Air Drop

By A'Myah Temple

Project Overview

Implement UDP-based file transfer between a client and server.

Client reads file in 1000-bytes chunks and assembles custom packets.

Server receives packets, validates type, saves data and acknowledges

Why UDP?

UDP is connectionless and does not delivery.

It’s faster and simpler than TCP

Requires manual handling or reliability, such as ACK responses.

Client Program Flow

1. Open file using fopen().
2. Use fread() to read 1000 bytes into a buffer.
3. Create packet: 4-byte Data Type + 1000 bytes of file data.
4. Send packet to server
5. Wait for ACK before sending next packet.
   When file ends, send ‘Done’ packet

Packet Format

First 4 bytes: DataType (‘Data’ or ‘Done’)

Next 1000 bytes file data

Done indicaties end of transmission

Reading File Chunks(fread)

fread(buffer,1,1000,pFile) reads up to 1000 bytes.

If fread returns < 1000, EOF is reached.

Loop stops when 0 bytes are read.

Remaining bytes in last packets are still sent.

Assembling a Packet

Create array of 1004 bytes

Copy 4-character DataType into Packet[0,-3]

Copy buffer data into packet[4,1003]

Use a loop: Packet [i+4] = buffer[i]

Send entire 1004 bytes packet to server

Client to Server Communication

After sending each packet, client waits for ACK via UDP receive.

ACK contains first 4 bytes of the received packet

Ensure server actually received each packet

Server Responsibilities

Receive packets via recvfrom()

Extra the first 4 bytes to determine packet type

If Data - write 1000 bytes to output file

Send ACK containing DataType

If Done - stop writing, send Done, and close file

Transmission Completion

Client sends last packet: DataType=”Done”

Server acknowledges with “Done”

Both sides close sockets and finish program

Video Demo

https://youtu.be/Q9saXN4YHEE