

FTP 程序设计报告

姓名：张凯

学号：2012013311

邮箱：nezharen@163.com

Contents

1 综述	2
2 编程环境	2
3 FTP 服务器	2
3.1 完成情况	2
3.1.1 基本功能	2
3.1.2 扩展功能	2
3.2 使用方法	2
4 FTP 客户端	3
4.1 完成情况	3
4.1.1 基本功能	3
4.1.2 扩展功能	3
4.2 使用方法	3
5 简答题：如何传输大文件而不会阻塞服务器	4

1 综述

本报告主要描述 FTP 服务器及客户端的完成情况和使用方法。

2 编程环境

处理器	4 * Intel Core i5-3230M CPU @ 2.60GHz
内存	4GB
操作系统	Kubuntu 14.04 64-bit
程序语言	Java
代码编辑器	Vim

3 FTP 服务器

3.1 完成情况

3.1.1 基本功能

通过多线程技术实现了一个多线程并发的 FTP 服务器的一些基本功能，对于每一个 FTP 请求都开启一个独立的线程进行处理，因此可以同时处理多个 FTP 请求并稳定工作。

FTP 服务器可以处理来自客户端的 USER, PASS, RETR, STOR, QUIT, SYST, TYPE, PORT 和 PASV 命令，并按照 FTP 协议完成相应的工作。特别的，服务器只实现了 TYPE 中的 binary 模式，如果客户端请求使用其他的模式，服务器会返回不支持此模式。

服务器可以设置工作目录。

3.1.2 扩展功能

FTP 服务器可以处理来自客户端的 LIST, PWD, CWD, CDUP, MKD, RMD, DELE, RNFR, RNTD 和 SIZE 命令，并使用 binary 模式返回相应数据。特别的，目录相关命令的实现依赖于 Linux 环境，因此此功能仅在非 Windows 环境下可以使用。

3.2 使用方法

在 “./server/program/” 文件夹中已存放打包好的 FTPServer.jar 文件，通过 “java -jar ./server/program/FTPServer.jar -d ”root-path”” 即可运行 FTP 服务器。服务器将监听主机的 21 号端口，可能需要以管理员权限运行并在防火

墙中开放 21 号端口权限。“-d”参数用于设置服务器的工作根目录，本参数如果忽略，则以 FTPServer.jar 所在的目录为工作根目录。

4 FTP 客户端

4.1 完成情况

4.1.1 基本功能

FTP 客户端可以使用 USER, PASS, RETR, STOR, QUIT, SYST, TYPE, PORT 和 PASV 命令，按照 FTP 协议向 FTP 服务器发送请求以完成相应的工作。

4.1.2 扩展功能

FTP 客户端可以使用 LIST, PWD, CWD, CDUP, MKD, RMD, DELE, RNFR, RNTD 和 SIZE 命令向服务器发送请求。

FTP 客户端模仿 linux 中的 FTP 命令编写，将 FTP 协议中的相应命令封装成了 open, get, put, system, binary, passive 等相应指令。用户不需要手动输入原始的 FTP 协议命令，只需要使用封装好的指令即可轻松操作客户端。

4.2 使用方法

在“./client/program/”文件夹中已存放打包好的 FTPClient.jar 文件，通过“java -jar ./client/program/FTPClient.jar”即可运行 FTP 客户端。FTP 客户端可以使用以下指令：

open host [port]	连接 FTP 服务器并登录，主机名为 host，端口为 port
user	切换用户，重新登录
system	显示 FTP 服务器系统信息
binary	切换为 binary 模式
ascii	切换为 ascii 模式
passive	设置被动模式开关
get remote-file	从服务器下载文件
put local-file	上传文件到服务器
delete remote-file	删除服务器的文件
ls [path]、dir [path]	列出服务器 path 路径下的文件
pwd	显示服务器工作路径
cd path	切换服务器工作路径到 path
cdup	切换服务器工作路径到上一层
mkdir path	在服务器创建目录
rmdir path	删除服务器的 path 目录
rename old-path new-path	修改服务器文件（夹）名
size file	显示服务器文件的大小
close	断开连接，但不退出 FTP 客户端
bye、exit、quit	断开连接，并退出 FTP 客户端

5 简答题：如何传输大文件而不会阻塞服务器

一方面，由于传输大文件需要长期占用固定端口，因此应尽量使用随机端口而不是固定端口，充分利用端口资源，并采用多线程处理 FTP 请求。另一方面，大文件传输会持续占用硬盘及网络资源，需要对传输速率进行控制，避免持久高速的硬盘读写及网络传输消耗大量的系统资源。