Cindy Zhou
cyzhou@usc.edu
EE 533 Network Processor Design and Programming

LAB 1

Concepts Covered:

1. How to simulate the environment of the client and server on a single computer.
2. Understand how applications communicate with one another. Server and client.
3. Know what types of sockets there are and how this affects communication.
4. How to write code to communicate between two applications on different computers.

Part 1

Following the instructions from the pdf, the line that said to download VMWare Fusion caused a lot of confusion because this seems to be the VMware download for Mac users. I eventually figured this out by asking ChatGPT what the correct version is. I double checked this by looking at these instructions, but there was more confusion because the VMware that the writer used couldn't be found, and so I just downloaded a random VMware Workstation.

After creating the two Virtual Machines on Workstation, getting them to communicate with one another did not require the Wikihow tutorial, which I never got through because I never figured out how to get past step 22. However, I was still able to ping one VM from another, so network communication seems to work fine.

Part 2

To understand how socket communication works, I read the section on communication in the lab manual. Though both the client and server use a socket, the way that you set up the socket is different depending on who you are.

Client Code:

1. create socket()
2. connect socket to server address using connect()
3. send and receive data using read() and write()

Server Code:

1. create socket()
2. bind() socket to an address
3. listen() for connections
4. accept() connections, and handle multiple requests if possible / when necessary
5. send and receive data

Part 3

To understand how socket the types of sockets used in communication, I read the section on communication in the lab manual and looked up the topic to fill in my knowledge gaps. There are the unix domains and internet domains, both of which were covered during class.

The Unix domain seems to only be communicating through the same computer, though different processes, since you need to know the file path.

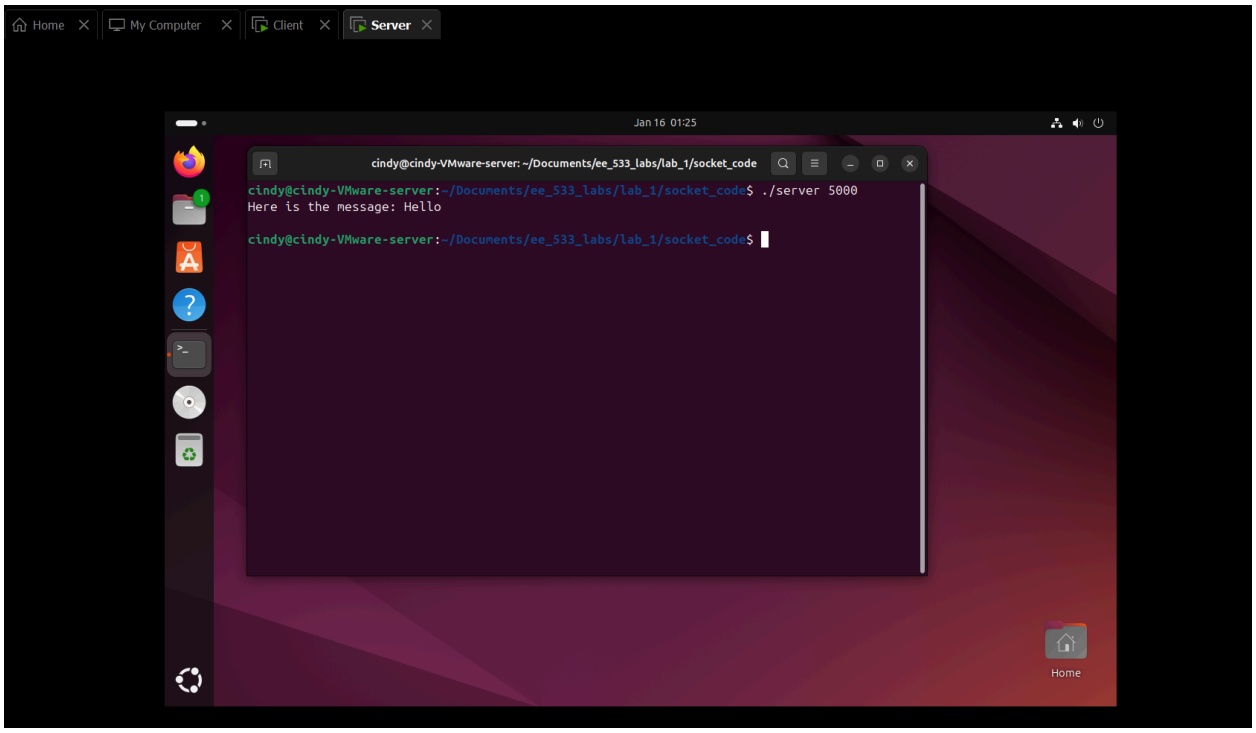
The Internet domain communicates through UDP or TCP protocols, and can easily happen between processes on two different devices. To do this, a server needs to specify the port that it will run on, and the client needs to know the Internet address of the server (IP), along with the port.

Part 4

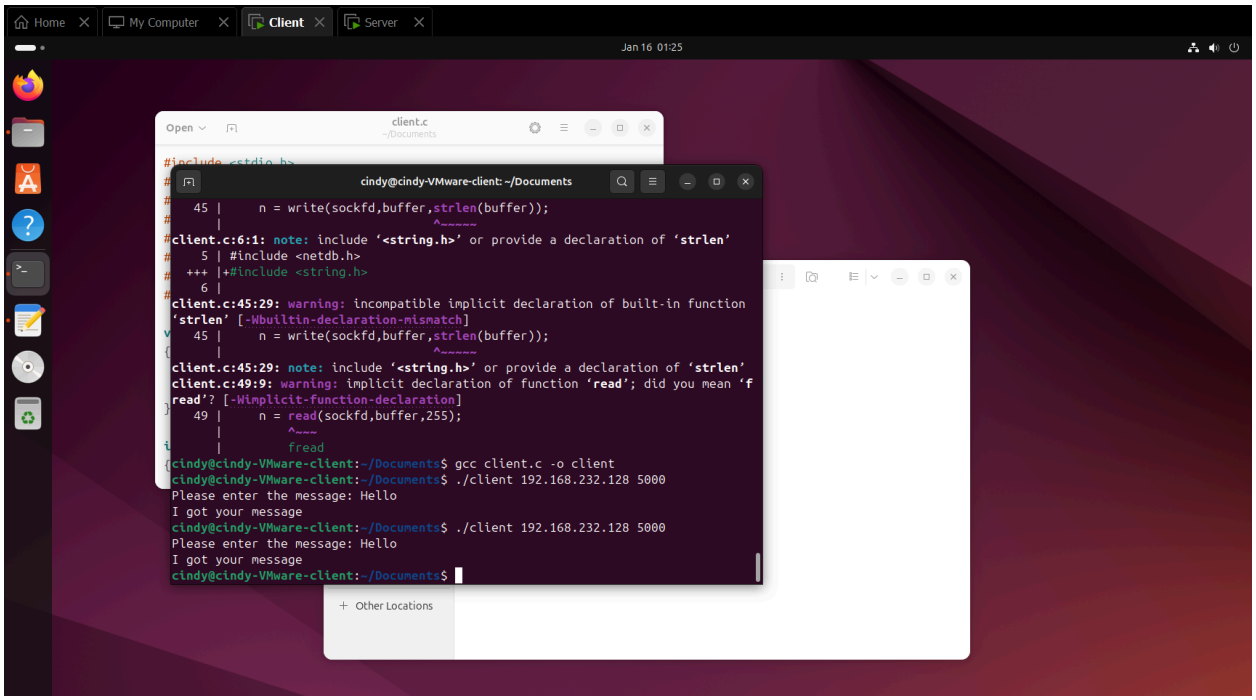
To complete this portion of the lab, I installed GCC on both the Virtual Machines. After this, I tried to compile the code, and it had errors. This shouldn't have been a big deal, but I added some imports just in case.

For the provided .c code, I first ran the server using ./server 5000. I then ran the client using ./client 192.168.232.128 5000. This meant that the server was running on port 5000, and when the client wants to talk to the server, it finds the server using the server's ip, and finds the specific application by the port.

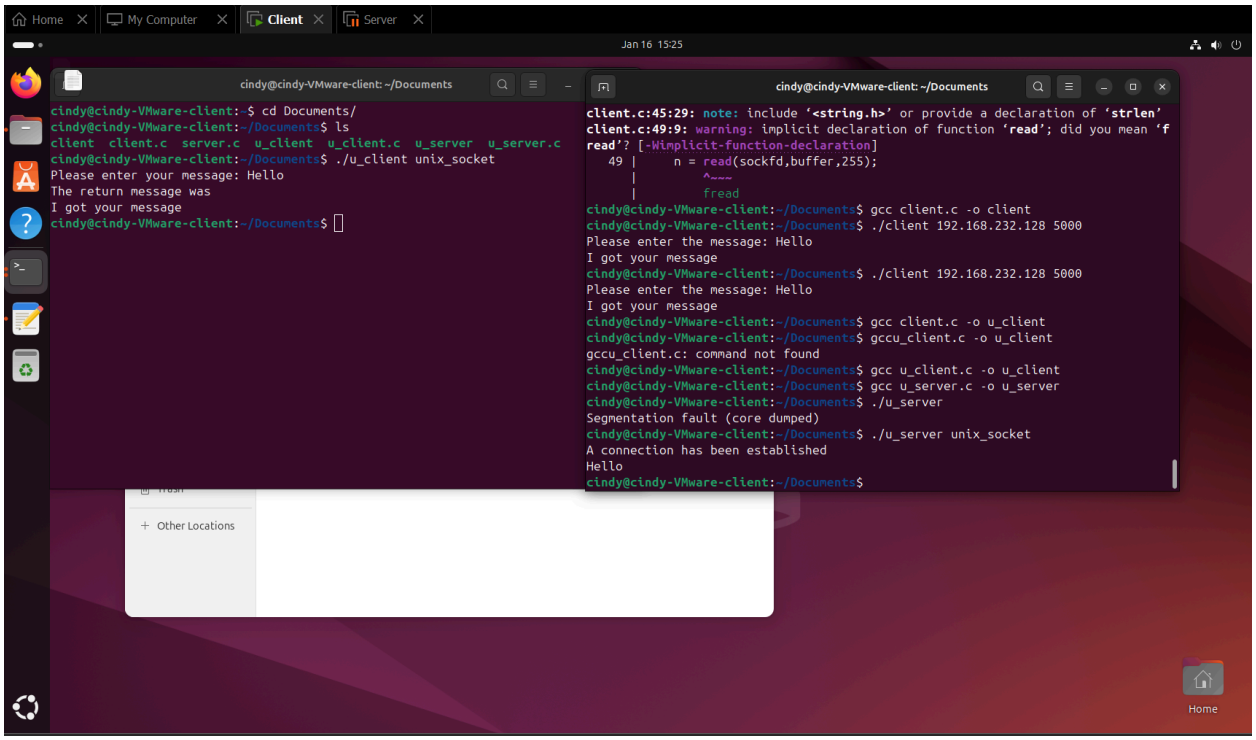
Server Code



Client Code



Unix Socket


Home
My Computer
Client
Server
Jan 16 15:25
cindy@cindy-VMware-client: ~/Documents
cindy@cindy-VMware-client:~$ cd Documents/
cindy@cindy-VMware-client:~/Documents$ ls
client client.c server.c u_client u_client.c u_server u_server.c
cindy@cindy-VMware-client:~/Documents$ ./u_client unix_socket
Please enter your message: Hello
The return message was
I got your message
cindy@cindy-VMware-client:~/Documents$
cindy@cindy-VMware-client: ~/Documents
client.c:45:29: note: include '<string.h>' or provide a declaration of 'strlen'
client.c:49:9: warning: implicit declaration of function 'read'; did you mean 'fread'? [-Wimplicit-function-declaration]
49 | n = read(sockfd,buffer,255);
| ^~~~
| fread
cindy@cindy-VMware-client:~/Documents$ gcc client.c -o client
cindy@cindy-VMware-client:~/Documents$ ./client 192.168.232.128 5000
Please enter the message: Hello
I got your message
cindy@cindy-VMware-client:~/Documents$ ./client 192.168.232.128 5000
Please enter the message: Hello
I got your message
cindy@cindy-VMware-client:~/Documents$ gcc client.c -o u_client
cindy@cindy-VMware-client:~/Documents$ gccu_client.c -o u_client
gccu_client.c: command not found
cindy@cindy-VMware-client:~/Documents$ gcc u_client.c -o u_client
cindy@cindy-VMware-client:~/Documents$ gcc u_server.c -o u_server
cindy@cindy-VMware-client:~/Documents$ ./u_server
Segmentation fault (core dumped)
cindy@cindy-VMware-client:~/Documents$ ./u_server unix_socket
A connection has been established
Hello
cindy@cindy-VMware-client:~/Documents$
Other Locations
Home