A detail project review on ChatChamber

***Group members*:**Gaurav Giri,Ajaya Chaudhary,Narayan Poudel,Iza K.C.

***Abstract* :**The objective of the ChatChamber is to create a chat application which provides a bidirectional communication between clients.The ChatChamber Program will enable the user to chat with the logged users in the chat room with an active server. It provides a very simple, fast, multithreaded platform with an independent web socket and client library that is used for the connection between client and server. This provides a robust way of communication between nodes. ChatChamber achieves a simple, reliable, multi-user in-terminal chatting service that supports sending and receiving of texts in a terminal based application.

Index terms - In-terminal application, IP and Ports,star topology, TCP Connection,web sockets, server, clients

Introduction

ChatChamber is a client-server system where the users exchange text messages between the system's users with the help of a central server. The user of the system is defined as client-server. Chat system is a distributed programming which consists of two distributed components,a chat server and a chat client. Chat client supports all communication including requesting chat server location information from a location server and displaying received chat messages. Chat server will conduct chat sessions and manage all chat clients. Basically, a chat client starts the chat session by requesting the communication parameter (server name and port number). In a topological way, this system is working on star topology because if the server is down, then the connection is over. Chat System is a form of communication that utilizes computer programs that allow for two-way conversations between users in real time (events that occur in cyberspace at the same speed that they would occur in real life). Typically the users will connect to a chat server using a chat client and meet in a chat room. Once the users are in the same chat room, they can converse with one another by typing messages into a window where all of the other users in the chat room can see the message. The user can also see all of the messages entered by the other users. Conversations are then carried on by reading the messages entered by the other users in the chat room and responding to them.

ChatChamber as a project helped us boost the knowledge regarding networking and multithreading.

It is an open source application so the users themselves can gain insight on the code of the program and can contribute to help better the program itself. The application provides encryption for secure message sharing. As it is a terminal based application no resources are required to render GUI and takes up minimal resources in the regards of memory and CPU uses.

Methodology

ChatChamber works upon the primitive sockets where there are TCP connections between the clients and a server.

A.*Transmission Control Protocol*

TCP is a standard that defines how to establish and maintain a network conversation by which applications can exchange data. TCP works with IP, which defines how computers send packets of data to each other. Together, TCP and IP are the basic rules that define the internet.TCP performs the following actions:

- Determines how to break application data into packets that network can deliver
- Sends packets to and accepts packets from the network layer
- Manages flow control
- Handles retransmission of dropped or garbled packets as it's meant to provide error-free data transmission
- Acknowledges all packets that arrives

Network packets are basic units of data that's grouped together.

B.*Server*:

A server is a software or hardware device that accepts the requests made over a network. This is the computer or a system which will accept or listen to the connection of other devices or systems to it. Servers are the main root of the network. If the server is down, the devices connected to it will lose their connection. Server handles and manages the devices connected to it and sets up various rules and protocols for the devices connected to it to make the network more secure, efficient and reliable.

C. *Clients:*

These are the devices or computers which are connected to a server. They will be limited to their functions that are assigned from the server. If the server rejects the clients data that violates the server's rule or protocol then the data will not be processed further

D.*Sockets*

A socket is one endpoint of a two way communication link between two programs running on the network. The socket mechanism provides a means of inter-process communication (IPC) by establishing named contact points between which the communication takes place. A socket connecting to the network is created at each end of the communication. Each socket has a specific address. This address is composed of an IP address and a port number.
Sockets are generally employed in client server applications. The server creates a socket, attaches it to a network port address then waits for the client to contact it. The client creates a socket and then attempts to connect to the server socket. When the connection is established, transfer of data takes place

E.*Threads*

A thread is an independent set of values for the processor registers (for a single core). Since this includes the Instruction Pointer (Program Counter), it controls what executes in what order. It also includes the Stack Pointer, which had better point to a unique area of memory for each thread or else they will interfere with each other. Threads are the software unit affected by control flow (function call, loop, goto), because those instructions operate on the Instruction Pointer, and that belongs to a particular thread. Threads are often scheduled according to some prioritization scheme (although it's possible to design a system with one thread per processor core, in which case every thread is always running and no scheduling is needed)

.
ChatChamber has a hosting server and clients will be able to join the server and share text data among the clients and some user-defined functions for the manipulation of the clients. Server manages the clients through linked lists and the processes for them will be handled by threads. So everytime, a client joins the server, a node and a thread is created in the linked list. First of all, a server will host a chatChamber by binding its ip (say XXX.XXX.XXX.XXX) and port (YYYY) and listening to the clients that will join it. The clients will then enter the ip and the port no of the server to connect the server. Then the server will prompt them two options, one will be for login and another will be for registering an account. Then User will choose accordingly, the verification is done in the server by reading the user's log and a new account will be appended to the data file if a user registers a new account. After that, the clients that are already present in the chatroom will get notified about the information of new client joining. And then the message one client sends will be sent to all the other clients through the server. If a client leaves, then the signal is caught in the server and then all the clients will get notified. All the chats will get stored in a log file and if any user wants to read the log file, they are freely allowed to see them.

Problems encountered

A.*Integrating SSL for encryption*

SSL is an security protocol that provides privacy, authentication, and integrity to Internet connections.
ChatChamber couldn't be properly integrated with SSL as we faced time management issues
B.*Parsing and Tokenizing commands*

We couldn't find an easy way to parse and tokenize commands because C programming language has extremely complicated ways for string manipulation.
C. *Creating a terminal UI*

While creating a terminal UI nCurses library was used. C being a low level language and ChatChamber itself a terminal based application achieving a somewhat pleasing UI itself was a hassle.
D. *Crossplatform*

The application uses linux api for networking which is not available for windows.Although the code can be compiled using Windows api, some parts of the code requires some changes. So ChatChamber is not yet a cross platform application.

Future prospects

ChatChamber hopes to achieve end-to-encryption through SSL.
We want the application to have a file sharing feature.
Having personal chat channels for more personalized app experience.
The resources used by the application can be better optimized by thread pulling.
Integrating different plugins, bots, and modes is one of the main priorities for ChatChamber.

Citations

[1] Lutkevich, Ben. “What Is Transmission Control Protocol (TCP)? Definition from Searchnetworking.” *Networking*, TechTarget, 4 Oct. 2021, https://www.techtarget.com/searchnetworking/definition/TCP
[2] “What Is a Server?” *Computer Hope*, 31 Dec. 2022, https://www.computerhope.com/jargon/s/server.htm
[3]“How Sockets Work.” *IBM*, https://www.ibm.com/docs/en/i/7.1?topic=programming-how-sockets-work.
.