



National University of Sciences and Technology (NUST)
School of Electrical Engineering and Computer Science

Department of Electrical Engineering

Faculty Member: Sir Hassan khaliq

Dated: 4/26/2023

Semester: 6th

Section: C

EE-357 Computer and Communication Networks

Wireshark – TCP (Transmission Control Protocol)

| Name | Reg. No | PLO5/ CLO3 | | PLO5/ CLO3 | PLO5/ CLO3 | PLO5/ CLO3 |
|--------------------------|---------|--|---|---------------------------------|------------------------------------|---|
| | | Viva / Quiz / Lab Performance 5 Marks | Analysis of data in Lab Report 5 Marks | Modern Tool Usage 5 Marks | Ethics and Safety 5 Marks | Individual and Team Work 5 Marks |
| Imran Haider | 332569 | | | | | |
| Muhammad Ahmed Mohsin | 333060 | | | | | |
| Amina Bashir | 343489 | | | | | |



1 TABLE OF CONTENTS

| | | |
|---|---|----|
| 2 | Objective of this lab:..... | 3 |
| 3 | Instructions: | 3 |
| 4 | Background:..... | 3 |
| 5 | A first look at the captured trace..... | 5 |
| 6 | TCP Basics..... | 8 |
| 7 | Conclusion | 11 |



Wireshark – TCP (Transmission Control Protocol)

2 OBJECTIVE OF THIS LAB:

In this lab, we will explore several aspects of the Transmission Control Protocol.

3 INSTRUCTIONS:

- Read carefully before starting the lab.
- These exercises are to be done individually.
- You are supposed to provide the answers to the in-line questions in this document and upload the completed document to your course's LMS site.
- **For all questions, you must not only answer the question, but also supply all necessary information regarding how you arrived at the answer (e.g., use screenshots/ accompanying text, etc.) Use red font color to distinguish your replies from the rest of the text.**
- Avoid plagiarism by copying from the Internet or from your peers. You may refer to source/ text but you must paraphrase the original work.

4 BACKGROUND:

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

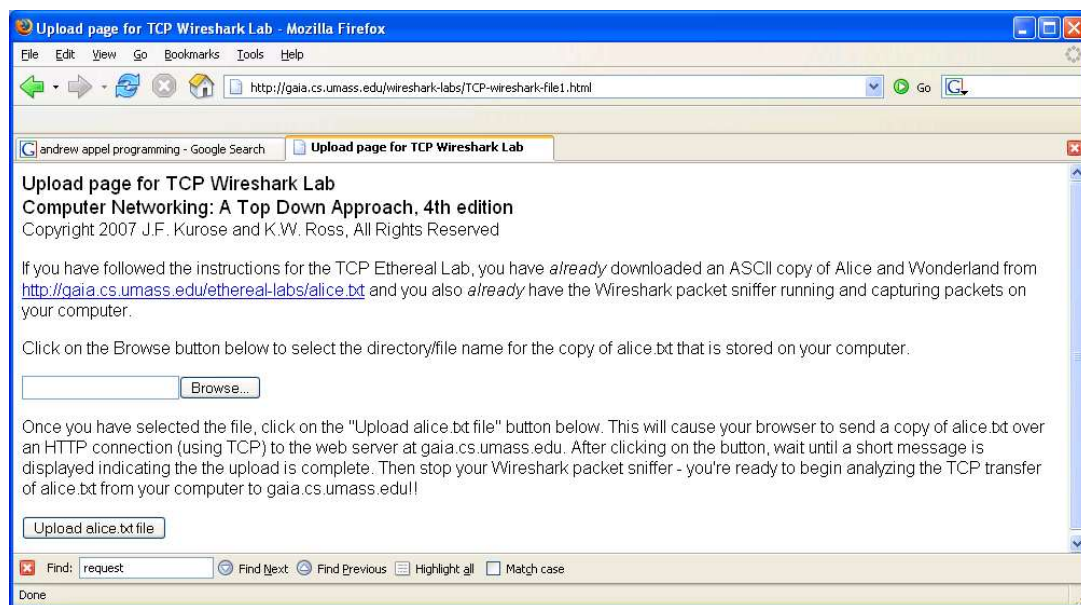
- Capturing a bulk TCP transfer from your computer to a remote server
- Before beginning our exploration of TCP, we will need to use Wireshark to obtain a packet trace of the TCP transfer of a file from your computer to a remote server. You will do so by accessing a Web page that will allow you



National University of Sciences and Technology (NUST) School of Electrical Engineering and Computer Science

to enter the name of a file stored on your computer (which contains an ASCII text), and then transfer the file to a Web server using the HTTP POST method. We are using the POST method rather than the GET method as we would like to transfer a large amount of data from your computer to another computer. Of course, we will be running Wireshark during this time to obtain the trace of the TCP segments sent and received from your computer.

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

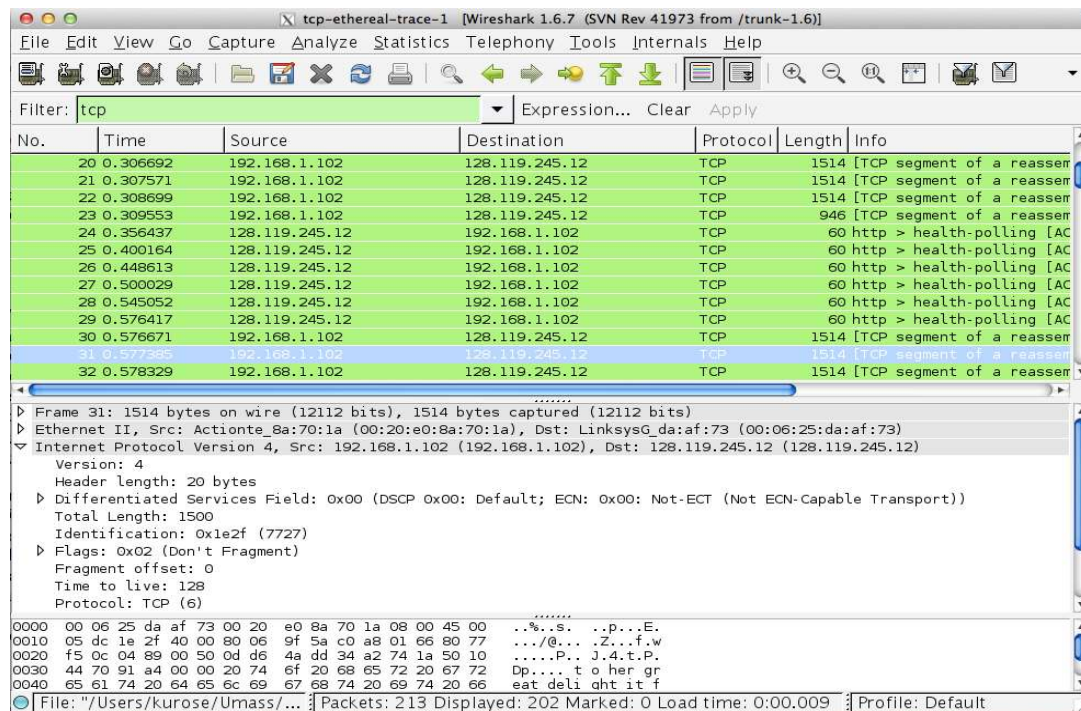


- Use the *Browse* button in this form to enter the name of the text file (full path name) on your computer (or do so manually). Do not yet press the Upload button.
- Now, startup Wireshark and begin packet capture (*Capture-→Start*)
- Returning to your browser, press the Upload button to upload the file to the server. Once the file has been uploaded, a short congratulations message will be displayed in your browser window.



National University of Sciences and Technology (NUST) School of Electrical Engineering and Computer Science

- Stop Wireshark packet capture. Your Wireshark window should look similar to the window shown below.



5 A FIRST LOOK AT THE CAPTURED TRACE

Before analyzing the behavior of the TCP connection in detail, you should take a high level view of the trace.

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

What you should see is series of TCP and HTTP messages between your computer and gaia.cs.umass.edu. You should see the initial three-way handshake



National University of Sciences and Technology (NUST) School of Electrical Engineering and Computer Science

containing a SYN message. You should see an HTTP POST message. Depending on the version of Wireshark you are using, you might see a series of “HTTP Continuation” messages being sent from your computer to `gaia.cs.umass.edu`. There is actually no such thing as an HTTP Continuation message – this is Wireshark’s way of indicating that there are multiple TCP segments being used to carry a single HTTP message. In more recent versions of Wireshark, You will see “[TCP segment of a reassembled PDU]” in the Info column of the Wireshark display to indicate that this TCP segment contained data that belonged to an upper layer protocol message (in our case here, HTTP). You should also see TCP ACK segments being returned from **`gaia.cs.umass.edu`** to your computer.

Answer the following questions, by opening the Wireshark captured packet file ***tcp-ethereal-trace-1***. Whenever possible, when answering a question, you should hand in a screenshot of the packet(s) within the trace that you used to answer the question asked. Annotate the screenshot to explain your answer. To print a packet, use *File->Print*, choose *selected packet only*, choose *Packet summary line*, and select the minimum amount of packet detail that you need to answer the question.

1. **What is the IP address and TCP port number used by the client computer (source) that is transferring the file to `gaia.cs.umass.edu`? To answer this question, it’s probably easiest to select an HTTP message and explore the details of the TCP packet used to carry this HTTP message, using the “details of the selected packet header window” (refer to Figure 2 in the “Getting Started with Wireshark” Lab if you’re uncertain about the Wireshark windows.**



National University of Sciences and Technology (NUST)

School of Electrical Engineering and Computer Science

Wireshark · Packet 3493 · Wi-Fi

> Frame 3493: 17574 bytes on wire (140592 bits), 17574 bytes captured (140592 bits) on interface \Device\NPF_{02BFF4AF-63}

▼ Ethernet II, Src: IntelCor_1e:c8:46 (dc:41:a9:1e:c8:46), Dst: HuaweiTe_40:6f:96 (28:a6:db:40:6f:96)

> Destination: HuaweiTe_40:6f:96 (28:a6:db:40:6f:96)

> Source: IntelCor_1e:c8:46 (dc:41:a9:1e:c8:46)

Type: IPv4 (0x0800)

> Internet Protocol Version 4, Src: 10.7.88.128, Dst: 128.119.245.12

▼ Transmission Control Protocol, Src Port: 63364, Dst Port: 80, Seq: 79548, Ack: 1, Len: 17520

Source Port: 63364

Destination Port: 80

[Stream index: 7]

[Conversation completeness: Incomplete, DATA (15)]

[TCP Segment Len: 17520]

Sequence Number: 79548 (relative sequence number)

Sequence Number (raw): 876050851

0000 28 a6 db 40 6f 96 dc 41 a9 1e c8 46 08 00 45 00 (..@.-A...F..E-
0010 00 00 03 2e 40 00 80 06 00 00 0a 07 58 80 80 77 ...@...-X..w
0020 f5 0c f7 84 00 50 34 37 79 a3 27 d3 d3 ce 50 18P47 y'...P-
0030 02 01 d8 11 00 00 20 20 60 54 77 6f 20 64 61 79`Two day
0040 73 20 77 72 6f 6e 67 21 27 20 73 69 67 68 65 64 s wrong! ' sighed
0050 20 74 68 65 20 48 61 74 74 65 72 2e 20 20 60 49 the Hat ter. `I
0060 20 74 6f 6c 64 20 79 6f 75 20 62 75 74 74 65 72 told yo u butter
0070 0d 0a 77 6f 75 6c 64 6e 27 74 20 73 75 69 74 20 ..wouldn 't suit
0080 74 68 65 20 77 6f 72 6b 73 21 27 20 68 65 20 61 the work s!' he a
0090 64 64 65 64 20 6c 6f 6f 6b 69 6e 67 20 61 6e 67 dded loo king ang
00a0 72 69 6c 79 20 61 74 20 74 68 65 20 4d 61 72 63 rily at the Marc
00b0 68 0d 0a 48 61 72 65 2e 0d 0a 0d 0a 20 20 60 49 h..Hare.`I
00c0 74 20 77 61 73 20 74 68 65 20 42 45 53 54 20 62 t was th e BEST b
00d0 75 74 74 65 72 2c 27 20 74 68 65 20 4d 61 72 63 utter,' the Marc
00e0 68 20 48 61 72 65 20 6d 65 65 6b 6c 79 20 72 65 h Hare m eekly re
00f0 70 6c 69 65 64 2e 0d 0a 0d 0a 20 20 60 59 65 73 plied... ..`Yes
0100 2c 20 62 75 74 20 73 6f 6d 65 20 63 72 75 6d 62 , but so me crumb

☒ Show packet bytes

| No. | Time | Delta | Source | Destination | Protocol | Length | Info |
|------|----------------------------|----------|----------------|----------------|----------|--------|--|
| 3492 | 2023-04-26 14:28:18.274857 | 0.000000 | 128.119.245.12 | 10.7.88.128 | TCP | 56 | 80 → 63364 [ACK] Seq=1 Ack=41588 Win=112384 Len=0 |
| 3493 | 2023-04-26 14:28:18.274890 | 0.000033 | 10.7.88.128 | 128.119.245.12 | TCP | 17574 | 63364 → 80 [PSH, ACK] Seq=79548 Ack=1 Win=131328 Len=17520 [TCP segment of |
| 3580 | 2023-04-26 14:28:18.752846 | 0.477956 | 128.119.245.12 | 10.7.88.128 | TCP | 56 | 80 → 63364 [ACK] Seq=1 Ack=48888 Win=126976 Len=0 |
| 3581 | 2023-04-26 14:28:18.752846 | 0.000000 | 128.119.245.12 | 10.7.88.128 | TCP | 56 | 80 → 63364 [ACK] Seq=1 Ack=56188 Win=141696 Len=0 |
| 3582 | 2023-04-26 14:28:18.752846 | 0.000000 | 128.119.245.12 | 10.7.88.128 | TCP | 56 | 80 → 63364 [ACK] Seq=1 Ack=62028 Win=151344 Len=0 |
| 3583 | 2023-04-26 14:28:18.752846 | 0.000000 | 128.119.245.12 | 10.7.88.128 | TCP | 56 | 80 → 63364 [ACK] Seq=1 Ack=66408 Win=162048 Len=0 |
| 3584 | 2023-04-26 14:28:18.752846 | 0.000000 | 128.119.245.12 | 10.7.88.128 | TCP | 56 | 80 → 63364 [ACK] Seq=1 Ack=69328 Win=167936 Len=0 |
| 3585 | 2023-04-26 14:28:18.752846 | 0.000000 | 128.119.245.12 | 10.7.88.128 | TCP | 56 | 80 → 63364 [ACK] Seq=1 Ack=73708 Win=176640 Len=0 |
| 3586 | 2023-04-26 14:28:18.752846 | 0.000000 | 128.119.245.12 | 10.7.88.128 | TCP | 56 | 80 → 63364 [ACK] Seq=1 Ack=75168 Win=179584 Len=0 |
| 3587 | 2023-04-26 14:28:18.752846 | 0.000000 | 128.119.245.12 | 10.7.88.128 | TCP | 56 | 80 → 63364 [ACK] Seq=1 Ack=78088 Win=182528 Len=0 |
| 3588 | 2023-04-26 14:28:18.752846 | 0.000000 | 128.119.245.12 | 10.7.88.128 | TCP | 56 | 80 → 63364 [ACK] Seq=1 Ack=79548 Win=183296 Len=0 |
| 3589 | 2023-04-26 14:28:18.752846 | 0.000000 | 128.119.245.12 | 10.7.88.128 | TCP | 56 | 80 → 63364 [ACK] Seq=1 Ack=86848 Win=179584 Len=0 |
| 3590 | 2023-04-26 14:28:18.752846 | 0.000000 | 128.119.245.12 | 10.7.88.128 | TCP | 56 | 80 → 63364 [ACK] Seq=1 Ack=92688 Win=180608 Len=0 |
| 3591 | 2023-04-26 14:28:18.752846 | 0.000000 | 128.119.245.12 | 10.7.88.128 | TCP | 56 | 80 → 63364 [ACK] Seq=1 Ack=94148 Win=183296 Len=0 |
| 3592 | 2023-04-26 14:28:18.752846 | 0.000000 | 128.119.245.12 | 10.7.88.128 | TCP | 56 | 80 → 63364 [ACK] Seq=1 Ack=97068 Win=182528 Len=0 |

According to above figure, the client computer (source)'s IP address is 10.7.88.128 and the TPC port number is 63364.

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

According to above figure, the IP address of gaia.cs.umass.edu is 128.119.245.12 and the TCP port number is 80.

Now, create your own trace and answer the following questions:



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

The screenshot shows a Wireshark packet capture. The packet list pane displays several packets, with packet 1257 highlighted. The packet details pane shows the structure of the selected packet, which is a TCP segment. The source IP address is 192.82.114.26 and the destination IP address is 10.7.88.128. The source port is 443 and the destination port is 443. The sequence number is 1099640792 and the acknowledgment number is 482661447. The window size is 78. The flags are SYN (0x00000002).

The TCP port number is 443 and the source IP address is 192.82.114.26.

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

6 TCP BASICS

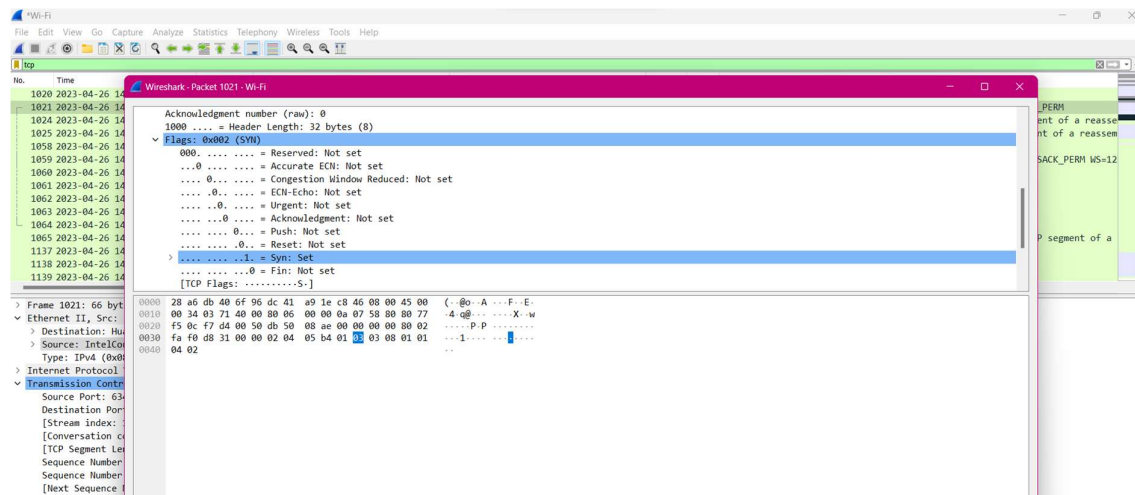
Answer the following questions for the TCP segments:

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



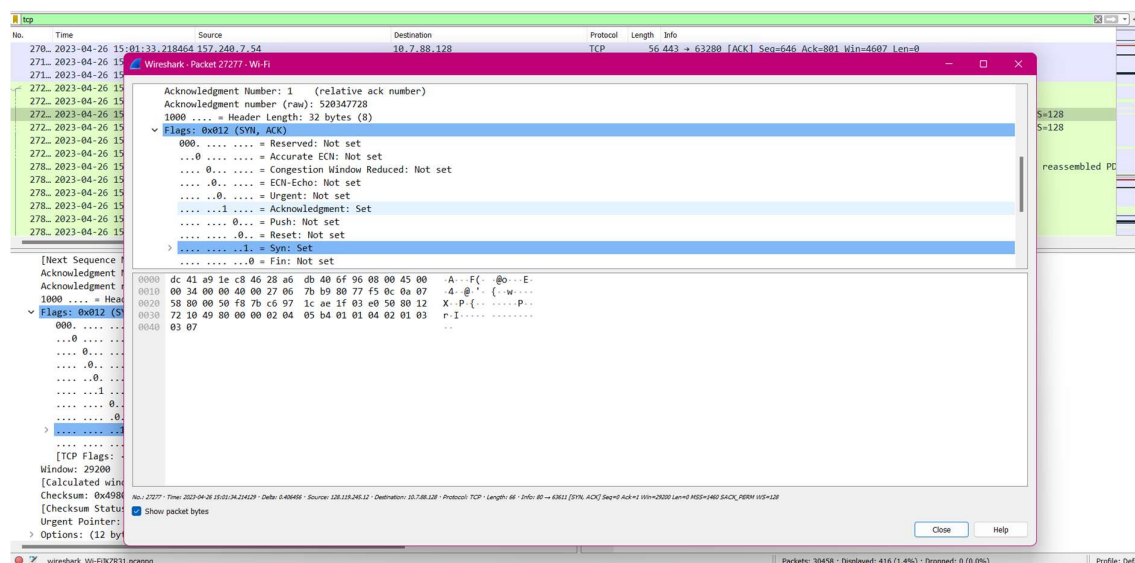
National University of Sciences and Technology (NUST)

School of Electrical Engineering and Computer Science



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 above figure, in the Flags section, the Syn flag is set to 1 which indicates that this segment is a SYN segment.

5. What is the sequence number of the SYNACK segment sent by gaia.cs.umass.edu to the client computer in reply to the SYN? What is the value of the Acknowledgement field in the SYNACK segment? How did gaia.cs.umass.edu determine that value? What is it in the segment that identifies the segment as a SYNACK segment?





According to the above figure, the sequence number of the SYNACK 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 SYNACK segment is 1. The value of the ACKnowledgement field in the SYNACK segment is determined by the server gaia.cs.umass.edu. The server adds 1 to the initial sequence number of SYN segment from the client computer. For this case, the initial sequence number of SYN segment from the client computer is 0, thus the value of the ACKnowledgement field in the SYNACK segment is 1. A segment will be identified as a SYNACK segment if both SYN flag and Acknowledgement in the segment are set to 1.

6. What is the sequence number of the TCP segment containing the HTTP POST command? Note that in order to find the POST command, you will need to dig into the packet content field at the bottom of the Wireshark window, looking for a segment with a "POST" within its DATA field.

The image shows a Wireshark packet capture of a TCP connection. The packet list on the left shows a packet with sequence number 705, which is highlighted in red. The packet details pane on the right shows the TCP segment's fields, including the sequence number (705), acknowledgment number (1), and flags (PSH, ACK). The packet bytes pane at the bottom shows the raw data of the segment, which is a POST command.

Sequence Number: 705 (relative sequence number)
Sequence Number (raw): 520348432
[Next Sequence Number: 918 (relative sequence number)]
Acknowledgment Number: 1 (relative ack number)
Acknowledgment number (raw): 3331792047
0101 = Header Length: 20 bytes (5)
Flags: 0x018 (PSH, ACK)
0000 = Reserved: Not set
...0 = Accurate ECH: Not set
...0 = Congestion Window Reduced: Not set
...0 = ECN-Echo: Not set

0000 28 a6 db 40 6f 96 dc 41 a9 1e c8 46 08 00 45 00 (..@..A...F..E..
0010 00 fd 04 74 40 00 00 05 00 0a 07 58 80 80 77 ...t@...X..w..
0020 f5 0c f8 7b 00 50 1f 03 e3 10 c6 97 1c af 50 18 ...{.P... ..P..
0030 02 01 d8 fa 00 00 2d 2d 2d 2d 2d 57 65 62 4bWebK

00c0 74 2f 70 6c 61 69 6e 0d 0a 0d 0a 4d 79 20 6e 61 t/plain-...My na
00d0 6d 65 20 69 73 20 41 68 6d 65 64 0d 0a 0d 0a 2d me is Ah med----
00e0 2d 2d 2d 2d 2d 57 65 62 4b 69 74 46 6f 72 6d 42 -----Web KitFormB
00f0 6f 75 6e 64 61 72 79 31 49 79 6f 48 54 62 64 73 oundary1 lyoHTbds
0100 79 70 39 4e 48 68 46 2d 2d 0d 0a yp9Wihf- ...

Frame (267 bytes) Reassembled TCP (917 bytes) Packets: 30458 · Displayed: 416 (1.4%) · Dropped: 0 (0.0%)

Hence, we can observe that it is 705th in sequence number.



7 CONCLUSION

In this lab report, I performed tracing of TCP elements and then worked inside the segments. I used the tcpdump command to capture TCP traffic and then used the Wireshark network analyzer to view the captured traffic. I was able to identify the different TCP elements in the captured traffic, including the source and destination IP addresses, the source and destination port numbers, the sequence numbers, and the acknowledgment numbers. I was also able to view the data that was being transmitted in the TCP segments.

I then worked inside the TCP segments to view the different fields that are present in each segment. I was able to identify the different fields, such as the source and destination IP addresses, the source and destination port numbers, the sequence numbers, the acknowledgment numbers, the window size, the checksum, and the urgent pointer. I was also able to view the data that was being transmitted in the TCP segments.

This lab report gave me a good understanding of how TCP works and how to trace TCP traffic. I was able to identify the different TCP elements in the captured traffic and view the data that was being transmitted in the TCP segments. This knowledge will be useful for me in my future studies and work.