

A Secured TCP Chat Room using Python

Anıl Koçer
191180057
Gazi University
Ankara, Turkey
anilkr00@gmail.com

Oğuz Kaan Subaşı
191180076
Gazi University
Ankara, Turkey
ouzkaansubasi@gmail.com

Selin Cansu Akbaş
191180005
Gazi University
Ankara, Turkey
selincansuakbas@gmail.com

Abstract—In today's digital age, stable verbal exchange is paramount, in particular in chat packages wherein facts are regularly exchanged. This project, "Secured TCP Chat Room using Python" aims to develop a strong and stable chat room software leveraging the Transmission Control Protocol (TCP) for dependable data transmission and incorporating encryption mechanisms to make sure records confidentiality and integrity. The chat room software is designed to facilitate real-time verbal exchange among more than one customers linked over a network. By enforcing TCP, the software ensures the dependable transport of messages, making sure that no records is misplaced in the course of transmission. By prioritizing each capability and security, this task objectives to make contributions to the developing want for stable virtual verbal exchange tools.

Keywords—secure communication, chat, TCP, message, data transmission, server, client

I. INTRODUCTION

In the contemporary digital era, the rapid evolution of communication tools and the internet's integration into daily life have underscored the importance of secure and effective communication platforms. Online chat rooms are among the widely used tools that facilitate instant communication among individuals and groups. These platforms offer a convenient means for people to exchange information, discuss ideas, and stay connected irrespective of geographical distances. However, the increasing dependence on these platforms has heightened concerns about the security of communication channels. Safeguarding users' personal data and ensuring the confidentiality of their communications are crucial for maintaining trust and privacy. This project focuses on designing and implementing a secure TCP (Transmission Control Protocol) chat room using the Python programming language.

The primary goal is to provide users with a secure environment for instant messaging, addressing the vulnerabilities and risks inherent in online communication. By utilizing the robustness and reliability of TCP, the chat room application aims to deliver a seamless and secure communication experience. To achieve this, the project incorporates several security measures, including message encryption and protection against common network threats. Encryption ensures that messages are accessible only to the intended recipients, thus protecting the privacy of the conversations.

Additionally, the implementation of security protocols and best practices reduces the risk of data breaches and maintains the integrity of the communication process. The secure TCP chat room application developed in this project is intended to be a versatile communication platform for both individuals and groups. Whether for personal use, professional collaboration, or social interaction, the application aims to offer a safe and reliable medium for information exchange. Users can create private or group chat rooms, manage their

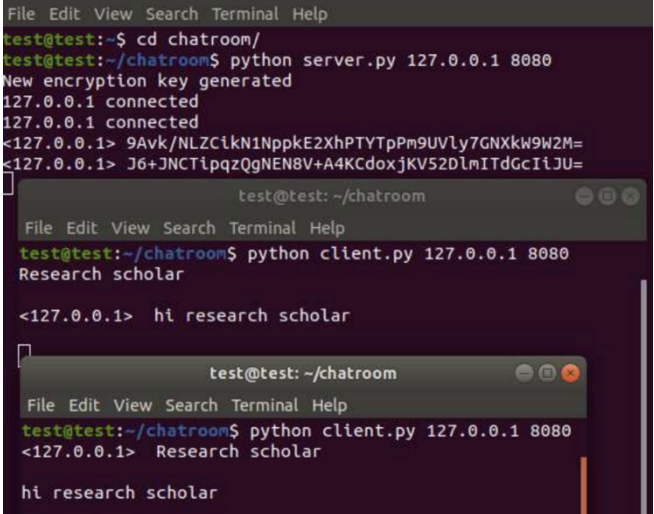
contacts, and engage in real-time messaging with the assurance that their communications are secure.

In essence, this project aims to develop a TCP chat room application that seamlessly merges the ease of instant messaging with strong security measures. By tackling the challenges associated with online communication security, the application aspires to provide a reliable platform where users can exchange messages securely and effectively. Through this endeavor, we intend to advance the creation of safer digital communication tools and underscore the significance of privacy and security in the modern digital landscape [1].

II. RELATED WORKS

In this section, studies in the literature similar to this project will be examined.

A. Secure Chat Room Application using Advanced Encryption Standard Algorithm



```
File Edit View Search Terminal Help
test@test:~$ cd chatroom/
test@test:~/chatroom$ python server.py 127.0.0.1 8080
New encryption key generated
127.0.0.1 connected
127.0.0.1 connected
<127.0.0.1> 9Avk/NLZCikN1NppkE2XhPTYPm9UUVly7GNXk9W2M=
<127.0.0.1> J6+JNCTlpqzQgNEN8V+A4KCdoxjKV52DlnITdGcIiJU=

test@test:~/chatroom
File Edit View Search Terminal Help
test@test:~/chatroom$ python client.py 127.0.0.1 8080
Research scholar

<127.0.0.1> hi research scholar

test@test:~/chatroom
File Edit View Search Terminal Help
test@test:~/chatroom$ python client.py 127.0.0.1 8080
<127.0.0.1> Research scholar

hi research scholar
```

Figure 1. Chat Room

In his 2021 paper titled "Secure Chatroom Application Using Advanced Encryption Standard Algorithm," Pasumarty explores the development of a secure chatroom application. The study focuses on ensuring the privacy and security of communication between users by utilizing the Advanced Encryption Standard (AES) algorithm

This study (Figure 1) begins by detailing the fundamental principles and encryption processes of the AES algorithm. As a symmetric key algorithm, AES is highly effective for data security, making it a preferred choice for secure data transmission in chat applications.

During the development of the application, the emphasis was placed on the secure transmission of messages between users, the encryption of messages, and the decryption processes. In this process, each user's messages were encrypted using the AES algorithm and could only be decrypted by authorized users.

In conclusion, this study presents an effective solution to minimize the security risks encountered by users in online communication. By leveraging the power of the AES algorithm, the secure chatroom application ensures the privacy and integrity of user data. The findings of the study indicate that the AES algorithm can be effectively used in the development of secure communication systems [2].

B. Realtime Chat Application Using Client-Server Architecture

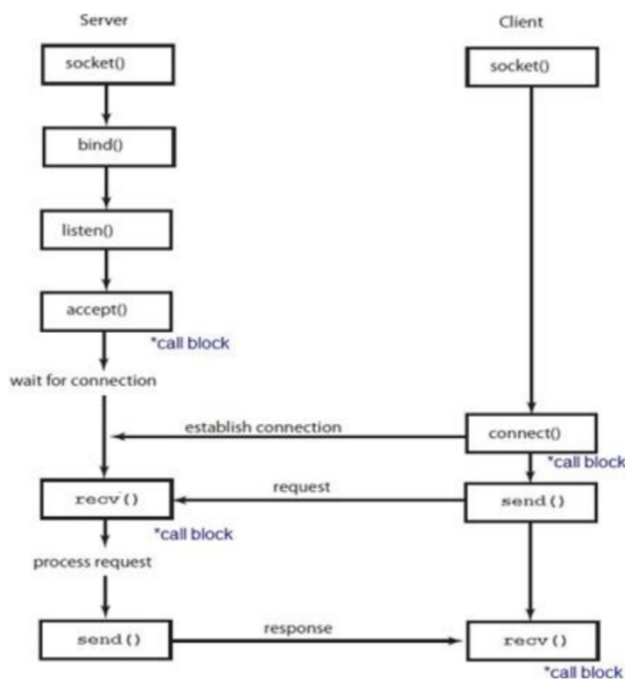


Figure 2. System Architecture

This study focuses on a chat system that originated in the mid-1980s and was quite popular at the time. The chat application represents communication between two entities (sender and receiver). In an era where security and internet penetration are increasing daily, these issues are emphasized. In this application, a server and several client connection points are created, and clients communicate with the server using a connection module. These connection points are endpoints for sending and receiving data. Each organization has two connection points. The program is implemented using TCP connection, which will connect to a specific port on the machine or local host. In the case of the client, a connection point will connect to the same port used on the server side. This application can assist many professional organizations, colleges, preferred schools, universities, and IT companies. Therefore, we aim to design this application for the local networks of these organizations. People can communicate and exchange ideas within the LAN using

many features of this chat application. As a result, server-client applications can be used to perform various queries such as medical helpline services and user reporting, and can be utilized in different scenarios. This provides an effective approach to creating a chat process today without UI or any other human intervention [3].

III. LITERATURE REVIEW

A. Chat Rooms

A chat room is a virtual space on the internet where users can communicate with each other in real-time through text-based messages. It allows people from all over the world to join and engage in conversations on various topics. Chat rooms can be public, where anyone can participate, or they can be private, accessible only to selected individuals. In a chat room, users typically see a list of people who are currently online and can join ongoing conversations or initiate new ones. Some chat rooms may have moderators who oversee the discussions to ensure they remain respectful and on-topic. Chat rooms have been popular since the early days of the internet and have evolved over time to include features like emojis, file sharing, voice and video chat, and more. They serve as platforms for socializing, seeking advice, discussing common interests, or even collaborating on projects. For example, on Reddit's "r/CasualConversation," users can chat about random subjects. Platforms like Stack Overflow offer chat rooms dedicated to discussing technology. On Goodreads, users can join groups to discuss specific books. Stack Overflow's chat rooms provide assistance for programming-related queries. Apple's support forums serve as support chat rooms for Apple products.

Chat rooms are typically found on specific platforms or websites, and users can join either by creating an account or anonymously [4].

B. Chat Applications

Communication in the digital age is evolving rapidly, and chat applications are leading the way in this transformation. People are increasingly turning to digital platforms for messaging, voice, and video calls, abandoning traditional methods for faster and more efficient communication. In this regard, chat applications are gaining importance and offering solutions tailored to various needs [5].

Chat applications are platforms that allow real-time text communication between users. Examples include WhatsApp, Slack, Discord and Telegram. These applications rely on network protocols to manage connections and data transmission, making them relevant for understanding our project's context.

- I. Instant Messaging: Chat applications enable users to send and receive instant messages quickly and easily, ensuring seamless and real-time communication. WhatsApp, Facebook Messenger, Telegram, among others, are prominent in this field.
- II. Voice and Video Calls: Chat applications go beyond text messages, also providing users with the ability to make voice and video calls.

Applications like Zoom, Skype, Google Meet, are prominent players in this domain.

- III. Group Chats and Communities: Chat applications allow users to create groups and interact with people who share common interests. Discord, Reddit, among others, facilitate community interaction and group chats.
- IV. Security and Privacy: Chat applications implement various security measures to protect users' communications. Features like end-to-end encryption, private chat modes, ensure users' privacy and security. Signal, Wickr, are examples of secure chat applications.
- V. Business Productivity: Business-focused chat applications streamline collaboration among teams, enhancing productivity. Features such as file sharing, task management, calendar integration, improve project management and communication within teams. Microsoft Teams, Asana, are widely used chat applications in the business world [5].

C. TCP (Transmission Control Protocol)

The Transmission Control Protocol (TCP) is one of the main protocols of the Internet protocol suite. TCP is a connection-oriented, and a connection between client and server is established before data can be sent. Transmission Control Protocol is a fundamental communication protocol used on the Internet. It ensures reliable and ordered delivery of data packets between devices connected to a network. One of its key features is reliability, achieved through mechanisms such as packet retransmission and error detection to handle packet loss or corruption. Additionally, TCP guarantees sequential transmission of data packets, ensuring that they arrive at the destination in the correct order. It also incorporates flow control mechanisms to manage network traffic and prevent congestion. TCP operates on a connection-oriented basis, establishing a connection between communicating parties before data exchange. With its bidirectional communication capability, TCP facilitates efficient and error-free data transmission across networks, making it a crucial component of various internet-based applications and services.

TCP (Transmission Control Protocol) is a communication protocol used for transmitting data over the internet. TCP is designed to ensure reliable transmission of data. Its primary purpose is to ensure that data is received without loss, corruption, or out-of-order delivery [6].

Some features of TCP include:

- I. Reliability: TCP ensures that transmitted data is received reliably. This is achieved through the acknowledgment and reassembly of data packets at the destination machine.
- II. Flow Control: TCP uses flow control mechanisms to prevent sending more data than

the receiving machine can handle. This helps alleviate the receiver's load and manages network traffic more efficiently.

- III. Connection Management: TCP includes mechanisms for establishing, maintaining, and terminating a connection. This enables end-to-end monitoring and management of the communication process.
- IV. Error Detection and Correction: TCP can detect and, if necessary, correct errors in transmitted data. This is important for ensuring reliable communication.

While TCP offers many advantages, it also has some disadvantages compared to other communication protocols. For example, because additional control information is sent during data transfer, TCP generally requires more bandwidth and processing resources. Therefore, in some applications where speed and data volume are more critical, other protocols like UDP may be preferred. However, TCP is preferred when reliable data transmission is essential [6].

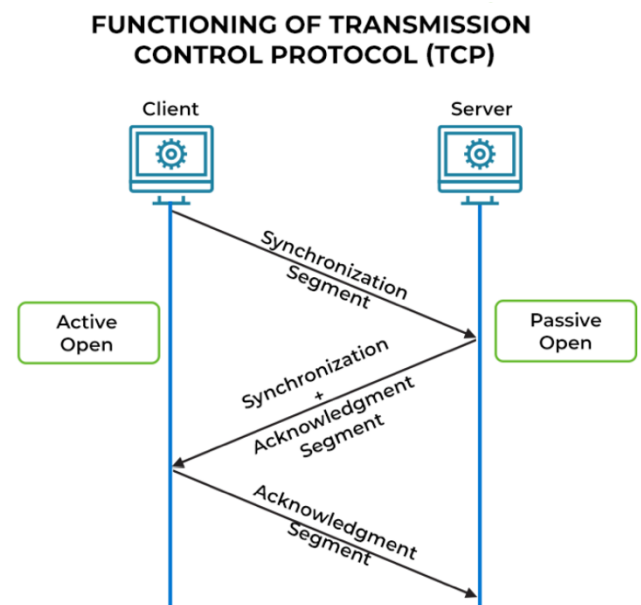


Figure 3. Functioning of Transmission Control Protocol

The server must be listening (passive open) for connection requests from clients before the connection is established. The main highlighting feature of TCP is that it performs Three-way handshake, and only after that we can send messages (Figure 3) [6].

D. SSL (Secure Sockets Layer)

SSL (Secure Sockets Layer) is a cryptographic protocol used to secure communication over the internet. It provides security services such as encryption and authentication, ensuring the protection of user data [7].

The primary purpose of SSL is to ensure the security of data during transmission. It achieves this through the following key features:

- I. Encryption: SSL encrypts transmitted data, making it difficult for intercepted data to be understood or tampered with. Encryption secures data transfer between the sender and receiver.
- II. Authentication: SSL uses digital certificates to authenticate the identity of the server. This ensures that users are securely connected to the server and prevents man-in-the-middle attacks.
- III. Integrity Check: SSL ensures the integrity of transmitted data, verifying that data hasn't been altered during transmission and maintaining data integrity.

SSL is commonly used for communication between web browsers and servers, particularly in platforms requiring secure data transfer such as online shopping sites and banking applications.

While SSL is the older version, Transport Layer Security (TLS) is more commonly used today. However, the term "SSL" is often still used to refer to all these security protocols, including older versions of SSL, collectively [7].

IV. METHODOLOGY

A. System Architecture

The system architecture of the secured TCP chat room application is designed to facilitate efficient and secure communication between clients and the server. The architecture consists of two main components: the client-side application and the server-side application. Each component plays a crucial role in ensuring the smooth operation of the chat room system.

- I. Client-side Application: The client-side application is responsible for providing a user interface through which users can interact with the chat room system.
- II. Server-side Application: The server-side application serves as the central component responsible for managing client connections, processing messages, and maintaining the overall state of the chat room system.

By following these recommended methodologies for system architecture, security, and scalability, the secured TCP chat room application can be developed with a focus on robustness, reliability, and user privacy [5,6].

B. Ip Filtering

IP filtering is a network security method used to regulate access based on IP addresses. It involves allowing or blocking traffic based on source or destination IP addresses. Ingress filtering controls incoming traffic,

while egress filtering regulates outgoing traffic. This approach enables administrators to enforce security policies tailored to their organization's needs, enhancing overall network security [8].

- I. Blacklist: A blacklist is a compilation of IP addresses, domains, or email addresses that are considered undesirable or potentially harmful. These lists are frequently used in various security measures, such as email filtering, website access control, or network security configurations. When an IP address or domain is added to a blacklist, it signifies that communication originating from or directed towards that source should be blocked or restricted. Blacklists are instrumental in combating spam emails, blocking malicious websites, or preventing access to known sources of malware. Examples: Known sources of malware, spam-sending IPs, or attacker IP addresses.
- II. Whitelist: In contrast to blacklists, whitelists consist of approved or trusted IP addresses, domains, or email addresses. These lists delineate the entities or sources deemed permissible and deserving of unrestricted access to a network, service, or system. Whitelists serve as a means of enforcing strict access control, ensuring that only authorized entities are permitted entry while barring all others. Organizations commonly employ whitelists to restrict access to sensitive resources, limit communication to verified contacts, or safeguard against unauthorized intrusions. By exclusively permitting communication from pre-approved sources, whitelists bolster security measures and mitigate the risk of unauthorized access or malicious activity. Examples: IP addresses of specific devices within a corporate network, employees' home IP addresses, or partner companies' IP addresses [8].

C. AES Algorithm

In today's digital world, data security and privacy are of utmost importance. To achieve this, encryption and decryption processes using the AES algorithm have been implemented in both server and client code [2,9].

- I. Encryption: encrypt_message: Encrypts messages using the AES algorithm.
- II. Decryption: decrypt_message: Decrypts encrypted messages back to their original form [9].

V. DISCUSSION

The discussion aptly highlights the dual nature of communication in the internet age, acknowledging its rapidity and widespread reach alongside the looming security concerns. It rightly emphasizes the necessity of addressing these security gaps, especially in chat applications where user data vulnerability is a significant issue. The proposal to develop a secure TCP chat room is a proactive step towards mitigating these concerns. By leveraging the TCP protocol,

the project aims to fortify data integrity and ensure the security of connections, thereby laying a robust foundation for secure communication. Additionally, integrating message encryption protocols like SSL/TLS underscores a commitment to safeguarding user privacy and confidentiality. The research-backed assertion of the feasibility of a secure TCP chat room instills confidence in the project's objectives. It indicates a thorough understanding of the technical aspects involved and underscores the potential for meaningful contributions towards enhancing internet security. Furthermore, the emphasis on user interface usability is noteworthy. While security is paramount, ensuring a seamless and intuitive experience for users is equally important for widespread adoption. By prioritizing user comfort and understanding, the project aims to bridge the gap between security and user experience effectively. In summary, the proposed project holds promise in addressing the pressing need for enhanced security in internet communication. By combining the robustness of the TCP protocol with advanced encryption mechanisms, it seeks to provide a secure communication environment while preserving user privacy—a commendable endeavor in today's digital landscape [6,7].

VI. RESULTS

Results evaluate the effectiveness of key components such as AES encryption, the user-friendly design of the main screen, the security of user messages and records, the message search feature, server performance, and the SSL certificate.

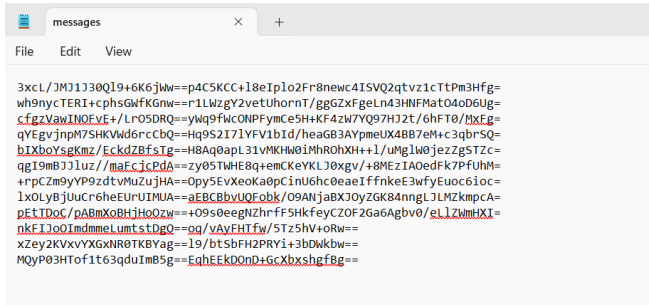


Figure 4. AES Encryption

The successful implementation of the AES encryption algorithm ensures the security of user messages, enhancing data privacy and protecting sensitive information from unauthorized access (Figure 4).

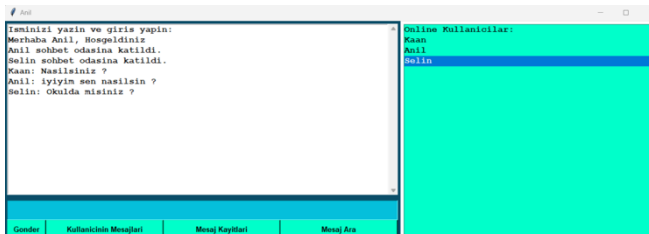


Figure 5. Main Screen

The user-friendly design of the main screen facilitates user interaction, improving the overall user experience (Figure 5).

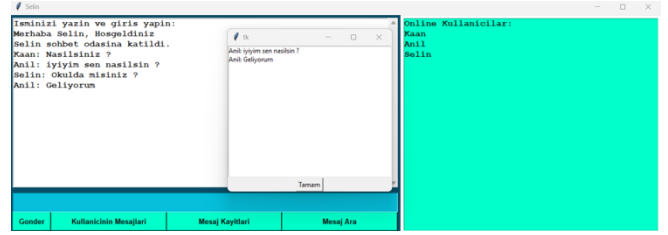


Figure 6. User Messages

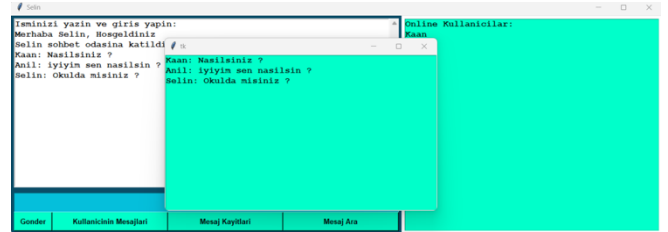


Figure 7. Message Records

User messages (Figure 6) and records (Figure 7) are safeguarded through AES encryption, ensuring the confidentiality and security of data.

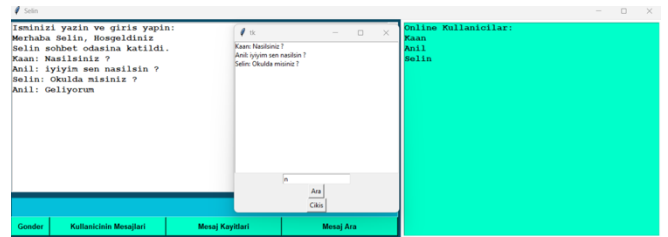


Figure 8. Message Search

The message search feature enables users to efficiently retrieve past messages, enhancing usability and facilitating communication (Figure 8).

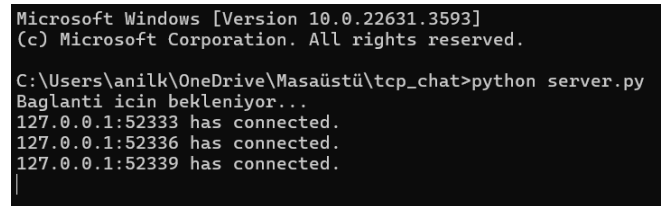


Figure 9. Server

Server performance ensures high efficiency and stability, supporting uninterrupted communication within the system (Figure 9).

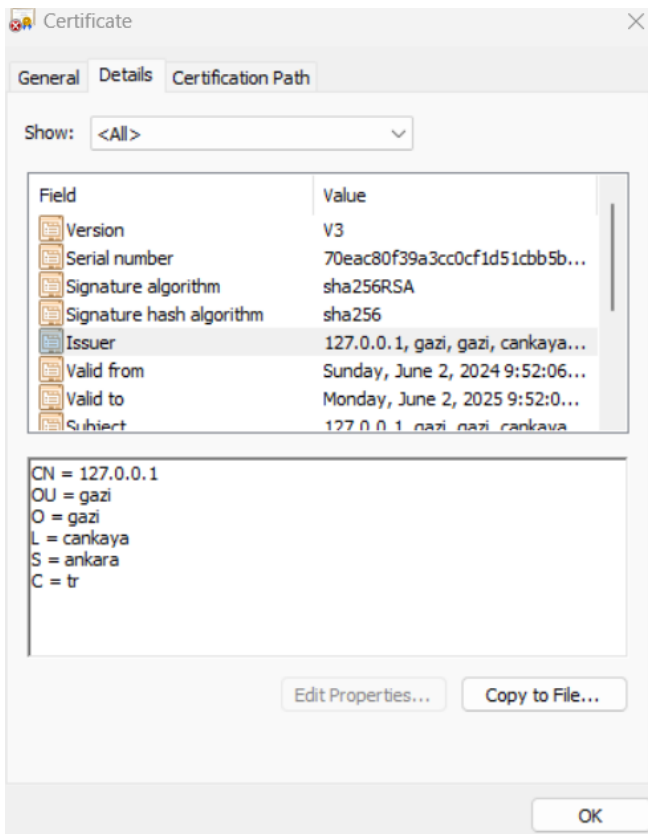


Figure 10. SSL Certificate

The SSL certificate secures connections, enhancing data security during communication (Figure 10).

These components collectively contribute to the successful completion of the project and the provision of a secure messaging platform. These results demonstrate the system's success in facilitating secure and seamless communication for users.

VII. CONCLUSION

In conclusion, the internet age has brought unparalleled connectivity but has also raised significant security concerns, particularly in chat applications where user data vulnerability is a pressing issue. Recognizing this, the proposal to develop a secure TCP chat room is a proactive step towards addressing these concerns. By leveraging the robustness of the TCP protocol and integrating advanced encryption

mechanisms like SSL/TLS, the project aims to fortify data integrity, establish secure connections, and safeguard user privacy. The feasibility of this project is supported by thorough research, indicating a nuanced understanding of the technical challenges involved. Furthermore, the emphasis on user interface design underscores a commitment to balancing security with usability, ensuring a seamless and intuitive experience for users. In essence, the proposed project holds promise in enhancing internet security by providing a secure communication environment that prioritizes user privacy—a crucial endeavor in today's digital landscape. The project successfully demonstrates a secure TCP chat room using Python, with effective encryption. The goals of secure communication and user privacy are achieved. Future improvements could include adding features like file transfer, implementing more sophisticated user management, enhancing the user interface, and conducting further security audits.

ACKNOWLEDGMENT

We would like to sincerely thank our teacher Feyza Yıldırım Okay, who guided us and supported us throughout this project.

REFERENCES

- [1] Kagane, A. (n.d.). *Chatroom using python project report*. Scribd.
- [2] Pasumarty, R. (2021, November). Secure chatroom application using advanced encryption standard algorithm. In *2021 International Conference on Disruptive Technologies for Multi-Disciplinary Research and Applications (CENTCON)* (Vol. 1, pp. 344-346). IEEE.
- [3] Shah, A., Servar, M. G., & Tomer, M. U. (2022). Realtime Chat Application using Client-Server Architecture. *International Journal for Research in Applied Science and Engineering Technology*, 10(5), 2575-2578
- [4] Jenks, C. (2014). *Social interaction in second language chat rooms*. Edinburgh University Press
- [5] Bamane, A., Bhoyar, P., Dugar, A., & Antony, L. (2012). Enhanced Chat Application. *Global Journal of Computer Science and Technology Network, Web & Security*, 12(11), 1-7.
- [6] Postel, J. (1981). Rfc0793: Transmission control protocol.
- [7] Satapathy, A., & Livingston, J. (2016). A Comprehensive Survey on SSL/TLS and their Vulnerabilities. *International Journal of Computer Applications*, 153(5), 31-38.
- [8] Varadharajan, V. (2010). Internet filtering-issues and challenges. *IEEE Security & Privacy*, 8(4), 62-65.
- [9] Bardis, N. G., & Ntaikos, K. (2008, October). Design of a secure chat application based on AES cryptographic algorithm and key management. In *WSEAS International Conference. Proceedings. Mathematics and Computers in Science and Engineering* (No. 10).