Jayden Cruz

Dr. Tisha Gaines

Computer Networks

21 April 2025

A Simple TCP-Based Chat Application: Purpose, Usage, and Networking Integration

For my CSC 2030 final project, I built a simple TCP-based chat application using Python. I had some prior experience with Tkinter from my internship at the Boys and Girls Club of Middle Tennessee, where I helped design a GUI for a device distribution system that tracked and managed tech inventory for students. That experience gave me a good foundation in event-driven programming and helped me better understand how to create intuitive interfaces for real users. Each part of the chat project includes a graphical interface using Tkinter, allowing users to send and receive real-time text messages. The project directly applies networking concepts we studied in class, especially socket programming and the client-server model.

TCP (Transmission Control Protocol) is a reliable, connection-based communication protocol that plays a foundational role in ensuring seamless data transmission between devices on a network. It establishes a connection, monitors the successful delivery of packets, and resends any that are lost in transit. This guarantees that the data reaches its destination correctly and in the right order. My application uses TCP sockets to allow multiple clients to connect to a server, which then relays messages to all connected users in real-time. This setup mirrors how real-world messaging systems like WhatsApp or Slack maintain consistent communication between users and servers.

TCP/IP networking originated in the 1970s as part of the ARPANET project, which was an experimental research network funded by the U.S. Department of Defense. It was designed to provide robust communication between computers in case of partial network failure. TCP became the protocol responsible for reliable end-to-end communication, while IP managed the routing of data between hosts. In 1981, TCP was officially standardized in RFC 793 by Jon Postel, making it a formal protocol for the growing internet. Today, TCP is still a cornerstone of internet communications, used for everything from web browsing and file transfers to streaming and, as demonstrated in my project, real-time chat systems.

The program's purpose is to demonstrate how TCP ensures reliable communication in a practical way. Users launch the server, then connect one or more clients on the same network. Messages are exchanged and displayed in real-time. The project helped me understand the TCP handshake, data flow, and connection handling.
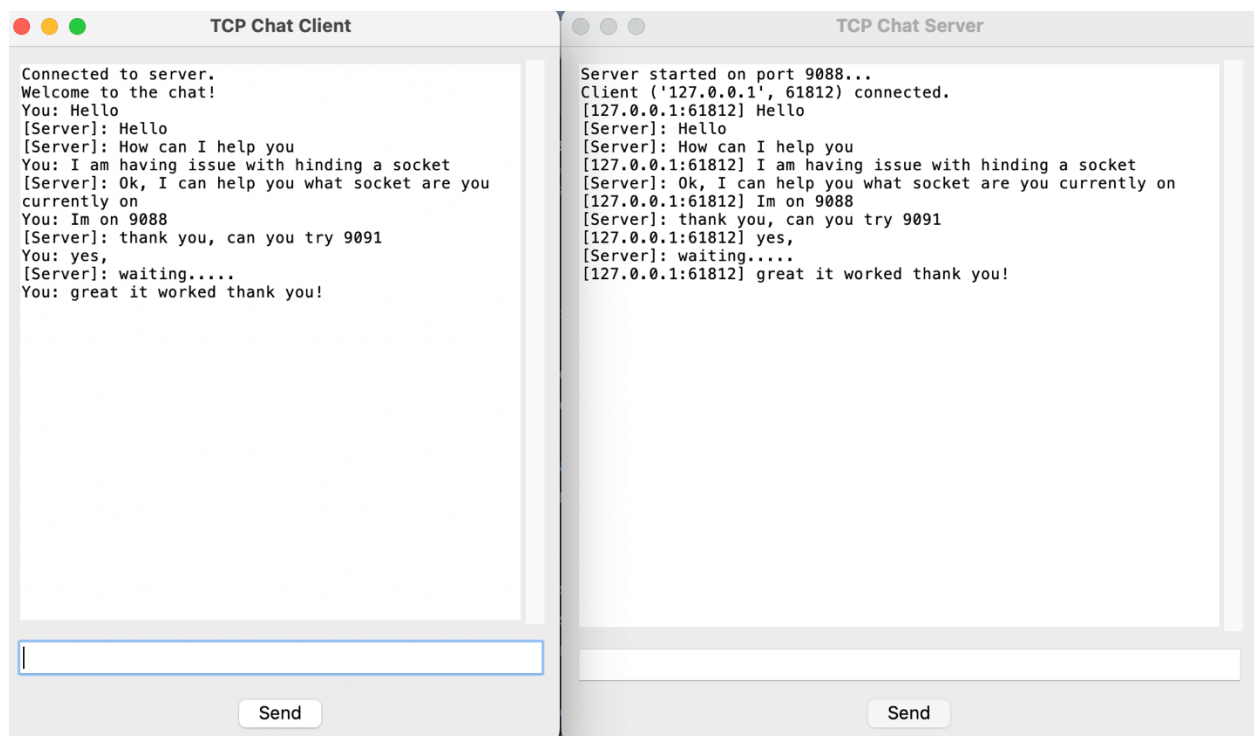
Figure 1: Screenshot of the TCP chat client and server GUI in use.

In the future, this project could be enhanced with features like user authentication, display names, message timestamps, and secure encrypted messaging using TLS. It could also be deployed to a remote server or cloud platform, enabling users across different networks to communicate. These additions would introduce more advanced networking topics such as security protocols, session handling, and cloud deployment strategies.

I chose this project because it allowed me to turn abstract networking ideas into something tangible and functional. It was engaging to build an interactive application where I could immediately see the results of my code and network logic. Working on it also gave me hands-on experience with how TCP actually works behind the scenes, deepening my overall understanding of network communication.

Works Cited

Kurose, James F., and Keith W. Ross. *Computer Networking: A Top-Down Approach*. 8th ed.,

Pearson, 2021, https://www.pearson.com/en-us/subject-catalog/p/computer-

networking/P200000003334/9780135928615?srsltid=AfmBOopXdZDF3lZG6cvlLkNpR

feI37XKO5BPYs1CnNpMHcNwm8324c33

Postel, Jon. "RFC 793 - Transmission Control Protocol." *Internet Engineering Task Force*, 1981,

https://tools.ietf.org/html/rfc793.

Stevens, W. Richard. *TCP/IP Illustrated, Volume 1: The Protocols*. Addison-Wesley, 2011,

https://www.r-5.org/files/books/computers/internals/net/Richard_Stevens-TCP-

IP_Illustrated-EN.pdf