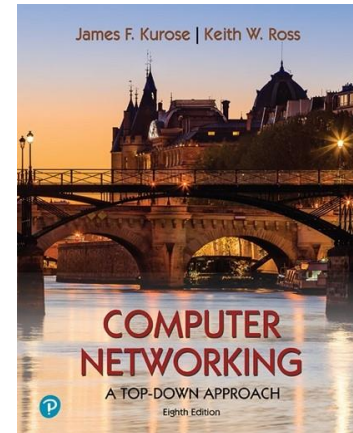


# Wireshark Lab: TCP v8.0

Supplement to *Computer Networking: A Top-Down Approach, 8<sup>th</sup> 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

© 2005-2020, J.F Kurose and K.W. Ross, All Rights Reserved



---

Student Name : Nguyễn Hữu Hiếu

Student ID : 2013149

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?

```

http
No.    Time    Source                                Destination                                Protocol    Length  Info
-----
20328 2023-02-28 22:13:10.961469 172.17.34.133 128.119.245.12 HTTP      534 GET /wireshark-labs/TCP-wireshark-file1.html HTTP/1.1
162   2023-02-28 22:11:11.997266 172.17.34.133 128.119.245.12 HTTP      519 GET /wireshark-labs/alice.txt HTTP/1.1
20357 2023-02-28 22:13:11.225607 128.119.245.12 172.17.34.133 HTTP      808 HTTP/1.1 200 OK (text/html)
25161 2023-02-28 22:14:07.897876 128.119.245.12 172.17.34.133 HTTP      831 HTTP/1.1 200 OK (text/html)
25133 2023-02-28 22:14:07.611372 172.17.34.133 128.119.245.12 HTTP      460 POST /wireshark-labs/lab3-1-reply.htm HTTP/1.1 (text/plain)

> Frame 25133: 460 bytes on wire (3680 bits), 460 bytes captured (3680 bits) on interface \Device\NPF_{5C0000
0010 01 be b2 c0 40 00 80 06 00 00 ac 11 22 85 04 00
0020 f5 0c de 5c 00 50 52 f8 52 ad 68 75 cb f9 50 12
0030 02 85 45 cb 00 00 72 20 61 62 6f 75 74 0d 65 21
0040 65 72 20 6f 74 68 65 72 20 6c 69 74 74 6c 65 21
0050 63 68 69 6c 54 72 65 6e 2c 20 61 6e 64 20 6d 6e
0060 6b 65 20 54 48 45 49 52 20 65 79 65 73 20 62 67
0070 69 67 68 74 20 61 6e 64 20 65 61 67 65 72 0d 00
0080 77 69 64 74 68 20 6d 61 6e 79 20 61 20 73 74 72 6e
0090 6e 67 65 20 74 61 6c 65 2c 20 70 65 72 68 61 20
00a0 73 20 65 76 65 6e 20 77 69 74 68 20 74 68 65 21
00b0 64 72 65 61 6d 20 6f 66 0d 0a 57 6f 6e 64 65 7f
00c0 6c 61 6e 64 20 6f 66 20 68 6f 6e 67 20 61 67 6e
00d0 3a 20 20 61 6e 64 20 68 6f 77 20 73 68 65 20 7f
00e0 6f 75 6c 64 20 66 65 65 65 6c 20 77 69 74 68 20 6e
00f0 6c 6c 20 74 68 65 69 72 0d 0a 73 69 6d 70 6c 6c

```

The IP address used by the client computer that is transferring the file to gaia.cs.umass.edu is 172.17.34.133

The TCP port is 56924

- What is the IP address of `gaia.cs.umass.edu`? On what port number is it sending and receiving TCP segments for this connection?

No.	Time	Source	Destination	Protocol	Length	Info
20328	2023-02-28 22:13:10.961469	172.17.34.133	128.119.245.12	HTTP	534	GET /wireshark-labs/TCP-wireshark-file1.html HTTP/1.1
162	2023-02-28 22:11:11.997266	172.17.34.133	128.119.245.12	HTTP	519	GET /wireshark-labs/alice.txt HTTP/1.1
20357	2023-02-28 22:13:11.225607	128.119.245.12	172.17.34.133	HTTP	808	HTTP/1.1 200 OK (text/html)
+ 25161	2023-02-28 22:14:07.897876	128.119.245.12	172.17.34.133	HTTP	831	HTTP/1.1 200 OK (text/html)
+ 25133	2023-02-28 22:14:07.611372	172.17.34.133	128.119.245.12	HTTP	460	POST /wireshark-labs/lab3.1-reply.htm HTTP/1.1 (text/plain)
>	Frame 25161: 831 bytes on wire (6648 bits), 831 bytes captured (6648 bits) on interface \Device\NPF_{5C0E0A4D-...}					0020 22 85 80 5d de 5c 68 75 cb bf 92 88 54 43 50 1
>	Ethernet II, Src: 5a:6e:0c:24:6c:47, Dst: IntelCor_b8:0f:ec (00:42:38:b8:0f:ec)					0030 05 86 5c ad d0 00 48 54 50 2f 31 2e 21 20 3
>	Internet Protocol Version 4, Src: 128.119.245.12, Dst: 172.17.34.133					0040 30 30 20 4f 4b 0d 0a 4d 61 74 65 3a 20 54 7
>	Transmission Control Protocol, Src Port: 80, Dst Port: 56924, Seq: 1, Ack: 153039, Len: 777					2c 20 32 38 46 05 6e 62 20 32 30 32 33 20 31
>	Source Port: 80					0060 3a 31 34 3a 30 37 20 47 ad 54 0d 0a 53 65 72 7
>	Destination Port: 56924					0070 65 72 3a 20 41 70 61 63 68 65 2f 32 2e 34 2e 3
>	[Stream index: 113]					0080 20 28 43 65 6e 74 4f 53 29 20 4f 70 75 6e 53 5
>	[Conversation completeness: Complete, WITH_DATA (31)]					0090 4c 2f 31 2e 30 2e 32 6b 2d 66 69 70 73 20 50 4
>	[TCP Segment Len: 777]					00a0 50 2f 32 2e 3a 33 31 20 6d 6f 64 5f 70 65 7
>						00b0 6c 2f 32 2e 30 2e 31 20 20 65 72 6c 2f 76 3

The IP address of `gaia.cs.umass.edu` is 128.119.245.12

The TCP port is 80

- What is the IP address and TCP port number used by your client computer (source) to transfer the file to gaia.cs.umass.edu?

No.	Time	Source	Destination	Protocol	Length	Info
20328	2023-02-28 22:13:10.961469	172.17.34.133	128.119.245.12	HTTP	534	GET /wireshark-labs/TCP-wireshark-file1.html HTTP/1.1
162	2023-02-28 22:11:11.997266	172.17.34.133	128.119.245.12	HTTP	519	GET /wireshark-labs/alice.txt HTTP/1.1
20357	2023-02-28 22:13:11.225607	128.119.245.12	172.17.34.133	HTTP	808	HTTP/1.1 200 OK (text/html)
25161	2023-02-28 22:14:07.897876	128.119.245.12	172.17.34.133	HTTP	831	HTTP/1.1 200 OK (text/html)
25133	2023-02-28 22:14:07.611372	172.17.34.133	128.119.245.12	HTTP	460	POST /wireshark-labs/lab3-1-reply.htm HTTP/1.1 (text/plain)

> Frame 25133: 460 bytes on wire (3680 bits), 460 bytes captured (3680 bits) on interface \Device\NPF_{5C000000-0000-0000-0000-000000000000}	0000 5a 6e 0c 24 6c 47 00 42 38 b8 0f ec 08 00 45 00
> Ethernet II, Src: IntelCor_b8:0f:ec (00:42:38:b8:0f:ec), Dst: 5a:6e:0c:24:6c:47 (5a:6e:0c:24:6c:47)	0010 01 be b2 c0 40 00 80 06 00 00 ac 11 22 85 80 7
> Internet Protocol Version 4, Src: 172.17.34.133, Dst: 128.119.245.12	0020 f5 0c de 5c 00 50 f2 88 52 ad 68 75 cb f9 50 1
> Transmission Control Protocol, Src Port: 56924, Dst Port: 80, Seq: 152633, Ack: 1, Len: 406	0030 02 05 45 cb 00 00 72 20 61 62 6f 75 74 0d 0a 6
Source Port: 56924	0040 65 72 20 6f 74 68 65 72 20 6c 69 74 74 6c 65 2
Destination Port: 80	0050 63 68 69 6c 64 72 65 6e 2c 20 61 6e 64 20 6d 6
[Stream index: 113]	0060 6b 65 20 54 48 45 49 52 20 65 79 65 73 20 62 7
[Conversation completeness: Complete, WITH_DATA (31)]	0070 69 67 68 74 20 61 6e 64 20 65 61 67 65 72 0d 0
[TCP Segment Len: 406]	0080 77 69 74 68 20 6d 61 6e 79 20 61 20 73 74 72 6
Sequence Number: 152633 (relative sequence number)	0090 6e 67 65 20 74 61 6c 65 2c 20 70 65 72 68 61 7
Sequence Number (raw): 4069020333	00a0 73 20 65 76 65 6e 20 77 69 74 68 20 74 68 65 2
[Next Sequence Number: 153039 (relative sequence number)]	00b0 64 72 65 61 6d 20 6f 66 0d 0a 57 6f 6e 64 65 7
Acknowledgment Number: 1 (relative ack number)	00c0 6c 61 6e 64 20 6f 66 20 6c 6f 6e 67 20 61 67 6
Acknowledgment number (raw): 1752550393	00d0 3a 20 20 61 6e 64 20 68 6f 77 20 73 68 65 20 7
0101 .... = Header Length: 20 bytes (5)	00e0 6f 75 6c 64 20 66 65 65 6c 20 77 69 74 68 20 6
	00f0 6c 6c 20 74 68 65 69 72 0d 0a 73 69 6d 70 6c 6

The IP address used by the client computer that is transferring the file to gaia.cs.umass.edu is 172.17.34.133

The TCP port is 56924

- 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?

74	2023-02-28 22:11:11.739536	172.17.34.133	128.119.245.12	TCP	66	56793 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
75	2023-02-28 22:11:11.739674	172.17.34.133	128.119.245.12	TCP	66	56794 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
76	2023-02-28 22:11:11.740051	172.17.34.133	8.8.8.8	TCP	66	56795 → 53 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
77	2023-02-28 22:11:11.740148	172.17.34.133	8.8.8.8	TCP	66	56796 → 53 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
78	2023-02-28 22:11:11.766103	8.8.8.8	172.17.34.133	TCP	66	53 → 56792 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1412 SACK_PERM WS=256
79	2023-02-28 22:11:11.766193	172.17.34.133	8.8.8.8	TCP	54	56792 → 53 [ACK] Seq=1 Ack=1 Win=131072 Len=0
80	2023-02-28 22:11:11.766297	172.17.34.133	8.8.8.8	TCP	56	56792 → 53 [PSH, ACK] Seq=1 Ack=1 Win=131072 Len=2 [TCP segment of a reassembled
82	2023-02-28 22:11:11.771200	8.8.8.8	172.17.34.133	TCP	66	53 → 56791 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1412 SACK_PERM WS=256
83	2023-02-28 22:11:11.771302	172.17.34.133	8.8.8.8	TCP	54	56791 → 53 [ACK] Seq=1 Ack=1 Win=131072 Len=0

[Conversation completeness: Complete, WITH_DATA (31)]	0000 5a 6e 0c 24 6c 47 00 42 38 b8 0f ec 08 00 45 00
[TCP Segment Len: 0]	0010 00 34 b2 04 40 00 80 06 00 00 ac 11 22 85 80 77
Sequence Number: 0 (relative sequence number)	0020 f5 0c dd d9 00 50 13 3a a4 e3 00 00 00 00 00 02
Sequence Number (raw): 322610403	0030 fa f0 44 a1 00 00 02 04 05 b4 01 03 03 08 01 01
[Next Sequence Number: 1 (relative sequence number)]	0040 04 02
Acknowledgment Number: 0	
Acknowledgment number (raw): 0	
1000 .... = Header Length: 32 bytes (8)	
> Flags: 0x002 (SYN)	
000. .... = Reserved: Not set	
...0 .... = Accurate ECN: Not set	
....0... = Congestion Window Reduced: Not set	
....0... = ECN-Echo: Not set	
.....0. = Urgent: Not set	
.....0. = Acknowledgment: Not set	
.....0.. = Push: Not set	
.....0.. = Reset: Not set	
> .....1. = Syn: Set	
.....0 = Fin: Not set	
[TCP Flags: .....S.]	
Window: 64240	
[Calculated window size: 64240]	

The sequence number of the TCP SYN segment is 0 since it is used to imitate the TCP connection between the client computer and gaia.cs.umass.edu. According to the screenshot below, in the Flags section, the SYN flag is set to 1 which indicates that this segment is a SYN segment

- 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?

76	2023-02-28	22:11:11.740051	172.17.34.133	8.8.8.8	TCP	66	56795 → 53 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
77	2023-02-28	22:11:11.740148	172.17.34.133	8.8.8.8	TCP	66	56796 → 53 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
78	2023-02-28	22:11:11.766103	8.8.8.8	172.17.34.133	TCP	66	53 → 56792 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1412 SACK_PERM WS=256
79	2023-02-28	22:11:11.766193	172.17.34.133	8.8.8.8	TCP	54	56792 → 53 [ACK] Seq=1 Ack=1 Win=131072 Len=0
80	2023-02-28	22:11:11.766297	172.17.34.133	8.8.8.8	TCP	56	56792 → 53 [PSH, ACK] Seq=1 Ack=1 Win=131072 Len=2 [TCP segment of a reassembled PDU]
82	2023-02-28	22:11:11.771200	8.8.8.8	172.17.34.133	TCP	66	53 → 56791 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1412 SACK_PERM WS=256
83	2023-02-28	22:11:11.771302	172.17.34.133	8.8.8.8	TCP	54	56791 → 53 [ACK] Seq=1 Ack=1 Win=131072 Len=0
84	2023-02-28	22:11:11.771429	172.17.34.133	8.8.8.8	TCP	56	56791 → 53 [PSH, ACK] Seq=1 Ack=1 Win=131072 Len=2 [TCP segment of a reassembled PDU]
86	2023-02-28	22:11:11.775963	8.8.8.8	172.17.34.133	TCP	66	53 → 56796 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1412 SACK_PERM WS=256
87	2023-02-28	22:11:11.776469	172.17.34.133	8.8.8.8	TCP	54	56796 → 53 [ACK] Seq=1 Ack=1 Win=131072 Len=0

[Conversation completeness: Complete, WITH_DATA (31)]	
[TCP Segment Len: 0]	
Sequence Number: 0	(relative sequence number)
Sequence Number (raw): 3146449026	
[Next Sequence Number: 1 (relative sequence number)]	
Acknowledgment Number: 1 (relative ack number)	
Acknowledgment number (raw): 3377543177	
1000 .... = Header Length: 32 bytes (8)	
▼ Flags: 0x012 (SYN, ACK)	
000. .... = Reserved: Not set	
...0. .... = Accurate ECN: Not set	
....0. .... = Congestion Window Reduced: Not set	
....0. .... = ECN-Echo: Not set	
....0. .... = Urgent: Not set	
...0...1. .... = Acknowledgment: Set	
....0. .... = Push: Not set	
....0. .... = Reset: Not set	
...0...1. .... = Syn: Set	
....0. .... = Fin: Not set	
[TCP Flags: .....A..S.]	

The sequence number of the SYN\_ACK segment sent by gaia.cs.umass.edu to the client computer in reply to the SYN is 0.

The value of the acknowledgement field in the SYN\_ACK segment is determined by the server gaia.cs.umass.edu.

The server adds 1 to the initial sequence number of the SYN segment from the client computer.

For this case, the initial sequence number of the SYN segment from the client computer is 0, thus the value of the acknowledgement field in the SYN\_ACK segment is 1. A segment will be identified as a SYN\_ACK segment if both SYN flag and ACKnowledgement flag in the segment are set to 1.

- 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.

Time	Source	Destination	Protocol	Length	Info
1	2023-02-28 23:00:42.1716..	172.17.34.133	TCP	1494	58665 → 80 [ACK] Seq=19437 Ack=1 Win=517 Len=1440 [TCP segment of a reassembled PDU]
2	2023-02-28 23:00:42.1716..	172.17.34.133	TCP	1494	58665 → 80 [ACK] Seq=17997 Ack=1 Win=517 Len=1440 [TCP segment of a reassembled PDU]
3	2023-02-28 23:00:42.1716..	172.17.34.133	TCP	1494	[TCP Retransmission] 80 → 58665 [ACK] Seq=1 Ack=2157 Win=477 Len=0 SLE=717 SRE=1441 SLE=3597 SRE=16557
4	2023-02-28 23:00:42.1715..	128.119.245.12	TCP	66	80 → 58665 [ACK] Seq=1 Ack=2157 Win=500 Len=0 SLE=3597 SRE=17997
5	2023-02-28 23:00:42.1715..	128.119.245.12	TCP	74	[TCP Window Update] 80 → 58665 [ACK] Seq=1 Ack=1 Win=454 Len=0 SLE=3597 SRE=16557 SLE=717 SRE=2157
6	2023-02-28 23:00:41.8637..	172.17.34.133	TCP	1494	58665 → 80 [PSH, ACK] Seq=16557 Ack=1 Win=517 Len=1440 [TCP segment of a reassembled PDU]
7	2023-02-28 23:00:41.8637..	172.17.34.133	TCP	1494	[TCP Out-Of-Order] 58665 → 80 [PSH, ACK] Seq=1 Ack=1 Win=517 Len=1440
8	2023-02-28 23:00:41.8635..	172.17.34.133	TCP	1494	58665 → 80 [ACK] Seq=15117 Ack=1 Win=517 Len=1440 [TCP segment of a reassembled PDU]
9	2023-02-28 23:00:41.8635..	172.17.34.133	TCP	1494	58665 → 80 [ACK] Seq=13677 Ack=1 Win=517 Len=1440 [TCP segment of a reassembled PDU]
10	2023-02-28 23:00:41.8635..	128.119.245.12	TCP	74	[TCP Window Update] 80 → 58665 [ACK] Seq=1 Ack=1 Win=409 Len=0 SLE=3597 SRE=13677 SLE=717 SRE=2157
11	2023-02-28 23:00:41.8635..	128.119.245.12	TCP	74	[TCP Window Update] 80 → 58665 [ACK] Seq=1 Ack=1 Win=431 Len=0 SLE=3597 SRE=13677 SLE=717 SRE=2157

Frame 719: 1494 bytes on wire (11952 bits), 1494 bytes captured (11952 bits) on interface DeviceVWF_{SCAAKBA-31ED-}	
Ethernet II, Src: IntelCor.B8:0F:ec (00:42:38:b8:0f:ec), Dst: Sa:6e:0c:24:6c:47 (Sa:6e:0c:24:6c:47)	
Internet Protocol Version 4, Src: 172.17.34.133, Dst: 128.119.245.12	
▼ Transmission Control Protocol, Src Port: 58665, Dst Port: 80, Seq: 1, Ack: 1, Len: 1440	
Source Port: 58665	
Destination Port: 80	
[Stream index: 32]	
[Conversation completeness: Incomplete (12)]	
[TCP Segment Len: 1440]	
Sequence Number: 1 (relative sequence number)	
Sequence Number (raw): 130797161	
[Next Sequence Number: 1441 (relative sequence number)]	
Acknowledgment Number: 1 (relative ack number)	
Acknowledgment number (raw): 17577437	
0101 .... = Header Length: 20 bytes (5)	
Flags: 0x018 (PSH, ACK)	
Window: 517	
[Calculated window size: 517]	
[Window size scaling factor: -1 (unknown)]	
Checksum: 0x49d5 [unverified]	
[Checksum Status: Unverified]	
Urgent Pointer: 0	
[Timestamps]	
[SEQ/ACK analysis]	
TCP payload (1440 bytes)	
Retransmitted TCP segment data (1440 bytes)	

0000	5a 6e 0c 24 6c 47 00 42 38 b8 0f ec 00 00 ac 11 22 85 80 77	Zn \$ig-B 8.....E
0010	05 c8 b2 de 40 00 80 06 00 00 ac 11 22 85 80 77	...@.....w
0020	f5 0c e5 20 00 50 07 cb ce 69 01 0c 35 dd 50 18	...).P...i..S.P..
0030	02 05 49 d5 00 00 50 4f 53 54 20 2f 77 69 72 65	...I...P...S.../wire
0040	73 68 61 72 6b 2d 6c 61 62 73 2f 6c 61 62 31 2d	shark-la bs/ta3-
0050	31 2d 62 65 70 6c 79 2e 68 74 6d 20 48 54 50	1-reply. hts HTTP
0060	2f 31 2e 31 0d 0a 48 6f 73 74 3a 20 67 61 69 61	/1.1. Ho st: gaia
0070	2e 63 73 2e 75 6d 61 73 73 2e 65 64 75 0d 0a 43	.cs.umass s.edu <C
0080	6f 6e 6e 65 63 74 69 6f 6e 3a 20 6b 65 65 70 2d	connectio n: keep-
0090	61 6c 69 76 65 0d 0a 43 6f 6e 74 65 6e 74 2d 4c	alive -C ontent-L
00a0	65 6e 67 74 68 3a 20 31 35 32 33 32 32 0d 0a 43	length: 1 52322-C
00b0	61 63 68 65 2d 43 6f 6e 74 72 6f 6c 3a 20 6d 61	ache-Con trol: ma
00c0	70 2d 61 67 65 3d 30 0d 0a 55 70 67 72 61 6d 65	x-age-0 Upgrade
00d0	2d 49 6e 73 65 63 75 72 65 2d 52 65 71 75 65 73	-Insecur e-Reques
00e0	74 73 3a 20 31 0d 0a 4f 72 69 6f 69 6e 3a 20 68	ts: 1-0 rigin: h
00f0	74 74 70 3a 2f 2f 67 61 69 61 2e 63 73 2e 75 6d	ttp://ga ia.cs.um
0100	61 73 73 2e 65 64 75 0d 0a 43 6f 6e 74 65 6e 74	ass.edu -Content
0110	2d 54 79 70 65 3a 20 6d 75 6c 74 69 70 61 72 74	-Type: m ultipart
0120	2f 66 6f 72 6d 2d 6d 61 74 61 3b 20 62 6f 75 6e	/form-da taj, bouu
0130	6d 61 72 79 3d 2d 2d 2d 2d 57 65 62 4b 69 74 46	darys-...WebKitF
0140	6f 72 6d 42 6f 75 6e 64 61 72 79 53 74 50 73 73	onbround arySfPss
0150	41 30 31 71 4e 55 33 45 68 65 58 0d 0a 55 73 65	A01qW3E heX- Use
0160	72 2d 41 67 65 6e 74 3a 20 4d 6f 7a 69 6e 6c 61	r-Agent: Mozilla
0170	2f 35 2e 30 20 28 57 69 6e 6d 6f 77 73 20 4e 54	/5.0 (X; ndoes HT
0180	20 31 30 2e 30 3b 20 57 69 6e 36 34 3b 20 78 36	10.0; W in64; x6
0190	34 29 20 41 70 70 6c 65 57 65 62 4b 69 74 2f 35	4) Apple WebKit/5
01a0	33 37 2e 33 36 20 28 4b 48 54 4d 4c 2c 20 6c 69	37.36 (K HTML, /1
01b0	6b 65 20 47 65 63 6b 6f 29 20 43 68 72 6f 6d 65	ke Gecko ) Chrome
01c0	2f 31 31 30 2e 30 2e 30 2e 30 20 53 61 66 61 72	/110.0.0 .0 Safari
01d0	69 2f 35 33 37 2e 33 36 0d 0a 41 63 63 65 70 74	/537.36 --Accept
01e0	3a 20 74 65 78 74 2f 68 74 6d 6c 2c 61 70 70 6c	: text/h tml,appl

The sequence number of the TCP segment containing the HTTP Post command is 1.

- 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*.

The image shows a Wireshark packet capture of a TCP connection. The packet list at the top shows several segments. The packet details pane on the right shows the details of a selected segment (Seq: 1). The details include:

- Frame 694: 770 bytes on wire (6160 bits), 770 bytes captured (6160 bits) on interface \Device\NPF\_{SCA4AC0A-31ED-4C8E}
- Ethernet II, Src: IntelCor\_b8:0f:ec (00:42:38:b8:0f:ec), Dst: Sa:6e:0c:24:6c:47 (5a:6e:0c:24:6c:47)
- Internet Protocol Version 4, Src: 172.17.34.133, Dst: 128.119.245.12
- Transmission Control Protocol, Src Port: 58665, Dst Port: 80, Seq: 1, Ack: 1, Len: 716
  - Source Port: 58665
  - Destination Port: 80
  - [Stream index: 32]
  - [Conversation completeness: Incomplete (12)]
  - [TCP Segment Len: 716]
  - Sequence Number: 1 (relative sequence number)
  - Sequence Number (raw): 130797161
  - Next Sequence Number: 717 (relative sequence number)
  - Acknowledgment Number: 1 (relative ack number)
  - Acknowledgment number (raw): 17577437
  - 0101 .... = Header Length: 20 bytes (5)
  - Flags: 0x018 (PSH, ACK)
  - Window: 517
  - [Calculated window size: 517]
  - [Window size scaling factor: -1 (unknown)]
  - Checksum: 0xd701 [unverified]

Sequence number for segment 1 is 1, sequence number for segment 2 is 717.

Seg1: Arrival Time: Feb 28, 2023 23:00:41.548950000 SE Asia Standard Time

Seg2: Arrival Time: Feb 28, 2023 23:00:41.549135000 SE Asia Standard Time



8. What is the length of each of the first six TCP segments?<sup>1</sup>

	Time	Source	Destination	Protocol	Length	Info
703	2023-02-28 23:00:41.549135	172.17.34.133	128.119.245.12	TCP	1494	58665 → 80 [ACK] Seq=12237 Ack=1 Win=517 Len=1440 [TCP segment of a reassembled PDU]
702	2023-02-28 23:00:41.549135	172.17.34.133	128.119.245.12	TCP	1494	58665 → 80 [ACK] Seq=10797 Ack=1 Win=517 Len=1440 [TCP segment of a reassembled PDU]
701	2023-02-28 23:00:41.549135	172.17.34.133	128.119.245.12	TCP	1494	58665 → 80 [ACK] Seq=9357 Ack=1 Win=517 Len=1440 [TCP segment of a reassembled PDU]
700	2023-02-28 23:00:41.549135	172.17.34.133	128.119.245.12	TCP	1494	58665 → 80 [ACK] Seq=7917 Ack=1 Win=517 Len=1440 [TCP segment of a reassembled PDU]
699	2023-02-28 23:00:41.549135	172.17.34.133	128.119.245.12	TCP	1494	58665 → 80 [ACK] Seq=6477 Ack=1 Win=517 Len=1440 [TCP segment of a reassembled PDU]
698	2023-02-28 23:00:41.549135	172.17.34.133	128.119.245.12	TCP	1494	58665 → 80 [ACK] Seq=5037 Ack=1 Win=517 Len=1440 [TCP segment of a reassembled PDU]
697	2023-02-28 23:00:41.549135	172.17.34.133	128.119.245.12	TCP	1494	58665 → 80 [ACK] Seq=3597 Ack=1 Win=517 Len=1440 [TCP segment of a reassembled PDU]
696	2023-02-28 23:00:41.549135	172.17.34.133	128.119.245.12	TCP	1494	58665 → 80 [ACK] Seq=2157 Ack=1 Win=517 Len=1440 [TCP segment of a reassembled PDU]
695	2023-02-28 23:00:41.549135	172.17.34.133	128.119.245.12	TCP	1494	58665 → 80 [ACK] Seq=717 Ack=1 Win=517 Len=1440 [TCP segment of a reassembled PDU]

the length of each of the first six TCP segments is 1440

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?

703	2023-02-28 23:00:41.549135	172.17.34.133	128.119.245.12	TCP	1494	58665 → 80 [ACK] Seq=12237 Ack=1 Win=517 Len=1440 [TCP segment of a reassembled PDU]
702	2023-02-28 23:00:41.549135	172.17.34.133	128.119.245.12	TCP	1494	58665 → 80 [ACK] Seq=10797 Ack=1 Win=517 Len=1440 [TCP segment of a reassembled PDU]
701	2023-02-28 23:00:41.549135	172.17.34.133	128.119.245.12	TCP	1494	58665 → 80 [ACK] Seq=9357 Ack=1 Win=517 Len=1440 [TCP segment of a reassembled PDU]
700	2023-02-28 23:00:41.549135	172.17.34.133	128.119.245.12	TCP	1494	58665 → 80 [ACK] Seq=7917 Ack=1 Win=517 Len=1440 [TCP segment of a reassembled PDU]
699	2023-02-28 23:00:41.549135	172.17.34.133	128.119.245.12	TCP	1494	58665 → 80 [ACK] Seq=6477 Ack=1 Win=517 Len=1440 [TCP segment of a reassembled PDU]
698	2023-02-28 23:00:41.549135	172.17.34.133	128.119.245.12	TCP	1494	58665 → 80 [ACK] Seq=5037 Ack=1 Win=517 Len=1440 [TCP segment of a reassembled PDU]
697	2023-02-28 23:00:41.549135	172.17.34.133	128.119.245.12	TCP	1494	58665 → 80 [ACK] Seq=3597 Ack=1 Win=517 Len=1440 [TCP segment of a reassembled PDU]
696	2023-02-28 23:00:41.549135	172.17.34.133	128.119.245.12	TCP	1494	58665 → 80 [ACK] Seq=2157 Ack=1 Win=517 Len=1440 [TCP segment of a reassembled PDU]
695	2023-02-28 23:00:41.549135	172.17.34.133	128.119.245.12	TCP	1494	58665 → 80 [ACK] Seq=717 Ack=1 Win=517 Len=1440 [TCP segment of a reassembled PDU]
694	2023-02-28 23:00:41.548950	172.17.34.133	128.119.245.12	TCP	770	58665 → 80 [PSH, ACK] Seq=1 Ack=1 Win=517 Len=716 [TCP segment of a reassembled PDU]

the minimum amount of available buffer space advertised at the received for the entire trace is 517

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

No there is no retransmitted segments in the trace file

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).

According to the screenshot below, we can see that the ACK numbers increase in the sequence of 1401, 2801, 4201, and so on. The ACK numbers increases by 1400 each time, indicating that the receiver is acknowledging 1400 bytes

933	2023-02-28 23:00:44.629370	128.119.245.12	172.17.34.133	TCP	60	80 → 58665 [ACK] Seq=1 Ack=115917 Win=2057 Len=0
934	2023-02-28 23:00:44.629370	128.119.245.12	172.17.34.133	TCP	60	80 → 58665 [ACK] Seq=1 Ack=120237 Win=2125 Len=0
935	2023-02-28 23:00:44.629370	128.119.245.12	172.17.34.133	TCP	60	80 → 58665 [ACK] Seq=1 Ack=127437 Win=2237 Len=0
936	2023-02-28 23:00:44.629370	128.119.245.12	172.17.34.133	TCP	60	80 → 58665 [ACK] Seq=1 Ack=128877 Win=2260 Len=0

<sup>1</sup> The TCP segments in the tcp-ethereal-trace-1 trace file are all less than 1460 bytes. This is because the computer on which the trace was gathered has an Ethernet card that limits the length of the maximum IP packet to 1500 bytes (40 bytes of TCP/IP header data and 1460 bytes of TCP payload). This 1500 byte value is the standard maximum length allowed by Ethernet. If your trace indicates a TCP length greater than 1500 bytes, and your computer is using an Ethernet connection, then Wireshark is reporting the wrong TCP segment length; it will likely also show only one large TCP segment rather than multiple smaller segments. Your computer is indeed probably sending multiple smaller segments, as indicated by the ACKs it receives. This inconsistency in reported segment lengths is due to the interaction between the Ethernet driver and the Wireshark software. We recommend that if you have this inconsistency, that you perform this lab using the provided trace file.