Java HTTP Server Report

Group 3: Emir D. & Peng J.

Server username/password: “syracuse” / “university”

Things we have implemented:

1. **Basic HTTP server:**

(1) Get HTTP request from client, then process it and generate an correct html response to the client. Can be used to serve HTML page. (by Emir)

(2) Serve other type of files dynamically, such as pdf, mp3, etc. According to the client request information, the server will decide the MIME type accordingly and attach correct response header. (by Peng)

2. **Multi-threading (by Emir):**

The server can be used by more than one user at the same time. Also, multi-threading increases the speed of the server socket acceptance.

3. **Logging (by Peng):**

When the server is running, for each critical process or possible error, the server will record the warning / information in a .log file in the root folder and display it in the console at the same time.

4. **Authorization (by Peng):**

When a new socket is established, the client is not authorized to access the files in the server. So the server will prompt a authorization challenge to the client and after the client input the correct username/password, which is “syracuse”/“university”, the client is then authorized and for the following request, the request header will automatically attach the authentication information so no need for authorization again.

5. **Server-side Scripting (by Emir):**

In this project, for demonstration purposes, we only implemented the “PHP Server-side Scripting” language. Therefore, we first, check that the file the client request is a “PHP” file or not. If it is, we execute the PHP file, if not, we generate a static response. To execute the PHP file, we use the “ProcessBuilder” class. Because, the php script will run on our “computer,” and then send an output to the browser. This is same as executing the php file from our command line. Once we get the output from the PHP file, we generate the header and send the output to the browser to print it to the user.