Lab 1 - TCP File transfer

BI12-149 - Kieu Huy Hai

March 2024

1 Protocol

The server and client will exchange information through the TCP socket:

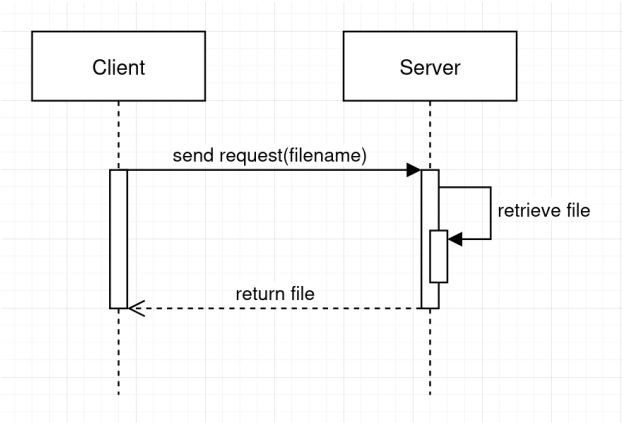

Figure 1: Protocol demonstration

The protocol includes these steps:

- The client will request the file name it wants to download from the server.
- The server acknowledges the request and sends the file to the client via socket.
- The client receives the file.

2 System Architecture

The system consists of 2 entities - client and server, with their respective sockets connected to each other over the network:

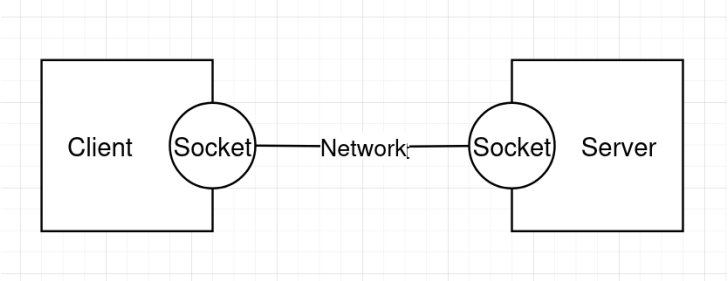

Figure 2: System Architecture

3 File Transfer

The server open and read the file content with `open(filepath,O_RDONLY)`. After that, it uses `write(socket_fd,buffer,length_of_buffer)` to send the file content to the client:

```
int file_fd = open(filepath, O_RDONLY);
if (file_fd < 0)
{
    perror("Error opening file for reading");
    exit(EXIT_FAILURE);
}
ssize_t bytes_read;
char buf[1024];
while ((bytes_read = read(file_fd, buf, sizeof(buf))) > 0)
{
    ssize_t bytes_written = write(fd, buf, bytes_read);
    if (bytes_written < 0)
    {
        perror("Error writing to socket");
        close(file_fd);
        exit(EXIT_FAILURE);
    }
}
```

```
if (bytes_read < 0)
{
    perror("Error reading from file");
    close(file_fd);
    exit(EXIT_FAILURE);
}
close(file_fd);
```

After the server sends the file content, the client reads the content from the socket using `read(socket_fd,buffer,length_of_buffer)` and store it in the location specified by the user:

```
int file_fd;
char buf[1024];
// Open file for writing, create if not exists, truncate if exists
file_fd = open(savelocation, O_WRONLY | O_CREAT | O_TRUNC, 0666);
if (file_fd == -1)
{
    perror("Error opening file\n");
    exit(EXIT_FAILURE);
}
ssize_t bytes_read;
while ((bytes_read = read(fd, buf, sizeof(buf))) > 0)
{
    ssize_t bytes_written = write(file_fd, buf, bytes_read);
    if (bytes_written < 0)
    {
        perror("Error writing to file\n");
        exit(EXIT_FAILURE);
    }
}
if (bytes_read < 0)
{
    perror("Error reading from socket\n");
    exit(EXIT_FAILURE);
}
printf("Received file!\n");
close(file_fd);
```