

# Client Server Application

## Thomas O'Brien

An exercise in TCP communications  
via Unix programming

### Server

- Establishes ownership over an available port on some UNIX based machine
- Listens for an incoming TCP request to make a 3 way handshake
- Decodes incoming files and stores them at a given existing directory

### Client

- Attempts to make a simulated TCP connection with a server
  - Utilizes a TCPHeader struct
- Sends encoded files from the client side to the server side
- Returns whether or not the file was sent over successfully

### Technologies

- Golang is a powerful systems programming language gaining popularity in modern tech stacks
- Unix programming libraries are utilized for lower level programming

### Justification

- To develop an in-depth understanding of the TCP/IP stack
- Relevant to Site Reliability/DevOps Engineering



```
ThomasO'Brien@MacBook-Pro-3:~/client-server-application]: go run unix-server.go
Let's listen
```

```
ThomasO'Brien@MacBook-Pro-3:~/client-server-application]: go run unix-client.go test.txt
```

UNIX®