

# 实验报告

王华清 软件 22 2012013317

## 实验简介

本次实验利用 Socket 实现了 UDPDemo 的修改以及 FTP 文件传输，两个工程分别基于 UDP 以及 TCP。

## 实验环境

- 操作系统: Windows7 64 位
- IDE: eclipse 4.4.1
- 程序语言: Java

## 功能实现

- 修改 UDPDemo

程序修改后实现了以下功能：客户端与服务器连接后，客户端向服务器以 UPD 方式发送 0 到 50。服务器每次收到数字后，在数字前加上已接受次数，找到客户端的 IP 和端口，并返回给客户端。

- FTP 文件传输

TCPClient 实现了以下功能：

1. 连接 FTP 服务器，向服务器发送指令，接受服务器的返回值，与服务器进行交互。
2. 容错性能良好，对错误返回不同语句，不会错误中断程序。

TCPServer 实现了以下功能：

1. 监听 FTP 客户端的连接，接受客户端的指令，向客户端发送返回值，与客户端进行交互。
2. 多线程工作，实现一对多服务。
3. 容错性能良好，对错误返回不同语句，不会错误中断程序。

TCPServer 与 TCPClient 实现的指令有：

1. USER：输入登录用户名。TCPServer 接受后，若已登录，则返回 500 指令无法识别。若未登录，则检查是否只有一个参数。若参数数量错误，则返回 501 参数错误。若参数数量正确，检查用户是否为 anonymous，若不是，则返回 530 登录失败。若是 anonymous，返回 331 输入密码，进入等待 PASS 状态。
2. PASS：输入登录密码。TCPServer 接受后，若不在 USER 后使用，则返回 500 指令无法识别。若在 USER 后使用，则检查是否只有一个参数。若参数数量错误，则返回 501 参数错误。若参数数量正确，TCPServer 接受所有密码，返回 230 登录成功，进入登录成功状态。
3. QUIT、ABOR：退出并断开连接。TCPClient 发送接受后，判断是否为 2\*\*，若是

则断开连接。TCPServer 接受后，检查是否没有参数。若参数数量错误，则返回 501 参数错误。若参数数量正确，TCPServer，返回 221 断开连接，关闭连接通道。

4. SYST :返回指定内容。TCPServer 接受后，检查是否没有参数。若参数数量错误，则返回 501 参数错误。若参数数量正确，返回 215 UNIX Type : L8。
5. TYPE :返回指定内容。TCPServer 接受后，检查是否只有一个参数。若参数数量错误，则返回 501 参数错误。若参数数量正确，检查参数是否为 I，若不是，则返回 501 参数错误。若是 I，返回 200 Type set to I。
6. PORT :发送客户端 IP 和端口，供下次数据传输使用。TCPClient 发送接受后，判断是否为 2\*\*，若是则存下 IP 和端口，置为 PORT 状态。TCPServer 接受后，检查是否只有一个参数，并用正则表达式检查是否为指定格式。若参数数量或格式错误，则返回 501 参数错误。若参数数量和格式正确，TCPServer 存下 IP 和端口，置为 PORT 状态，并返回 220 准备完毕。
7. PASV :请求返回服务器 IP 和端口，供下次数据传输使用。TCPClient 发送接受后，判断是否为 2\*\*，若是则寻找并存下 IP 和端口，置为 PASV 状态。TCPServer 接受后，检查是否没有参数。若参数数量错误，则返回 501 参数错误。若参数数量正确，TCPServer 新建 ServerSocket 并返回 227IP 和端口，同时存下 IP 和端口，置为 PASV 状态。
8. RETR :客户端下载文件。TCPClient 发送指令，判断是否在 PORT 或 PASV 状态和返回是否为 1\*\* ( 125 或 150 )。若是则建立连接下载，否则表明连接失败不进行操作。同时，TCPServer 接受后，检查是否只有一个参数。若参数数量错误，则返回 501 参数错误。否则，TCPServer 判断是否在 PORT 或 PASV 状态和文件是否存在，若是则建立连接并上传文件，返回 226。若不是或传输中出错，则返回 425、426 或 550。TCPClient 和 TCPServer 清除 PORT 和 PASV 状态。
9. STOR :客户端上传文件。大致与 RETR 相同，区别为：TCPClient 判断文件是否存在，若不存在不发送指令并提示用户。TCPClient 上传，TCPServer 下载。
10. CWD :进入指定目录，支持绝对路径和相对路径，“..\"表示进入父目录。TCPServer 接受后，检查是否只有一个参数。若参数数量错误，则返回 501 参数错误。若参数数量正确，判断目录是否存在，不存在返回 550。若存在，更改当前目录地址，返回 250。
11. CDUP :进入父目录。TCPServer 接受后，检查是否没有参数。若参数数量错误，则返回 501 参数错误。若参数数量正确，判断父目录是否存在，不存在返回 550。若存在，更改当前目录地址，返回 200。
12. DELE :删除文件。TCPServer 接受后，检查是否只有一个参数。若参数数量错误，则返回 501 参数错误。若参数数量正确，判断文件是否存在，不存在返回 550。若存在，删除文件，返回 250。
13. MKD :创建文件夹。TCPServer 接受后，检查是否只有一个参数。若参数数量错误，则返回 501 参数错误。若参数数量正确，判断文件夹是否存在，已存在返回

- 550。若不存在，创建文件夹，返回 257。
14. PWD：返回当前目录地址。TCPServer 接受后，检查是否没有参数。若参数数量错误，则返回 501 参数错误。若参数数量正确，返回 257 和目录地址。
15. RMD：删除文件夹。TCPServer 接受后，检查是否只有一个参数。若参数数量错误，则返回 501 参数错误。若参数数量正确，判断文件夹是否存在，不存在返回 550。若存在，深度优先搜索删除所有子文件和子文件夹后删除目标文件夹，返回 250。
16. RNFR：指定重命名目标。TCPServer 接受后，检查是否只有一个参数。若参数数量错误，则返回 501 参数错误。若参数数量正确，判断文件夹是否存在，不存在返回 550。若存在，置为 RN 状态，记录其地址和文件（夹）名，返回 250。
17. RNT0：重命名。TCPServer 接受后，检查是否只有一个参数。若参数数量错误，则返回 501 参数错误。若参数数量正确，判断是否为 RN 状态，若不是返回 550。判断原文件（夹）和目标文件（夹）是否存在，不存在返回 550。若存在，更改文件（夹）名，返回 250。TCPServer 清除 RN 状态。
18. LIST：获取当前目录文件（夹）列表。TCPClient 发送指令，判断是否在 PORT 或 PASV 状态和返回是否为 1\*\*（125 或 150）。若是则建立连接收取文件列表信息，否则表明连接失败不进行操作。同时，TCPServer 接受后，检查是否只有一个参数。若参数数量错误，则返回 501 参数错误。否则，TCPServer 判断是否在 PORT 或 PASV 状态，若是则建立连接并发送文件列表信息，返回 226。若不是或传输中出错，则返回 425、426 或 550。TCPClient 和 TCPServer 清除 PORT 和 PASV 状态。

## 困难和解决方案

- PORT 命令报错

编写过程中遇到只有 PORT 命令执行时报错的现象。排除代码 bug 的原因后，最终发现防火墙会阻止 PORT 命令，断开客户端和服务器的连接。

- 程序容错性

编写过程中程序 bug 导致很多时候无法调试需要的部分。最终通过 throw 和 catch 解决的许多问题，修复了许多 bug。

- 文件加锁

在用户 RETR 和 STOR 过程中，若文件被其他用户 DELE、RMD 或 RNT0，则会发生错误。考虑过的解决方法是：在文件中保存一个加锁文件表明那些文件正在被操作，这些加锁文件无法被操作，但可能覆盖有用的文件，而且需要考虑 RMD 命令在所有父文件夹加锁。现在使用的方法为 try-catch 返回 550。

- 服务器安全和用户权限

由于 CWD 支持绝对路径和相对路径和 CDUP 无限制，现在命令可以进入服务器所有目录，且可以对服务器所有文件操作，这对服务器安全有一定的影响。但考虑到用户的自由度，暂时对用户没有限制。