Emre Can Sen - 21902516
Mert Pehlivan – 21703139

CS421 - PROGRAMMING ASSIGNMENT 1

1. INTRODUCTION

In this project assignment, a program called TextEditor is designed, where the client side of the collaborative text editing program is implemented. The programming language used is Python, and OS is Windows 11. The program does not include any GUI, however it utilizes console window to choose from 5 options. In the report, the work done for these different operations' implementation is discussed. Additionally, the protocols used are tracked from the Wireshark as the program is executed.

2. WORK AND RESULTS

The work done can be separated into two parts. In the 1st part, the clients are connected from terminals via TCP connection and they are connected from/to localhost. Both clients and the server are connected through localhosts, therefore they are on the same machine however TCP connections are not common. In the 2nd part, the analysis done with Wireshark is provided.

2.1 Work Done on Terminal

The TCP connection is established by running the server program as a first step. The command is given as

```
python Server.py localhost 60000
```

and then the client program is run through a similar command given as

```
python TextEditor.py localhost 60000
```

After these commands, a menu appears as follows.

```
Press 1 to Enter a Username:
 Press 2 to Enter a Password:
 Press 3 to Enable Write Mode:
 Press 4 to  Enable Append Mode:
 Press 5 to Exit:
```

Figure 1: Menu Screen

Since an authentication is required for the clients, unless 1 and 2 is pressed sequentially the program throws an exception and it returns an error on the command window. In other words, it is necessary to enter username first and then the password correctly to be able write/append data to text file. When the entries are correctly received the program returns "OK" message. The below figure shows a sample "OK" message returned after entering the correct password.

```
2
Type your password:cs421f2022
Received b'OK \r\n'
```

Figure 2: "OK" Message after Entering "cs421f2022"

From now on, users can either write content into the text file with specifying line number or they can append new content at the end of the file. Alternatively, they do not change the file and exit from the program. For the write mode case, users enter 3 and specify the line number which is an integer. Then they enter data and when the server receives the message the file is updated. Using the below codes, the text file's initial row appears as follows.

```
3
Type your line to write:0
Type your text to write:Hello World
Received b'OK 1\r\n'
```

Figure 3: Updating Text File with Write Mode

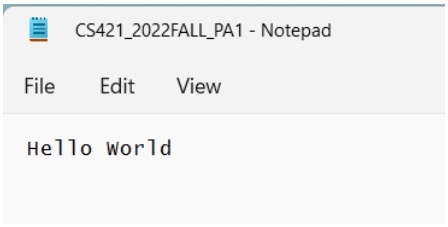

Figure 4: Updated Text File for Above Operation

Now, a second user (from a different terminal) uses the program and enters data to append some text to the file. Entering the following inputs, the text file is updated as follows.

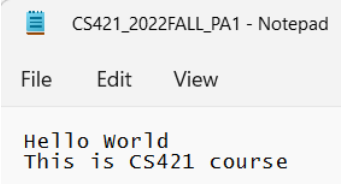

Figure 5: Updated Text File for Above Operation

Now for the testing purposes, the first user changes the second line (line number is 1) and writes “This is a test program”. As similar to above examples, the operation and the output is provided next.

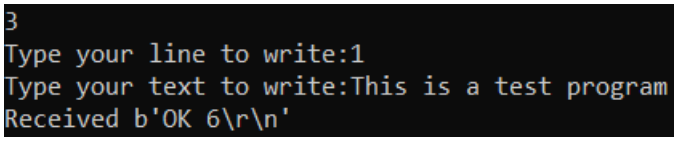

Figure 6: Updating Text File with Write Mode #2

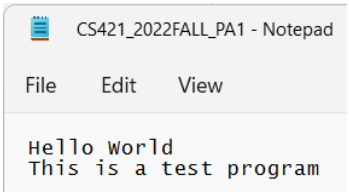

Figure 7: Updated Text File for Above Operation

When the users enter an invalid number a warning message indicating an invalid number entry is returned, and they are redirected to menu.

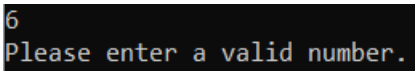

Figure 8: Wrong Number Entry

When users try to enter 1 or 2 and enters wrong data, program returns the following message indicating the username/password data is already received.

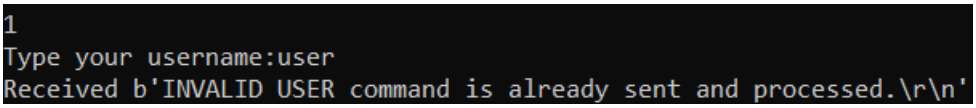

Figure 9: Message after Re-entering Username Wrong

After modifying the text file, users exit the program and the connection is terminated.

Here is also the client and server console prompts at the same time.

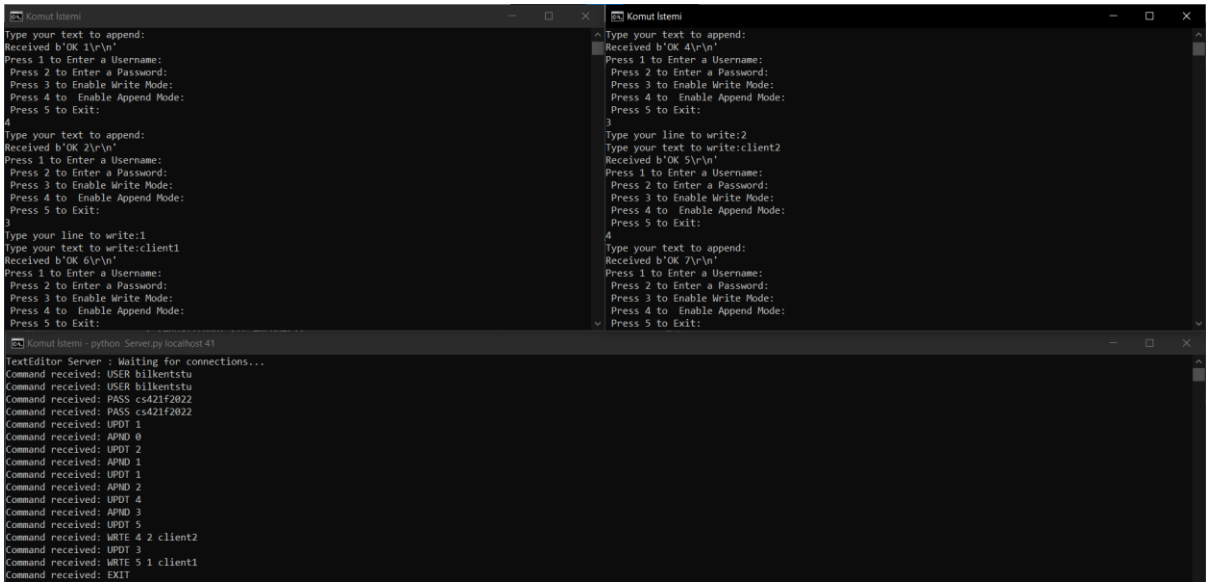

Figure 10: Server, Client 1 and Client 2 Console Windows

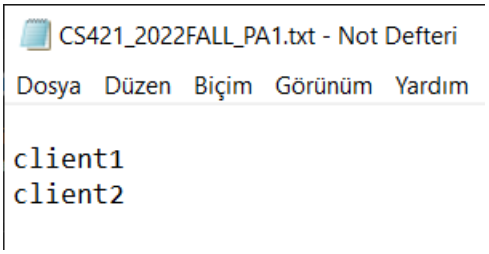

Figure 11: Multiple Client Output

2.2 Wireshark Analysis

In this part of the report, the protocols are tracked as the user uses the program from the Wireshark. The operations done in the command prompt are very similar to previous part, therefore here the only Wireshark results are discussed. Now, assume initially the text file is modified as follows.

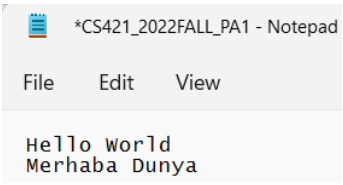

Figure 12: Initial Text File

This appears in the Wireshark as follows.

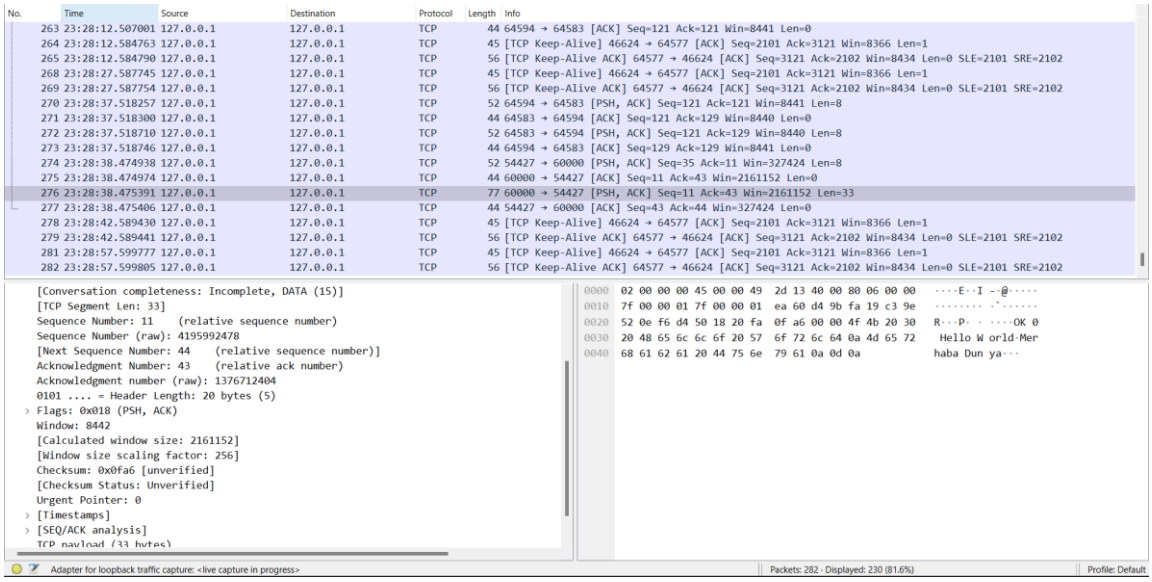

Figure 13: Initial Text File on Wireshark

As a starting point, both the server and client programs are run consecutively and the connection is established. After this step, as usual users enter username and password data to be authenticated for the text file editing. The below figure shows only the username verification TCP protocol message. The password authentication is similar.

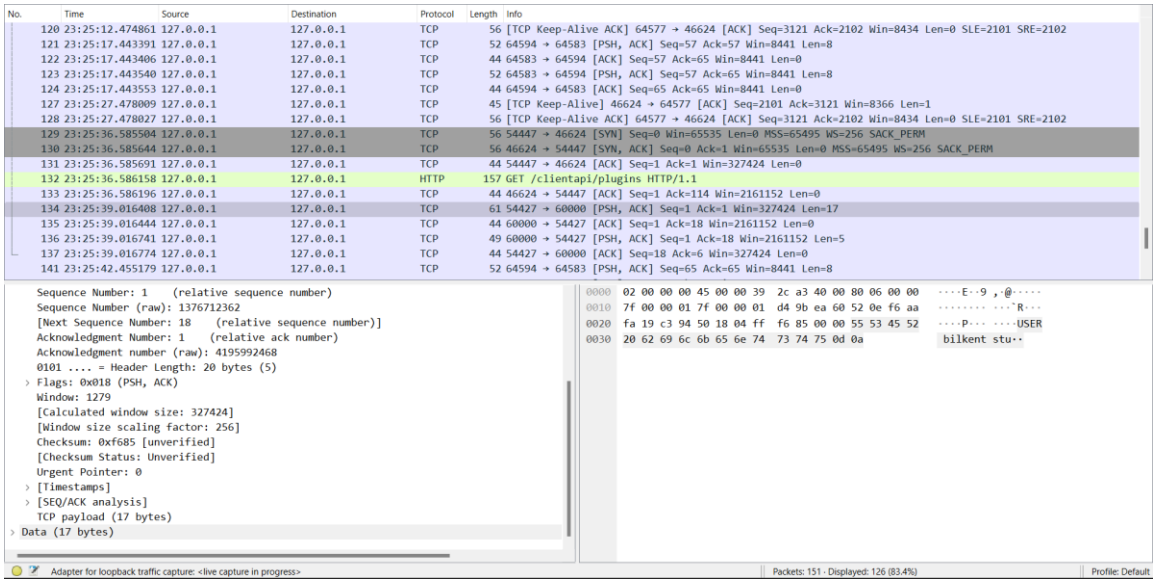

Figure 14: Correct Username Entry

The write/append mode of the program messages can be traced in Wireshark as well. Suppose a user wants to replace the initial line (line number is 0) with the message “CS421 is a good course”, through write mode. The following TCP protocol can be observed.

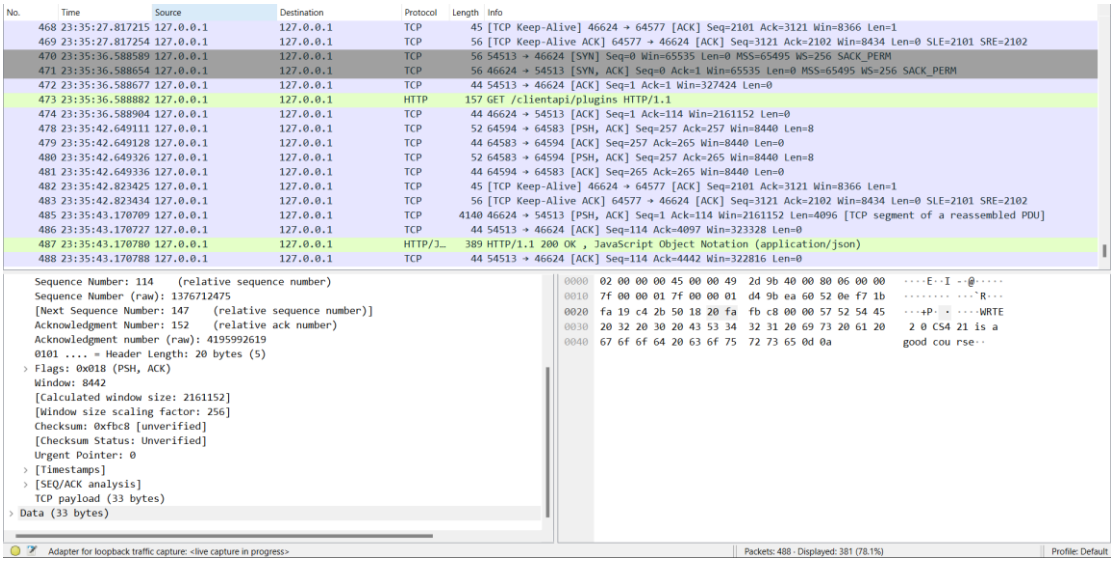

Figure 15: Write Mode Entry TCP Message

As a final demonstration, user exits the program and tries to re-enter. However, the user provides a wrong username information. In that case the following message is returned.

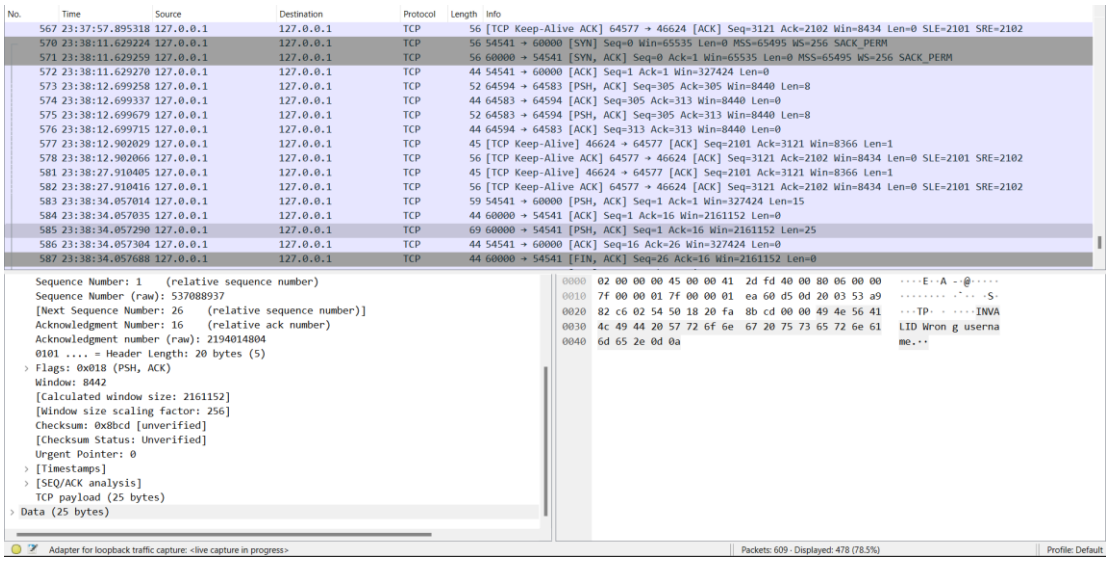

Figure 16: Wrong Username Entry TCP Message

From another point of view, it is seen that there are some properties characterised with the TCP protocols. For example, for the above TCP messages all of them have a header size of 20 bytes. They may have different TCP payload size, which indicates the length of the data. Also, it is seen from the figures that most TCP messages have a window size of 8442.

3. CONCLUSION

In this project assignment, the client side of the TextEditor program is implemented. After both server and client programs run consecutively, a menu appears to request from user to select different operations requiring correct username and password information. Unless these steps (1st and 2nd steps) are correctly done, no users can change the text file. The authenticated users can modify the text file by either writing a new content to a specified line or appending data to end of the file. Throughout the process, it is shown that 2 users can edit the file with updated content. Also, it is provided that when users try to re-enter username/password entries with wrong information, program returns an indication message that the data is already processed. For the final part, as the program is executed and different operations are tested, the TCP protocol messages appeared in Wireshark are tracked for the according operations.