Homework 2

Oct 31, 2018 (Wed)

Eunggu Yun

1. Introduction

There are two programs: TCPServer and TCPClient. These two programs are a server program and a client program which are using TCP protocol to communicate each other. TCPServer program is a server program that waiting on the specified port for just one client and send the specified file to that client. TCPClient program is a client program that connects to server using the specified IP address and the port. When these two programs get contact to each other, they transfer **the file name**, **the size of the file**, and **the file data** using one TCP socket. After all transmissions are ended, both programs are terminated.

These programs compare the transmitted data size and the original data size. If two values are different, they transmit the remaining data that have not been transmitted.

2. Development Environment

1) Platform (OS): Ubuntu 16.04.1 LTS (GNU/Linux 4.4.0-31-generic x86_64)

2) Compiler: gcc (Ubuntu 5.4.0-6ubuntu1~16.04.10) 5.4.0 20160609

This is the SKKU server's development environment. (swye.skku.edu)

3. How to build and run

1) Build

To build the server and client program:

```
$ make all
```

To build the server program only:

```
$ make server
```

To build the client program only:

```
$ make client
```

To clean up output files:

```
$ make clean
```

2) Run

To run the server program,

```
$ ./TCPServer [port_number] [file_name]
```

For example,

```
$ ./TCPServer 4000 sample.pdf
```

To run the client program,

```
$ ./TCPClient [server_ip] [port_number]
```

For example,

```
$ ./TCPClient 127.0.0.1 4000
```

4. Project Structure

```
Project Folder
 ├ Makefile
 ├ TCPClient.c
 └ TCPServer.c
```

1) Makefile

File for GNU Make. It gives the knowledge of how to build the TCPServer program and the TCPClient program to GCC compiler.

2) TCPClient.c

A source code of TCPClient. It has no dependency on TCPServer.

3) TCPServer.c

A source code of TCPServer. It has no dependency on TCPClient.

5. Implementation Details

TCPServer program gets two arguments: port number and file name. When it starts, it reads the file and stores it at the memory. And then, it creates a TCP socket, binds the socket, and listens for client connection. When a client makes a connection, it accepts a client and opens new TCP socket for data transmission. Because the server no longer needs to wait for another client, it closes the listen socket. After the socket for data transmission created, it sends three data: the file name, the size of the file, and the file data. It terminates when it sends all data to client.

TCPClient program gets two arguments: server IP address and port number. When it starts, it creates a TCP socket and connects to the server. When the connection is accepted by the server, it first receives the file name and opens the file. And then, it receives the file size and allocates the memory to store the file data. Finally, it receives the file data and stores it to the memory. After the transmission ends, it writes the file data to the file and terminates.

The protocol for data transmission is defined as below. It allocates 128 bytes for the file name, 13 bytes for the file size, and the others for the file data.

File Name	File Size	File Data
128 Bytes	13 Bytes	Variable Length

Because the send function and recv function in C socket library don't guarantee the whole data we want to transmit will be transmitted, we should check whether the whole data are transmitted or not. So, TCPServer program and TCPClient program always check the size of data they send and receive and re-transmit the remaining data if the data transmission is incomplete.