**Secure Group Chat Server**

For

**NETWORK PROGRAMMING LABORATORY**

**T. E. Computer Engineering**

**Dwarkadas J. Sanghvi College of Engineering, Mumbai**

By

**Arnav Jalui 60004150036**

**Jahanvi Jasani 60004150037**

**Yash Lahoti 60004150050**

**Zahan Shahana 60004150113**

Name of the Group Members **Arnav Jalui (60004150036) A-4**

with SAP ID with Batch No: **Jahanvi Jasani (60004150037) A-4**

**Yash Lahoti (60004150050) A-4**

**Zahan Shahana (60004150113) A-4**

Class with Division: **TE (A)**

Name of the Supervisor/Guide: **Poonam Joshi**

Title of Report:  **Secure Group Chat Server**

Field of Project:  **Networking**

Area of the project: **Mitigation of Networking Threats**

**Abstract:**

This project, in all its essence, is a group chat program, between a variable number of users, who are connected to each other, by virtue of being a member of the same network. A central server acts as a service to accept messages from a user and forward it to all other users, who are participating in the group chat by being connected to the server at that particular point of time. Given that the server is up and running, new users can join the group chat server at any given time by providing the current local IP address and the port on which the server is running, although, this can only be done if the active members in a group chat does not exceed the specified and declared capacity of the server. If a large number of users is expected, the server can be configured to listen for a higher number of incoming connection requests, which should be carefully determined, since, performing an alteration in the capacity of the server requires the server to be restarted for the new capacity to come into effect. This project has been implemented in Python 2 programming language. The unique selling point of this group chat server is that it is a secure group chat server. The messages that are being sent and received are encrypted. When the server is started, a random cipher key is generated for encryption and decryption of messages. This cipher key is the secret key that is confidential within the boundaries of the system. When a message originates from a user, it is encrypted before being sent to the server. The server receives this encrypted message and forwards it to the other users that are currently connected to the group chat server. The other users receive an encrypted message, which is decrypted by the user terminal using the randomly generated cipher text that is being used system-wide.

**Background art of the work/prior history:**

By World War II, mechanical and electromechanical cipher machines were in wide use, although—where such machines were impractical—manual systems continued in use. Great advances were made in both cipher design and cryptanalysis, all in secrecy. Information about this period has begun to be declassified as the official British 50-year secrecy period has come to an end, as US archives have slowly opened, and as assorted memoirs and articles have appeared. Encryption in modern times is achieved by using algorithms that have a key to encrypt and decrypt information. These keys convert the messages and data into "digital gibberish" through encryption and then return them to the original form through decryption. In general, the longer the key is, the more difficult it is to crack the code. This holds true because deciphering an encrypted message by brute force would require the attacker to try every possible key. Beginning around 1990, the use of the Internet for commercial purposes and the introduction of commercial transactions over the Internet called for a widespread standard for encryption. Before the introduction of the Advanced Encryption Standard (AES), information sent over the Internet, such as financial data, was encrypted if at all, most commonly using the Data Encryption Standard (DES)

**Description of the project work:**

1. **Introduction**

Group chats have become an indispensable part of the communication aspect of our lives. Individual chats, as the name suggests are P2P chat programs. Group chats provide the users with the ability of having simultaneous conversations with all the other users. Group chats are extensively used by organizations to collaborate and communicate for official purposes. Official communications, related to the work that an organization does, needs to be confidential. It should be confined within the bounds of the organization, or if possible, only limited to the participants of the chat. Dispersion of information external to the participants of the chat, or employees of the organization is harmful for the concerns of the organization. One technique of keeping communication confidential is to encrypt the messages being sent and received on the server. Encryption of messages ensure that they reside in an unintelligible format, which does not make sense, even if they are intercepted or acquired by sources who are not meant to accessing the information and have no authority to do so. These encrypted messages should be decrypted before they assume the original body and intended meaning of the message in question. This process of decryption can be carried out only by systems that have access to a secret key, known as the cipher key. This key is only accessible by authorized users. This key is essential to carry out the decryption of messages. Without the possession of this secret key, unauthorized sources cannot decipher the original message even if the get access to the encrypted messages. All messages that flow in the system, between users, are encrypted. They are encrypted before being sent out by a sender and are decrypted only when they reach an authorized and intended user, who is in possession of the cipher key. This makes sure that unencrypted messages do not travel in the system, and the security of the system along with the confidentiality of information is maintained.

1. **Aim & Objective**

The main objective of the project is to encrypt/decrypt messages for personal and professional security as well as protect against unauthorized data access.

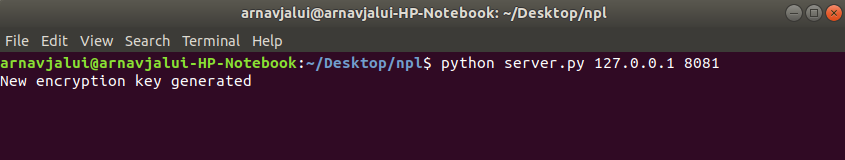
1. **Platforms used**

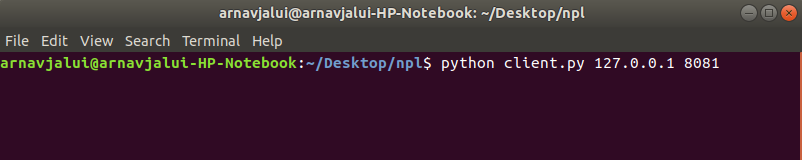
The server and clients communicate through the use of *send* and *recv* system calls, which is present in the Linux OS lineup. This product has been developed and tested on Ubuntu 17.10. Python 2 is the choice of programming language due to the easy availability of modules that help in making the development and maintenance of such a system much more straightforward. Messages are not backed up or stored anywhere, keeping in mind the confidentiality requirement of such a system. This eliminates the need of a database system.

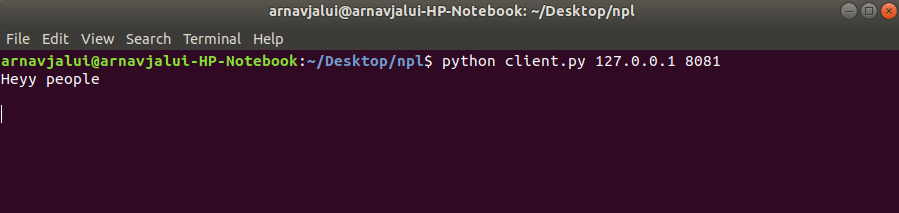
1. **Working**

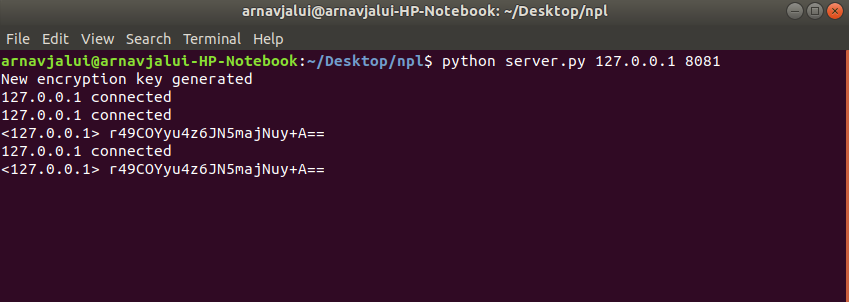
When the server is started, a random cipher key is generated and stored in the file *key.txt*, which is only accessible to machines in the same network as that of the server. The local IP address of the server is the IP address from which users are allowed to join the group chat by providing IP address and the port number on which the server is running. Upon a successful connection, a log is displayed on the server screen about the newly connected client. When a client sends a message, the message is encrypted at client’s machine. This encrypted message is sent to the group chat server, which then forwards it to all other machines connected to the server, except the machine which sent the message in question. The encrypted message sent out to all clients is accompanied by the originator address of the message, which is in encrypted form as well. Upon reaching the destination client machines, this encrypted machine is then decrypted with the help of the cipher key in *key.txt* file and the original message and the sender of the message is shown to the client. At no point of time does a message exist in an unencrypted form while being transferred over the network. Encryption is done before being sent out and this encrypted message is decrypted only when message reaches the intended client on the network. This ensures confidentiality of a message even if it is intercepted in an illegal manner by parties who are not supposed to have access to these messages.

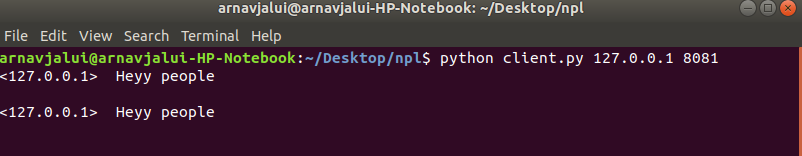
1. **Screenshots**

*****Group chat server is started and new cipher key is generated*

*Client started and connected*

*Message sent from client machine*

*Encrypted messages encountered by server*

*Message received by other client/clients*

1. **Results**

The messages are being sent out in an encrypted form, and are decrypted and restored for retrieving original message only after reaching the destination. This is evident from the fact that the messages displayed on the server are in an encrypted and unintelligible format. Messages sent out from a client are received by all other connected clients, along with IP address of sender of message for identification purposes.

1. **Conclusion**

The system performs exactly as desired and stands up to our expectations of providing a secure and reliable group chat server, that can be used in various organizations and institutes, where secrecy of internal communications is a major concern. Bearing in mind the growing public demand of better privacy in a highly digitized age, this secure group chat server has the potential to replace traditional group chat servers.