

# FTP 实验报告

吴寅秋 2017013563

## 1、FTP概述

FTP是基于TCP实现的文件传输协议，具体定义在RFC959，FTP协议具有两个端口，一个控制端口（通常是21端口）用来传输命令和返回信息，另外一个端口用来传输文件。同时FTP具有主动和被动两种模式，主动模式即PORT模式，是客户端建立并告诉服务端文件传输端口，服务端进行连接。而被动模式即PASV模式，是服务端建立端口并告诉客户端。客户端在这个端口上与服务端进行文件传输。

## 2、FTP服务端实现

实现的命令有USER, PASS, RETR, STOR, QUIT, SYST, TYPE, PORT, PASV, MKD, CWD, PWD, LIST, RMD, RNFR, RNTD。以上命令的实现位置均在代码中有注释

主要流程如下：

创建一个 Socket → 监听 → 接收连接的请求 → 使用fork () 新建进程 → 接受并处理命令 → ctrl+c关闭Socket

运行方式：

在server目录下“make”，之后在终端输入“./server -port(可选) -root(可选)”。如果不输入“-port”“-root”默认为“/tmp”目录和21端口。

最终运行autograde.py能得到满分40分

```
/usr/local/bin/python3.7 "/Users/wuyinqiu/作业/计算机网络/For Student/ftp/server/autograde.py"
Your credit is 40

Process finished with exit code 0
```

## 3、FTP客户端实现

与服务端一样实现了USER, PASS, RETR, STOR, QUIT, SYST, TYPE, PORT, PASV, MKD, CWD, PWD, LIST, RMD, RNFR, RNTD等命令。以上命令的实现位置均在代码中有注释。

主要流程如下：

创建一个 Socket → 与服务器连接 → 命令输入和应答 → QUIT命令关闭 Socket

运行方式：

客户端:在client目录下“make”，之后在终端输入“./client host port”，如：“./client 127.0.0.1 21”

## 4、实验感悟

通过本次实验使用socket编程，将课上学习到的理论通过实践转化为成果非常有成就感。自己动手实现可以让学习到的知识能不浮于表面，感觉自己对UDP和TCP协议的理解更深入了一层。包括TCP和UDP客户端和服务端的流程，以及FTP客户端服务端的流程。还有了解以及能够使用协议内部的数据结构，如sockaddr\_in等。