PETRUCCI Titouan - 2210009

Creating a Basic File Transfer Protocol

using the C programming language

University of Sciences and Technology of Hanoi

Contents

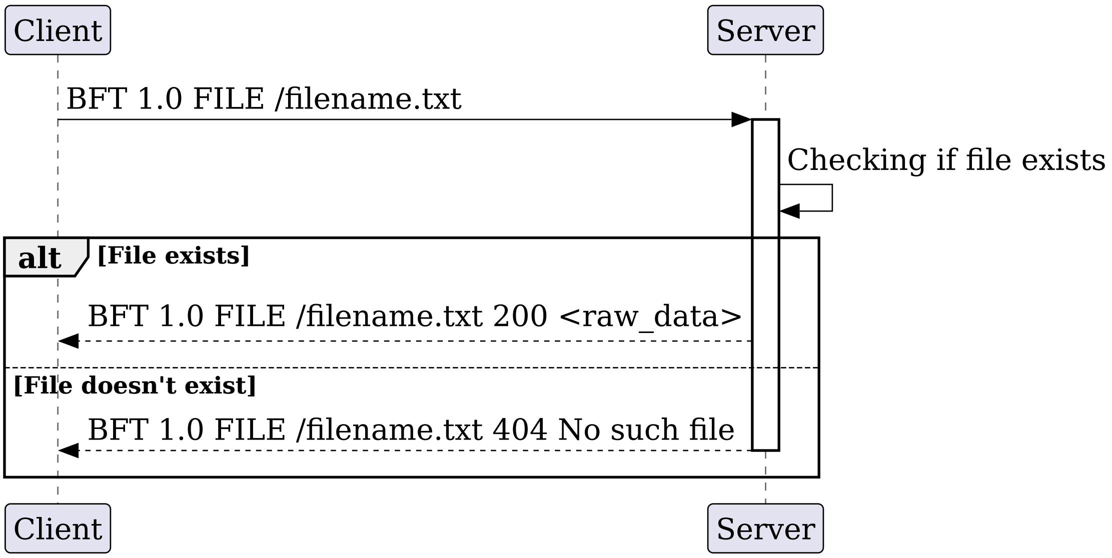
I/ Protocol

To create the protocol, we need to establish what are the real needs

I.1/ Requirements

- Clients would be able to download files from the server
- Clients can download multiple files in a row
- Protocol need to support upgrade (if we want to upgrade it)

I.2/ Diagram


Client
Server
BFT 1.0 FILE /filename.txt
Checking if file exists
alt
[File exists]
BFT 1.0 FILE /filename.txt 200 <raw_data>
[File doesn't exist]
BFT 1.0 FILE /filename.txt 404 No such file
Client
Server

Error code 200 for success and 404 for non-existant data are taken from the HTTP protocol, because it works well and everyone knows them.

II/ Basic Implementation

II.1/ Server

```c
#include <sys/socket.h>
#include <stdlib.h>
#include <stdio.h>
#include <unistd.h>
#include <arpa/inet.h>
#include <signal.h>


void handle_request(int connfd);

char **FILES;



int main(int argc, char *argv[]) {
  int sfd;
  struct sockaddr_in hostinfo = {
    .sin_family = AF_INET,
    .sin_addr.s_addr = INADDR_ANY
  };
```

```
if (argc < 2) {
       perror("No port specified\n");
       exit(EXIT_FAILURE);
}

int port = atoi(argv[1]);

if (port == 0) {
  perror("Bad Port Specified\n");
  exit(EXIT_FAILURE);
}

hostinfo.sin_port = htons(port);

if ((sfd = socket(AF_INET, SOCK_STREAM, 0)) == 0) {
  perror("Unable to init socket\n");
  exit(EXIT_FAILURE);
}

if (bind(sfd, (struct sockaddr *) &hostinfo, sizeof (hostinfo)) < 0) {
  perror("Port already in use or reserved\n");
  exit(EXIT_FAILURE);
}

if (listen(sfd, 3) < 0) {
  perror("Can't listen on port");
  exit(EXIT_FAILURE);
}

printf("Listening on port: %d!!\n", port);

while (1) {
  if ((newconn = accept(sfd, (struct sockaddr *) &hostinfo, sizeof (hostinfo)))
```

II.2/ Client