-1-reply.htm HTTP/1.1  (text/plain)
    200 5.389471  128.119.245.12     192.168.1.102     TCP         60 80 → 1161 [ACK] Seq=1 Ack=162309 Win=62780 Len=0
    201 5.447887  128.119.245.12     192.168.1.102     TCP         60 80 → 1161 [ACK] Seq=1 Ack=164041 Win=62780 Len=0
    202 5.455830  128.119.245.12     192.168.1.102     TCP         60 80 → 1161 [ACK] Seq=1 Ack=164091 Win=62780 Len=0
    203 5.461175  128.119.245.12     192.168.1.102     HTTP       784 HTTP/1.1 200 OK  (text/html)
    204 5.598090  192.168.1.100      192.168.1.1       SSDP       174 M-SEARCH * HTTP/1.1
    205 5.599082  192.168.1.100      192.168.1.1       SSDP       175 M-SEARCH * HTTP/1.1
    206 5.651141  192.168.1.102      128.119.245.12    TCP         54 1161 → 80 [ACK] Seq=164091 Ack=731 Win=16790 Len=0
    207 6.101044  192.168.1.100      192.168.1.1       SSDP       174 M-SEARCH * HTTP/1.1
    208 6.102069  192.168.1.100      192.168.1.1       SSDP       175 M-SEARCH * HTTP/1.1
    209 6.600152  192.168.1.100      192.168.1.1       SSDP       174 M-SEARCH * HTTP/1.1
    210 6.601063  192.168.1.100      192.168.1.1       SSDP       175 M-SEARCH * HTTP/1.1
    211 7.102852  192.168.1.100      192.168.1.1       SSDP       174 M-SEARCH * HTTP/1.1
    212 7.103780  192.168.1.100      192.168.1.1       SSDP       175 M-SEARCH * HTTP/1.1
    213 7.595557  192.168.1.102      199.2.53.206      TCP         62 1162 → 631 [SYN] Seq=0 Win=16384 Len=0 MSS=1460 SACK_PERM=1
```

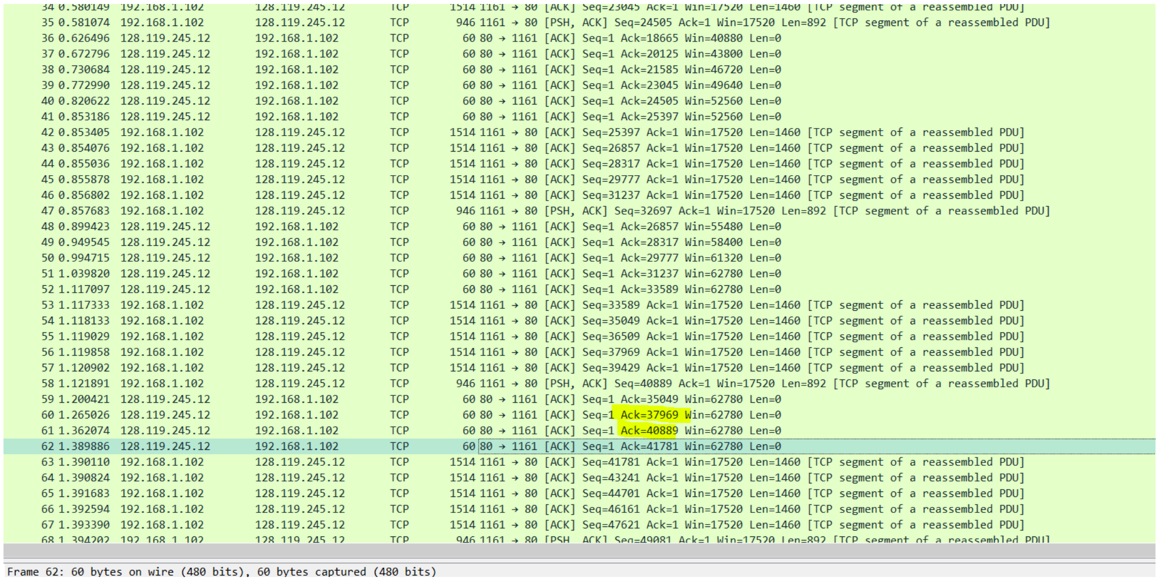
10. Are there any retransmitted segments in the trace file? What did you check for (in the trace) in order to answer this question?
Based on the time-sequence we can verify that there is no retransmitted data, because the graph is always increasing. If there were retransmitted data than the sequence number of retransmitted data would be smaller than the neighbor plot.



11. 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).

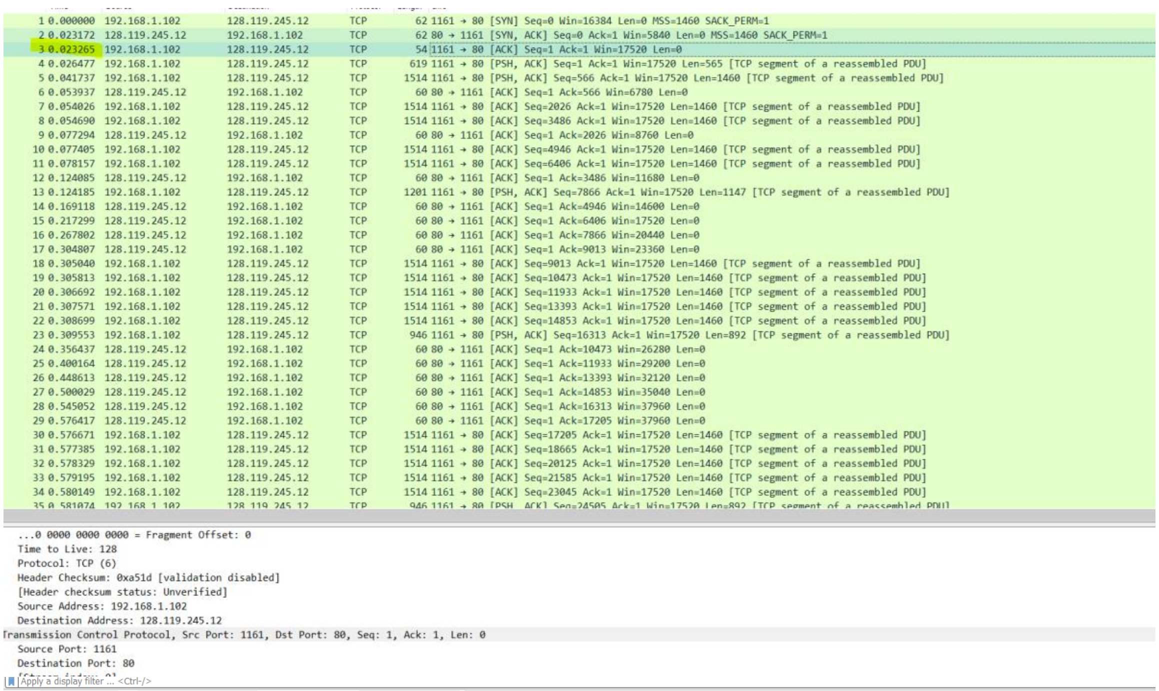
Number	Sequence Acknowledgement Number	Acknowledgment Data Size
1	566	566
2	2026	1460
3	3486	1460
4	4946	1460
5	6406	1460
6	7866	1460
7	9013	1460
8	10473	1460
9	11933	1460
10	14853	1460
11	16313	1460
12	17205	1460
13	18665	1460
14	20125	1460
15	21585	1460

The difference between the consecutive ACKS indicates the data that was received in bytes. In the screenshot, there is an instance highlighted that shows this.



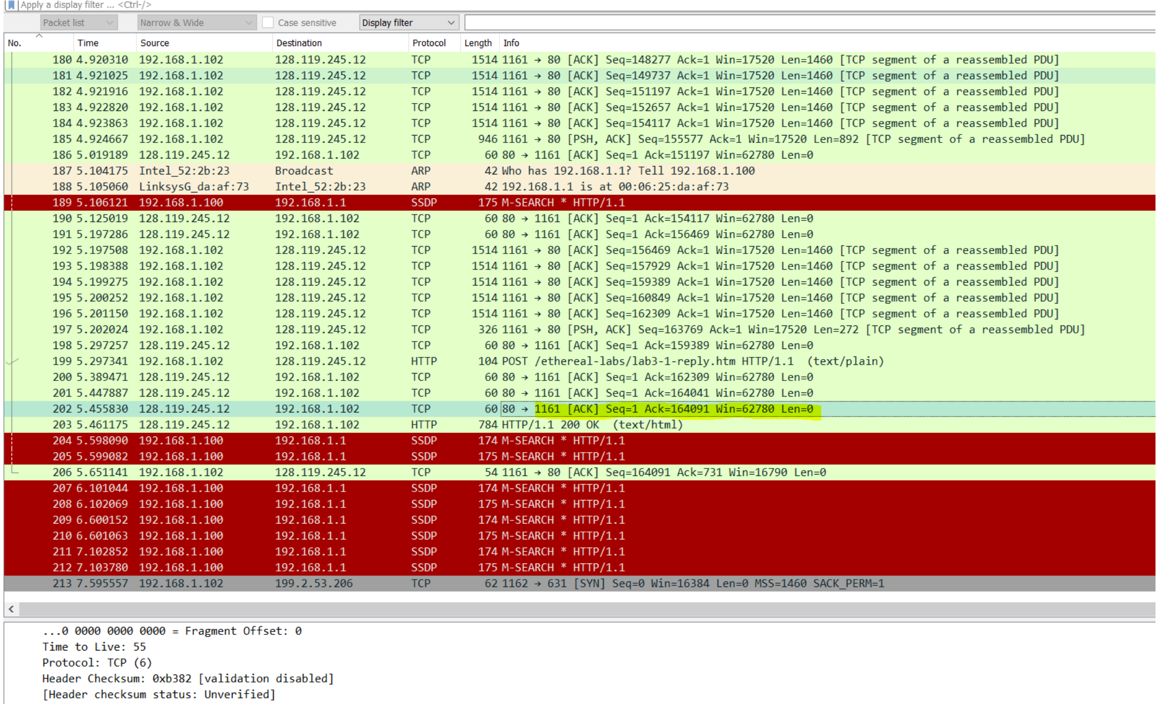
```
34 0.580149  192.168.1.102    128.119.245.12   TCP   1514 1161 → 80 [ACK] Seq=23045 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
35 0.581074  192.168.1.102    128.119.245.12   TCP    946 1161 → 80 [PSH, ACK] Seq=24505 Ack=1 Win=17520 Len=892 [TCP segment of a reassembled PDU]
36 0.626496  128.119.245.12   192.168.1.102    TCP     60 80 → 1161 [ACK] Seq=1 Ack=18665 Win=40880 Len=0
37 0.672796  128.119.245.12   192.168.1.102    TCP     60 80 → 1161 [ACK] Seq=1 Ack=20125 Win=43800 Len=0
38 0.730684  128.119.245.12   192.168.1.102    TCP     60 80 → 1161 [ACK] Seq=1 Ack=21585 Win=46720 Len=0
39 0.772990  128.119.245.12   192.168.1.102    TCP     60 80 → 1161 [ACK] Seq=1 Ack=23045 Win=49640 Len=0
40 0.820622  128.119.245.12   192.168.1.102    TCP     60 80 → 1161 [ACK] Seq=1 Ack=24505 Win=52560 Len=0
41 0.853186  128.119.245.12   192.168.1.102    TCP     60 80 → 1161 [ACK] Seq=1 Ack=25397 Win=52560 Len=0
42 0.853405  192.168.1.102    128.119.245.12   TCP   1514 1161 → 80 [ACK] Seq=25397 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
43 0.854076  192.168.1.102    128.119.245.12   TCP   1514 1161 → 80 [ACK] Seq=26857 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
44 0.855036  192.168.1.102    128.119.245.12   TCP   1514 1161 → 80 [ACK] Seq=28317 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
45 0.855878  192.168.1.102    128.119.245.12   TCP   1514 1161 → 80 [ACK] Seq=29777 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
46 0.856802  192.168.1.102    128.119.245.12   TCP   1514 1161 → 80 [ACK] Seq=31237 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
47 0.857683  192.168.1.102    128.119.245.12   TCP    946 1161 → 80 [PSH, ACK] Seq=32697 Ack=1 Win=17520 Len=892 [TCP segment of a reassembled PDU]
48 0.899423  128.119.245.12   192.168.1.102    TCP     60 80 → 1161 [ACK] Seq=1 Ack=26857 Win=55480 Len=0
49 0.949545  128.119.245.12   192.168.1.102    TCP     60 80 → 1161 [ACK] Seq=1 Ack=28317 Win=58400 Len=0
50 0.994715  128.119.245.12   192.168.1.102    TCP     60 80 → 1161 [ACK] Seq=1 Ack=29777 Win=61320 Len=0
51 1.039820  128.119.245.12   192.168.1.102    TCP     60 80 → 1161 [ACK] Seq=1 Ack=31237 Win=62780 Len=0
52 1.117097  128.119.245.12   192.168.1.102    TCP     60 80 → 1161 [ACK] Seq=1 Ack=33589 Win=62780 Len=0
53 1.117333  192.168.1.102    128.119.245.12   TCP   1514 1161 → 80 [ACK] Seq=33589 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
54 1.118133  192.168.1.102    128.119.245.12   TCP   1514 1161 → 80 [ACK] Seq=35049 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
55 1.119029  192.168.1.102    128.119.245.12   TCP   1514 1161 → 80 [ACK] Seq=36509 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
56 1.119858  192.168.1.102    128.119.245.12   TCP   1514 1161 → 80 [ACK] Seq=37969 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
57 1.120902  192.168.1.102    128.119.245.12   TCP   1514 1161 → 80 [ACK] Seq=39429 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
58 1.121891  192.168.1.102    128.119.245.12   TCP    946 1161 → 80 [PSH, ACK] Seq=40889 Ack=1 Win=17520 Len=892 [TCP segment of a reassembled PDU]
59 1.200421  128.119.245.12   192.168.1.102    TCP     60 80 → 1161 [ACK] Seq=1 Ack=35049 Win=62780 Len=0
60 1.265026  128.119.245.12   192.168.1.102    TCP     60 80 → 1161 [ACK] Seq=1 Ack=37969 Win=62780 Len=0
61 1.362074  128.119.245.12   192.168.1.102    TCP     60 80 → 1161 [ACK] Seq=1 Ack=40889 Win=62780 Len=0
62 1.389886  128.119.245.12   192.168.1.102    TCP     60 80 → 1161 [ACK] Seq=1 Ack=41781 Win=62780 Len=0
63 1.390110  192.168.1.102    128.119.245.12   TCP   1514 1161 → 80 [ACK] Seq=41781 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
64 1.390824  192.168.1.102    128.119.245.12   TCP   1514 1161 → 80 [ACK] Seq=43241 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
65 1.391683  192.168.1.102    128.119.245.12   TCP   1514 1161 → 80 [ACK] Seq=44701 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
66 1.392594  192.168.1.102    128.119.245.12   TCP   1514 1161 → 80 [ACK] Seq=46161 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
67 1.393390  192.168.1.102    128.119.245.12   TCP   1514 1161 → 80 [ACK] Seq=47621 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
68 1.394202  192.168.1.102    128.119.245.12   TCP    946 1161 → 80 [PSH, ACK] Seq=49081 Ack=1 Win=17520 Len=892 [TCP segment of a reassembled PDU]

Frame 62: 60 bytes on wire (480 bits), 60 bytes captured (480 bits)
```

12. What is the throughput (bytes transferred per unit time) for the TCP connection? Explain how you calculated this value.
Total transmitted in bytes = 164091 – 1 = 164090 bytes
Time from first ACK to last ACK = 5.455830 - .023265= 5.432565 seconds
Throughput = 164090 bytes / 5.432565 seconds = 30,204.88 bytes / second



```
  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=1
  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=1
  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  TCP   619 1161 → 80 [PSH, ACK] Seq=1 Ack=1 Win=17520 Len=565 [TCP segment of a reassembled PDU]
  5 0.041737 192.168.1.102   128.119.245.12  TCP  1514 1161 → 80 [PSH, ACK] Seq=566 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  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  TCP  1514 1161 → 80 [ACK] Seq=2026 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  8 0.054690 192.168.1.102   128.119.245.12  TCP  1514 1161 → 80 [ACK] Seq=3486 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  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  TCP  1514 1161 → 80 [ACK] Seq=4946 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 11 0.078157 192.168.1.102   128.119.245.12  TCP  1514 1161 → 80 [ACK] Seq=6406 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 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  TCP  1201 1161 → 80 [PSH, ACK] Seq=7866 Ack=1 Win=17520 Len=1147 [TCP segment of a reassembled PDU]
 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  TCP  1514 1161 → 80 [ACK] Seq=9013 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 19 0.305813 192.168.1.102   128.119.245.12  TCP  1514 1161 → 80 [ACK] Seq=10473 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 20 0.306692 192.168.1.102   128.119.245.12  TCP  1514 1161 → 80 [ACK] Seq=11933 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 21 0.307571 192.168.1.102   128.119.245.12  TCP  1514 1161 → 80 [ACK] Seq=13393 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 22 0.308699 192.168.1.102   128.119.245.12  TCP  1514 1161 → 80 [ACK] Seq=14853 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 23 0.309553 192.168.1.102   128.119.245.12  TCP   946 1161 → 80 [PSH, ACK] Seq=16313 Ack=1 Win=17520 Len=892 [TCP segment of a reassembled PDU]
 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  TCP  1514 1161 → 80 [ACK] Seq=17205 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 31 0.577385 192.168.1.102   128.119.245.12  TCP  1514 1161 → 80 [ACK] Seq=18665 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 32 0.578329 192.168.1.102   128.119.245.12  TCP  1514 1161 → 80 [ACK] Seq=20125 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 33 0.579195 192.168.1.102   128.119.245.12  TCP  1514 1161 → 80 [ACK] Seq=21585 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 34 0.580149 192.168.1.102   128.119.245.12  TCP  1514 1161 → 80 [ACK] Seq=23045 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
 35 0.581074 192.168.1.102   128.119.245.12  TCP   946 1161 → 80 [PSH, ACK] Seq=24505 Ack=1 Win=17520 Len=892 [TCP segment of a reassembled PDU]
```

```
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 128
    Protocol: TCP (6)
    Header Checksum: 0xa51d [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 192.168.1.102
    Destination Address: 128.119.245.12
Transmission Control Protocol, Src Port: 1161, Dst Port: 80, Seq: 1, Ack: 1, Len: 0
    Source Port: 1161
    Destination Port: 80
```



```
No.   Time     Source          Destination     Protocol Length Info
  180 4.920310 192.168.1.102   128.119.245.12  TCP    1514 1161 → 80 [ACK] Seq=148277 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  181 4.921025 192.168.1.102   128.119.245.12  TCP    1514 1161 → 80 [ACK] Seq=149737 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  182 4.921916 192.168.1.102   128.119.245.12  TCP    1514 1161 → 80 [ACK] Seq=151197 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  183 4.922820 192.168.1.102   128.119.245.12  TCP    1514 1161 → 80 [ACK] Seq=152657 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  184 4.923863 192.168.1.102   128.119.245.12  TCP    1514 1161 → 80 [ACK] Seq=154117 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  185 4.924667 192.168.1.102   128.119.245.12  TCP     946 1161 → 80 [PSH, ACK] Seq=155577 Ack=1 Win=17520 Len=892 [TCP segment of a reassembled PDU]
  186 5.019189 128.119.245.12  192.168.1.102   TCP      60 80 → 1161 [ACK] Seq=1 Ack=151197 Win=62780 Len=0
  187 5.104175 Intel_52:2b:23  Broadcast       ARP      42 Who has 192.168.1.1? Tell 192.168.1.100
  188 5.105060 LinksysG_da:af:73 Intel_52:2b:23 ARP     42 192.168.1.1 is at 00:06:25:da:af:73
  189 5.106121 192.168.1.100   192.168.1.1     SSDP    175 M-SEARCH * HTTP/1.1
  190 5.125019 128.119.245.12  192.168.1.102   TCP      60 80 → 1161 [ACK] Seq=1 Ack=154117 Win=62780 Len=0
  191 5.197286 128.119.245.12  192.168.1.102   TCP      60 80 → 1161 [ACK] Seq=1 Ack=156469 Win=62780 Len=0
  192 5.197508 192.168.1.102   128.119.245.12  TCP    1514 1161 → 80 [ACK] Seq=156469 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  193 5.198388 192.168.1.102   128.119.245.12  TCP    1514 1161 → 80 [ACK] Seq=157929 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  194 5.199275 192.168.1.102   128.119.245.12  TCP    1514 1161 → 80 [ACK] Seq=159389 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  195 5.200252 192.168.1.102   128.119.245.12  TCP    1514 1161 → 80 [ACK] Seq=160849 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  196 5.201150 192.168.1.102   128.119.245.12  TCP    1514 1161 → 80 [ACK] Seq=162309 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
  197 5.202024 192.168.1.102   128.119.245.12  TCP     326 1161 → 80 [PSH, ACK] Seq=163769 Ack=1 Win=17520 Len=272 [TCP segment of a reassembled PDU]
  198 5.297257 128.119.245.12  192.168.1.102   TCP      60 80 → 1161 [ACK] Seq=1 Ack=159389 Win=62780 Len=0
  199 5.297341 192.168.1.102   128.119.245.12  HTTP    104 POST /ethereal-labs/lab3-1-reply.htm HTTP/1.1  (text/plain)
  200 5.389471 128.119.245.12  192.168.1.102   TCP      60 80 → 1161 [ACK] Seq=1 Ack=162309 Win=62780 Len=0
  201 5.447887 128.119.245.12  192.168.1.102   TCP      60 80 → 1161 [ACK] Seq=1 Ack=164041 Win=62780 Len=0
  202 5.455830 128.119.245.12  192.168.1.102   TCP      60 80 → 1161 [ACK] Seq=1 Ack=164091 Win=62780 Len=0
  203 5.461175 128.119.245.12  192.168.1.102   HTTP    784 HTTP/1.1 200 OK  (text/html)
  204 5.598090 192.168.1.100   192.168.1.1     SSDP    174 M-SEARCH * HTTP/1.1
  205 5.599082 192.168.1.100   192.168.1.1     SSDP    175 M-SEARCH * HTTP/1.1
  206 5.651141 192.168.1.102   128.119.245.12  TCP      54 1161 → 80 [ACK] Seq=164091 Ack=731 Win=16790 Len=0
  207 6.101044 192.168.1.100   192.168.1.1     SSDP    174 M-SEARCH * HTTP/1.1
  208 6.102069 192.168.1.100   192.168.1.1     SSDP    175 M-SEARCH * HTTP/1.1
  209 6.600152 192.168.1.100   192.168.1.1     SSDP    174 M-SEARCH * HTTP/1.1
  210 6.601063 192.168.1.100   192.168.1.1     SSDP    175 M-SEARCH * HTTP/1.1
  211 7.102852 192.168.1.100   192.168.1.1     SSDP    174 M-SEARCH * HTTP/1.1
  212 7.103780 192.168.1.100   192.168.1.1     SSDP    175 M-SEARCH * HTTP/1.1
  213 7.595557 192.168.1.102   199.2.53.206    TCP      62 1162 → 631 [SYN] Seq=0 Win=16384 Len=0 MSS=1460 SACK_PERM=1
```

```
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 55
    Protocol: TCP (6)
    Header Checksum: 0xb382 [validation disabled]
    [Header checksum status: Unverified]
```

To calculate the throughput rate, we take the first ACK that contains Seq=1 and Len=0. Then subtract the first ACK from the last ACK. The last contains ACK Seq=1, the final sequence number in Ack= portion of the info, and Len=0. Finally subtract 1. This gives us the bytes transferrer. Then divide by time which is the time in seconds from the second ACK we pulled minus the time in seconds from the first ACK we pull. This will give us the bytes/sec or throughput.

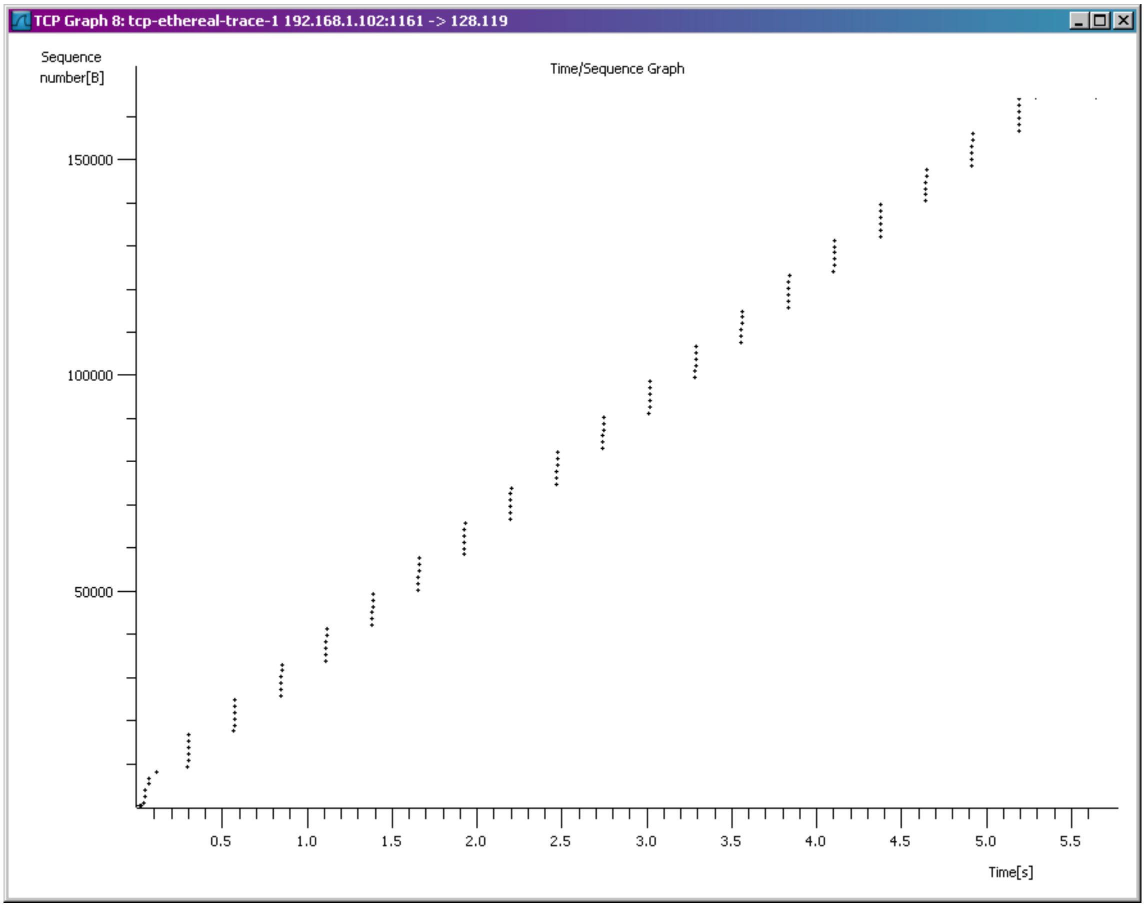
https://madpackets.com/2018/05/18/finding-throughput-with-wireshark/

4. TCP congestion control in action

Let's now examine the amount of data sent per unit time from the client to the server. Rather than (tediously!) calculating this from the raw data in the Wireshark window, we'll use one of Wireshark's TCP graphing utilities - *Time-Sequence-Graph(Stevens*) - to plot out data.

Note: you may need to click switch direction to get a more meaningful graph!

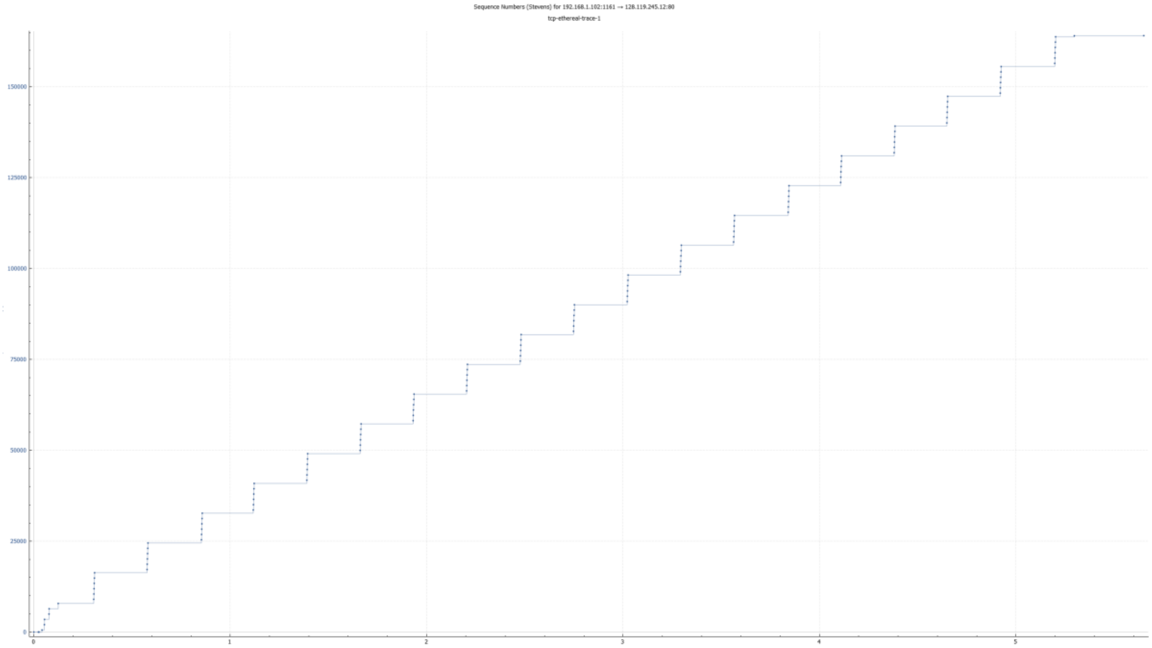
- Select a TCP segment in the Wireshark's "listing of captured-packets" window. Then select the menu : *Statistics->TCP Stream Graph-> Time-Sequence-Graph(Stevens*). You should see a plot that looks similar to the following plot, which was created from the captured packets in the packet trace *tcp-ethereal-trace-1* in http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip (see earlier footnote):



Here, each dot represents a TCP segment sent, plotting the sequence number of the segment versus the time at which it was sent. Note that a set of dots stacked above each other represents a series of packets that were sent back-to-back by the sender.

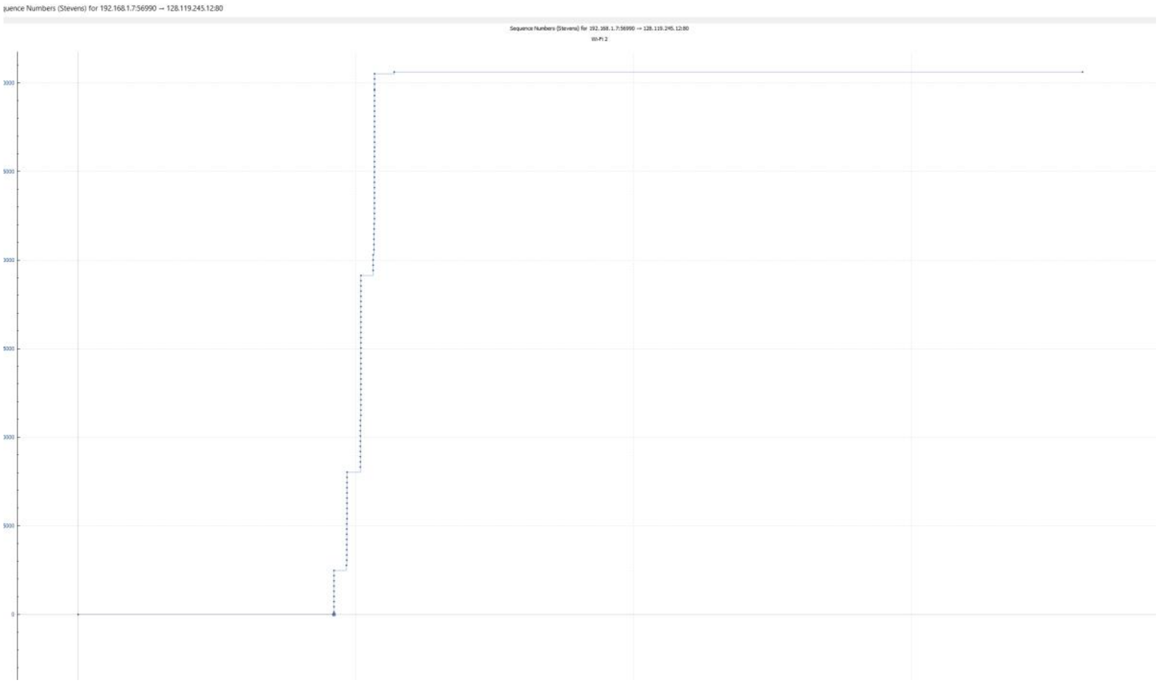
Answer question 13 for the TCP segments the packet trace *tcp-ethereal-trace-1* in http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip

13. 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.



The slow start phase begins and ends at 0 and ends around .1 or .15. Congestion control happens around .3 and continues every vertical bar of 6 packets until about 5.2. The graph that was pulled from wireshark doesn't look like the slow start exponential graphs shown in the book. This graph has a zigzag like pattern that is uniform.

14. Answer Question 13 for the trace that you captured when you transferred a file from your ***own*** computer to gaia.cs.umass.edu



Based on the trace I pulled, there is a slow start and then appears to grow much faster and somewhat exponentially. This is quite different than the graph provided from the zip file. The window in my trace doesn't decrease so it is closer to an ideal scenario with no congestion control.