

# Socket Programming

2021011823 张国威

## 一、MyFTP 说明

我的 FTP 运行环境为 M1 芯片的 MacOS 系统，实现了实验要求的所有 16 个命令，server 端在 autograde 评测中得到了 40 分，用同学的电脑上的 client 测试无误；client 端在本机和助教提供的服务器上评测无误，且对于 STOR 和 RETR 文件不存在的情况具备较好的异常处理能力。

下面对实现的命令进行说明：

- 1、USER, PASS 必须全部通过校验后才可输入其他命令，否则显示 550 Invalid Command 错误，必须重新输入 USER, PASS 才可继续。
- 2、如果检查到用户输入路径中有 .. 则返回 550 Permission denied，因此若要回到上级目录则需要输入其绝对路径。
- 3、PORT 命令要求端口在 20000 与 65535 之间，格式类似 PORT 127,0,0,1,123,23，否则无法连接。
- 3、客户端控制台输出的消息中产生自服务器和客户端的均有，服务器的带返回码，客户端的不带。
- 4、PASV 命令由服务器设置 ip 地址为 get\_ip() 函数返回值，port 在 20000 和 65535 之间随机取值。
- 5、RMD 命令进行 rm -r 系统调用，可删除非空文件夹，但删除没有访问权限的文件及文件夹需要服务器端确认。
- 6、MKD 命令调用 mkdir 函数，若所创建目录已存在则会提示错误。
- 7、RNTD 命令调用 rename 函数，可重命名文件及文件夹。
- 8、LIST 命令进行 ls -l 系统调用，当目录不存在时会显示错误。
- 9、RETR 和 STOR 命令调用 fopen 函数，读取文件时，若找不到文件会提示错误；写入文件时，若找不到文件则会创建一个新的文件。

## 二、心得体会

本次实验难度非常大，从开始写到完成总共花了一周左右的时间，每天从起床到睡觉，除了上课和吃饭基本就是在 code。在没有接触过网络编程的情况下，一上来就要面对 socket programming 的各种概念，我感到完全无从下手。于是我在 CSDN 等平台上搜集各种介绍各种 api 具体信息的文章，并仔细阅读文档中参考资料的内容，看不懂的地方就问舍友和 chatGPT，终于搭出来了基本框架。Debug 阶段，错误主要集中在 send 与 recv 的对应以及路径处理上：为了解决有时 recv 不到消息而卡死，或者 send 的消息没有被正确的函数接收，我不得不反复调整 client 和 server 端的代码，尽量保证对应关系；对于路径处理，我开始没有弄清根目录的含义，导致实现结果与实际要求不符，因此不得不对各个函数的相关部分进行大幅度的改动。

通过本次实验，我对网络这一陌生领域的基础知识有了一定了解，并入门了网络编程，也由亲身实践体会到了网络这个领域情况的复杂性，以及安全问题的普遍性。

