

Indian Institute of Technology Jodhpur

Computer Science and Engineering Department

Lab 2

CSL6010 - Cyber Security

Date: 02-02-2023

Marks: 10

Aim: Build a simple client-server system, where you use the client to chat with a "math" server. The protocol between the client and server is as follows:

- The server is first started on a known port.
- The client program is started (server IP and port is provided on the commandline).
- The client connects to the server, and then asks the user for input. The user enters a simple arithmetic expression string (e.g., "1 + 2", "5 - 6", "3 * 4"). The user's input is sent to the server via the connected socket.
- The server reads the user's input from the client socket, evaluates the expression, and sends the result back to the client.
- The client should display the server's reply to the user, and prompt the user for the next input, until the user terminates the client program with Ctrl+C.

You will write two versions of the server:

- a. Your server program "server1" will be a single process server that can handle only one client at a time. If a second client tries to chat with the server while one client's session is already in progress, the second client's socket operations should see an error.
- b. Your server program "server2" will be a multi-process or multi-threaded server that will fork a process for every new client it receives. Multiple clients should be able to simultaneously chat with the server.

At the very minimum, all your servers should be able to handle addition, multiplication, subtraction, and division operations on two integer operands. You may also assume that the two operands are separated by a space. So, some sample test cases are:

User types: 1 + 2, server replies 3

User types: 2 * 3, server replies 6

User types: 4 - 7, server replies -3

User types: 30 / 10, server replies 3

Note that the actual test cases we will use may be different from the ones shown above: your servers should correctly work with any numbers, not just the ones shown above, as long as they conform to this format. Handling non-integer operands, other arithmetic operations, or operations with more than 2 operands (e.g., "1 + 2 + 3") is purely optional.

Several good tutorials and useful documentation on socket programming and designing servers for concurrent clients (using both fork and select) are available online. Please make use of these resources to learn the intricacies of socket programming on your own during this assignment.

Please write your code for this assignment in **python**. Please talk to me if you have a compelling reason to use any other language.

References:

https://www.cse.iitb.ac.in/~mythili/teaching/cs641_autumn2015/pa/pa1/index.html