

Ian Armada
CSE156/L
Lab5
iarmada
1717147

Lab 5: Proxy Server

Overview:

In this lab, I created a proxy server that accepts HTTP requests from a client and forwards the request as a HTTPS request via OpenSSL to the main server. The proxy will then receive the response from the server and forward the response back to the client and log the request into the provided log file path. Also there is a forbidden sites file that sends error codes to the client if it request a forbidden site mentioned in the file.

In order to run this program you must first run make in the top directory. Then open two separate terminals in the bin directory.

In one terminal run:

```
./myserver <port_number> <forbidden_path> <log_path>
```

In another terminal run any curl or wget to the server.

Design:

The design of this lab was taken from Lab1 as it does similar functionality but it instead forwards the request via ssl to another server. I basically just set up a normal http connection at first and for each new connection I would fork a child process to handle new connections. Then for each connection I would parse the request and error check the method and host name/ip and return responses back to the client accordingly. For valid requests I set up a SSL connection and forward the request to the server. Once the server responds to the proxy I log the response and forward the response back to the client and close the connection.

Test Cases:

Here are the following test cases I used to test my programs functionality:

1. Concept: Invalid Inputs

- a. `./myserver -1 0 a`
 - i. Invalid Port Number
- b. `./myserver 10 a a a`
 - i. Too many arguments
- c. `./myserver`
 - i. Too little arguments

2. Concept: Good Inputs

- a. `./myserver 8080 forbidden.txt log.txt`
- b. `./myserver 8081 forbidden.txt log.txt`

3. Concept: Inputs Causing Timeout

- a. `curl -x http://localhost:8082 http://adfasd`
`./myserver 1234 forbidden.txt test1`
 - i. Timeout occurred cause server was not detected (client started prior to server)

4. Concept: Multiple Clients Transmitting to Proxy at Same Time

- a. `./myserver 8081 forbidden.txt log.txt`
`curl -x http://localhost:8082 http://google.com/`
`curl -x http://localhost:8082 http://google.com/`
`curl -x http://localhost:8082 http://google.com/`
`curl -x http://localhost:8082 http://google.com/`
`curl -x http://localhost:8082 http://google.com/`

5. Concept: Massive Document Size

- a. `./myserver 8080 forbidden.txt log.txt`
`curl -x http://localhost:8082 http://worldslongestwebsite.com/`