MY FTP

FILE TRANSFER PROTOCOL SERVER

MY FTP

Preliminaries

binary name: myftp
language: C/C++
compilation: via Makefile, including re, clean and fclean rules

All the standard C library is authorized except **recv**, **send** and **all system calls that make a socket non-blocking**.

The goal of this project is to create a **FTP server**.

You MUST implement the given protocol (bonuses must be RFC959 compliant).
You MUST implement request sequences that are explained in the protocol (more information could be find in RFC959).

The network communication will be achieved through the use of TCP sockets.

{ EPITECH }

Server

```
▽                              Terminal                        –  +  X
~/MANDATORY MODULE CODE HERE> ./myftp -help
USAGE: ./myftp port path
    port is the port number on which the server socket listens
    path is the path to the home directory for the Anonymous user
```

The server MUST be able to handle several clients at the same time by using **poll** for command management and **fork** for data transfer.

> A good use of **poll** is expected for both reading and writing on sockets.
> Any bad use of **poll** would cause point loss.

The server MUST have an authentication with an *Anonymous* account and an empty password. Data transfers MUST use *active* or *passive* mode, as explained in the bootstrap.

> Use actual ftp clients to see if your code holds to the rfc

> If your program works with a real ftp client it does not mean it is strictly RFC959 compliant. Even if it works on an ftp client it might not pass the automated tests, therefore read the RFC carefully and test accordingly.

You can install a FTP server such as **proftpd** or **vsftpd** on your machine to understand how it works.

{ EPITECH }

Bonuses

Here is a list of possible bonuses;

- ✓ an FTP client with the following commands: ls, pwd, cd, get and put
- ✓ lcd + lpwd + lls
- ✓ mget + mput
- ✓ rights management
- ✓ a different home directory for each use account
- ✓ handle "bin" and "asc" modes
- ✓ automatic completion for get
- ✓ automatic completion for put

and as a super bonus:

- ✓ total compliance with the RFC. (spoiler alert: it will take a while…)

{ EPITECH }