Authors

Areeyoka John(adj69), Joelany Morales(jlm765), Christian Espinal(cje84)

Implementation Summary

Handshake logic
The server establishes the connection

Data transfer logic
Client splits messages into chunks and assigns sequence numbers
Sends data along with the sequence numbers and waits for the matching ACK from the server
Retries until it receives its ACK
Server receives data and checks to make sure the sequence and the expected sequence match
Prints the data, simulates a delay, sends back an ACK, and increments the expected sequence
If its out of order it will resend the last correct ACK and decrement its expected ACK

Teardown logic
For teardown, client sends message FIN telling the server to close the connection
Server sends back FIN-ACK
Client receives final ACK and exits

Capture Analysis

project3_handshake.pcap
In the handshake.pcap you can see the packets that are exchanged between the server and client to establish a connection.

No 1: The client (127.0.0.1, 64141) sends a SYN to the server (127.0.0.1, 30045)
The first byte of the Data section of the UDP packet is 1 which is the code that identifies it as a SYN

No 2: The server then replies back by sending a SYN-ACK
The first byte of the Data section of the UDP packet is 2 which is the code that identifies it as a SYN-ACK

No 3: The client then sends an ACK to the server
The first byte of the Data section of the UDP packet is 3 which is the code that identifies it is a ACK

project3_data.pcap
In the data.pcap you can see the data packets being transferred with the WireGuard protocol. The retransmissions of the data can be seen when the ACK doesn't arrive from the server within 0.5 seconds. All the ACKs are sent using the UDP protocol with the first byte of the DATA section in the UDP packet being 5 which is the code that identifies it as a DATA ACK.

Project3_teardown.pcap
In the teardown.pcap you see the FIN udp packet being sent with data code code being in the first byte which is 6. It is retransmitted a couple of times before the server finally sends back the FIN ACK udp packet, with the data code being 7 which is the first byte of the udp packet.

Discussion

Ack Delay Triggers Retransmission:

Client employs timeout-based retransmissions of 0.5 seconds
If the client doesn't receive an ACK for the packet in 0.5 seconds, the client assumes the packet or ACK was lost
It will then retransmit the packet

RUDP Ensuring Reliable Delivery:

- Sequence numbers: every data packet and ACK share a sequence number to ensure the correct amount of packets are transmitted and makes sure the packets are also in order
- Retransmission through timeout in case of missing packets