CISC 335 – Computer Networks – Winter 2021
Wireshark Assignment 1: TCP

Due March 12, 2021, 11:59 pm

Objective

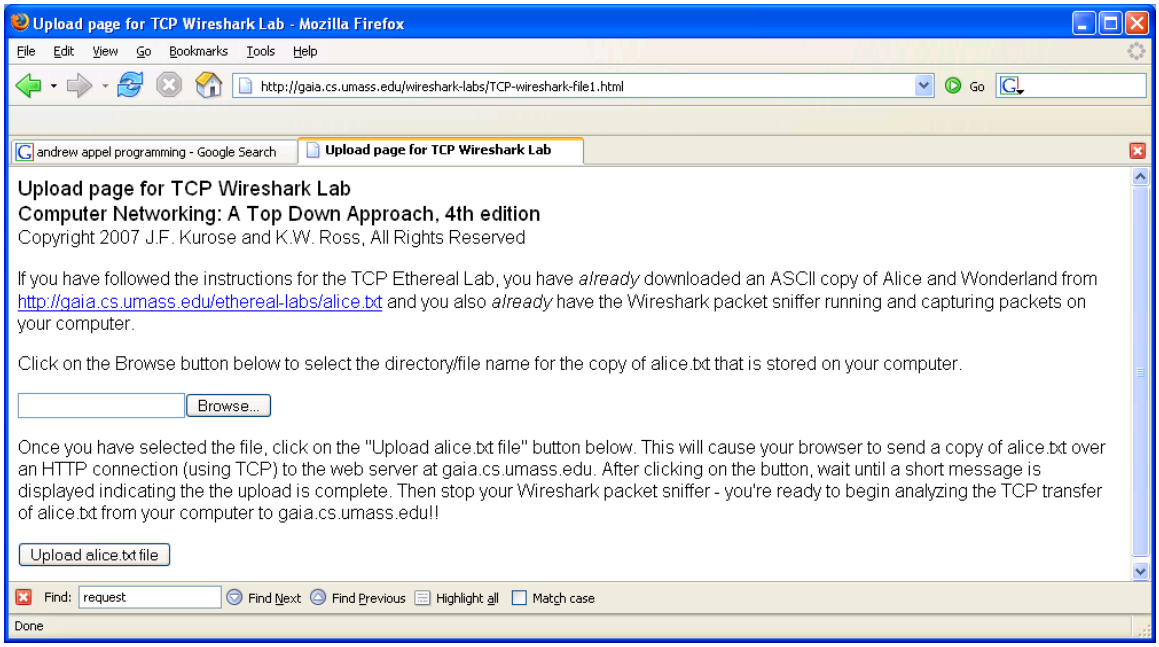
In this lab, you will capture a bulk TCP segment transfer from your computer to a remote server to explore the TCP protocol.

Whenever possible, when answering a question, you should hand in a printout or a screenshot of the packet(s) within the trace that you used to answer the question asked. Annotate the printout or screenshot to explain your answer. To print a packet, use *File->Print*, choose *Selected packet only*, choose *Packet summary line*, and select the minimum amount of packet detail that you need to answer the question (or take a screenshot showing this minimum amount of packet detail).

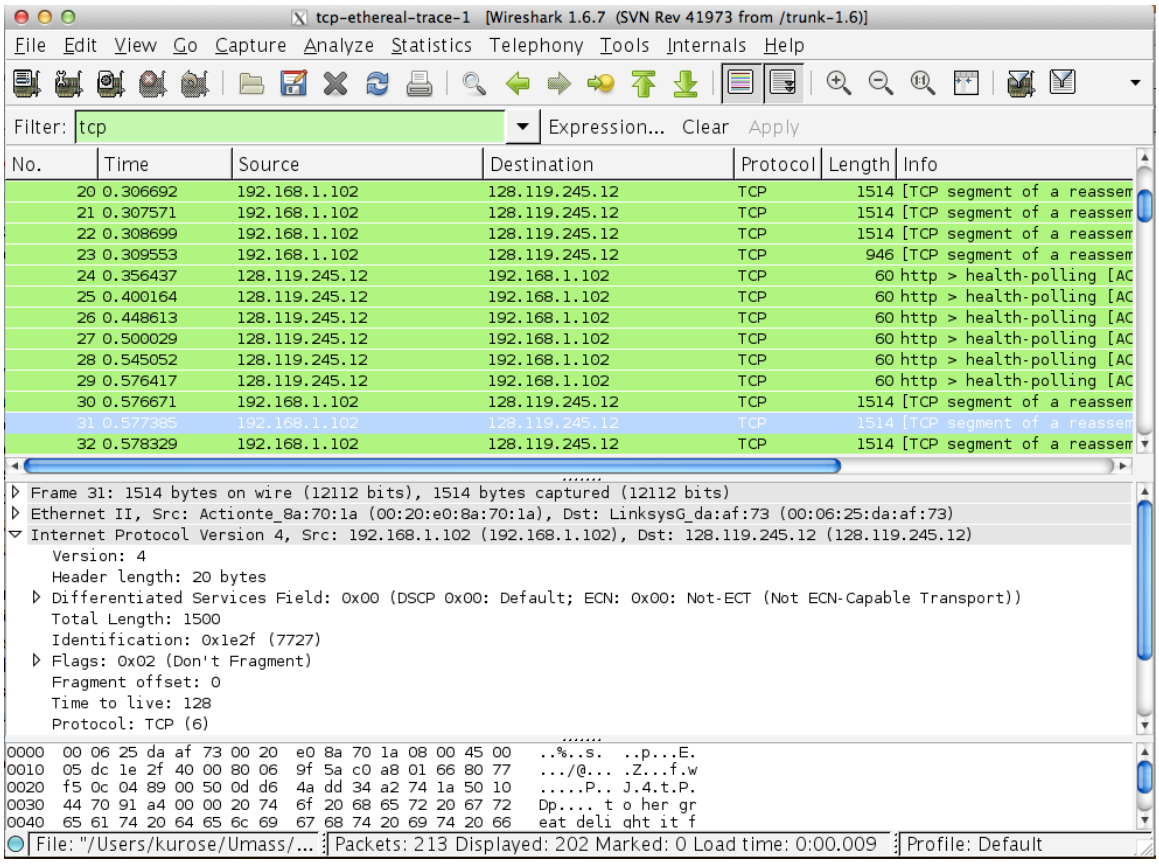
Capturing a bulk of TCP Segment Transfer

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:



- 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.



If you face any problems with the aforementioned method, please do the following:
Download the zip file http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip and extract. The traces in this zip file were collected by Wireshark running on the author's computers. Please load the trace tcp-ethereal-trace-1 into Wireshark and view the trace using the File pull down menu, choosing Open, and then selecting the tcp-ethereal-trace-1 trace file.

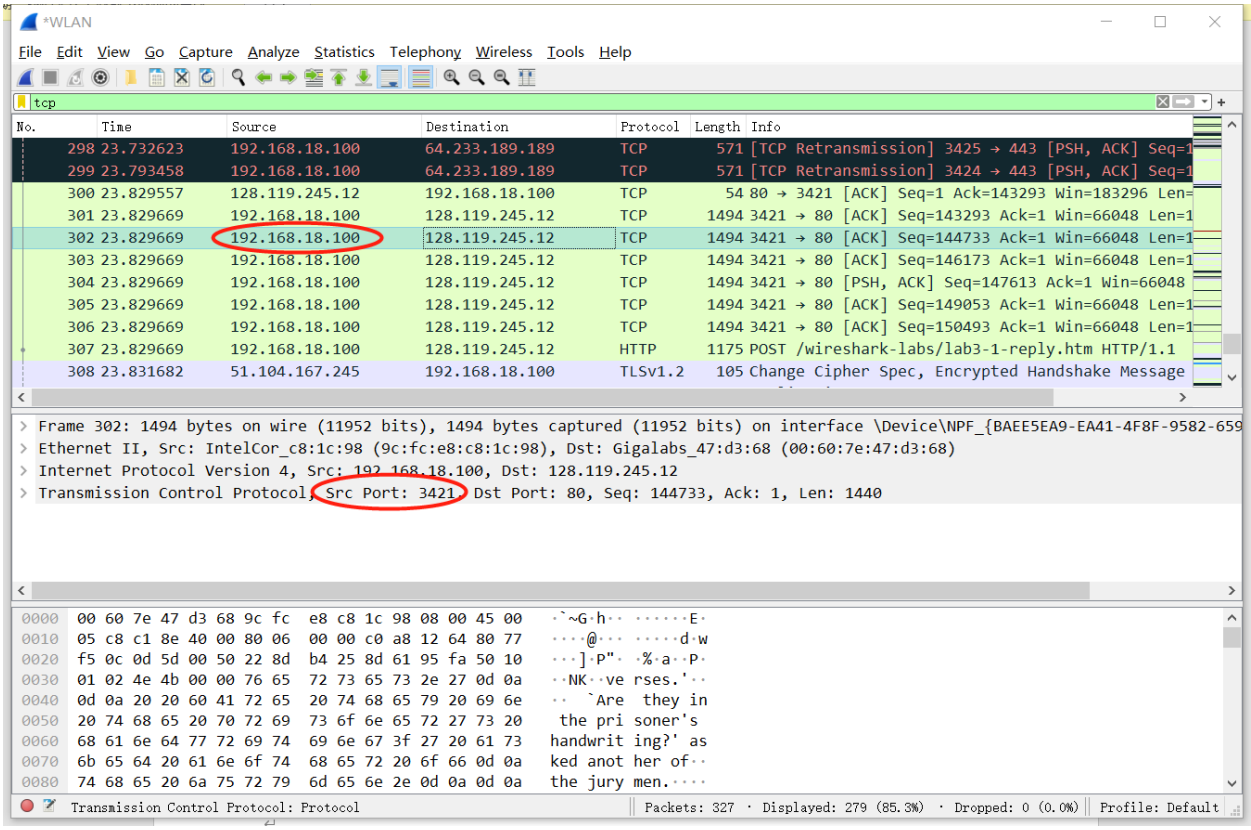
What you should see is a series of TCP and HTTP messages between your computer and gaia.cs.umass.edu. You should see the initial three-way handshake containing a SYN message. You should see an HTTP POST message being sent from your computer to gaia.cs.umass.edu. You should also see TCP ACK segments being returned from gaia.cs.umass.edu to your computer.

Recall that you can filter the packets displayed in the Wireshark window based on protocol, so you can enter “tcp” for example (or “http”) to view these packets.

Please answer the following questions based on the displayed trace:

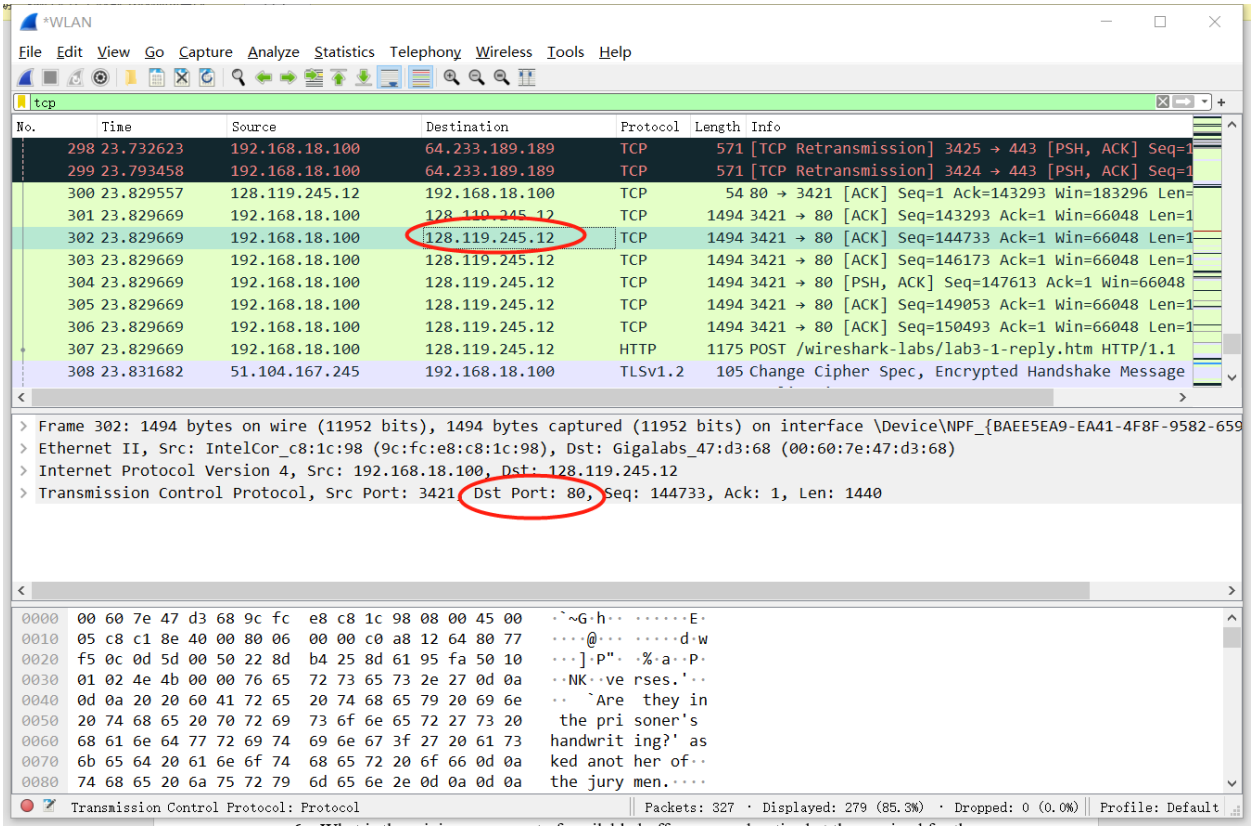
1. What is the IP address and TCP port number used by your client computer (source) to transfer the file to gaia.cs.umass.edu? (Please include a screenshot and highlight your answer on that screenshot).

IP address: **192.168.18.100**
TCP port number: **3421**



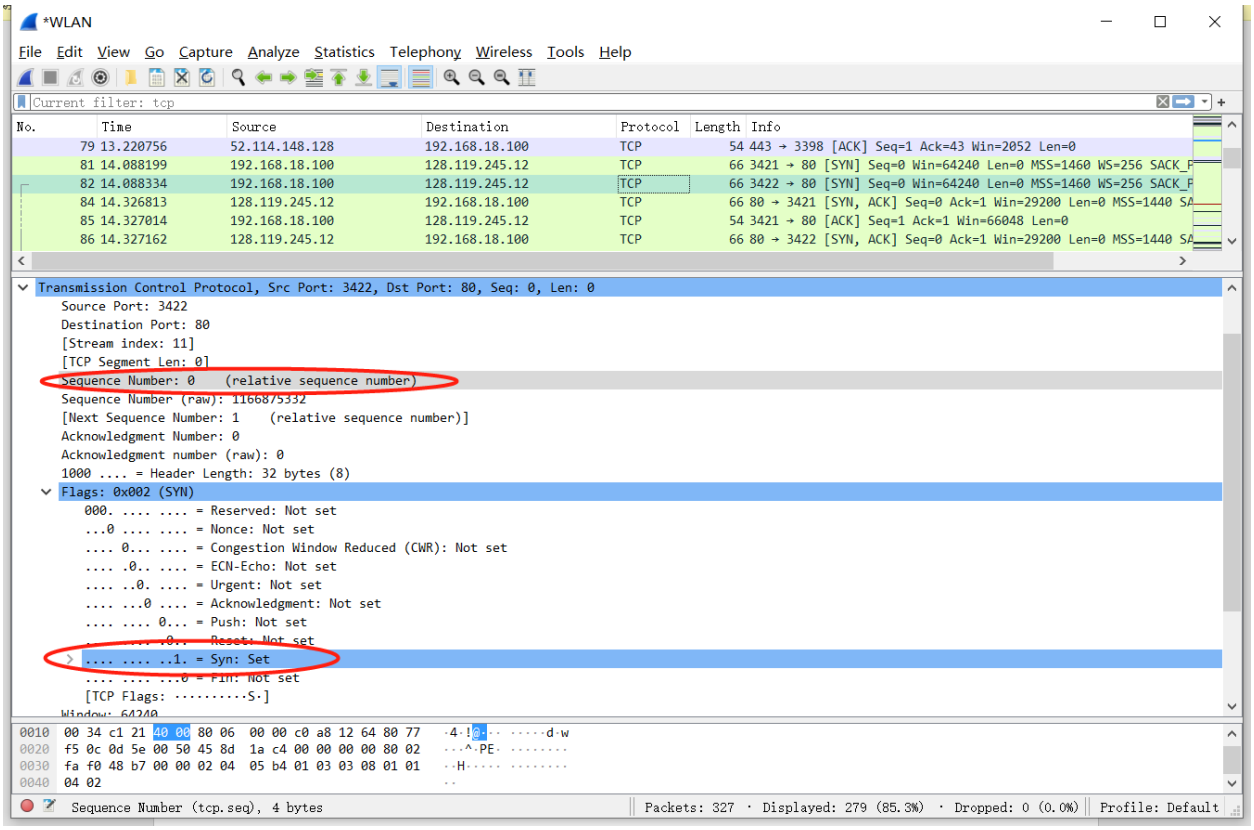
2. What is the IP address of gaia.cs.umass.edu? On what port number is it sending and receiving TCP segments for this connection? (Please include a screenshot and highlight your answer on that screenshot)

IP address: **128.119.245.12**
Port number: **80**



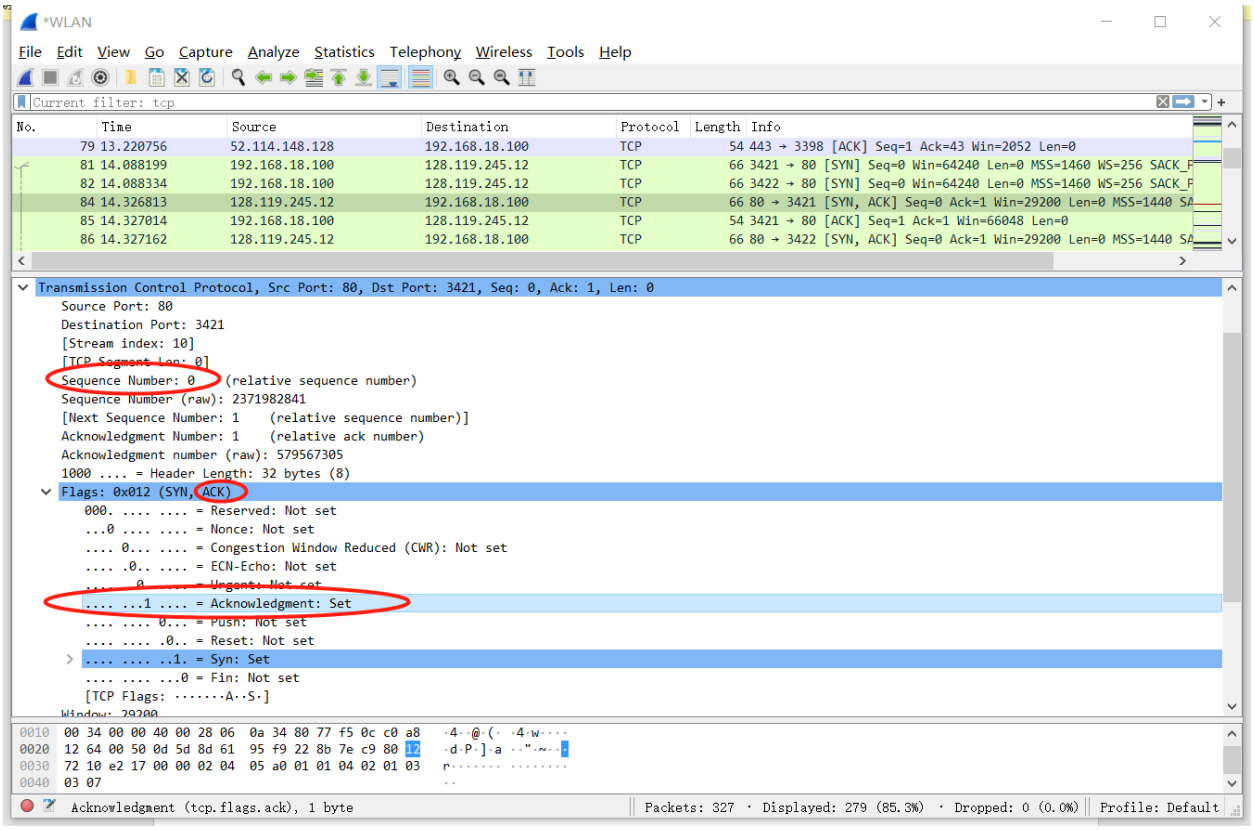
3. What is the sequence number of the TCP SYN segment that is used to initiate the TCP connection between the client computer and gaia.cs.umass.edu (Please include a screenshot and highlight your answer on that screenshot)? What is it in the segment that identifies the segment as a SYN segment?

Sequence number: 0
There's an SYN flag which identifies this, and its detail shows that its 'Syn' field is set to 1, as circled below. This identifies the segment as a SYN segment.



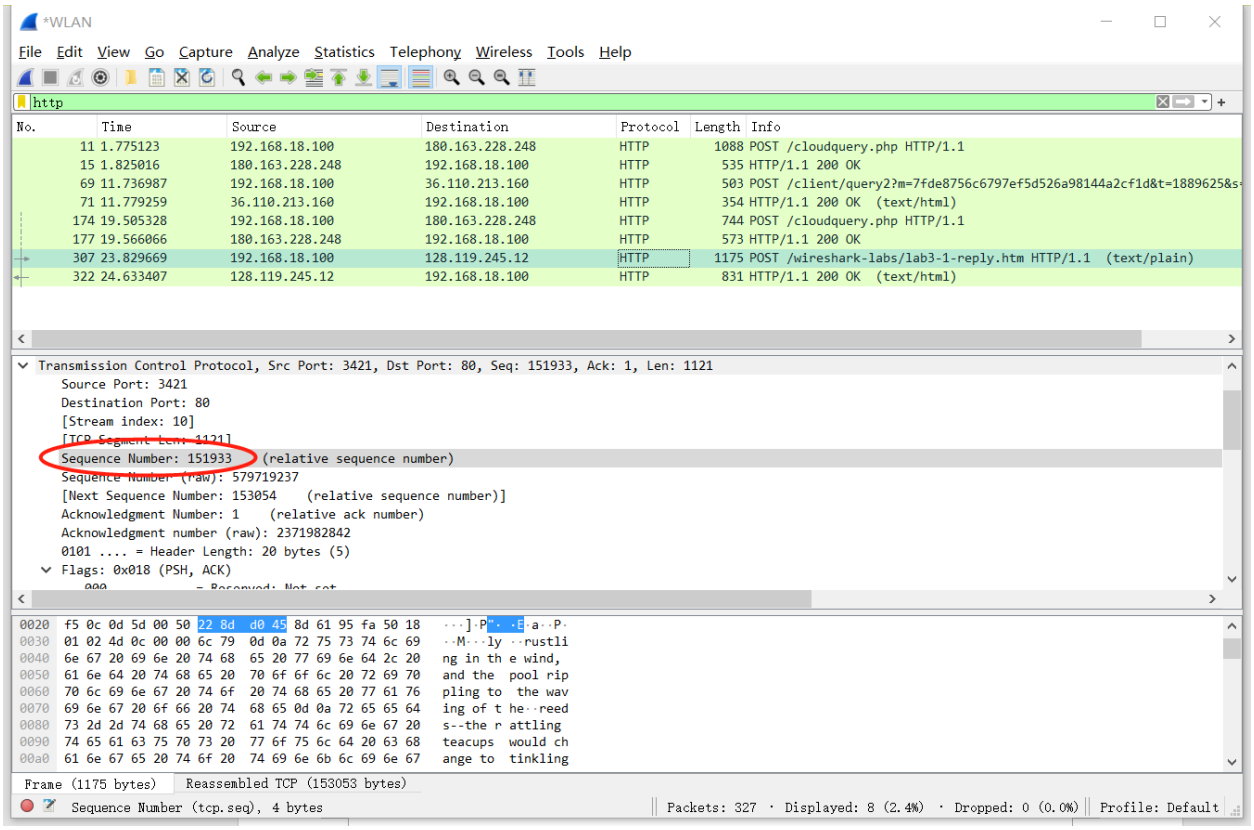
4. What is the sequence number of the SYNACK segment sent by gaia.cs.umass.edu to the client computer in reply to the SYN? What is the value of the Acknowledgement field in the SYNACK segment? (Please include a screenshot and highlight your answer on that screenshot) What is it in the segment that identifies the segment as a SYNACK segment?

Sequence number: **0**
There's an ACK flag which identifies this connection has been acknowledged, and its detail shows that its acknowledgment field is set to 1, as circled below. This identifies the segment as a SYNACK segment.



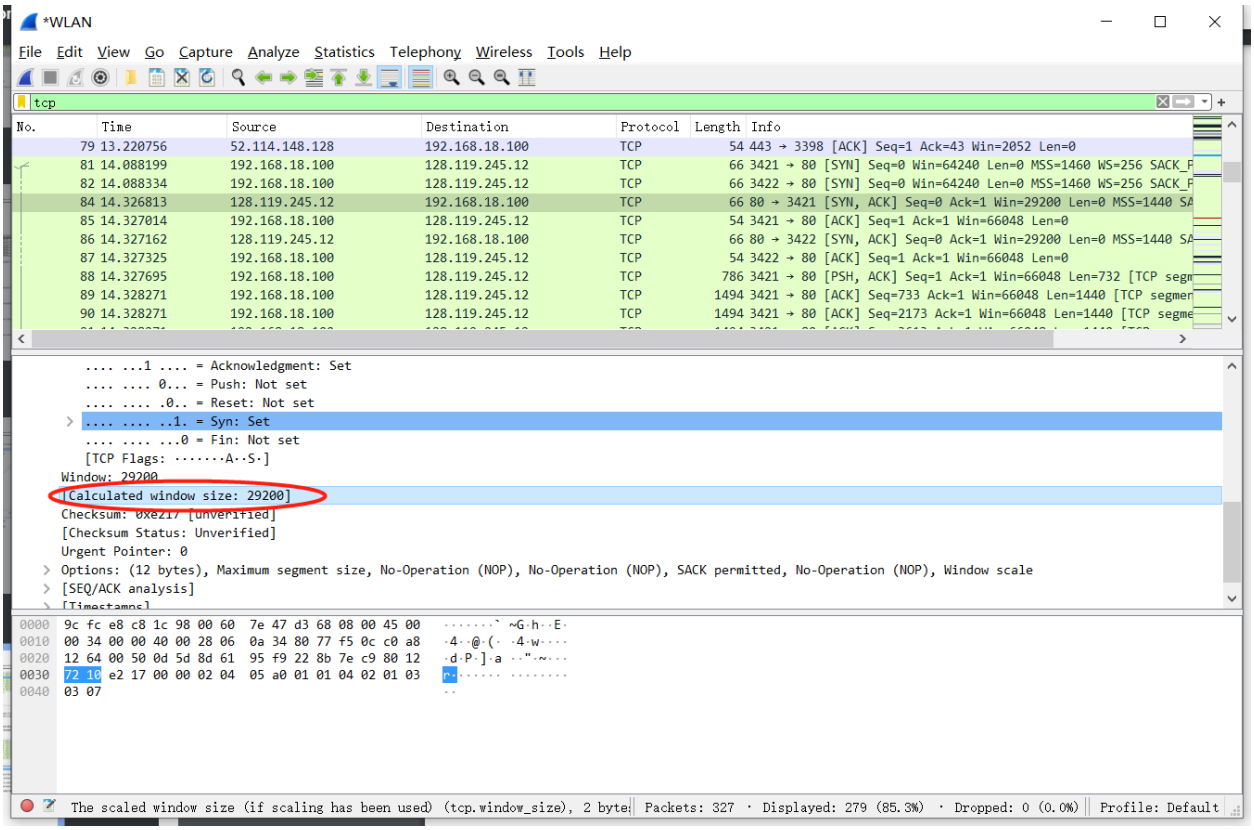
5. What is the sequence number of the TCP segment containing the HTTP POST command? (Please include a screenshot and highlight your answer on that screenshot).

Sequence number: **151933**



6. What is the minimum amount of available buffer space advertised at the received for the entire trace? (Please include a screenshot and highlight your answer on that screenshot)

Minimum amount of available buffer space: **29200 bytes.**
This is indicated by the first ACK from the server. The screenshot is shown below.

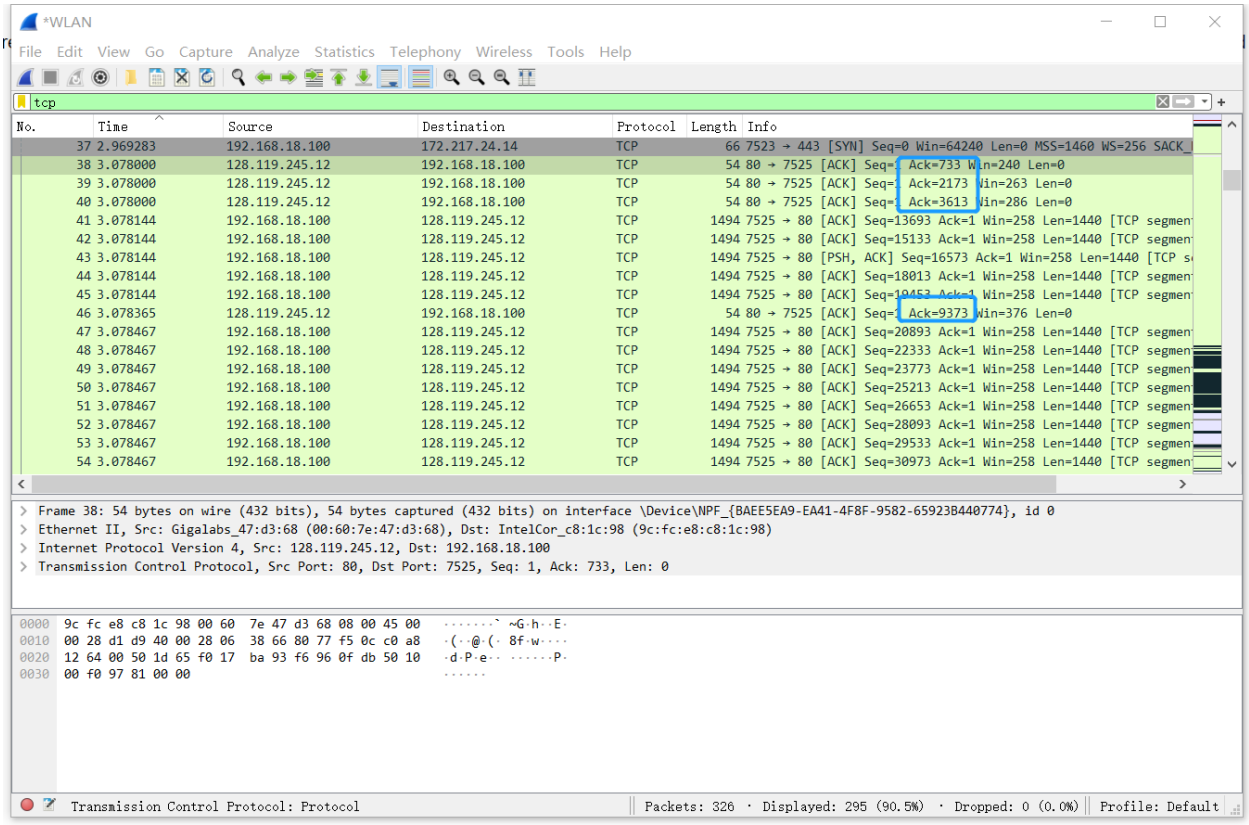


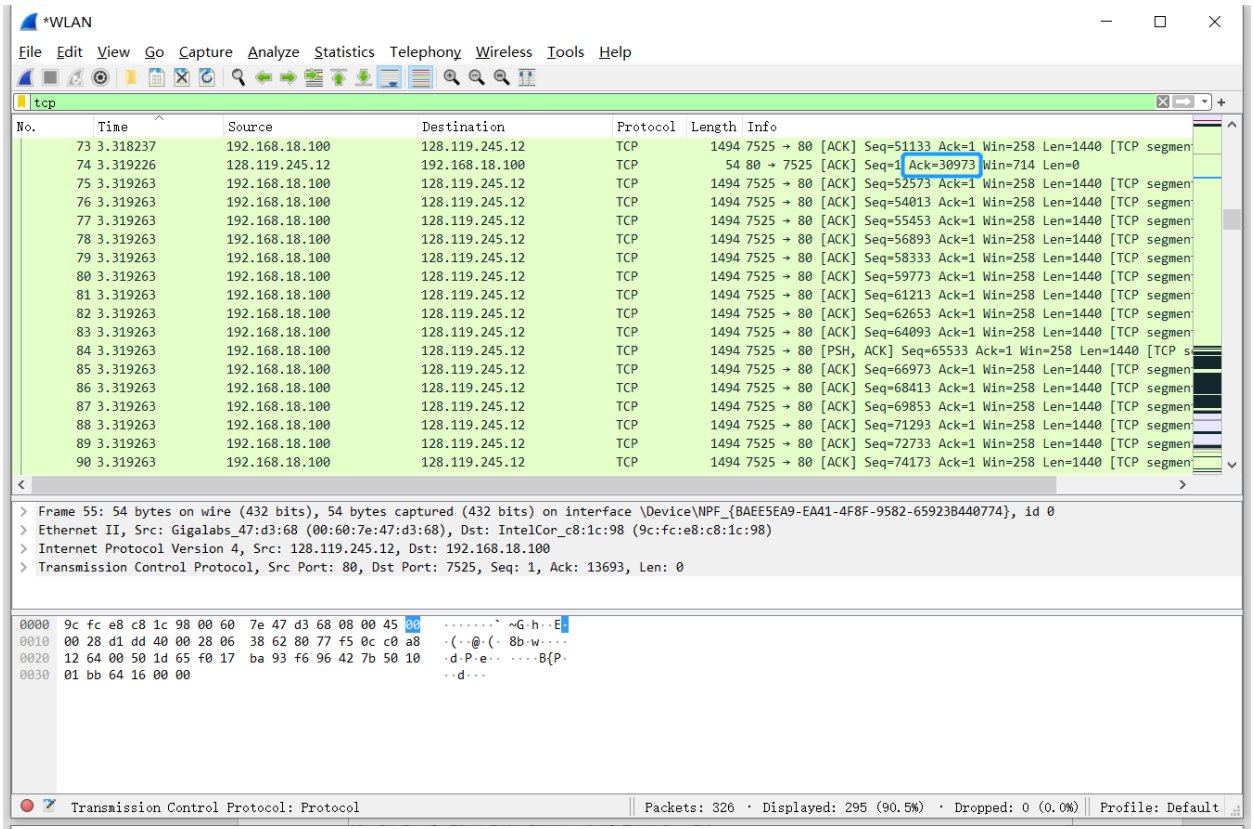
7. How much data does the receiver typically acknowledge in an ACK? To answer this question, please record the acknowledgment number and the acknowledged data of the first 12 acknowledgments generated by the receiver in the table below and state the data size that appears the most? (Please include a screenshot and highlight your answer on that screenshot)

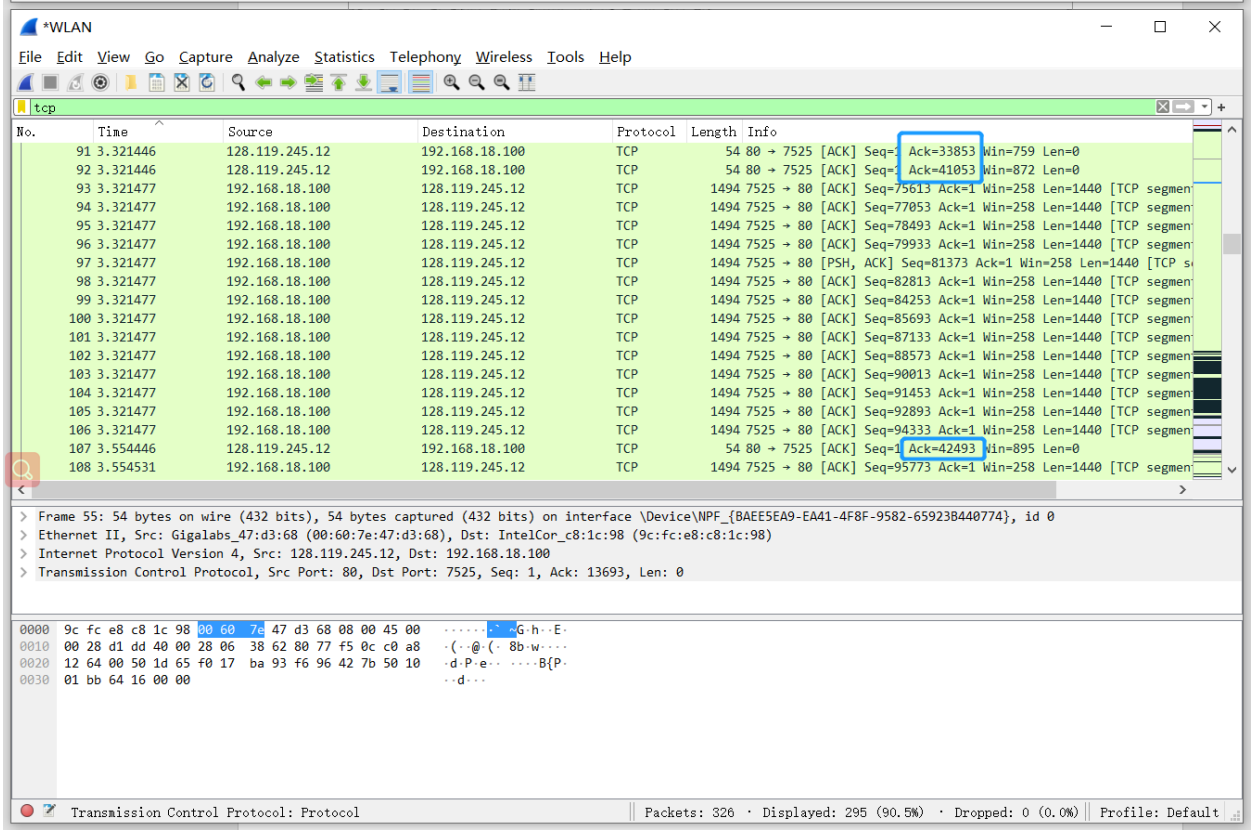
	acknowledged sequence number	acknowledged data
ACK 1	733	733
ACK 2	2173	1440
ACK 3	3613	1440
ACK 4	9373	5760
ACK 5	13693	4320
ACK 6	15133	1440
ACK 7	16573	1440
ACK 8	18013	1440
ACK 9	19453	1440
ACK 10	30973	11520
ACK 11	33853	2880
ACK 12	41053	7200

Base on the above table, I use the difference between the two acknowledged sequence numbers to calculate the acknowledged data on the right. However, it was strange to me that the TCP connections rarely show consecutive ACKs – they are mostly spread out, as shown below, which leads to the above uncommon results, as shown in the above table. The possible reason is probably because I am browsing overseas, and some possible timeout may occur, leading different inconsecutive rows. However, there are some rows where consecutive ACKs are shown either, and these rows' acknowledged data are of 1440.

Hence, based on this information, the data the receiver typically acknowledge in an ACK is 1440 bytes.




*WLAN
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
tcp
No. Time Source Destination Protocol Length Info
73 3.318237 192.168.18.100 128.119.245.12 TCP 1494 7525 → 80 [ACK] Seq=51133 Ack=1 Win=258 Len=1440 [TCP segmen
74 3.319226 128.119.245.12 192.168.18.100 TCP 54 80 → 7525 [ACK] Seq=1 Ack=30973 Win=714 Len=0
75 3.319263 192.168.18.100 128.119.245.12 TCP 1494 7525 → 80 [ACK] Seq=52573 Ack=1 Win=258 Len=1440 [TCP segmen
76 3.319263 192.168.18.100 128.119.245.12 TCP 1494 7525 → 80 [ACK] Seq=54013 Ack=1 Win=258 Len=1440 [TCP segmen
77 3.319263 192.168.18.100 128.119.245.12 TCP 1494 7525 → 80 [ACK] Seq=55453 Ack=1 Win=258 Len=1440 [TCP segmen
78 3.319263 192.168.18.100 128.119.245.12 TCP 1494 7525 → 80 [ACK] Seq=56893 Ack=1 Win=258 Len=1440 [TCP segmen
79 3.319263 192.168.18.100 128.119.245.12 TCP 1494 7525 → 80 [ACK] Seq=58333 Ack=1 Win=258 Len=1440 [TCP segmen
80 3.319263 192.168.18.100 128.119.245.12 TCP 1494 7525 → 80 [ACK] Seq=59773 Ack=1 Win=258 Len=1440 [TCP segmen
81 3.319263 192.168.18.100 128.119.245.12 TCP 1494 7525 → 80 [ACK] Seq=61213 Ack=1 Win=258 Len=1440 [TCP segmen
82 3.319263 192.168.18.100 128.119.245.12 TCP 1494 7525 → 80 [ACK] Seq=62653 Ack=1 Win=258 Len=1440 [TCP segmen
83 3.319263 192.168.18.100 128.119.245.12 TCP 1494 7525 → 80 [ACK] Seq=64093 Ack=1 Win=258 Len=1440 [TCP segmen
84 3.319263 192.168.18.100 128.119.245.12 TCP 1494 7525 → 80 [PSH, ACK] Seq=65533 Ack=1 Win=258 Len=1440 [TCP s
85 3.319263 192.168.18.100 128.119.245.12 TCP 1494 7525 → 80 [ACK] Seq=66973 Ack=1 Win=258 Len=1440 [TCP segmen
86 3.319263 192.168.18.100 128.119.245.12 TCP 1494 7525 → 80 [ACK] Seq=68413 Ack=1 Win=258 Len=1440 [TCP segmen
87 3.319263 192.168.18.100 128.119.245.12 TCP 1494 7525 → 80 [ACK] Seq=69853 Ack=1 Win=258 Len=1440 [TCP segmen
88 3.319263 192.168.18.100 128.119.245.12 TCP 1494 7525 → 80 [ACK] Seq=71293 Ack=1 Win=258 Len=1440 [TCP segmen
89 3.319263 192.168.18.100 128.119.245.12 TCP 1494 7525 → 80 [ACK] Seq=72733 Ack=1 Win=258 Len=1440 [TCP segmen
90 3.319263 192.168.18.100 128.119.245.12 TCP 1494 7525 → 80 [ACK] Seq=74173 Ack=1 Win=258 Len=1440 [TCP segmen
Frame 55: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface \Device\NPF_{BAEE5EA9-EA41-4F8F-9582-65923B440774}, id 0
Ethernet II, Src: Gigalabs_47:d3:68 (00:60:7e:47:d3:68), Dst: IntelCor_c8:1c:98 (9c:fc:e8:c8:1c:98)
Internet Protocol Version 4, Src: 128.119.245.12, Dst: 192.168.18.100
Transmission Control Protocol, Src Port: 80, Dst Port: 7525, Seq: 1, Ack: 13693, Len: 0
0000 9c fc e8 c8 1c 98 00 60 7e 47 d3 68 08 00 45 00
0010 00 28 d1 dd 40 00 28 06 38 62 80 77 f5 0c c0 a8
0020 12 64 00 50 1d 65 f0 17 ba 93 f6 96 42 7b 50 10
0030 01 bb 64 16 00 00
Transmission Control Protocol: Protocol
Packets: 326 · Displayed: 295 (90.5%) · Dropped: 0 (0.0%)
Profile: Default


*WLAN
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
tcp
No. Time Source Destination Protocol Length Info
91 3.321446 128.119.245.12 192.168.18.100 TCP 54 80 → 7525 [ACK] Seq=1 Ack=33853 Win=759 Len=0
92 3.321446 128.119.245.12 192.168.18.100 TCP 54 80 → 7525 [ACK] Seq=1 Ack=41053 Win=872 Len=0
93 3.321477 192.168.18.100 128.119.245.12 TCP 1494 7525 → 80 [ACK] Seq=75613 Ack=1 Win=258 Len=1440 [TCP segmen
94 3.321477 192.168.18.100 128.119.245.12 TCP 1494 7525 → 80 [ACK] Seq=77053 Ack=1 Win=258 Len=1440 [TCP segmen
95 3.321477 192.168.18.100 128.119.245.12 TCP 1494 7525 → 80 [ACK] Seq=78493 Ack=1 Win=258 Len=1440 [TCP segmen
96 3.321477 192.168.18.100 128.119.245.12 TCP 1494 7525 → 80 [ACK] Seq=79933 Ack=1 Win=258 Len=1440 [TCP segmen
97 3.321477 192.168.18.100 128.119.245.12 TCP 1494 7525 → 80 [PSH, ACK] Seq=81373 Ack=1 Win=258 Len=1440 [TCP s
98 3.321477 192.168.18.100 128.119.245.12 TCP 1494 7525 → 80 [ACK] Seq=82813 Ack=1 Win=258 Len=1440 [TCP segmen
99 3.321477 192.168.18.100 128.119.245.12 TCP 1494 7525 → 80 [ACK] Seq=84253 Ack=1 Win=258 Len=1440 [TCP segmen
100 3.321477 192.168.18.100 128.119.245.12 TCP 1494 7525 → 80 [ACK] Seq=85693 Ack=1 Win=258 Len=1440 [TCP segmen
101 3.321477 192.168.18.100 128.119.245.12 TCP 1494 7525 → 80 [ACK] Seq=87133 Ack=1 Win=258 Len=1440 [TCP segmen
102 3.321477 192.168.18.100 128.119.245.12 TCP 1494 7525 → 80 [ACK] Seq=88573 Ack=1 Win=258 Len=1440 [TCP segmen
103 3.321477 192.168.18.100 128.119.245.12 TCP 1494 7525 → 80 [ACK] Seq=90013 Ack=1 Win=258 Len=1440 [TCP segmen
104 3.321477 192.168.18.100 128.119.245.12 TCP 1494 7525 → 80 [ACK] Seq=91453 Ack=1 Win=258 Len=1440 [TCP segmen
105 3.321477 192.168.18.100 128.119.245.12 TCP 1494 7525 → 80 [ACK] Seq=92893 Ack=1 Win=258 Len=1440 [TCP segmen
106 3.321477 192.168.18.100 128.119.245.12 TCP 1494 7525 → 80 [ACK] Seq=94333 Ack=1 Win=258 Len=1440 [TCP segmen
107 3.554446 128.119.245.12 192.168.18.100 TCP 54 80 → 7525 [ACK] Seq=1 Ack=42493 Win=895 Len=0
108 3.554531 192.168.18.100 128.119.245.12 TCP 1494 7525 → 80 [ACK] Seq=95773 Ack=1 Win=258 Len=1440 [TCP segmen
Frame 55: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface \Device\NPF_{BAEE5EA9-EA41-4F8F-9582-65923B440774}, id 0
Ethernet II, Src: Gigalabs_47:d3:68 (00:60:7e:47:d3:68), Dst: IntelCor_c8:1c:98 (9c:fc:e8:c8:1c:98)
Internet Protocol Version 4, Src: 128.119.245.12, Dst: 192.168.18.100
Transmission Control Protocol, Src Port: 80, Dst Port: 7525, Seq: 1, Ack: 13693, Len: 0
0000 9c fc e8 c8 1c 98 00 60 7e 47 d3 68 08 00 45 00
0010 00 28 d1 dd 40 00 28 06 38 62 80 77 f5 0c c0 a8
0020 12 64 00 50 1d 65 f0 17 ba 93 f6 96 42 7b 50 10
0030 01 bb 64 16 00 00
Transmission Control Protocol: Protocol
Packets: 326 · Displayed: 295 (90.5%) · Dropped: 0 (0.0%)
Profile: Default