- **How many packets were exchanged during TCP communication?**

A total of 7 packets were exchanged during the TCP communication. The first three packets handled the connection setup: SYN, SYN-ACK, and ACK. Next, two packets were used for data transfer. One from the client to the server and one from the server to the client. The final two packets were used to close the connection, involving a FIN and an ACK.

- **How many packets were exchanged during UDP communication?**

UDP communication involved just 2 packets. One sent from the client to the server and a response from the server back to the client. Unlike TCP, UDP does not require a connection to be established or terminated beforehand.

- **What differences do you observe in packet size and protocol overhead?**

TCP packets are generally larger than UDP packets because they contain extra header fields such as sequence numbers, acknowledgment numbers, and control flags. These additions increase overhead but are essential for reliable communication. In contrast, UDP uses a simpler header with minimal overhead, making it lighter. Since TCP is connection-oriented, it demands more processing power and bandwidth, whereas UDP is connectionless and better suited for quick, straightforward data transfers.

- **What happens if the UDP client sends a message to a server that's not running?**

When a UDP client sends a message to a server that isn’t running, the packet is still transmitted, but no reply is received. The client doesn’t receive any error message and might wait endlessly for a response. This occurs because UDP doesn't establish a connection or check if the destination is available before sending data.

- **Does TCP retransmit any segments? Why or why not?**

TCP did not have to resend any segments since all packets were delivered successfully. However, TCP includes built-in mechanisms to retransmit data if any packets are lost or if acknowledgments are not received within a certain timeout. This retransmission capability is a key part of TCP’s reliability.

Socket Programming with TCP and UDP – Report Summary

1. Code Behavior

In the TCP version, the server starts first and waits for the client to initiate a connection. After the connection is established, the client sends a lowercase message like "hello world." The server processes the message by converting it to uppercase and sends the response back. The client then displays the server's reply and closes the connection. Because TCP is connection-oriented, it requires a handshake before any data is exchanged and properly closes the connection afterward.

In contrast, the UDP version is more straightforward. The client sends a message directly to the server without establishing a connection. The server receives it, converts the text to uppercase, and replies. The client then prints the response. Since UDP doesn't involve connection setup or termination, it's faster and more lightweight, but it does not ensure the message will be delivered.

2. Key Differences between TCP and UDP

TCP and UDP are both transport layer protocols but operate in different ways:

- TCP is connection-oriented, meaning it establishes a connection before data transfer begins. It ensures reliable delivery, maintains the correct order of data, and uses mechanisms like handshakes, acknowledgments, and retransmissions. This makes TCP ideal for applications where data integrity matters, such as web browsing and file transfers.

- UDP is connectionless, sending data without first checking if the receiver is available or if the data arrives successfully. It has minimal overhead and delivers faster performance, but without reliability guarantees. UDP is commonly used in applications like online gaming, video streaming, and voice communication, where speed is more critical than perfect accuracy.

3. Wireshark Findings

- **TCP Communication: A total of 7 packets were involved:**

  - 3 packets for establishing the connection (SYN, SYN-ACK, ACK)
  - 2 packets for exchanging data (one from client to server, one from server to client)
  - 2 packets for closing the connection (FIN, ACK)

    TCP packets were larger due to additional header information like sequence and acknowledgment numbers. In this test, all packets were delivered successfully, so no retransmissions were necessary.

- **UDP Communication:**

  Only 2 packets were exchanged—one sent by the client and one response from the server. UDP packets had smaller headers with minimal overhead. If the server isn't running, the client still sends the packet but gets no reply and receives no error notification.

Krittapat Chumwattana 6581225