COMP4621 Lab 4

Name: CHAN, Chun Hin
Student ID: 20853893
Email: chchanec@connect.ust.hk

- ***Does it work if we replace "serverName" from "localhost" to "127.0.0.1" in tcp/udp client?***

For the tcp client, if we replace “serverName” from “localhost” to “127.0.0.1”, yes, it still works and the server can return the upper case version of what the client has input.

For the udp client, if we replace “serverName” from “localhost” to “127.0.0.1”, yes, it still works and the server can return the upper case version of what the client has input.

- ***What will happen if we change "serverPort" to a number less than "1024" like "22" in both tcp/udp client and server? Why?***

For tcp client and server, if we change “serverPort” to a number less than “1024” to “22”, the tcp client will receive the message from the SSH which written “SSH-2.0-OpenSSH_8.7”, which means the tcp client is successfully connect to port 22 (the SSH port) and get the response from it, while the tcp server will encounter Permission Error: [Errorno 13] Permission denied since port 22 is reserved for SSH (Secure Shell).

For udp client and server, if we change “serverPort” to a number less than “1024” to “22”, the udp client will get stuck as it is waiting to connect to port 22 (i.e. the SSH service) and wait for the response, while the udp server will encounter Permission Error: [Errorno 13] Permission denied since port 22 is reserved for SSH (Secure Shell).

- ***Change the messageSize to "1" and see what will happen? Why?***

For tcp client and tcp server, when I type “hello” in tcp client, the tcp server can only receive the first character “h” of the input message in the tcp client side, and later the tcp client can only get back “H” in response.

For udp client and udp server, when I type “hello” in udp client, the udp server can only receive the first character “h” of the input message in the udp client side, and later the udp client can only get back “H” in response.

The same phenomenon observed in both tcp and udp client-server connect is because only one character can be sent and received in each time.

- ***Describe the what needs to be changed if we want to implement a web server that handles one request at a time in the TCP code.***

First of all, we need to add the import statement “import http.server” in the tcp_server.py file.

Secondly, remove the statements “serverSocket = socket(AF_INET,SOCK_STREAM)” and “serverSocket.bind(("localhost",serverPort))”. Replace this line of code with “serverWeb = HTTPServer((“localhost”, serverPort), BaseHTTPRequestHandler)”.

Thirdly, in the while loop, remove the statement “connectionSocket, addr = serverSocket.accept()” and replace it with “serverWeb. handle_one_request()”.

This should completes the setting of the web server.