

Assignment 1 Report

Name: Jordan Tam
Student ID: z5254001

Language: Python 3.9

Design Idea:

1. Basically all prompts, messages and processes are sent and done on the server. Clients are only required to send requests to the server. I got this idea from the browsers which only send requests to the domain server.
2. Everytime the server sends a message to the client, the client would send a response to the server in order to notify the server that the client has received the message successfully. And the client would wait for a new message from the server (The idea of the function: *send_message*).
3. The pattern of this program:
server(send) → client(receive) → client(send) → server(receive) → server(send) →...

Design of Server:

1. Server will bind a TCP socket with provided port number and local host IP address in the beginning
2. 2 global variable is used : *auth_data*, *devices_count* (Counting active device)
3. Using Dictionary to store all valid user in *auth_data* with key "username"(str) to store a inner dictionary with key "password"(str) and "locked"(Boolean)
4. If the username got locked because clients have try to login with a certain times, the "locked" would set in false, in order to block other new attempts from clients
5. Lock account function is done in a new created thread so that the operation of server would not get interrupted
6. The server can handle multiple connection from clients with multi-threading design
7. With command EDG, the integer would generate from 1 to target value in ascending order.

Design of Client:

1. Client will connect the server with TCP in the beginning
2. Client will bind a UDP socket by provided and unique port number and the IP from os in a second thread
3. The UDP thread will be terminated if the client log out or got blocked by sending a request to the socket
4. The response from server will trigger different operation in client:
 - a. "Start_UVF": Start connecting with other client directly by UDP (P2P), and upload the file
 - b. "REQUEST": The server is waiting for the command from user
 - c. "USERNAME_LOCKED": Notice the user that the username has blocked and terminate the UDP thread

