Name - Vijay Tadikamalla and Sahil Shah
Roll. no- CS17BTECH11040 and CS17BTECH11035

# Assignment 1 - WireShark
# Computer Networks 2

## TCP

Here is the copy of our trace extracted for the TCP assignment. By default the author's trace provided in the assignment statement is used to answer all questions unless explicitly stated in the problem statement for example in the 3rd question and 14th question.

/home/megatron/Desktop/Networks_assign/down/tcp-ethereal-trace-1 213 total packets, 202 shown

```
No.     Time           Source                Destination           Protocol Length Info
      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]
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
Transmission Control Protocol, Src Port: 1161, Dst Port: 80, Seq: 1, Ack: 1, Len: 565
```

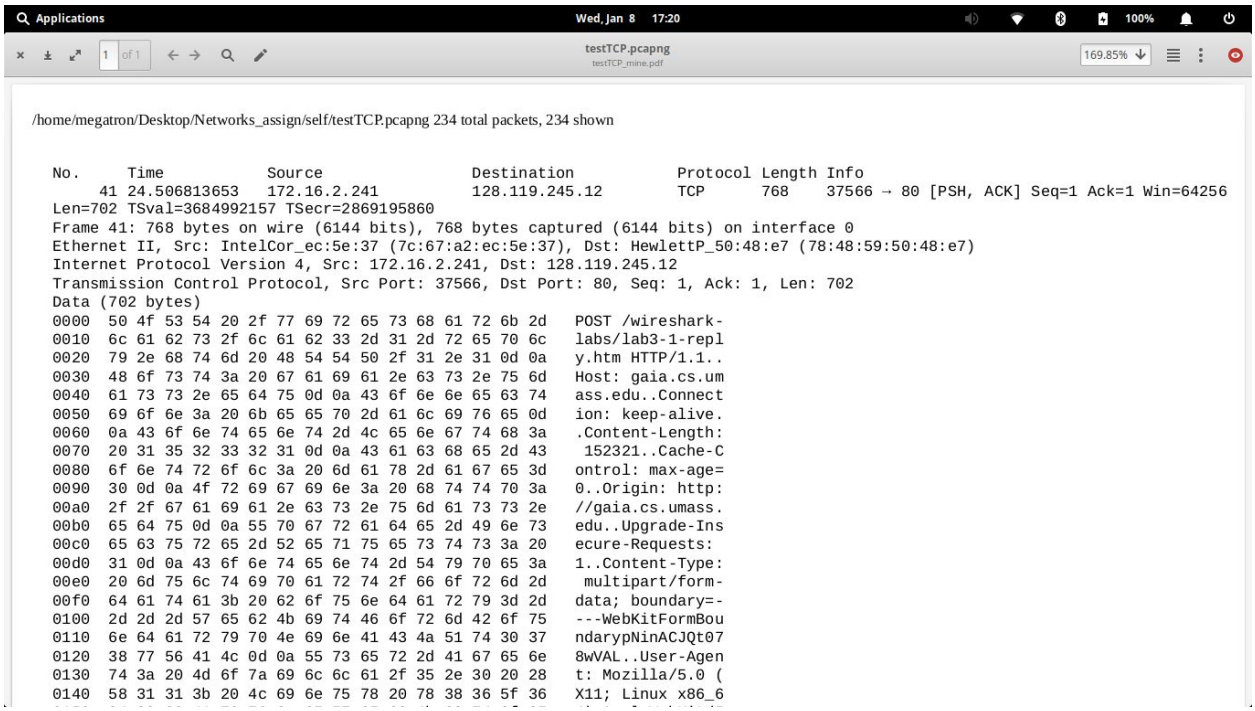
This is the screenshot of the HTTP POST request packet. It clearly shows Source and Destination IPs and 1161 -> 80 (under INFO) denotes the concerned port numbers asked for in the first 2 questions.

**Q1. 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?**

**A1.** IP of Source :- 192.168.1.102. TCP port no of source :- 1161.

**Q2. What is the IP address of gaia.cs.umass.edu? On what port number is it sending and receiving TCP segments for this connection?**

**A2.** IP of Destination :- 128.119.245.12. TCP port no of destination :- 80.



```
/home/megatron/Desktop/Networks_assign/self/testTCP.pcapng 234 total packets, 234 shown

No.     Time           Source                Destination           Protocol Length Info
     41 24.506813653   172.16.2.241          128.119.245.12        TCP      768    37566 → 80 [PSH, ACK] Seq=1 Ack=1 Win=64256
Len=702 TSval=3684992157 TSecr=2869195860
Frame 41: 768 bytes on wire (6144 bits), 768 bytes captured (6144 bits) on interface 0
Ethernet II, Src: IntelCor_ec:5e:37 (7c:67:a2:ec:5e:37), Dst: HewlettP_50:48:e7 (78:48:59:50:48:e7)
Internet Protocol Version 4, Src: 172.16.2.241, Dst: 128.119.245.12
Transmission Control Protocol, Src Port: 37566, Dst Port: 80, Seq: 1, Ack: 1, Len: 702
Data (702 bytes)
0000  50 4f 53 54 20 2f 77 69 72 65 73 68 61 72 6b 2d   POST /wireshark-
0010  6c 61 62 73 2f 6c 61 62 33 2d 31 2d 72 65 70 6c   labs/lab3-1-repl
0020  79 2e 68 74 6d 20 48 54 54 50 2f 31 2e 31 0d 0a   y.htm HTTP/1.1..
0030  48 6f 73 74 3a 20 67 61 69 61 2e 63 73 2e 75 6d   Host: gaia.cs.um
0040  61 73 73 2e 65 64 75 0d 0a 43 6f 6e 6e 65 63 74   ass.edu..Connect
0050  69 6f 6e 3a 20 6b 65 65 70 2d 61 6c 69 76 65 0d   ion: keep-alive.
0060  0a 43 6f 6e 74 65 6e 74 2d 4c 65 6e 67 74 68 3a   .Content-Length:
0070  20 31 35 32 33 32 31 0d 0a 43 61 63 68 65 2d 43    152321..Cache-C
0080  6f 6e 74 72 6f 6c 3a 20 6d 61 78 2d 61 67 65 3d   ontrol: max-age=
0090  30 0d 0a 4f 72 69 67 69 6e 3a 20 68 74 74 70 3a   0..Origin: http:
00a0  2f 2f 67 61 69 61 2e 63 73 2e 75 6d 61 73 73 2e   //gaia.cs.umass.
00b0  65 64 75 0d 0a 55 70 67 72 61 64 65 2d 49 6e 73   edu..Upgrade-Ins
00c0  65 63 75 72 65 2d 52 65 71 75 65 73 74 73 3a 20   ecure-Requests:
00d0  31 0d 0a 43 6f 6e 74 65 6e 74 2d 54 79 70 65 3a   1..Content-Type:
00e0  20 6d 75 6c 74 69 70 61 72 74 2f 66 6f 72 6d 2d    multipart/form-
00f0  64 61 74 61 3b 20 62 6f 75 6e 64 61 72 79 3d 2d   data; boundary=-
0100  2d 2d 2d 57 65 62 4b 69 74 46 6f 72 6d 42 6f 75   ---WebKitFormBou
0110  6e 64 61 72 79 70 4e 69 6e 41 43 4a 51 74 30 37   ndarypNinACJQt07
0120  38 77 56 41 4c 0d 0a 55 73 65 72 2d 41 67 65 6e   8wVAL..User-Agen
0130  74 3a 20 4d 6f 7a 69 6c 6c 61 2f 35 2e 30 20 28   t: Mozilla/5.0 (
0140  58 31 31 3b 20 4c 69 6e 75 78 20 78 38 36 5f 36   X11; Linux x86_6
```

This screenshot helps to answer question 3 using my trace (as explicitly stated).

**Q3. What is the IP address and TCP port number used by your client computer (source) to transfer the file to gaia.cs.umass.edu?**

**A3.** My IP :- 172.16.2.241. Used TCP port number :- 37566.

tcp-ethereal-trace-1
syn-1.pdf

/home/megatron/Desktop/Networks_assign/down/tcp-ethereal-trace-1 213 total packets, 202 shown

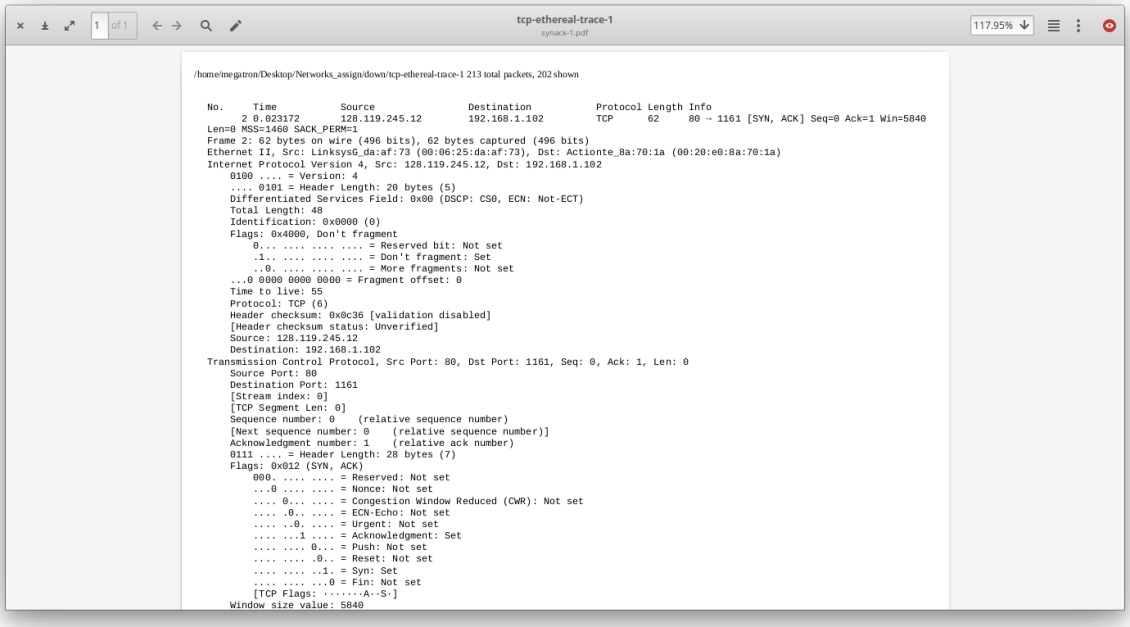
```
No.     Time           Source                Destination           Protocol Length Info
      1 0.000000       192.168.1.102         128.119.245.12        TCP      62     1161 → 80 [SYN] Seq=0 Win=16384 Len=0
MSS=1460 SACK_PERM=1
Frame 1: 62 bytes on wire (496 bits), 62 bytes captured (496 bits)
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
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
    Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 48
    Identification: 0x1e1d (7709)
    Flags: 0x4000, Don't fragment
        0... .... .... .... = Reserved bit: Not set
        .1.. .... .... .... = Don't fragment: Set
        ..0. .... .... .... = More fragments: Not set
    ...0 0000 0000 0000 = Fragment offset: 0
    Time to live: 128
    Protocol: TCP (6)
    Header checksum: 0xa518 [validation disabled]
    [Header checksum status: Unverified]
    Source: 192.168.1.102
    Destination: 128.119.245.12
Transmission Control Protocol, Src Port: 1161, Dst Port: 80, Seq: 0, Len: 0
    Source Port: 1161
    Destination Port: 80
    [Stream index: 0]
    [TCP Segment Len: 0]
    Sequence number: 0    (relative sequence number)
    [Next sequence number: 0    (relative sequence number)]
```

tcp-ethereal-trace-1
syn-1.pdf

```
    [Header checksum status: Unverified]
    Source: 192.168.1.102
    Destination: 128.119.245.12
Transmission Control Protocol, Src Port: 1161, Dst Port: 80, Seq: 0, Len: 0
    Source Port: 1161
    Destination Port: 80
    [Stream index: 0]
    [TCP Segment Len: 0]
    Sequence number: 0    (relative sequence number)
    [Next sequence number: 0    (relative sequence number)]
    Acknowledgment number: 0
    0111 .... = Header Length: 28 bytes (7)
    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
        [TCP Flags: ··********S·]
    Window size value: 16384
    [Calculated window size: 16384]
    Checksum: 0xf6e9 [unverified]
    [Checksum Status: Unverified]
    Urgent pointer: 0
    Options: (8 bytes), Maximum segment size, No-Operation (NOP), No-Operation (NOP), SACK permitted
    [Timestamps]
```

**Q4. 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?**

**A4.** In the given trace TCP segment 1 is used to initiate the TCP connection between the client computer and gaia.cs.umass.edu. The value of the sequence number of the TCP SYN segment is 0. The SYN flag is set to 1 which indicates that this segment is a SYN segment.



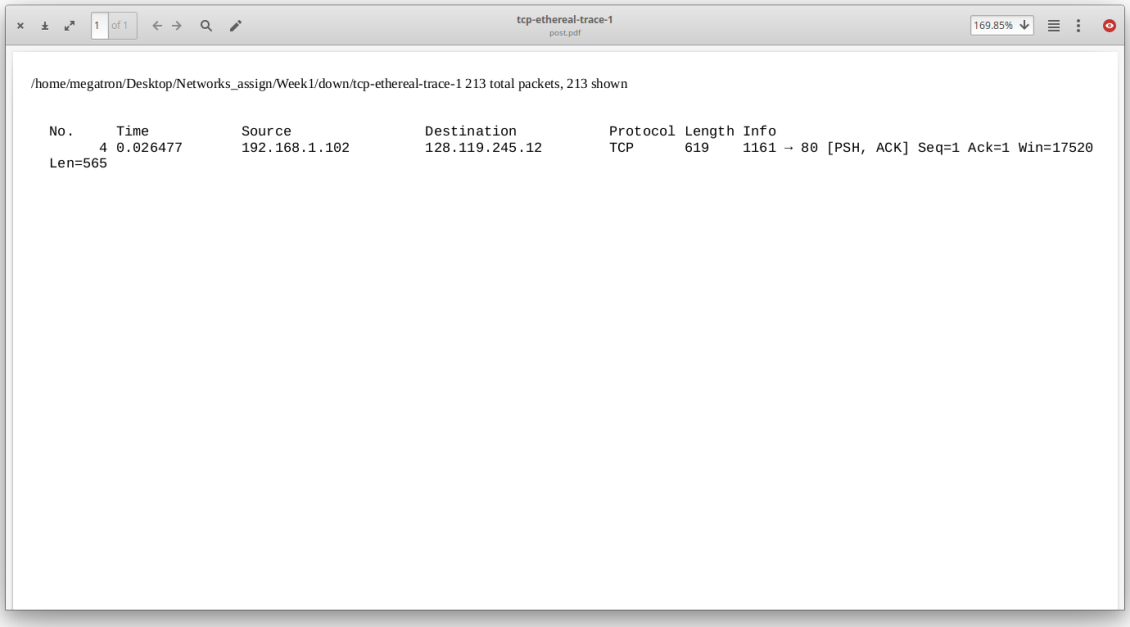
tcp-ethereal-trace-1
synack-1.pdf

/home/megatron/Desktop/Networks_assign/down/tcp-ethereal-trace-1 213 total packets, 202 shown

```
No.     Time           Source                Destination           Protocol Length Info
      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
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
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
    Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 48
    Identification: 0x0000 (0)
    Flags: 0x4000, Don't fragment
        0... .... .... .... = Reserved bit: Not set
        .1.. .... .... .... = Don't fragment: Set
        ..0. .... .... .... = More fragments: Not set
    ...0 0000 0000 0000 = Fragment offset: 0
    Time to live: 55
    Protocol: TCP (6)
    Header checksum: 0x0c36 [validation disabled]
    [Header checksum status: Unverified]
    Source: 128.119.245.12
    Destination: 192.168.1.102
Transmission Control Protocol, Src Port: 80, Dst Port: 1161, Seq: 0, Ack: 1, Len: 0
    Source Port: 80
    Destination Port: 1161
    [Stream index: 0]
    [TCP Segment Len: 0]
    Sequence number: 0    (relative sequence number)
    [Next sequence number: 0    (relative sequence number)]
    Acknowledgment number: 1    (relative ack number)
    0111 .... = Header Length: 28 bytes (7)
    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 size value: 5840
```

**Q5. 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?**

**A5.** Sequence number of the SYNACK segment from gaia.cs.umass.edu to the client computer in reply to the SYN has the value of 0. The value of the ACKnowledgement field in the SYNACK segment is 1. The value of the ACKnowledgement field in the SYNACK segment is determined by gaia.cs.umass.edu by adding 1 to the initial sequence number of SYN segment from the client computer (i.e. the sequence number of the SYN segment initiated by the client computer is 0, which is exactly what is asked in the previous question). The SYN flag and Acknowledgement flag in the segment are set to 1 and they indicate that this segment is a SYNACK segment.



tcp-ethereal-trace-1
post.pdf

/home/megatron/Desktop/Networks_assign/Week1/down/tcp-ethereal-trace-1 213 total packets, 213 shown

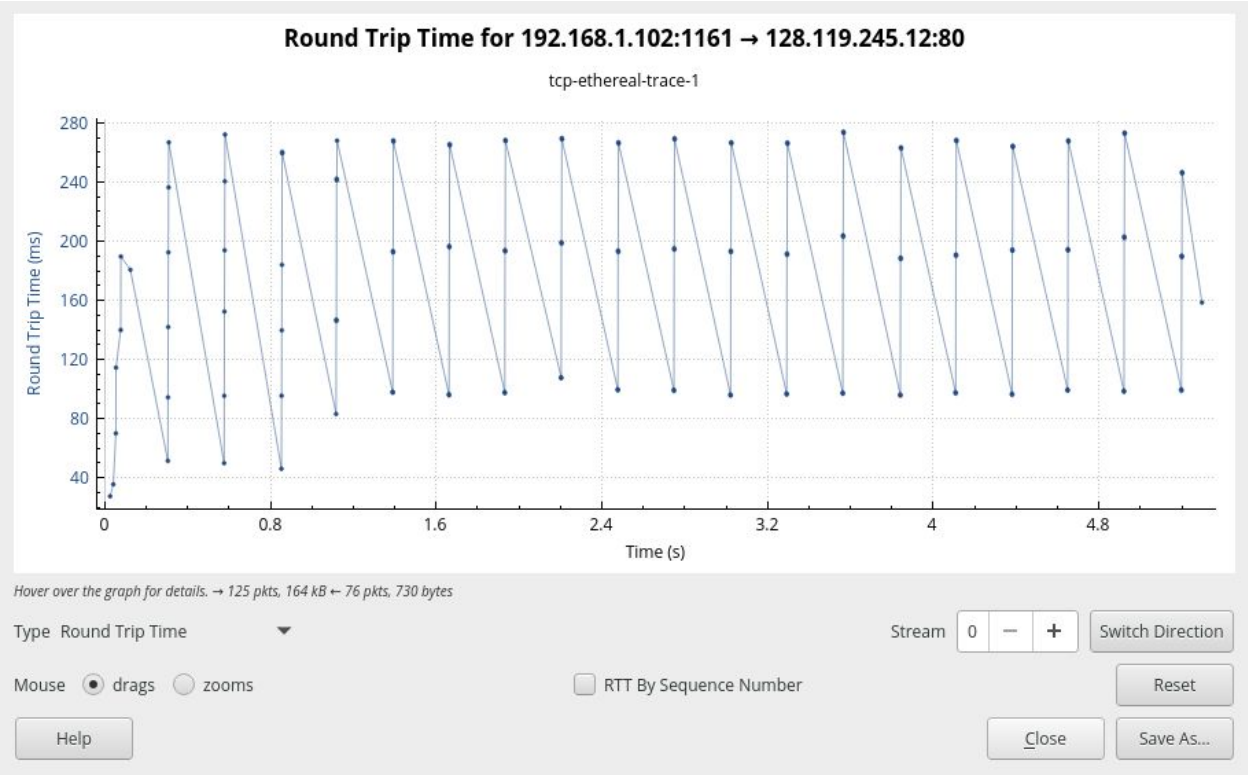
| No. | Time | Source | Destination | Protocol | Length | Info |
|---|---|---|---|---|---|---|
| 4 | 0.026477 | 192.168.1.102 | 128.119.245.12 | TCP | 619 | 1161 → 80 [PSH, ACK] Seq=1 Ack=1 Win=17520 Len=565 |

**Q6. What is the sequence number of the TCP segment containing the HTTP POST command?**

**A6.** TCP segment number 4 contains the HTTP POST command. The sequence number of this segment has a value of 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 |
| 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 |
| 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 |
| 8 | 0.054690 | 192.168.1.102 | 128.119.245.12 | TCP | 1514 | 1161 → 80 [ACK] Seq=3486 Ack=1 Win=17520 Len=1460 |
| 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 |
| 11 | 0.078157 | 192.168.1.102 | 128.119.245.12 | TCP | 1514 | 1161 → 80 [ACK] Seq=6406 Ack=1 Win=17520 Len=1460 |
| 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 |

List of TCP segments. We will solve the question below with the selected segment (4) as the first segment. The lengths can be seen in this image.




Round trip time graph with increasing segment numbers, starting from segment 4.

**Q7. 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 page 249 in text) after the receipt of each ACK?**

**A7.** The HTTP POST segment is considered as the first segment. Segments 1 – 6 are No. 4, 5, 7, 8, 10, and 11 in this trace respectively. The ACKs of segments 1 – 6 are No. 6, 9, 12, 14, 15, and 16 in this trace.

EstimatedRTT = 0.875 * EstimatedRTT + 0.125 * SampleRTT. Using this formula and the measurements in table we calculate the below values. All measurements are in seconds.

| Segment No. | Sequence No. | SendTime | ACK-Time | RTT | Estimated RTT |
|---|---|---|---|---|---|
| 1 | 1 | 0.026477 | 0.053937 | 0.02746 | 0.02746 |
| 2 | 566 | 0.041737 | 0.077294 | 0.035557 | 0.028472125 |
| 3 | 2026 | 0.054026 | 0.124085 | 0.070059 | 0.03367048438 |
| 4 | 3486 | 0.05469 | 0.169118 | 0.114428 | 0.04376517383 |
| 5 | 4946 | 0.077405 | 0.217299 | 0.139894 | 0.0557812771 |
| 6 | 6406 | 0.078157 | 0.267802 | 0.189645 | 0.07251424246 |

Estimated RTT :- 0.725s

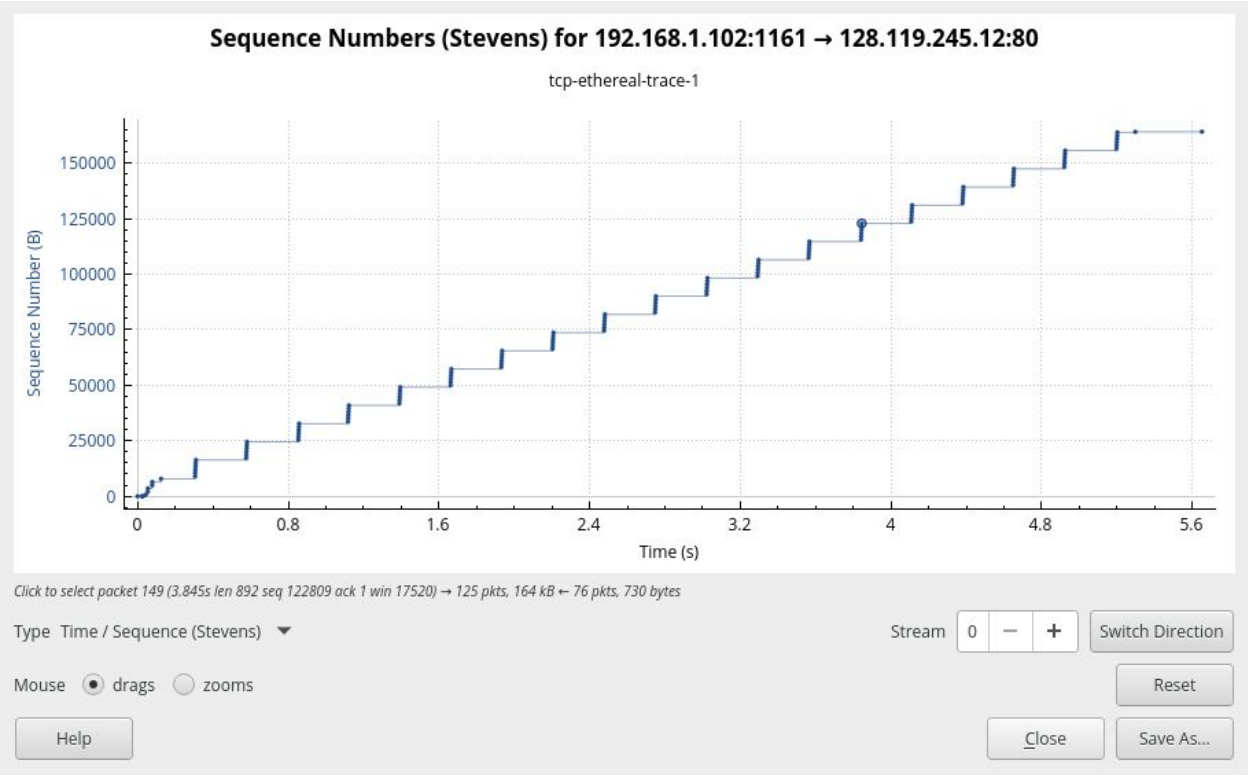
**Q8. What is the length of each of the first six TCP segments?**

**A8.** Length of the first TCP segment 565 bytes Length of each of the other five TCP segments 1460 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 |
| 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 |
| 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 |
| 8 | 0.054690 | 192.168.1.102 | 128.119.245.12 | TCP | 1514 | 1161 → 80 [ACK] Seq=3486 Ack=1 Win=17520 Len=1460 |
| 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 |
| 11 | 0.078157 | 192.168.1.102 | 128.119.245.12 | TCP | 1514 | 1161 → 80 [ACK] Seq=6406 Ack=1 Win=17520 Len=1460 |
| 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 |
| 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 |

**Q9. 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?**

**A9.** The minimum amount of buffer space (at the receiver window) advertised at gaia.cs.umass.edu for the entire trace is 5840 bytes, which shows in the first acknowledgement from the server. This receiver window grows until a maximum receiver buffer size of 62780 bytes. The receiver window size doesn't cause any issues as it is safely less than the max available at all times throughout the connection, thus it doesn't throttle the sender.




**Q10. Are there any retransmitted segments in the trace file? What did you check for (in the trace) in order to answer this question?**

**A10.** There are no retransmitted segments in the trace file. We can verify this by checking the sequence numbers of the TCP segments in the trace file. In the TimeSequence-Graph (Stevens) of this trace, all sequence numbers from the source (192.168.1.102) to the destination (128.119.245.12) are increasing monotonically with respect to time. This clearly shows that the client keeps sending packets from the file from top to bottom, without failing at any packet. Incase some packet would have been lost/not ACK by the receiver, the sender would have to resend it, breaking the monotonicity of the seq number.

| | | | | | | |
|---|---|---|---|---|---|---|
| 83 | 1.932757 | 192.168.1.102 | 128.119.245.12 | TCP | 1514 | 1161 → 80 [ACK] Seq=61085 Ack=1 Win=17520 Len=1460 |
| 84 | 1.933636 | 192.168.1.102 | 128.119.245.12 | TCP | 1514 | 1161 → 80 [ACK] Seq=62545 Ack=1 Win=17520 Len=1460 |
| 85 | 1.934770 | 192.168.1.102 | 128.119.245.12 | TCP | 1514 | 1161 → 80 [ACK] Seq=64005 Ack=1 Win=17520 Len=1460 |
| 86 | 1.935586 | 192.168.1.102 | 128.119.245.12 | TCP | 946 | 1161 → 80 [PSH, ACK] Seq=65465 Ack=1 Win=17520 Len=892 |
| 87 | 2.029069 | 128.119.245.12 | 192.168.1.102 | TCP | 60 | 80 → 1161 [ACK] Seq=1 Ack=61085 Win=62780 Len=0 |
| 88 | 2.126682 | 128.119.245.12 | 192.168.1.102 | TCP | 60 | 80 → 1161 [ACK] Seq=1 Ack=64005 Win=62780 Len=0 |
| 89 | 2.293195 | 128.119.245.12 | 192.168.1.102 | TCP | 60 | 80 → 1161 [ACK] Seq=1 Ack=66257 Win=62780 Len=0 |

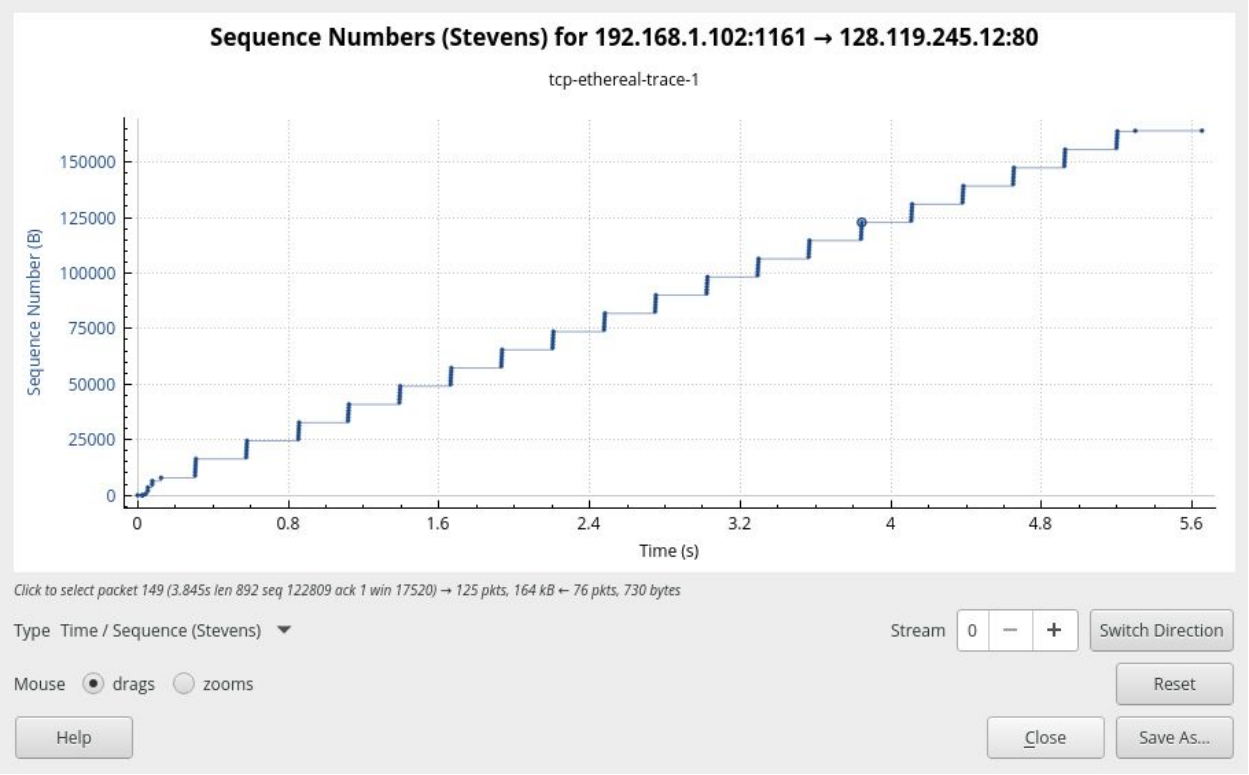
**Q11. How much data does the receiver typically acknowledge in an ACK? Can you identify cases where the receiver is ACKing every other received segment (see Table 3.2 on page 257 in the text).**

**A11.** Typically the receiver ACKs data sent from one segment at once. The difference between the acknowledged sequence numbers of two consecutive ACKs indicates the data received by the server between these two ACKs. By inspecting the amount of acknowledged data by each ACK, there are cases where the receiver is ACKing every other segment. For example, segments highlighted above. The receiver here skips ACK value 62545 by ACKing two packets at once.

| No. | Time | Source | Destination | Protocol | Length | Info |
|---|---|---|---|---|---|---|
| 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 |
| 202 | 5.455830 | 128.119.245.12 | 192.168.1.102 | TCP | 60 | 80 → 1161 [ACK] Seq=1 Ack=164091 Win=62780 Len=0 |

**Q12. What is the throughput (bytes transferred per unit time) for the TCP connection? Explain how you calculated this value.**

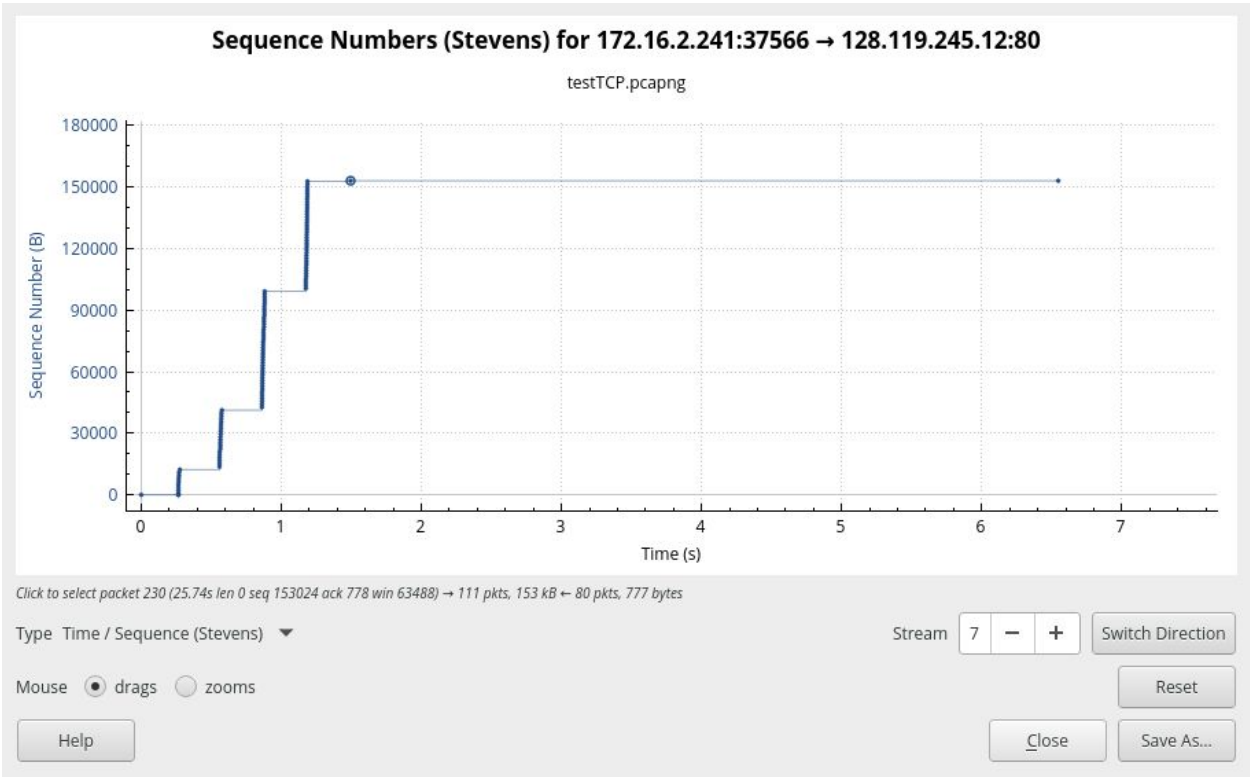
**A12.** We compute the TCP throughput as the total data transferred divided by the whole connection time (not including the handshake time). The total data transmitted can be computed by the difference between the sequence number of the first TCP segment (i.e. 1 byte for No. 4 segment) and the acknowledged sequence number of the last ACK (164091 bytes for No. 202 segment). Therefore, the total data are 164091 - 1 = 164090 bytes. Therefore, the total transmission time is 5.455830 - 0.026477 = 5.4294 seconds. Hence, the throughput for the TCP connection is computed as 164090/5.4294 = 30.222 KByte/sec.




**Q13. 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.**

**A13.** The slow start period starts at very close to 0 when the first packet is sent. The slow start period ends when the congestion window size crosses the set threshold. Upto this point the packets are sent slowly as the window size is low, but it is increasing exponentially. As soon as it crosses the threshold, we get sets of packets in the graph sent in bursts, denoting that multiple packets are sent at almost the same time and the number of packets in these bursts can increase linearly with time as the window size in this stage increases linearly with time. At this point Congestion Avoidance stage has begun (around 0.25s).

The below question as stated in the problem statement is answered by the generated trace.




**Q14. Answer each of two questions above for the trace that you have gathered when you transferred a file from your computer to gaia.cs.umass.edu**

**A14.** In the generated trace the Congestion Avoidance state is encountered much later at around 1s. Up To this mark the window size grows exponentially as is supposed to in the Slow start stage. After this the window size increases linearly (as is seen the number of packets sent in one burst just after 1s and just before 1s are almost the same).

# HTTP

**1. Is your browser running HTTP version 1.0 or 1.1? What version of HTTP is the server running?**

My browser and server both are running on http version 1.1

```
374 8.421105137   192.168.105.228   128.119.245.12   HTTP   526 GET /wireshark-labs/HTTP-wire
384 8.694872595   128.119.245.12    192.168.105.228  HTTP   552 HTTP/1.1 200 OK  (text/html)

▸ Frame 384: 552 bytes on wire (4416 bits), 552 bytes captured (4416 bits) on interface 0
▸ Ethernet II, Src: HewlettP_a9:8b:f2 (5c:8a:38:a9:8b:f2), Dst: QuantaCo_46:de:fb (a8:1e:84:46:de:fb)
▸ Internet Protocol Version 4, Src: 128.119.245.12, Dst: 192.168.105.228
▸ Transmission Control Protocol, Src Port: 80, Dst Port: 60474, Seq: 1, Ack: 461, Len: 486
▾ Hypertext Transfer Protocol
  ▸ HTTP/1.1 200 OK\r\n
    Date: Wed, 08 Jan 2020 16:53:06 GMT\r\n
```

**2. What languages (if any) does your browser indicate that it can accept to the server?**

My browser indicates that it will accept English-US and English languages from the server

```
▾ Hypertext Transfer Protocol
  ▸ GET /wireshark-labs/HTTP-wireshark-file1.html HT
    Host: gaia.cs.umass.edu\r\n
    Connection: keep-alive\r\n
    Upgrade-Insecure-Requests: 1\r\n
    User-Agent: Mozilla/5.0 (X11; Linux x86_64) Appl
    Accept: text/html,application/xhtml+xml,applicat
    Accept-Encoding: gzip, deflate\r\n
    Accept-Language: en-GB,en-US;q=0.9,en;q=0.8\r\n
```

**3. What is the IP address of your computer? Of the gaia.cs.umass.edu server?**

The IP address of my computer is 192.168.105.228 and IP address of the server is 128.119.245.12.

| | Time | Source | Destination | Protocol |
|---|---|---|---|---|
| 374 | 8.421105137 | 192.168.105.228 | 128.119.245.12 | HTTP |
| 384 | 8.694872595 | 128.119.245.12 | 192.168.105.228 | HTTP |

**4. What is the status code returned from the server to your browser?**

The status code returned was 200 OK

| Protocol | Length | Info |
|---|---|---|
| HTTP | 526 | GET /wireshark-labs/HTTP-wire |
| HTTP | 552 | HTTP/1.1 200 OK (text/html) |

**5. When was the HTML file that you are retrieving last modified at the server?**

The file was last modified on Wed,08 Jan 2020 16:53:06 GMT

```
Hypertext Transfer Protocol
  HTTP/1.1 200 OK\r\n
  Date: Wed, 08 Jan 2020 16:53:06 GMT\r\n
  Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips
  Last-Modified: Wed, 08 Jan 2020 06:59:01 GMT\r\n
```

**6. How many bytes of content are being returned to your browser?**

128 bytes of content are being returned

```
Content-Length: 128\r\n
Keep-Alive: timeout=5, max=100\r\n
Connection: Keep-Alive\r\n
```

**7. By inspecting the raw data in the packet content window, do you see any headers within the data that are not displayed in the packet-listing window? If so, name one.**

No difference.

**8. Inspect the contents of the first HTTP GET request from your browser to the server. Do you see an "IF-MODIFIED-SINCE" line in the HTTP GET?**

No there is no IF-MODIFIED-SINCE line in the HTTP GET message

**9. Inspect the contents of the server response. Did the server explicitly return the contents of the file? How can you tell?**

We can see all the contents of the file in the section Line-based text data.

```
Line-based text data: text/html (10 lines)
    \n
    <html>\n
    \n
    Congratulations again!  Now you've downloaded the file lab2-2.html. <br>\n
    This file's last modification date will not change.  <p>\n
    Thus  if you download this multiple times on your browser, a complete copy <br>\n
    will only be sent once by the server due to the inclusion of the IN-MODIFIED-SINCE<br>\n
    field in your browser's HTTP GET request to the server.\n
    \n
    </html>\n
```

**10. Now inspect the contents of the second HTTP GET request from your browser to the server. Do you see an "IF-MODIFIED-SINCE:" line in the HTTP GET? If so, what information follows the "IF-MODIFIED-SINCE:" header?**

Yes in the second HTTP message an IF-MODIFIED-SINCE line is included. The information that follows is the date and time that I last accessed the webpage

```
Hypertext Transfer Protocol
  GET /wireshark-labs/HTTP-wireshark-file2.html HTTP/1.1\r\n
  Host: gaia.cs.umass.edu\r\n
  Connection: keep-alive\r\n
  Cache-Control: max-age=0\r\n
  Upgrade-Insecure-Requests: 1\r\n
  User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chr
  Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/
  Accept-Encoding: gzip, deflate\r\n
  Accept-Language: en-GB,en-US;q=0.9,en;q=0.8\r\n
  If-None-Match: "173-59b9b6d6f5286"\r\n
  If-Modified-Since: Wed, 08 Jan 2020 06:59:01 GMT\r\n
  \r\n
  [Full request URI: http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file2.html]
  [HTTP request 2/2]
  [Prev request in frame: 507]
  [Response in frame: 976]
```

**11. What is the HTTP status code and phrase returned from the server in response to this second HTTP GET? Did the server explicitly return the contents of the file? Explain.**

The HTTP status code is "304: Not Modified". The server did not return the contents of the file because the browser simply retrieved the contents from its cache.

**12. How many HTTP GET request messages did your browser send? Which packet number in the trace contains the GET message for the Bill or Rights?**

Only 1 GET request message was made to the server. The packer no. 402 contained the GET message.

| | | | | | |
|---|---|---|---|---|---|
| 402 | 6.287301462 | 192.168.105.228 | 128.119.245.12 | HTTP | 526 GET /wireshark-l |
| 416 | 6.557308785 | 128.119.245.12 | 192.168.105.228 | HTTP | 583 HTTP/1.1 200 OK |

**13. Which packet number in the trace contains the status code and phrase associated with the response to the HTTP GET request?**

The packer no. 416 contains the status code and phrase.

| | | | | | |
|---|---|---|---|---|---|
| 402 | 6.287301462 | 192.168.105.228 | 128.119.245.12 | HTTP | 526 GET /wireshark-l |
| 416 | 6.557308785 | 128.119.245.12 | 192.168.105.228 | HTTP | 583 HTTP/1.1 200 OK |

**14. What is the status code and phrase in the response?**

The code and phrase in the response was 200 OK.

| | | | | | |
|---|---|---|---|---|---|
| 416 | 6.557308785 | 128.119.245.12 | 192.168.105.228 | HTTP | 583 HTTP/1.1 200 OK (text/html) |

**15. How many data-containing TCP segments were needed to carry the single HTTP response and the text of the Bill of Rights?**

The data was sent in 4 TCP segments to the browser, then reassembled.

```
▸ Transmission Control Protocol, Src Port: 80, Dst Port: 32820, Seq: 4345, Ack: 461, Len: 517
▸ [4 Reassembled TCP Segments (4861 bytes): #410(1448), #412(1448), #414(1448), #416(517)]
▾ Hypertext Transfer Protocol
  ▸ HTTP/1.1 200 OK\r\n
```

**16. How many HTTP GET request messages did your browser send? To which Internet addresses were these GET requests sent?**

My browser sent 3 http GET message requests. Each GET message request was made to IP address: 128.119.245.12.

| No. | Time | Source | Destination | Protocol | Length | Info |
|---|---|---|---|---|---|---|
| 2232 | 36.268827516 | 192.168.105.228 | 128.119.245.12 | HTTP | 526 | GET /wireshark-la |
| 2256 | 36.539733823 | 128.119.245.12 | 192.168.105.228 | HTTP | 1139 | HTTP/1.1 200 OK |
| 2258 | 36.654135285 | 192.168.105.228 | 128.119.245.12 | HTTP | 464 | GET /pearson.png |
| 2274 | 36.919954708 | 192.168.105.228 | 128.119.245.12 | HTTP | 478 | GET /~kurose/cove |
| 2279 | 36.921621689 | 128.119.245.12 | 192.168.105.228 | HTTP | 781 | HTTP/1.1 200 OK |
| 2461 | 38.131402659 | 128.119.245.12 | 192.168.105.228 | HTTP | 1472 | HTTP/1.1 200 OK |

**17. Can you tell whether your browser downloaded the two images serially, or whether they were downloaded from the two websites in parallel? Explain.**

The images were downloaded serially because the request for the first image was sent before the second image.

**18. What is the server's response (status code and phrase) in response to the initial HTTP GET message from your browser?**

The servers initial response was "401 Unauthorized".

| No. | Time | Source | Destination | Protocol | Length | Info |
|---|---|---|---|---|---|---|
| 1156 | 25.345907046 | 192.168.105.228 | 128.119.245.12 | HTTP | 542 | GET /wireshark-labs/protected_pages/HTTP-wires |
| 1163 | 25.626143788 | 128.119.245.12 | 192.168.105.228 | HTTP | 783 | HTTP/1.1 401 Unauthorized (text/html) |
| 2659 | 70.430210490 | 192.168.105.228 | 128.119.245.12 | HTTP | 601 | GET /wireshark-labs/protected_pages/HTTP-wires |
| 2671 | 70.720735299 | 128.119.245.12 | 192.168.105.228 | HTTP | 556 | HTTP/1.1 200 OK (text/html) |

**19. When your browser sends the HTTP GET message for the second time, what new field is included in the HTTP GET message?**

The new field that is now included is the authorization field.

```
▾ Hypertext Transfer Protocol
  ▾ GET /wireshark-labs/protected_pages/HTTP-wireshark-file5.html HTTP/1.1\r\n
    ▸ [Expert Info (Chat/Sequence): GET /wireshark-labs/protected_pages/HTTP-w
      Request Method: GET
      Request URI: /wireshark-labs/protected_pages/HTTP-wireshark-file5.html
      Request Version: HTTP/1.1
    Host: gaia.cs.umass.edu\r\n
    Connection: keep-alive\r\n
  ▸ Authorization: Basic d2lyZXNoYXJrLXN0dWRlbnRzOm5ldHdvcms=\r\n
    Upgrade-Insecure-Requests: 1\r\n
    User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like
    Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,im
    Accept-Encoding: gzip, deflate\r\n
    Accept-Language: en-GB,en-US;q=0.9,en;q=0.8\r\n
```

# DNS

1. **Run nslookup to obtain the IP address of a Web server in Asia.**

```
➜ ~ nslookup www.iith.ac.in
Server:         127.0.0.53
Address:        127.0.0.53#53

Non-authoritative answer:
Name:   www.iith.ac.in
Address: 192.168.35.56
```

2. **Run nslookup to determine the authoritative DNS servers for a university in Europe.**

```
➜ ~ nslookup -type=NS cam.ac.uk
Server:         127.0.0.53
Address:        127.0.0.53#53

Non-authoritative answer:
cam.ac.uk       nameserver = authdns0.csx.cam.ac.uk.
cam.ac.uk       nameserver = sns-pb.isc.org.
cam.ac.uk       nameserver = dns0.eng.cam.ac.uk.
cam.ac.uk       nameserver = dns0.cl.cam.ac.uk.
cam.ac.uk       nameserver = ns2.ic.ac.uk.

Authoritative answers can be found from:
```

3. **Run nslookup so that one of the DNS servers obtained in Question 2 is queried for the mail servers for Yahoo! mail.**

```
➜ ~ nslookup cam.ac.uk mail.yahoo.com
;; connection timed out; no servers could be reached
```

4. **Locate the DNS query and response messages. Are then sent over UDP or TCP?**

   The DNS query and response messages are sent over UDP.

ip.addr == 192.168.105.228 Express

| No. | Time | Source | Destination | Protocol | Length | Info |
|---|---|---|---|---|---|---|
| 315 | 3.495175069 | 172.217.163.132 | 192.168.105.228 | TLSv1.3 | 313 | Application Data |
| 316 | 3.495182157 | 192.168.105.228 | 172.217.163.132 | TCP | 66 | 51548 → 443 [ACK] Seq=1904 Ack=1352 Win=6412 |
| 317 | 3.495658938 | 172.217.163.132 | 192.168.105.228 | TLSv1.3 | 97 | Application Data |
| 318 | 3.495677337 | 192.168.105.228 | 172.217.163.132 | TCP | 66 | 51548 → 443 [ACK] Seq=1904 Ack=1383 Win=6412 |
| 319 | 3.495682509 | 172.217.163.132 | 192.168.105.228 | TLSv1.3 | 105 | Application Data |
| 320 | 3.495687678 | 192.168.105.228 | 172.217.163.132 | TCP | 66 | 51548 → 443 [ACK] Seq=1904 Ack=1422 Win=6412 |
| 321 | 3.496412758 | 192.168.105.228 | 172.217.163.132 | TLSv1.3 | 101 | Application Data |
| 322 | 3.496463511 | 192.168.105.228 | 172.217.163.132 | TLSv1.3 | 105 | Application Data |
| 324 | 3.506862489 | 192.168.105.228 | 192.168.36.53 | DNS | 83 | Standard query 0xa43e A www.ietf.org OPT |
| 325 | 3.516362354 | 172.217.163.132 | 192.168.105.228 | TCP | 66 | 443 → 51548 [ACK] Seq=1422 Ack=1939 Win=6374 |
| 326 | 3.516395696 | 172.217.163.132 | 192.168.105.228 | TCP | 66 | 443 → 51548 [ACK] Seq=1422 Ack=1978 Win=6374 |
| 333 | 3.731813102 | 192.168.36.53 | 192.168.105.228 | DNS | 160 | Standard query response 0xa43e A www.ietf.or |
| 334 | 3.732568473 | 192.168.105.228 | 104.20.0.85 | TCP | 74 | 58766 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1 |
| 335 | 3.756859625 | 192.168.105.228 | 104.20.0.85 | TCP | 74 | 58768 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1 |
| 366 | 3.811572575 | 104.20.0.85 | 192.168.105.228 | TCP | 66 | 80 → 58766 [SYN, ACK] Seq=0 Ack=1 Win=29200 |
| 367 | 3.811647188 | 192.168.105.228 | 104.20.0.85 | TCP | 54 | 58766 → 80 [ACK] Seq=1 Ack=1 Win=64256 Len=0 |
| 368 | 3.811934330 | 192.168.105.228 | 104.20.0.85 | HTTP | 469 | GET / HTTP/1.1 |
| 370 | 3.835082852 | 104.20.0.85 | 192.168.105.228 | TCP | 66 | 80 → 58768 [SYN, ACK] Seq=0 Ack=1 Win=29200 |
| 371 | 3.835175355 | 192.168.105.228 | 104.20.0.85 | TCP | 54 | 58768 → 80 [ACK] Seq=1 Ack=1 Win=64256 Len=0 |
| 373 | 3.889920292 | 104.20.0.85 | 192.168.105.228 | TCP | 60 | 80 → 58766 [ACK] Seq=1 Ack=416 Win=30720 Len |
| 386 | 4.261281296 | 104.20.0.85 | 192.168.105.228 | TCP | 705 | 80 → 58766 [PSH, ACK] Seq=1 Ack=416 Win=3072 |
| 387 | 4.261326785 | 192.168.105.228 | 104.20.0.85 | TCP | 54 | 58766 → 80 [ACK] Seq=416 Ack=652 Win=64128 L |
| 388 | 4.261334787 | 104.20.0.85 | 192.168.105.228 | HTTP | 60 | HTTP/1.1 302 Found (text/html) |
| 389 | 4.261343622 | 192.168.105.228 | 104.20.0.85 | TCP | 54 | 58766 → 80 [ACK] Seq=416 Ack=657 Win=64128 L |
| 390 | 4.271730513 | 192.168.105.228 | 104.20.0.85 | TCP | 74 | 39816 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS= |
| 391 | 4.350150528 | 104.20.0.85 | 192.168.105.228 | TCP | 66 | 443 → 39816 [SYN, ACK] Seq=0 Ack=1 Win=29200 |
| 392 | 4.350206322 | 192.168.105.228 | 104.20.0.85 | TCP | 54 | 39816 → 443 [ACK] Seq=1 Ack=1 Win=64256 Len= |
| 393 | 4.350792704 | 192.168.105.228 | 104.20.0.85 | TLSv1.3 | 602 | Client Hello |
| 397 | 4.428064980 | 104.20.0.85 | 192.168.105.228 | TCP | 60 | 443 → 39816 [ACK] Seq=1 Ack=549 Win=30720 Le |
| 398 | 4.434548049 | 104.20.0.85 | 192.168.105.228 | TLSv1.3 | 266 | Server Hello, Change Cipher Spec, Applicatio |
| 399 | 4.434563968 | 192.168.105.228 | 104.20.0.85 | TCP | 54 | 39816 → 443 [ACK] Seq=549 Ack=213 Win=64128 |

```
▸ Frame 324: 83 bytes on wire (664 bits), 83 bytes captured (664 bits) on interface 0
▸ Ethernet II, Src: QuantaCo_46:de:fb (a8:1e:84:46:de:fb), Dst: HewlettP_a9:8b:f2 (5c:8a:38:a9:8b:f2)
▸ Internet Protocol Version 4, Src: 192.168.105.228, Dst: 192.168.36.53
▾ User Datagram Protocol, Src Port: 45024, Dst Port: 53
    Source Port: 45024
    Destination Port: 53
    Length: 49
    Checksum: 0x5085 [unverified]
    [Checksum Status: Unverified]
    [Stream index: 57]
  ▸ [Timestamps]
▸ Domain Name System (query)
```

**5. What is the destination port for the DNS query message? What is the source port of DNS response message?**

The destination port is 53 and the source port is 45024

```
▾ User Datagram Protocol, Src Port: 45024, Dst Port: 53
    Source Port: 45024
    Destination Port: 53
    Length: 49
    Checksum: 0x5085 [unverified]
    [Checksum Status: Unverified]
    [Stream index: 57]
  ▸ [Timestamps]
▸ Domain Name System (query)
```

**6. To what IP address is the DNS query message sent? Use ipconfig to determine the IP address of your local DNS server. Are these two IP addresses the same?**

The DNS query was sent to IP address 192.168.36.53. Yes it is the same IP address as that of my local DNS server.

| | | | | | |
|---|---|---|---|---|---|
| 324 | 3.506862489 | 192.168.105.228 | 192.168.36.53 | DNS | 83 Standard query 0xa43e A www.ietf.org OPT |

```
→ ~ nmcli --fields ip4.dns con show 'Wired connection 1'
IP4.DNS[1]:                                192.168.36.53
IP4.DNS[2]:                                192.168.35.52
```

**7. Examine the DNS query message. What "Type" of DNS query is it? Does the query message contain any "answers"?**

The query message was a type A query, but the message did not contain any "answers"

```
▾ Domain Name System (query)
    Transaction ID: 0xa43e
  ▸ Flags: 0x0100 Standard query
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 1
  ▾ Queries
    ▸ www.ietf.org: type A, class IN
```

**8. Examine the DNS response message. How many "answers" are provided? What do each of these answers contain?**

The response message contained three answer. Each answer contains a site and some site details.

```
▾ Queries
  ▸ www.ietf.org: type A, class IN
▾ Answers
  ▾ www.ietf.org: type CNAME, class IN, cname www.ietf.org.cdn.cloudflare.net
      Name: www.ietf.org
      Type: CNAME (Canonical NAME for an alias) (5)
      Class: IN (0x0001)
      Time to live: 1800
      Data length: 33
      CNAME: www.ietf.org.cdn.cloudflare.net
  ▾ www.ietf.org.cdn.cloudflare.net: type A, class IN, addr 104.20.1.85
      Name: www.ietf.org.cdn.cloudflare.net
      Type: A (Host Address) (1)
      Class: IN (0x0001)
      Time to live: 300
      Data length: 4
      Address: 104.20.1.85
  ▾ www.ietf.org.cdn.cloudflare.net: type A, class IN, addr 104.20.0.85
      Name: www.ietf.org.cdn.cloudflare.net
      Type: A (Host Address) (1)
      Class: IN (0x0001)
      Time to live: 300
      Data length: 4
      Address: 104.20.0.85
```

**9. Consider the subsequent TCP SYN packet sent by your host. Does the destination IP address of the SYN packet correspond to any of the IP addresses provided in the DNS response message?**

The destination of the SYN packet is 104.20.0.85, the same address that was provided in the DNS response message as the type “A” address of the webpage.

| | | | | | |
|---|---|---|---|---|---|
| 333 | 3.731813102 | 192.168.36.53 | 192.168.105.228 | DNS | 160 Standard query response 0xa43e A www.ietf.org |
| 334 | 3.732568473 | 192.168.105.228 | 104.20.0.85 | TCP | 74 58766 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=14 |
| 335 | 3.756859625 | 192.168.105.228 | 104.20.0.85 | TCP | 74 58768 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=14 |
| 366 | 3.811572575 | 104.20.0.85 | 192.168.105.228 | TCP | 66 80 → 58766 [SYN, ACK] Seq=0 Ack=1 Win=29200 Le |

**10. This web page contains images. Before retrieving each image, does your host issue new DNS queries?**

No.

**11. What is the destination port for the DNS query message? What is the source port of DNS response message?**

Destination Port: 53 and Source Port: 56599

```
▾ User Datagram Protocol, Src Port: 56599, Dst Port: 53
    Source Port: 56599
    Destination Port: 53
    Length: 62
    Checksum: 0xb7c5 [unverified]
    [Checksum Status: Unverified]
    [Stream index: 13]
  ▸ [Timestamps]
▾ Domain Name System (query)
```

**12. To what IP address is the DNS query message sent? Is this the IP address of your default local DNS server?**

The DNS query message is sent to IP address 192.168.36.52, the same address as my default local DNS server.

| | | | | | | |
|---|---|---|---|---|---|---|
| 51 | 2.832780186 | 192.168.105.228 | 192.168.35.52 | DNS | 94 | Standard query 0x41b9 A |
| 52 | 2.890866804 | 192.168.35.52 | 192.168.105.228 | DNS | 146 | Standard query response |
| 53 | 2.892394018 | 192.168.105.228 | 192.168.35.52 | DNS | 96 | Standard query 0x933a AA |
| 54 | 2.919461996 | 192.168.35.52 | 192.168.105.228 | DNS | 152 | Standard query response |

```
→ ~ nmcli --fields ip4.dns con show 'Wired connection 1'
IP4.DNS[1]:                                192.168.36.53
IP4.DNS[2]:                                192.168.35.52
```

**13. Examine the DNS query message. What "Type" of DNS query is it? Does the query message contain any "answers"?**

The DNS query message is a type "A" query, containing only one question and not containing any answers.

```
▾ Domain Name System (query)
    Transaction ID: 0x41b9
  ▸ Flags: 0x0100 Standard query
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 1
  ▾ Queries
    ▾ www.mit.edu.edgekey.net: type A, class IN
        Name: www.mit.edu.edgekey.net
        [Name Length: 23]
        [Label Count: 5]
        Type: A (Host Address) (1)
        Class: IN (0x0001)
  ▾ Additional records
```

**14. Examine the DNS response message. How many “answers” are provided? What do each of these answers contain?**

The response message contains one answer to the aforementioned query which is the type “A” address of http://www.mit.edu or 104.114.106.91.

```
▸ Internet Protocol Version 4, Src: 192.168.35.52, Dst: 192.168.105.228
▸ User Datagram Protocol, Src Port: 53, Dst Port: 50868
▾ Domain Name System (response)
    Transaction ID: 0x41b9
  ▸ Flags: 0x8180 Standard query response, No error
    Questions: 1
    Answer RRs: 2
    Authority RRs: 0
    Additional RRs: 1
  ▾ Queries
    ▾ www.mit.edu.edgekey.net: type A, class IN
        Name: www.mit.edu.edgekey.net
        [Name Length: 23]
        [Label Count: 5]
        Type: A (Host Address) (1)
        Class: IN (0x0001)
  ▾ Answers
    ▾ www.mit.edu.edgekey.net: type CNAME, class IN, cname e9566.dscb.akamaiedge.net
        Name: www.mit.edu.edgekey.net
        Type: CNAME (Canonical NAME for an alias) (5)
        Class: IN (0x0001)
        Time to live: 60
        Data length: 24
        CNAME: e9566.dscb.akamaiedge.net
    ▾ e9566.dscb.akamaiedge.net: type A, class IN, addr 104.114.106.91
        Name: e9566.dscb.akamaiedge.net
        Type: A (Host Address) (1)
        Class: IN (0x0001)
        Time to live: 20
        Data length: 4
        Address: 104.114.106.91
  ▾ Additional records
```

**15. Provide a screenshot.**

**16. To what IP address is the DNS query message sent? Is this the IP address of you default local DNS server?**

The query is sent to 192.168.36.52, the same IP address as that of my default local DNS server.

| | | | | | |
|---|---|---|---|---|---|
| 89 | 2.465441789 | 192.168.105.228 | 192.168.35.52 | DNS | 94 Standard query 0x710c NS www.mit.edu. |
| 92 | 2.526219149 | 192.168.35.52 | 192.168.105.228 | DNS | 194 Standard query response 0x710c NS www |
| 93 | 2.526760327 | 192.168.105.228 | 192.168.35.52 | DNS | 96 Standard query 0x05f3 NS e9566.dscb.a |
| 94 | 2.528213970 | 192.168.35.52 | 192.168.105.228 | DNS | 160 Standard query response 0x05f3 NS e95 |

**17. Examine the DNS query message. What "Type" of DNS query is it? Does the query message contain any "answers".**

The DNS query is a type "NS" message including one question. The query message did not contain any answers.

```
▾ Domain Name System (query)
    Transaction ID: 0x710c
  ▸ Flags: 0x0100 Standard query
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 1
  ▾ Queries
    ▾ www.mit.edu.edgekey.net: type NS, class IN
        Name: www.mit.edu.edgekey.net
        [Name Length: 23]
        [Label Count: 5]
        Type: NS (authoritative Name Server) (2)
        Class: IN (0x0001)
  ▾ Additional records
```

**18. Examine the DNS response message. What MIT nameservers does the response message provide? Does this response message also provide the IP addresses of the MIT namesers?**

**19. Provide a screenshot.**

**20. To what IP address is the DNS query message sent? Is this the IP address of your default local DNS server? If not, what does the IP address correspond to?**

**21. Examine the DNS query message. What "Type" of DNS query is it? Does the query message contain any "answers"?**

**22. Examine the DNS response message. How many "answers" are provided? What does each of these answers contain?**

**23. Provide a screenshot.**

```
➜ ~ nslookup www.aiit.or.kr bitsy.mit.edu
;; connection timed out; no servers could be reached
```