

Computer Networks CSE 5344 Project 2
Transmission Control Protocol Analysis using Wireshark
Name: Sushmitha Nagarajan
Student ID: 1001556348

Objective:

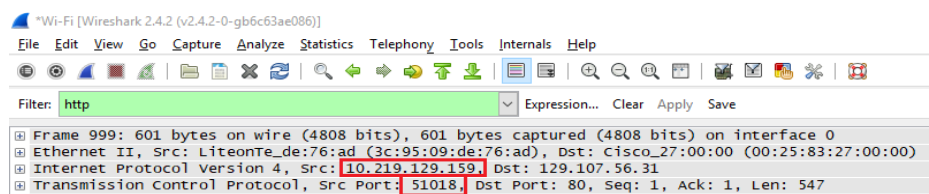
Problem Set 1:

1. What is the IP address and TCP port number used by your client computer (source) to browse the page uta.edu? Use the 'GET' message to answer the following questions.

Ans:

IP address used by the client: 10.219.129.159

TCP Port number used by the client: 51018



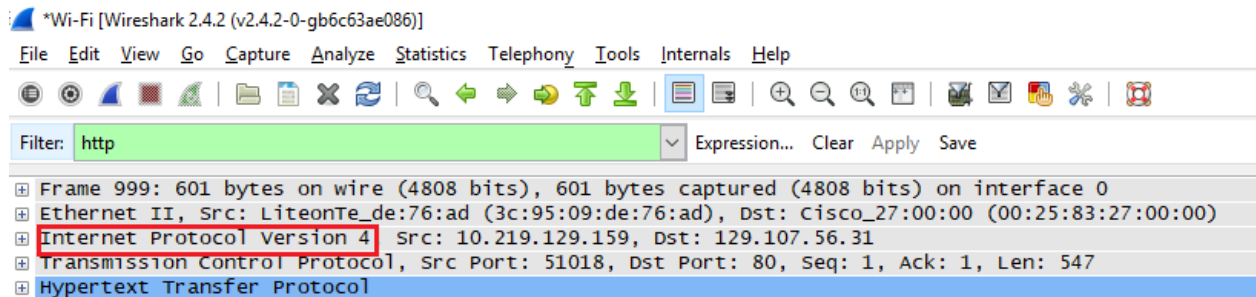
2. What is the TTL value that is used in this communication?

Ans. TTL: 128 secs

```
Internet Protocol Version 4, Src: 10.219.129.159, Dst: 129.107.56.31
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 587
    Identification: 0x4d43 (19779)
  Flags: 0x02 (Don't Fragment)
    0... .... = Reserved bit: Not set
    .1.. .... = Don't fragment: Set
    ..0. .... = More fragments: Not set
  Fragment offset: 0
  Time to live: 128
```

3. Did you use IPV4 or IPV6 for communication?

Ans. The client uses IPV4 for communication.

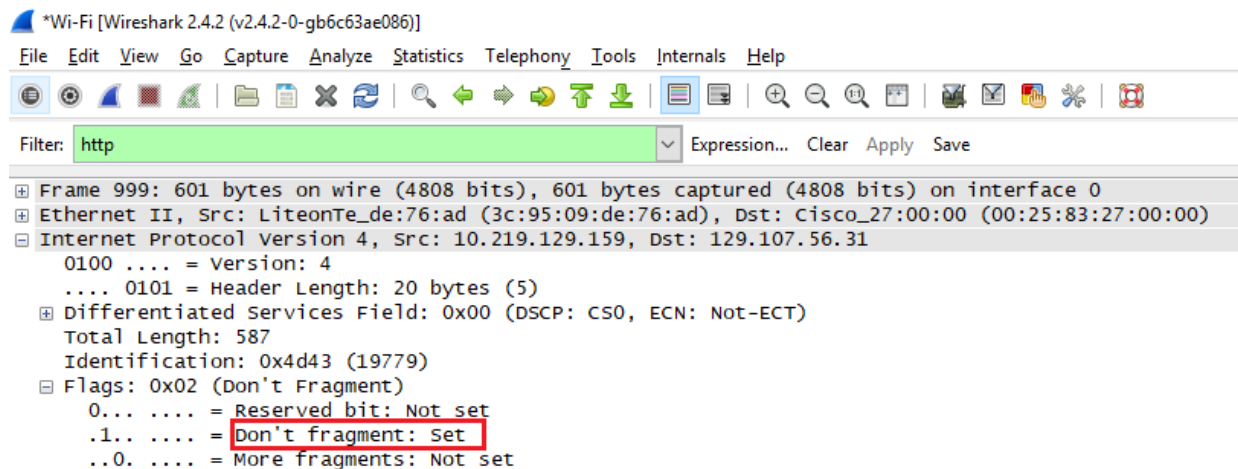


4 Do your optional field has some information or not?

Ans: The optional field is not available in HTTP segment.

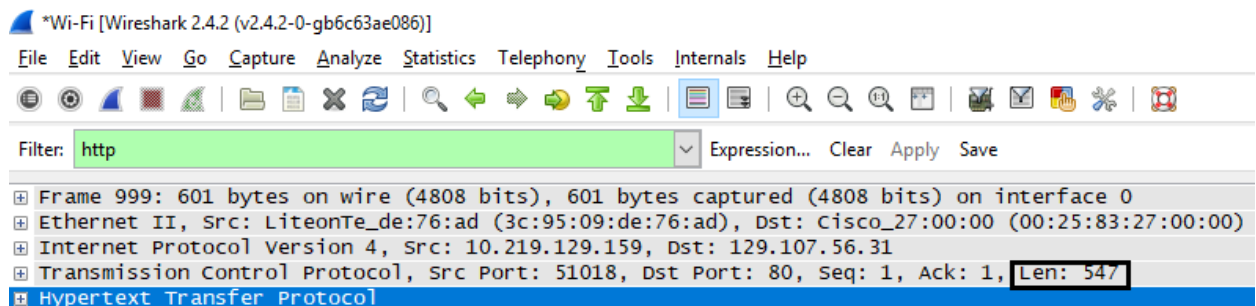
5. Is the Packet Fragmented

Ans. Packet is not fragmented as the 'Don't Fragment' packet is Set.



6. What is the TCP segment length?

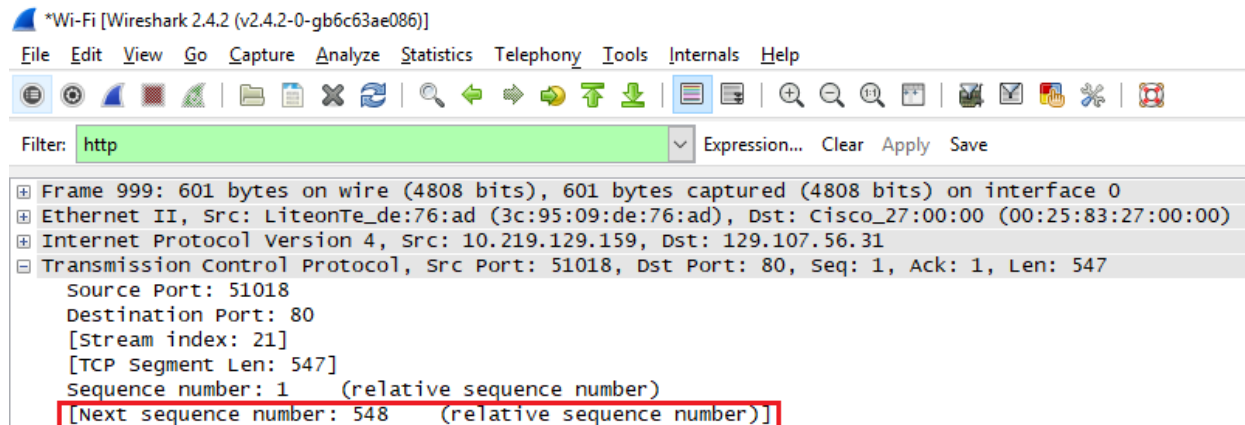
Ans. The length of the TCP segment is 547 bytes.



7. What is the Sequence Number of TCP segment (you can use the relative sequence number)?

Ans. Sequence Number of TCP segment: 1

Next sequence number: 548

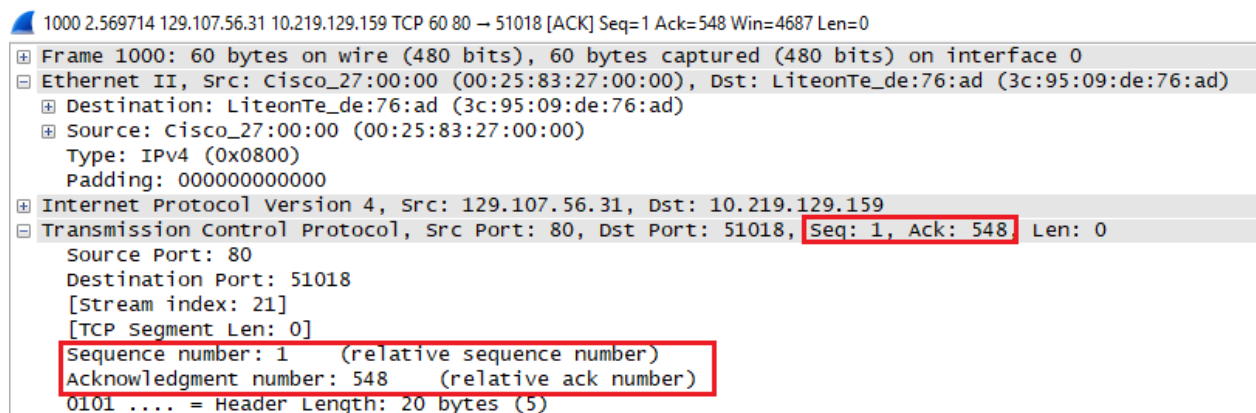


8. Calculate the acknowledgement number based on the two questions above. Verify your solution with the Wireshark values.

Ans. From the previous question, we can derive that: Initial Sequence number = 1, ACK = 1,

Acknowledgment number = (ACK number of HTTP segment + TCP Segment length) = 1+547= 548.

So, ACK number should be equal to Next Sequence number i.e. 548.



9. What are the fields in the TCP Flags? No need to give any values but give the field names given in Wireshark

Ans. TCP Flags are: PSH and ACK bit are set

- Reserved

- Nonce
- Congestion Window Reduced
- ECN- Echo
- Urgent
- Acknowledgment
- Push
- Reset
- Syn
- Fin

```

999 2.561338 10.219.129.159 129.107.56.31 HTTP 601 GET / HTTP/1.1
* Frame 999: 601 bytes on wire (4808 bits), 601 bytes captured (4808 bits) on interface 0
+ Ethernet II, Src: LiteonTe_de:76:ad (3c:95:09:de:76:ad), Dst: Cisco_27:00:00 (00:25:83:27:00:00)
+ Internet Protocol Version 4, Src: 10.219.129.159, Dst: 129.107.56.31
+ Transmission Control Protocol, Src Port: 51018, Dst Port: 80, Seq: 1, Ack: 1, Len: 547
  Source Port: 51018
  Destination Port: 80
  [Stream index: 21]
  [TCP segment Len: 547]
  Sequence number: 1 (relative sequence number)
  [Next sequence number: 548 (relative sequence number)]
  Acknowledgment number: 1 (relative ack number)
  0101 .... = Header Length: 20 bytes (5)
  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...]

```

10. What is the IP address of uta.edu? On what port number is it sending and receiving TCP segments for this connection?

Ans. IP address of uta.edu: 129.107.56.31, Port number used by uta.edu to send TCP segments: 80

```

999 2.561338 10.219.129.159 129.107.56.31 HTTP 601 GET / HTTP/1.1
+ Frame 999: 601 bytes on wire (4808 bits), 601 bytes captured (4808 bits) on interface 0
+ Ethernet II, Src: LiteonTe_de:76:ad (3c:95:09:de:76:ad), Dst: Cisco_27:00:00 (00:25:83:27:00:00)
+ Internet Protocol Version 4, Src: 10.219.129.159, Dst: 129.107.56.31
+ Transmission Control Protocol, Src Port: 51018, Dst Port: 80, Seq: 1, Ack: 1, Len: 547
+ Hypertext Transfer Protocol

```

Section 2: Analyzing the Connection Parameters in TCP

Problem Set 2

1. What is the sequence number (absolute) of the TCP SYN segment that is used to initiate the TCP connection between the client computer and youtube.com?

Ans. The absolute sequence number of the of the TCP SYN segment is **3177954973**.

314	22.809147	10.219.129.159	216.58.194.142	TCP	54 51435 → 443 [ACK] Seq=9267020 Ack=1561851566 Win=
315	22.809213	8.8.8.8	10.219.129.159	DNS	133 Standard query response 0x88ac A upload.youtube.c
316	22.809467	10.219.129.159	216.58.194.142	TLSv1.2	258 Client Hello
317	22.810796	10.219.129.159	161.69.92.62	TLSv1	411 Application Data
318	22.818436	216.58.194.142	10.219.129.159	TCP	60 443 → 51435 [ACK] Seq=1561851566 Ack=9267224 win=
319	22.829386	216.58.194.142	10.219.129.159	TLSv1.2	1440 Server Hello
320	22.829386	216.58.194.142	10.219.129.159	TLSv1.2	1388 Certificate, Server Key Exchange, Server Hello Dc
321	22.829472	10.219.129.159	216.58.194.142	TCP	54 51435 → 443 [ACK] Seq=9267224 Ack=1561854286 win=
322	22.837930	161.69.92.62	10.219.129.159	TLSv1	155 Application Data
323	22.838914	10.219.129.159	216.58.194.143	TCP	66 51436 → 443 [SYN] Seq=3177954973 win=65535 Len=0
324	22.838926	10.219.129.159	216.58.194.143	TCP	66 51437 → 443 [SYN] Seq=2685057543 win=65535 Len=0

```
323 22.838914 10.219.129.159 216.58.194.143 TCP 66 51436 → 443 [SYN] Seq=3177954973 Win=65535 Len=0 MSS=1460 WS=8 SACK_PERM=1
+ Frame 323: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0
+ Ethernet II, Src: LiteonTe_de:76:ad (3c:95:09:de:76:ad), Dst: Cisco_27:00:00 (00:25:83:27:00:00)
+ Internet Protocol version 4, Src: 10.219.129.159, Dst: 216.58.194.143
+ Transmission Control Protocol, Src Port: 51436, Dst Port: 443, Seq: 3177954973, Len: 0
  Source Port: 51436
  Destination Port: 443
  [Stream index: 9]
  [TCP Segment Len: 0]
  Sequence number: 3177954973
  Acknowledgment number: 0
  1000 .... = Header Length: 32 bytes (8)
  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
```

2. What is it in the segment that identifies the segment as a SYN segment?

Ans. Flag in the TCP protocol depicts that it is a SYN bit.

Flags – 0x002 (SYN)

```

323 22.838914 10.219.129.159 216.58.194.143 TCP 66 51436 → 443 [SYN] Seq=3177954973 Win=65535 Len=0 MSS=1460 WS=8 SACK_PERM=1
+ Frame 323: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0
+ Ethernet II, Src: LiteonTe_de:76:ad (3c:95:09:de:76:ad), Dst: Cisco_27:00:00 (00:25:83:27:00:00)
+ Internet Protocol Version 4, Src: 10.219.129.159, Dst: 216.58.194.143
+ Transmission Control Protocol, Src Port: 51436, Dst Port: 443, Seq: 3177954973, Len: 0
  Source Port: 51436
  Destination Port: 443
  [Stream index: 9]
  [TCP Segment Len: 0]
  Sequence number: 3177954973
  Acknowledgment number: 0
  1000 .... = Header Length: 32 bytes (8)
+ 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

```

3. What is the sequence number of the SYNACK segment sent by youtube.com to the client computer in reply to the SYN? What is the value of the Acknowledgement field in the SYNACK segment?

Ans. **Sequence number** of the SYNACK segment sent by youtube.com to the client computer - 1611531606

Acknowledgement number in the SYNACK segment – 3177954974

```

326 22.843450 216.58.194.143 10.219.129.159 TCP 66 443 → 51436 [SYN, ACK] Seq=1611531606 Ack=3177954974 Win=42780 Len=0 MSS=1380 SACK_PERM=1 WS=256
+ Frame 326: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0
+ Ethernet II, Src: Cisco_27:00:00 (00:25:83:27:00:00), Dst: LiteonTe_de:76:ad (3c:95:09:de:76:ad)
+ Internet Protocol Version 4, Src: 216.58.194.143, Dst: 10.219.129.159
+ Transmission Control Protocol, Src Port: 443, Dst Port: 51436, Seq: 1611531606, Ack: 3177954974, Len: 0
  Source Port: 443
  Destination Port: 51436
  [Stream index: 9]
  [TCP Segment Len: 0]
  Sequence number: 1611531606
  Acknowledgment number: 3177954974
  1000 .... = Header Length: 32 bytes (8)
+ 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

```

pcapng [Wireshark (v2.4.2-0-gb6c63ae086)]

Capture Analyze Statistics Telephony Tools Internals Help

Source	Destination	Protocol	Length	Info
10.219.129.159	216.58.194.143	TCP	54	51435 → 443 [ACK] Seq=9267224 Ack=1561854286 win=262144 Len=0
161.69.92.62	10.219.129.159	TLSv1	155	Application Data
10.219.129.159	216.58.194.143	TCP	66	51436 → 443 [SYN] Seq=3177954973 Win=65535 Len=0 MSS=1460 WS=8 SACK_PERM=1
10.219.129.159	216.58.194.143	TCP	66	51437 → 443 [SYN] Seq=2685057543 Win=65535 Len=0 MSS=1460 WS=8 SACK_PERM=1
10.219.129.159	8.8.8.8	DNS	73	Standard query 0xc8ad A ocsipki.google.com
216.58.194.143	10.219.129.159	TCP	66	443 → 51436 [SYN, ACK] Seq=1611531606 Ack=3177954974 Win=42780 Len=0
10.219.129.159	216.58.194.143	TCP	54	51436 → 443 [ACK] Seq=3177954974 Ack=1611531607 win=262144 Len=0
10.219.129.159	216.58.194.143	TLSv1.2	261	Client Hello
216.58.194.142	10.219.129.159	TCP	60	443 → 51429 [ACK] Seq=2753065531 Ack=1284885685 win=45056 Len=0

4. How did youtube.com determine that value? What is it in the segment that identifies the segment as a SYNACK segment?

Ans. Youtube.com determines the value using Flag bit.

The segment that identifies SYNACK segment in the Flag bit.

Flag – 0x012 (SYN,ACK)

```
326 22.843450 216.58.194.143 10.219.129.159 TCP 66 443 → 51436 [SYN, ACK] Seq=1611531606 Ack=3177954974 Win=42780 Len=0 MSS=1380 SACK_PERM=1 WS=256
  Frame 326: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface 0
  Ethernet II, Src: Cisco_27:00:00 (00:25:83:27:00:00), Dst: LiteonTe_de:76:ad (3c:95:09:de:76:ad)
  Internet Protocol Version 4, Src: 216.58.194.143, Dst: 10.219.129.159
  Transmission Control Protocol, Src Port: 443, Dst Port: 51436, Seq: 1611531606, Ack: 3177954974, Len: 0
    Source Port: 443
    Destination Port: 51436
    [Stream index: 9]
    [TCP Segment Len: 0]
    Sequence number: 1611531606
    Acknowledgment number: 3177954974
    1000 .... = Header Length: 32 bytes (8)
    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
```

Section 3: Analysis of the trace provided

Problem Set 3:

1. What is the sequence number of the TCP segment containing the first HTTP POST command?

Ans. Relative Sequence number for the TCP segment containing the first HTTP POST command is **1415**.

```
Wireshark · Packet 140 · kayak(1)
  > Frame 140: 1454 bytes on wire (11632 bits), 1454 bytes captured (11632 bits) on interface 0
  > Ethernet II, Src: Apple_8b:6e:80 (6c:40:08:8b:6e:80), Dst: fa:cf:9c:21:5f:64 (fa:cf:9c:21:5f:64)
  > Internet Protocol Version 4, Src: 172.20.10.2, Dst: 23.235.44.231
  ✓ Transmission Control Protocol, Src Port: 55790, Dst Port: 80, Seq: 1415, Ack: 56130, Len: 1388
    Source Port: 55790
    Destination Port: 80
    [Stream index: 4]
    [TCP Segment Len: 1388]
    Sequence number: 1415 (relative sequence number)
    [Next sequence number: 2803 (relative sequence number)]
    Acknowledgment number: 56130 (relative ack number)
```


0000	fa cf 9c 21 5f 64 6c 40 08 8b 6e 80 08 00 45 00	...!_dl@ ..n...E.
0010	05 a0 d5 66 40 00 40 06 65 09 ac 14 0a 02 17 eb	...f@.@. e.....
0020	2c e7 d9 ee 00 50 12 fc 0d cf 3f 57 a5 79 80 10	,....P.. ...?W.y..
0030	10 00 36 9a 00 00 01 01 08 0a 06 4f 76 33 03 0d	..6..... ...0v3..
0040	d1 4e 50 4f 53 54 20 2f 76 73 2f 70 61 67 65 2f	.NPOST / vs/page/
0050	68 6f 74 65 6c 2f 72 65 73 75 6c 74 73 20 48 54	hotel/re sults HT
0060	54 50 2f 31 2e 31 0d 0a 48 6f 73 74 3a 20 77 77	TP/1.1.. Host: ww
0070	77 2e 6b 61 79 61 6b 2e 63 6f 6d 0d 0a 41 63 63	w.kayak. com..Acc
0080	65 70 74 3a 20 2a 2f 2a 0d 0a 58 2d 52 65 71 75	ept: */* ..X-Requ
0090	65 73 74 65 64 2d 57 69 74 68 3a 20 58 4d 4c 48	ested-Wi th: XMLH
00a0	74 74 70 52 65 71 75 65 73 74 0d 0a 41 63 63 65	ttpReque st..Acce
00b0	70 74 2d 4c 61 6e 67 75 61 67 65 3a 20 65 6e 2d	pt-Langu age: en-
00c0	75 73 0d 0a 41 63 63 65 70 74 2d 45 6e 63 6f 64	us..Acce pt-Encod
00d0	69 6e 67 3a 20 67 7a 69 70 2c 20 64 65 66 6c 61	ing: gzi p, defla
00e0	74 65 0d 0a 43 6f 6e 74 65 6e 74 2d 54 79 70 65	te..Cont ent-Type
00f0	3a 20 61 70 70 6c 69 63 61 74 69 6f 6e 2f 78 2d	: applic ation/x-
0100	77 77 77 2d 66 6f 72 6d 2d 75 72 6c 65 6e 63 6f	www-form -urlenco
0110	64 65 64 3b 20 63 68 61 72 73 65 74 3d 55 54 46	ded; cha rset=UTF

2. Consider the TCP segment containing the HTTP POST as the first segment in the TCP connection.

i) What are the sequence numbers of the first four segments in the TCP connection (including the segment containing the HTTP POST)?

Ans. TCP segment having HTTP POST: (140)

TCP Segment 1: (140)

Sequence number: 1415 (Relative), Next Sequence number: (Relative)

TCP Segment 2: (141)

Sequence number: 2803 (Relative), Next Sequence number: (Relative)

TCP Segment 3: (142)

Sequence number: 2938 (Relative), Next Sequence number: (Relative)

TCP Segment 4: (174)

Sequence number: 2947 (Relative), Next Sequence number: (Relative)

Length	Info	Sequence Number	Acknowledgement Number
66 80 → 55789	[ACK] Seq=5152 Ack=2946 win=85 Len=0 TSval=917517929 TSecr=105870856	5152	2946
66 80 → 55789	[ACK] Seq=5152 Ack=3138 win=90 Len=0 TSval=917517929 TSecr=105870856	5152	3138
74 443 → 55806	[SYN, ACK] Seq=0 Ack=1 win=8192 Len=0 MSS=1400 WS=256 SACK_PERM=1 TSval=98461199 TSecr=105870899	0	1
66 55806 → 443	[ACK] Seq=1 Ack=1 win=131840 Len=0 TSval=105870899 TSecr=98461199	1	1
1454 55790 → 80	[ACK] Seq=1415 Ack=56130 win=4096 Len=1388 TSval=105870899 TSecr=51237198	1415	56130
201 55790 → 80	[PSH, ACK] Seq=2803 Ack=56130 win=4096 Len=135 TSval=105870899 TSecr=512371	2803	56130
75	POST /vs/page/hotel/results HTTP/1.1 (application/x-www-form-urlencoded)	2938	56130
583	Client Hello	1	1
66 80 → 55790	[ACK] Seq=56130 Ack=2803 win=72 Len=0 TSval=51237230 TSecr=105870899	56130	2803

66	55806	-	443	[ACK]	Seq=1014	Ack=759	win=131072	Len=0	TSval=105871146	TSecr=98461224	1014	759
78	[TCP Dup ACK 168#1]	443	-	55806	[ACK]	Seq=759	Ack=1014	win=131328	Len=0	TSval=98461224	759	1014
78	55807	-	443	[SYN]	Seq=0	win=65535	Len=0	MSS=1460	WS=32	TSval=105871374	TSecr=0	SACK_PER
1454	55789	-	80	[ACK]	Seq=4703	Ack=7633	win=4096	Len=1388	TSval=105871380	TSecr=917517986	4703	7633
210	55789	-	80	[PSH, ACK]	Seq=6091	Ack=7633	win=4096	Len=144	TSval=105871380	TSecr=917517986	6091	7633
1454	55790	-	80	[ACK]	Seq=2947	Ack=56939	win=4096	Len=1388	TSval=105871380	TSecr=51237247	2947	56939
175	GET	/s/run/recentsearchhistory/gethistory?searchType=hotel&maxSearchHistoryNum=30&sear									4335	56939

ii) At what time was each segment sent?

Ans. Time for each segment is as follows:

TCP segment having HTTP POST: 4.081502 secs

TCP Segment 1: 4.081503 secs

TCP Segment 2: 4.081603 secs

136	4.079317	23.235.44.231	172.20.10.2	TCP	66	80	-	55789	[ACK]	Seq=5152	Ack=2946	win=85	Len=0
137	4.079650	23.235.44.231	172.20.10.2	TCP	66	80	-	55789	[ACK]	Seq=5152	Ack=3138	win=90	Len=0
138	4.080945	209.105.248.3	172.20.10.2	TCP	74	443	-	55806	[SYN, ACK]	Seq=0	Ack=1	win=8192	Len=0
139	4.081018	172.20.10.2	209.105.248.3	TCP	66	55806	-	443	[ACK]	Seq=1	Ack=1	win=131840	Len=0
140	4.081502	172.20.10.2	23.235.44.231	TCP	1454	55790	-	80	[ACK]	Seq=1415	Ack=56130	win=4096	Len=0
141	4.081503	172.20.10.2	23.235.44.231	TCP	201	55790	-	80	[PSH, ACK]	Seq=2803	Ack=56130	win=4096	Len=0
142	4.081603	172.20.10.2	23.235.44.231	HTTP	75	POST	/vs/page/hotel/results	HTTP/1.1	(application/x-www-form-urlencoded)				
143	4.082853	172.20.10.2	209.105.248.3	TLSv1	583	client	Hello						
144	4.135363	23.235.44.231	172.20.10.2	TCP	66	80	-	55790	[ACK]	Seq=56130	Ack=2803	win=72	Len=0
145	4.135713	23.235.44.231	172.20.10.2	TCP	66	80	-	55790	[ACK]	Seq=56130	Ack=2938	win=77	Len=0
146	4.135716	23.235.44.231	172.20.10.2	TCP	66	80	-	55790	[ACK]	Seq=56130	Ack=2947	win=77	Len=0

TCP Segment 3: 4.570353 secs

No.	Time	Source	Destination	Protocol	Length	Sequence number	Next sequence number	Info
173	4.570191	172.20.10.2	23.235.44.231	TCP	210	6091	6235	[TCP segment of a reassembled PDU]
174	4.570353	172.20.10.2	23.235.44.231	TCP	1454	2947	4335	[TCP segment of a reassembled PDU]
175	4.570354	172.20.10.2	23.235.44.231	HTTP	175	4335	4444	GET /s/run/recentsearchhistory/gethistory?searchType=hotel&maxSearchHistoryNum=30&searchHistoryNum=30
176	4.571636	172.20.10.2	23.235.44.231	HTTP	85	6235	6254	POST /s/run/hotelbookmsg HTTP/1.1 (application/x-www-form-urlencoded)
177	4.572121	172.20.10.2	216.58.218.194	TLSv1.2	352	1	287	Application Data

iii) When was the ACK for each segment received?

Ans.

TCP containing HTTP Post (140): Sequence number: 1415, ACK: 56130

ACK for HTTP (144): Sequence number: 56130 ACK: 2803, Time: 4.135363 secs

No.	Time	Source	Destination	Protocol	Length	Sequence number	Next sequence number	Acknowledgment number	Info
140	4.081502	172.20.10.2	23.235.44.231	TCP	1454	1415	2803	56130	[TCP segment of a reassembled PDU]
141	4.081503	172.20.10.2	23.235.44.231	TCP	201	2803	2938	56130	[TCP segment of a reassembled PDU]
142	4.081603	172.20.10.2	23.235.44.231	HTTP	75	2938	2947	56130	POST /vs/page/hotel/results HTTP/1.1 (application/x-www-form-urlencoded)
143	4.082853	172.20.10.2	209.105.248.3	TLSv1	583	1	518		1 Client Hello
144	4.135363	23.235.44.231	172.20.10.2	TCP	66	56130			2803 80->55790 [ACK] Seq=56130 Ack=2803 Win=72 Len=0 TSval=105871380 TSecr=917517986
145	4.135713	23.235.44.231	172.20.10.2	TCP	66	56130			2938 80->55790 [ACK] Seq=56130 Ack=2938 Win=77 Len=0 TSval=105871380 TSecr=917517986

TCP Segment 1 (141): Sequence number: 2803, ACK: 56130

ACK for TCP Segment 1(145): Sequence number: 56130, ACK: 2938, Time: 4.135713 secs

We have considered this packet as the next sequence number of the ACK packet is equal to sequence number of the next TCP segment.

No.	Time	Source	Destination	Protocol	Length	Sequence number	Next sequence number	Acknowledgment number	Info
140	4.081502	172.20.10.2	23.235.44.231	TCP	1454	1415	2803	56130	[TCP segment of a reassembled PDU]
141	4.081503	172.20.10.2	23.235.44.231	TCP	201	2803	2938	56130	[TCP segment of a reassembled PDU]
142	4.081603	172.20.10.2	23.235.44.231	HTTP	75	2938	2947	56130	POST /vs/page/hotel/results HTTP/1.1 (application/x-www-form-urlencoded)
143	4.082853	172.20.10.2	209.105.248.3	TLSv1	583	1	518		1 Client Hello
144	4.135363	23.235.44.231	172.20.10.2	TCP	66	56130		2803	80+55790 [ACK] Seq=56130 Ack=2803 Win=72 Len=0 TSval=2803
145	4.135713	23.235.44.231	172.20.10.2	TCP	66	56130		2938	80+55790 [ACK] Seq=56130 Ack=2938 Win=77 Len=0 TSval=2938
146	4.135716	23.235.44.231	172.20.10.2	TCP	66	56130		2947	80+55790 [ACK] Seq=56130 Ack=2947 Win=77 Len=0 TSval=2947

TCP Segment 2 (142): Sequence number: 2938, ACK: 56130

ACK for TCP Segment 2 (146): Sequence number: 56130, ACK: 2947, Time: 4.135716 Secs

No.	Time	Source	Destination	Protocol	Length	Sequence number	Next sequence number	Acknowledgment number	Info
140	4.081502	172.20.10.2	23.235.44.231	TCP	1454	1415	2803	56130	[TCP segment of a reassembled PDU]
141	4.081503	172.20.10.2	23.235.44.231	TCP	201	2803	2938	56130	[TCP segment of a reassembled PDU]
142	4.081603	172.20.10.2	23.235.44.231	HTTP	75	2938	2947	56130	POST /vs/page/hotel/results HTTP/1.1 (application/x-www-form-urlencoded)
143	4.082853	172.20.10.2	209.105.248.3	TLSv1	583	1	518		1 Client Hello
144	4.135363	23.235.44.231	172.20.10.2	TCP	66	56130		2803	80+55790 [ACK] Seq=56130 Ack=2803 Win=72 Len=0 TSval=2803
145	4.135713	23.235.44.231	172.20.10.2	TCP	66	56130		2938	80+55790 [ACK] Seq=56130 Ack=2938 Win=77 Len=0 TSval=2938
146	4.135716	23.235.44.231	172.20.10.2	TCP	66	56130		2947	80+55790 [ACK] Seq=56130 Ack=2947 Win=77 Len=0 TSval=2947
147	4.136062	209.105.248.3	172.20.10.2	TLSv1	211	1	146		518 Server Hello, Change Cipher Spec, Encrypted Handshake

TCP Segment 3 (174): Sequence number: 4335, ACK: 56939

ACK for TCP Segment 3: Sequence number: 56939, ACK: 4335, Time: 4.646868 Secs

No.	Time	Source	Destination	Protocol	Length	Sequence number	Next sequence number	Acknowledgment number	Info
173	4.570191	172.20.10.2	23.235.44.231	TCP	210	6091	6235	7633	[TCP segment of a reassembled PDU]
174	4.570353	172.20.10.2	23.235.44.231	TCP	1454	2947	4335	56939	[TCP segment of a reassembled PDU]
175	4.570354	172.20.10.2	23.235.44.231	HTTP	175	4335	4444	56939	GET /s/run/recentsearchhistory/gethistory?searchType=
176	4.571636	172.20.10.2	23.235.44.231	HTTP	85	6235	6254	7633	POST /s/run/hotelbookmsg HTTP/1.1 (application/x-www-form-urlencoded)
177	4.572121	172.20.10.2	216.58.218.194	TLSv1.2	352	1	287		1 Application Data
178	4.575141	172.20.10.2	172.20.10.1	DNS	74				Standard query 0xcceb2 A www.google.com
179	4.640828	173.194.115.60	172.20.10.2	TCP	74	0		1	443+55807 [SYN, ACK] Seq=0 Ack=1 Win=42540 Len=0 MSS=
180	4.640909	172.20.10.2	173.194.115.60	TCP	66	1		1	55807+443 [ACK] Seq=1 Ack=1 Win=131840 Len=0 TSval=
181	4.641019	172.20.10.1	172.20.10.2	DNS	154				Standard query response 0xcceb2 A www.google.com A 17
182	4.643313	172.20.10.2	173.194.115.51	TCP	78	0		0	55808+00 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=32
183	4.643917	172.20.10.2	173.194.115.60	TLSv1.2	300	1	235		1 Client Hello
184	4.646397	23.235.44.231	172.20.10.2	TCP	66	7633		6091	80+55789 [ACK] Seq=7633 Ack=6091 Win=106 Len=0 TSval=
185	4.646590	23.235.44.231	172.20.10.2	TCP	66	7633		6235	80+55789 [ACK] Seq=7633 Ack=6235 Win=112 Len=0 TSval=
186	4.646868	23.235.44.231	172.20.10.2	TCP	66	56939		4335	80+55790 [ACK] Seq=56939 Ack=4335 Win=82 Len=0 TSval=
187	4.649228	216.58.218.194	172.20.10.2	TCP	66	1		287	443+55496 [ACK] Seq=1 Ack=287 Win=455 Len=0 TSval=24

iv) Given the difference between when each TCP segment was sent, and when its acknowledgement was received,

Ans.

Segment	Time (secs)	Time (ACK received) (secs)	Difference (Time(ACK)- Time) (secs)
TCP Segment with HTTP post	4.081502	4.135363	0.053861
TCP Segment 1	4.081503	4.135713	0.05421
TCP Segment 2	4.081603	4.135716	0.054113
TCP Segment 3	4.570353	4.646868	0.076518

v) what is the RTT value for each of the four segments?

Ans.

Segment	Time (secs)	Time (ACK received) (secs)	Sample RTT (Time(ACK)- Time) (secs)
TCP Segment with HTTP post	4.081502	4.135716	0.053861
TCP Segment 1	4.081503	4.135713	0.05421
TCP Segment 2	4.081603	4.135716	0.054113
TCP Segment 3	4.570373	4.646868	0.076518

vi) What is the EstimatedRTT value (see Section 3.5.3, page 239 in text) after the receipt of each ACK?

Ans. Estimated RTT of first segment is equal to sample RTT

Segment	Time (secs)	Time (ACK received) (secs)	Sample RTT (secs)	Estimated RTT (secs)
TCP Segment with HTTP post	4.081502	4.135716	0.53861	0.053861
TCP Segment 1	4.081503	4.135713	0.05421	0.053904
TCP Segment 2	4.081603	4.135716	0.054113	0.053930
TCP Segment 3	4.570373	4.646868	0.076518	0.056754

vii) Assume that the value of the EstimatedRTT is equal to the measured RTT for the first segment, and then is computed using the Estimated RTT equation on page 239 for all subsequent segments.

Ans.

Segment	Time (secs)	Time (ACK received) (secs)	Sample RTT (secs)	Estimated RTT (secs)
TCP Segment with HTTP post	4.081502	4.135716	0.53861	0.053861
TCP Segment 1	4.081503	4.135713	0.05421	0.053904
TCP Segment 2	4.081603	4.135716	0.054113	0.053930
TCP Segment 3	4.570373	4.646868	0.076518	0.056754

3. What is the length of each of the first four TCP segments?

Ans.

Segment	Length (bytes)	Column number	Sequence number
TCP Segment with HTTP post	1388	140	1415
TCP Segment 1	135	141	2803
TCP Segment 2	9	142	2938
TCP Segment 3	1388	174	4335

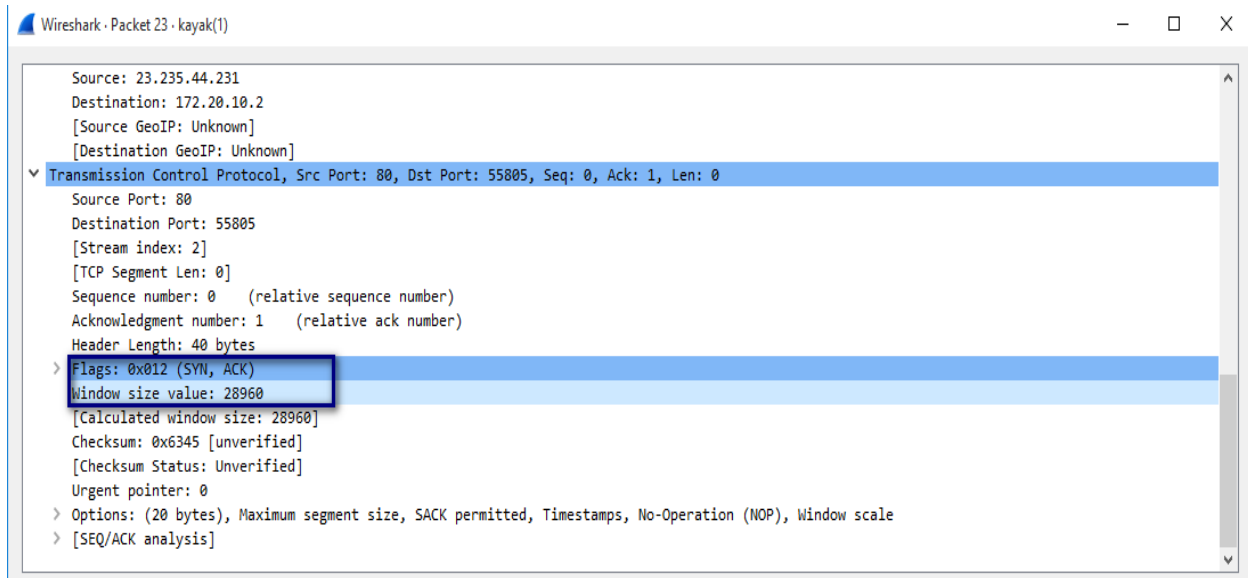
No.	Time	Source	Destination	Protocol	Length	Sequence number	Next sequence number	TCP Segment Len	Info
140	4.081502	172.20.10.2	23.235.44.231	TCP	1454	1415	2803	1388	[TCP segment of a reassembled PDU]
141	4.081503	172.20.10.2	23.235.44.231	TCP	201	2803	2938	135	[TCP segment of a reassembled PDU]
142	4.081603	172.20.10.2	23.235.44.231	HTTP	75	2938	2947	9	POST /vs/page/hotel/results HTTP/1.1 (application/x-www-f
160	4.201866	172.20.10.2	23.235.44.231	TCP	66	2947			0 55790+80 [ACK] Seq=2947 Ack=56939 Win=4070 Len=0 TSval=105
174	4.570353	172.20.10.2	23.235.44.231	TCP	1454	2947	4335	1388	[TCP segment of a reassembled PDU]
175	4.570354	172.20.10.2	23.235.44.231	HTTP	175	4335	4444		109 GET /s/run/recentsearchhistory/gethistory?searchType=hotel
202	4.715684	172.20.10.2	23.235.44.231	TCP	66	4444			0 55790+80 [ACK] Seq=4444 Ack=57783 Win=4069 Len=0 TSval=105

4. What is the minimum amount of available buffer space advertised at the receiver for the entire trace?

Ans. The first TCP [SYN, ACK] bit in the TCP segment gives the minimum amount of available buffer size.

Window Size: 28960

Packet number: 23



5. Does the lack of receiver buffer space ever throttle the sender?

Ans. The minimum amount of available buffer space advertised at the receiver for the entire trace is 28960 bytes (Packet number: 23).

No.	Time	Source	Destination	Protocol	Length	Sequence number	Window size value	Info
21	2.913690	172.20.10.2	23.235.44.231	TCP	78	0	65535	55805→80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=32 TSval=105869757 TSecr=0...
22	3.004363	172.20.10.1	172.20.10.2	DNS	117			Standard query response 0x9cb8 A mathid-origin.mathtag.com A 74.121.139.80 ...
23	3.005770	23.235.44.231	172.20.10.2	TCP	74	0	28960	80→55805 [SYN, ACK] Seq=0 Ack=1 Win=28960 Len=0 MSS=1400 SACK_PERM=1 TSval=...
24	3.005854	172.20.10.2	23.235.44.231	TCP	66	1	4120	55805→80 [ACK] Seq=1 Ack=1 Win=131840 Len=0 TSval=105869849 TSecr=335022806

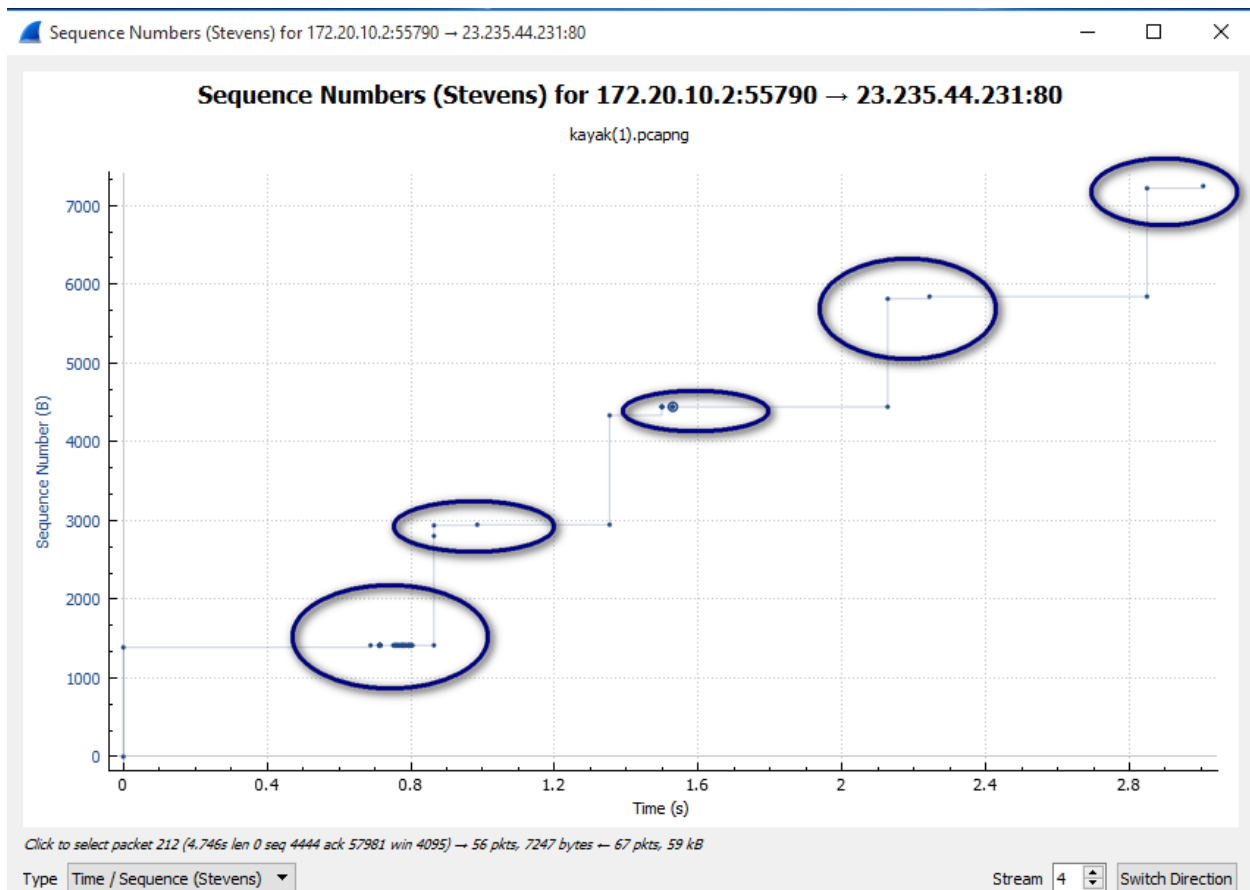
The maximum amount of available buffer space advertised at the receiver for the entire trace is 65535 bytes (Packet number: 1104).

No.	Time	Source	Destination	Protocol	Length	Sequence number	Window size value	Info
1113	9.821203	172.20.10.2	216.58.218.163	TCP	78	0	65535	55838→80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=32 TSval=105876462 TSecr=0...
1104	9.800917	93.184.216.182	172.20.10.2	TCP	74	0	65535	80→55837 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1400 SACK_PERM=1 TSval=...
1103	9.783520	172.20.10.2	129.107.56.31	TCP	66	4948	65535	55827→80 [ACK] Seq=4948 Ack=1038 Win=65535 Len=0 TSval=105876428 TSecr=3732...

This receiver window grows until it reaches the maximum receiver buffer size. By examining the trace, the sender is never throttled due to lack of receiver buffer space. Even when the advertised receiver window is having its lowest value i.e. 28960, the sender is constrained by congestion window. Window size will increase up to 65535 bits, allowing sufficient buffer size for the sender to send the data.

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

Ans.



We have considered Stevens graph to check the retransmissions. In the above, graph it clearly represents the multiple re-transmissions of the same segment.

We have also used “tcp.analysis.retransmission” filter to find out the number of retransmissions.

No.	Time	Source	Destination	Protocol	Length	Sequence number	Window size value	Info
8	0.059891	173.194.115.90	172.20.10.2	TLSv1.2	129	1	375	[TCP Spurious Retransmission] Application Data
226	4.767929	216.58.218.194	172.20.10.2	TLSv1.2	112	682	455	[TCP Spurious Retransmission] Application Data
6	0.025943	172.20.10.2	173.194.115.90	TCP	66	1	4096	[TCP Spurious Retransmission] 55720+443 [FIN, ACK] Seq=1 Ack=65 Win=4096 Len=0 TSv...
7	0.025944	172.20.10.2	173.194.115.90	TCP	66	1	4096	[TCP Spurious Retransmission] 55720+443 [FIN, ACK] Seq=1 Ack=65 Win=4096 Len=0 TSv...
9	0.059976	172.20.10.2	173.194.115.90	TCP	78	1	4096	[TCP Spurious Retransmission] 55720+443 [FIN, ACK] Seq=1 Ack=65 Win=4096 Len=0 TSv...
15	1.757791	172.20.10.2	104.72.237.125	TCP	394	1	4096	[TCP Retransmission] 55795+80 [PSH, ACK] Seq=1 Ack=1 Win=4096 Len=328 TSval=105868...
163	4.384811	172.20.10.2	209.105.248.3	TCP	503	577	4116	[TCP Retransmission] 55806+443 [PSH, ACK] Seq=577 Ack=146 Win=131712 Len=437 TSval...
1145	10.288426	172.20.10.2	64.6.21.1	TCP	1454	3455	65535	[TCP Retransmission] 55826+443 [PSH, ACK] Seq=3455 Ack=4382 Win=65535 Len=1388 TSv...

7. How much data does the receiver typically acknowledge in an ACK? Can you identify cases where the receiver is ACK-ing every other received segment (see Table 3.2 on page 247 in the text)?

Ans. Typically, Acknowledge data size is as follows;

No.	Time	Source	Destination	Protocol	Length	Sequence number	Acknowledgment number	Info
414	5.779592	64.6.21.1	172.20.10.2	TCP	66	1	2176	80+55813 [ACK] Seq=1 Ack=2176 Win=6375 Len=0 TSval=3732542352 TSecr=10587...
415	5.813839	68.67.129.43	172.20.10.2	TCP	66	3459	1121	443+55812 [ACK] Seq=3459 Ack=1121 Win=30208 Len=0 TSval=3353254775 TSecr=...
418	5.874897	172.20.10.2	64.6.21.1	TCP	66	2176	1467	55813+80 [ACK] Seq=2176 Ack=1467 Win=65535 Len=0 TSval=105872641 TSecr=37...
421	5.876519	172.20.10.2	64.6.21.1	TCP	66	2176	2915	55813+80 [ACK] Seq=2176 Ack=2915 Win=65535 Len=0 TSval=105872643 TSecr=37...
425	5.877615	172.20.10.2	64.6.21.1	TCP	66	2176	4303	55813+80 [ACK] Seq=2176 Ack=4303 Win=65535 Len=0 TSval=105872644 TSecr=37...
426	5.877616	172.20.10.2	64.6.21.1	TCP	66	2176	4363	55813+80 [ACK] Seq=2176 Ack=4363 Win=65535 Len=0 TSval=105872644 TSecr=37...
427	5.877616	172.20.10.2	64.6.21.1	TCP	66	2176	4382	55813+80 [ACK] Seq=2176 Ack=4382 Win=65535 Len=0 TSval=105872644 TSecr=37...
430	5.971340	172.20.10.2	64.6.21.1	TCP	66	2176	5811	55813+80 [ACK] Seq=2176 Ack=5811 Win=65535 Len=0 TSval=105872735 TSecr=37...
433	5.971752	172.20.10.2	64.6.21.1	TCP	66	2176	7259	55813+80 [ACK] Seq=2176 Ack=7259 Win=65535 Len=0 TSval=105872735 TSecr=37...
436	5.974066	172.20.10.2	64.6.21.1	TCP	66	2176	8700	55813+80 [ACK] Seq=2176 Ack=8700 Win=65535 Len=0 TSval=105872737 TSecr=37...
443	6.070443	172.20.10.2	23.235.44.231	TCP	78	0	0	55814+80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=32 TSval=105872831 TSecr=...

Packet number	Sequence number	Acknowledgment number	Acknowledgement data size (Next ACK number-ACK)
418	2176	1467	1467
421	2176	2915	1488
425	2176	4303	1388
426	2176	4363	60
427	2176	4382	19
430	2176	5811	1429
433	2176	7259	1488
436	2176	8700	1441

Approximately, 1097 is the Acknowledgment data size.

While examining the packets, we find that multiple acknowledgements has the same sequence number.

The following sample of data ACKs every other received segment:

No.	Time	Source	Destination	Protocol	Length	Sequence number	Acknowledgment number	Info
616	6.595728	172.20.10.2	68.67.128.233	TCP	66	1174	3226	55820+443 [ACK] Seq=1174 Ack=3226 Win=131072 Len=0 TSval=105873330 TSecr=...
617	6.596336	172.20.10.2	68.67.128.233	TCP	66	1174	3226	55820+443 [FIN, ACK] Seq=1174 Ack=3226 Win=131072 Len=0 TSval=105873330 T...
620	6.668593	68.67.128.233	172.20.10.2	TCP	66	3226	1175	443+55820 [ACK] Seq=3226 Ack=1175 Win=30208 Len=0 TSval=1062508368 TSecr=...
622	6.716879	216.58.218.194	172.20.10.2	TCP	66	1165	1056	443+55496 [ACK] Seq=1165 Ack=1056 Win=477 Len=0 TSval=2418027250 TSecr=10...
623	6.717144	216.58.218.194	172.20.10.2	TCP	66	1165	1454	443+55496 [ACK] Seq=1165 Ack=1454 Win=487 Len=0 TSval=2418027250 TSecr=10...
625	6.735630	172.20.10.2	216.58.218.194	TCP	66	1454	1774	55496+443 [ACK] Seq=1454 Ack=1774 Win=4076 Len=0 TSval=105873468 TSecr=24...
628	6.735773	172.20.10.2	216.58.218.194	TCP	66	1454	1854	55496+443 [ACK] Seq=1454 Ack=1854 Win=4093 Len=0 TSval=105873468 TSecr=24...
629	6.735811	172.20.10.2	216.58.218.194	TCP	66	1454	1900	55496+443 [ACK] Seq=1454 Ack=1900 Win=4092 Len=0 TSval=105873468 TSecr=24...
634	6.744776	173.194.115.60	172.20.10.2	TCP	66	5033	924	443+55807 [ACK] Seq=5033 Ack=924 Win=44800 Len=0 TSval=995258842 TSecr=10...
635	6.744788	172.20.10.2	216.58.218.194	TCP	66	1500	2454	55496+443 [ACK] Seq=1500 Ack=2454 Win=4078 Len=0 TSval=105873477 TSecr=24...
636	6.744903	172.20.10.2	216.58.218.194	TCP	66	1500	2534	55496+443 [ACK] Seq=1500 Ack=2534 Win=4076 Len=0 TSval=105873477 TSecr=24...
637	6.744903	172.20.10.2	216.58.218.194	TCP	66	1500	2580	55496+443 [ACK] Seq=1500 Ack=2580 Win=4074 Len=0 TSval=105873477 TSecr=24...
642	6.757425	172.20.10.2	173.194.115.60	TCP	66	924	5194	55807+443 [ACK] Seq=924 Ack=5194 Win=130880 Len=0 TSval=105873489 TSecr=9...
643	6.757426	172.20.10.2	173.194.115.60	TCP	66	924	5274	55807+443 [ACK] Seq=924 Ack=5274 Win=130816 Len=0 TSval=105873489 TSecr=9...
644	6.757469	172.20.10.2	173.194.115.60	TCP	66	924	5320	55807+443 [ACK] Seq=924 Ack=5320 Win=130784 Len=0 TSval=105873489 TSecr=9...
648	6.789845	216.58.218.194	172.20.10.2	TCP	66	2580	1546	443+55496 [ACK] Seq=2580 Ack=1546 Win=487 Len=0 TSval=2418027321 TSecr=10...
649	6.797816	173.194.115.48	172.20.10.2	TCP	66	652	855	443+55310 [ACK] Seq=652 Ack=855 Win=1175 Len=0 TSval=995266436 TSecr=1058...
651	6.814929	172.20.10.2	173.194.115.48	TCP	66	1262	703	55310+443 [ACK] Seq=1262 Ack=703 Win=4094 Len=0 TSval=105873544 TSecr=995...
654	6.815207	172.20.10.2	173.194.115.48	TCP	66	1262	783	55310+443 [ACK] Seq=1262 Ack=783 Win=4093 Len=0 TSval=105873544 TSecr=995...
655	6.815259	172.20.10.2	173.194.115.48	TCP	66	1262	829	55310+443 [ACK] Seq=1262 Ack=829 Win=4092 Len=0 TSval=105873544 TSecr=995...
657	6.827026	173.194.115.60	172.20.10.2	TCP	66	5320	970	443+55807 [ACK] Seq=5320 Ack=970 Win=44800 Len=0 TSval=995258929 TSecr=10...
659	6.853230	172.20.10.2	173.194.115.48	TCP	66	1308	880	55310+443 [ACK] Seq=1308 Ack=880 Win=4094 Len=0 TSval=105873582 TSecr=995...
662	6.853427	172.20.10.2	173.194.115.48	TCP	66	1308	960	55310+443 [ACK] Seq=1308 Ack=960 Win=4091 Len=0 TSval=105873582 TSecr=995...
663	6.853428	172.20.10.2	173.194.115.48	TCP	66	1308	1006	55310+443 [ACK] Seq=1308 Ack=1006 Win=4090 Len=0 TSval=105873582 TSecr=99...
665	6.886490	173.194.115.48	172.20.10.2	TCP	66	1006	1354	443+55310 [ACK] Seq=1006 Ack=1354 Win=1197 Len=0 TSval=995266526 TSecr=10...
671	6.954294	172.20.10.2	199.59.150.44	TCP	78	0	0	55821+80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 WS=32 TSval=105873682 TSecr=...

8. What is the throughput (bytes transferred per unit time) for the TCP connection (Just consider a single connection)? Think on how to calculate the throughput!

Ans.

We consider TCP trace, first sequence number to be packet number 29, and the last packet to be packet number 464.

So, the Throughput = Size of the data/Time

Size of the data = Last ACK Sequence number – First Sequence number

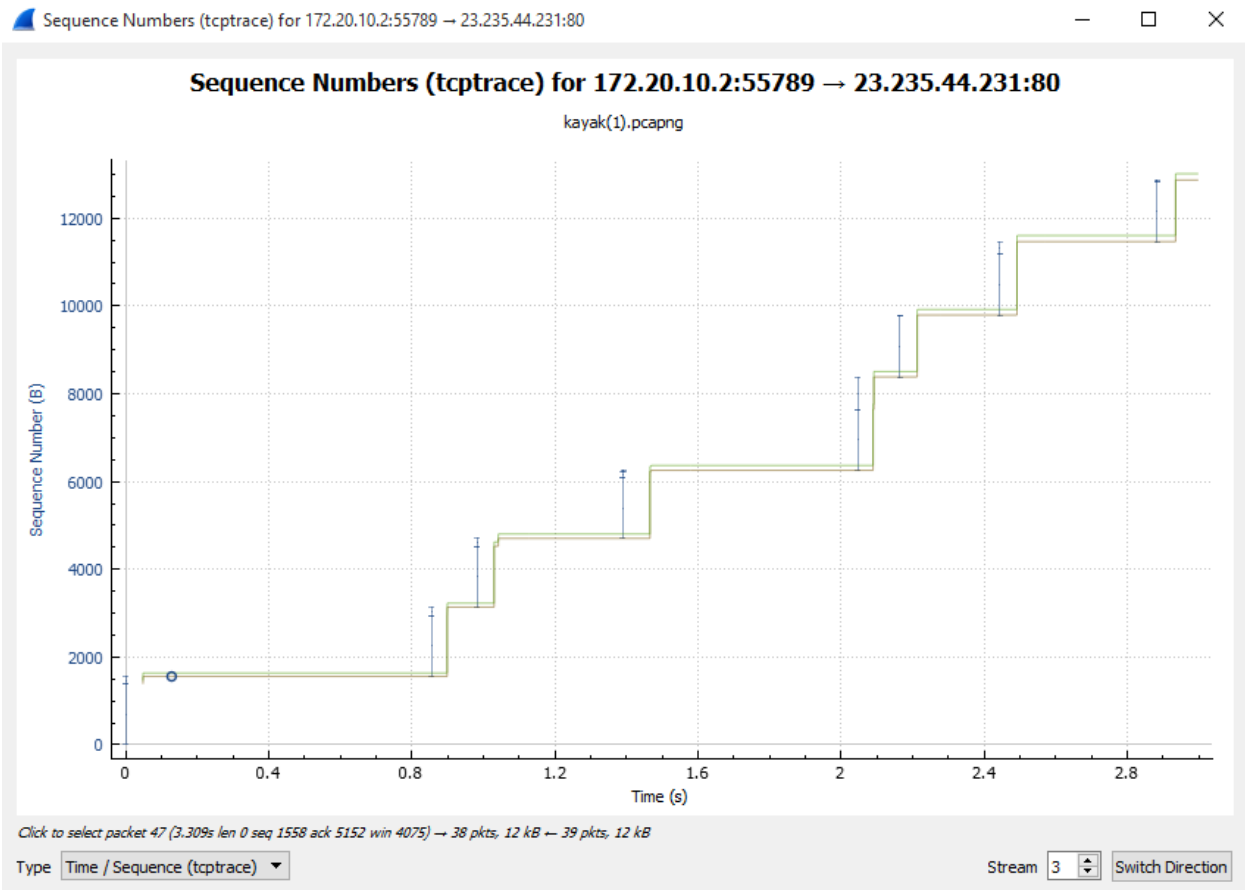
= 12870-1 = 12869

Time = 6.179502 – 3.180418

= 2.999

Throughput= 12869/2.999

= 4291 bytes/sec



9. Explain how you calculated this value.

Ans. We chose the first TCP packet with the Sequence and ACK number as 1.

By using the TCP trace report from **Statistics -> TCP Trace graphs -> TCP Trace** (as per the above image), we found that last ACK packet is 464 with the sequence number 12870.

In order to check the Throughput, we need the size of the data and the time taken.

Size of the data = Last ACK Sequence number – First Sequence number = $12870 - 1 = 12869$

Total Time Taken = $6.179502 - 3.180418 = 2.999$

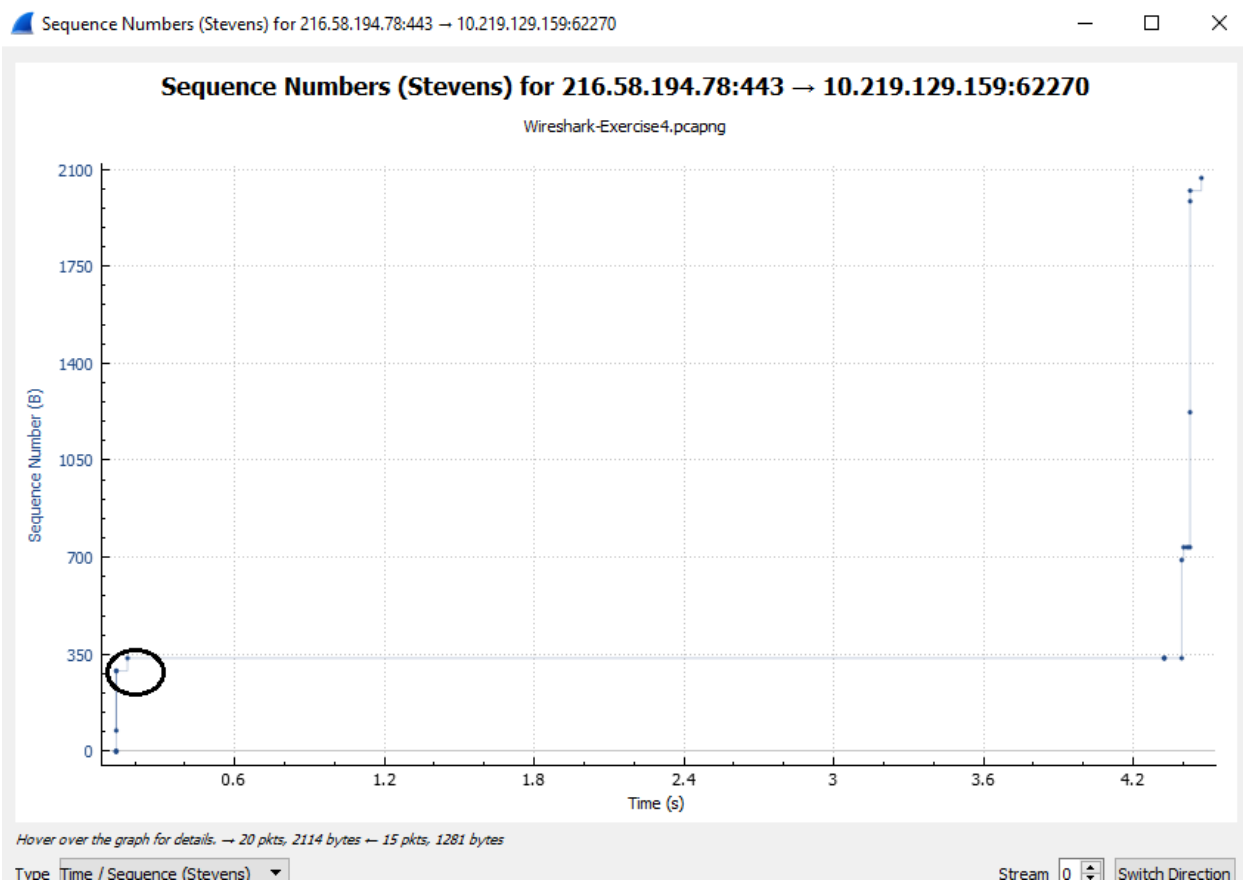
Throughput = $12869 / 2.999 = 4291$ bytes/sec

Problem Set 4

Use the Time-Sequence-Graph(Stevens) plotting tool to view the sequence number versus time plot of segments being sent from youtube.com to your computer. Answer each of three questions below for the trace that you have gathered when you transferred a le to your computer from youtube.com.

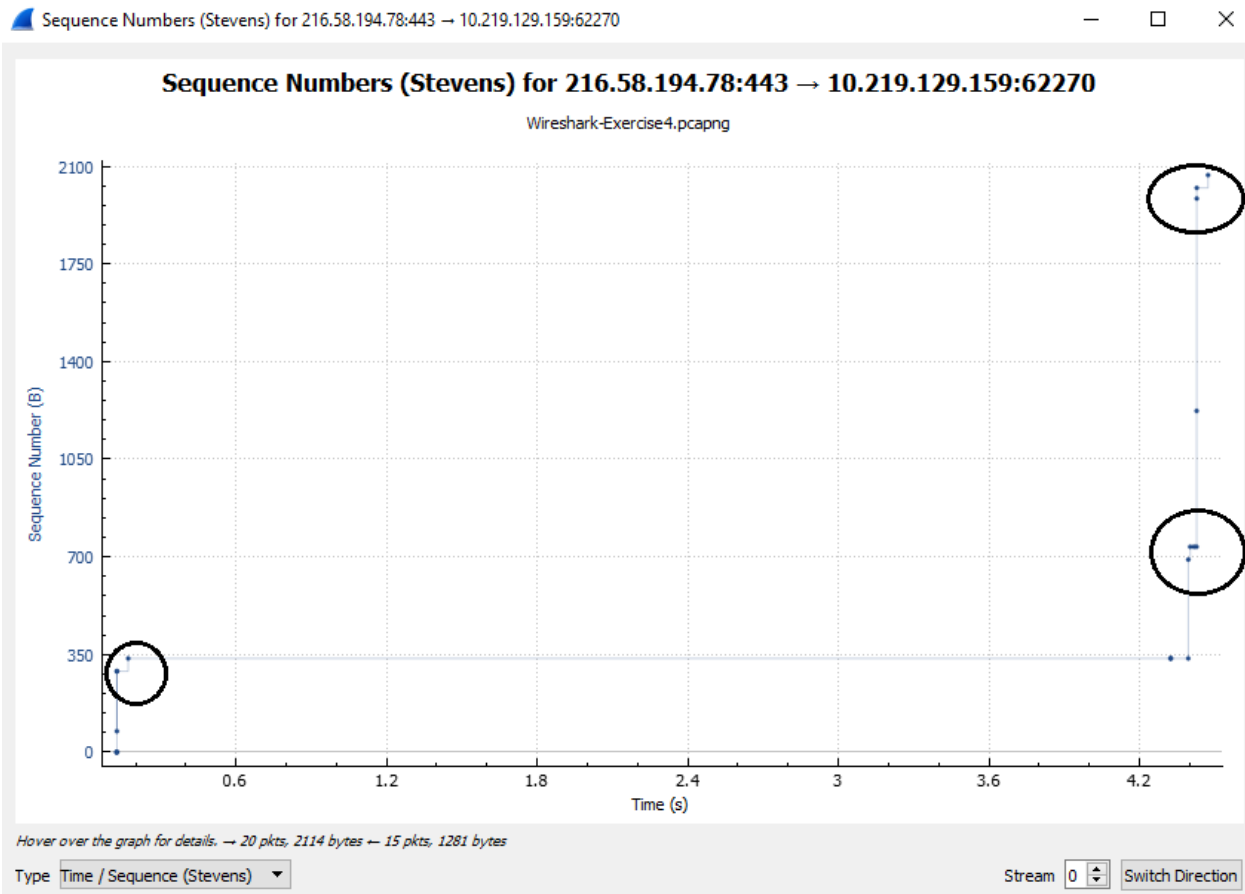
1. Can you identify where TCP's slow-start phase begins and ends.

Ans. TCP slow start phase begin from 0.1 secs



2. Where congestion avoidance takes over? Highlight these areas.

Ans. Congestion takes over at 0.1, 4.45 secs and 4.5 secs



3. Comment on ways in which the measured data differs from the idealized behavior of TCP that we've studied in the text.

Ans. As per the ideal behavior of TCP, the sender is travelling in the same medium. Whereas, the data is travelled in multiple media. There can be more loss of data and retransmissions. And, practically TCP segments travel faster and aggressive in sending data.