SPL-1 Project Report, 2019

LAN Chat

SE 305: Software Project Lab 1

Submitted by

Atkia Akila Karim Sobah

BSSE Roll No. : 1015

BSSE Session: 2017-18

Supervised by

Prof. Dr. Md. Shariful Islam

Designation: Director

Institute of Information Technology

Institute of Information Technology

University of Dhaka

[29-05-2019]

Table of Contents

1. Introduction

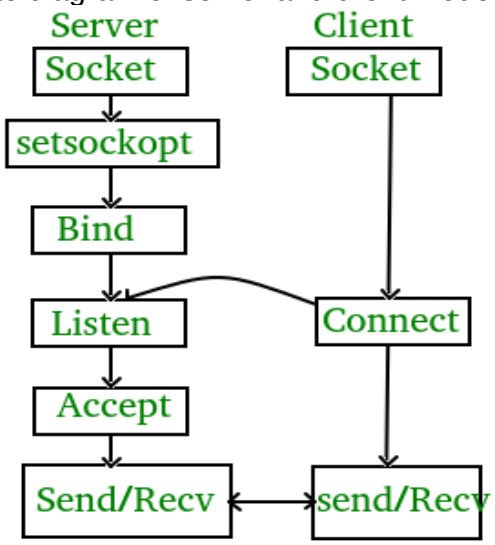
LAN Chat defined as an instant messaging application used within office local area network (LAN). As external users cannot access LAN, the office chat will be more secure in LAN Chat than public messenger. Moreover it doesn't require any internet connection. In my project it has One-to-One Chat, Group Chat. It is an example of a chat server. It is made up of 2 applications the client application, which runs on the user's PC and server application, which run on any PC on the network. To start chatting client should get connected to server where they can practice two kinds of chatting, single chat and Group chat.

1.1 Background Study

Socket Programming

Socket programming is a way of connecting two nodes on a network to communicate with each other. One socket (node) listens on a particular port at an IP, while other socket reaches out to the other to form a connection. Server forms the listener socket while client reaches out to the server.

State diagram of server and client model:



Socket creation:

int sockfd = socket(domain, type, protocol)

sockfd: socket descriptor, an integer (like a file-handle)

domain: integer, communication domain e.g., AF_INET (IPv4 protocol) , AF_INET6 (IPv6 protocol)
type: communication type

SOCK_STREAM: TCP(reliable, connection oriented)
SOCK_DGRAM: UDP(unreliable, connectionless)
protocol: Protocol value for Internet Protocol(IP), which is 0. This is the same number which appears on protocol field in the IP header of a packet.(man protocols for more details)

. Bind:

```
int bind(int sockfd, const struct sockaddr *addr, socklen_t addrlen);
```

After creation of the socket, bind function binds the socket to the address and port number specified in addr(custom data structure). In the example code, we bind the server to the localhost, hence we use INADDR_ANY to specify the IP address.

. Listen:

```
int listen(int sockfd, int backlog);
```

It puts the server socket in a passive mode, where it waits for the client to approach the server to make a connection. The backlog, defines the maximum length to which the queue of pending connections for sockfd may grow. If a connection request arrives when the queue is full, the client may receive an error with an indication of ECONNREFUSED.

. Accept:

```
int new_socket= accept(int sockfd, struct sockaddr *addr, socklen_t *addrlen);
```

It extracts the first connection request on the queue of pending connections for the listening socket, sockfd, creates a new connected socket, and returns a new file descriptor referring to that socket. At this point, connection is established between client and server, and they are ready to transfer data.

Stages for Client

. Socket connection:
Exactly same as that of server's socket creation
. Connect:

```
int connect(int sockfd, const struct sockaddr *addr,socklen_t addrlen);
```

The connect() system call connects the socket referred to by the file descriptor sockfd to the address specified by addr. Server's address and port is specified in addr.

1.1 Challenges

Implementing a new software solution carries with it a number of challenges. The process can be overwhelming, confusing and lengthy. For implementing this project there are lot of challenges that I have faced. Some of them are

- Handling large code for the first time
- learning and understanding socket programming
- lan chat code writing in c

2. Project Overview

I have divided my whole project into three different parts. They are

- Implementation of socket programming
- Multiple client handling
- Single chat
- Group chat

2.1 Implementation of Socket programming

This is the core part of my project. LAN Chat is basically established through this. In this Socket programming I have used TCP connection to connect server and client. Client send connection request to server. Server first create socket in which client will communicate. Then it has bind and listen function then it has accept function which accept the request of client part. Then both can communicate with each other. That is basic part of socket programming.

2.2 Multiple Client handling

For multiple client handling I have used fork function which help server to handle multiple client at a time.

2.3 Single Chat

In my project one can chat with server. It requires username and password to log in one can also sign up to create its account.

2.4 Group Chat

In my project one can connect with multiple user. Server broadcast the message of client to all of its connected client. Through this one can forward message faster.

3. User Manual

I have implemented my project in C. So it is not that much user friendly. This program needs two run side as one is client and other is as server. First one need to run server part then client part it doesn't require port no as it is given in my program. Client initiates the process of sending and receiving.

4. Conclusion

Implementing LAN Chat helps me to improve my coding skill and I have learned to handle large code for the first time. I hope it will help me to deal with difficulties in future. This

project was quiet challenging and I gained a lot of experience from it. I want to thank my supervisor for guiding me a lot during this project.

5. Appendix

In this project, I have implemented single chat and group chat. In future I want to implement file transformation and also implement Graphics User Interface.

6. Reference

1. https://www.youtube.com/watch?v=hptViBE23fI

2. https://en.wikipedia.org/wiki/LAN_messenger

3. https://www.youtube.com/watch?v=BIJGSQEipEE

4. https://www.geeksforgeeks.org/socket-programming-cc