Jay Offerdahl
EECS 678
Lab 11 Report

Lab 11: Sockets

1. Briefly describe the design of the program. How does your program control when the client runs and when the server runs?

- My program decides when the client runs and when the server runs by first connecting the sockets over which they will communicate. Next, in my server.c file, I have designated the while condition to be “read(session_sockfd, buf, 256)”. This means the while loop will wait for data to be available on the socket at session_sockfd. Once data is available, it will be read into buf, and the desired operation of capitalizing the string and sending it back will occur. After this, in client.c, the program will be waiting on the read() system call in the for loop. This pattern continues until all the data has been sent and received in the client.c program. After this, client.c closes the session socket, causing server.c’s while loop to terminate, ending the program.

2. What is the purpose of the handshake socket? Why not have the server create and bind session sockets that clients may connect to directly?

- This is for the purpose of multiple communication points with the server. If we had the server create and bind session sockets that clients connected to directly, the server would not be able to have discrete connections with other clients. The reason we have the handshake socket is to have a unique point of connection with clients. This allows the server to serve multiple clients at a time.

3. For the simple / client server program, we chose to use sockets instead of pipes to send messages between the client and server. Why are sockets preferred over pipes for this program? Give at least two reasons.

- Sockets are preferred over pipes in this example because it allows the client and server to be on different hosts. If we were to implement a pipe, the data endpoints would have to exist on the same server.
- Next, sockets allow multiple connections (as with this example). If we were to use a pipe, we could only communicate between two processes. Because we use sockets, we have the potential to open multiple communication points between a client and server.