

http_server_and_downloader 项目操作手册

项目描述

首先配置 xinetd, 使其能提供正常的服务监听以及多线程服务功能, 接着编写 http 服务子程序以及 http 终端下载器程序, 处理 http 请求, 并且该 http 服务程序分为浏览器和命令后终端两个版本, 因此, 最终可通过浏览器或者命令行终端对服务器上的文件进行访问下载。

(注: 由于 http 服务程序有两个版本, 但是两个程序相似性非常高, 下面说明以浏览器版本为例, 终端版本的可参考作业文档说明里的图片, 其余的程序结构基本一致)

项目文件

—— http_downloader.c	// 命令行终端 http 下载器
—— myhttpd_browser_version.c	// http 服务程序支持浏览器版本
—— myhttpd_terminal_version.c	// http 服务程序支持命令行终端版本
—— Makefile	// Makefile 文件
—— 项目操作手册.pdf	// 项目操作手册
—— 3G 测试视频.mp4	// 较大视频测试文件
—— 9M 测试视频.mp4	// 小视频测试文件
—— 30K 测试文本.txt	// 小文本测试文件
—— 168K 测试图片.jpg	// 小图片测试文件
—— 作业文档说明.pdf	// 作业文档说明

项目环境

- 操作系统: lubuntu 18.04.1
- 编译器: gcc version 7.3.0
- 编辑器: vim 8.0.1453, sublime_text_3

xinetd 配置

- cd /etc/xinetd.d/
- sudo vim myhttpd // 注意加上 sudo 以及保证 myhttpd 和所生成的可执行程序名字一样
- 添加如下内容: // 注意修改相应的路径参数

service myhttpd

```
{
    disable            = no
    server_args        = /home/xiao/myhttpd/ l
    server             = /home/xiao/myhttpd/myhttpd
    socket_type        = stream
    protocol           = tcp
    user               = xiao
    wait               = no
    flags              = IPv4
}
```

- sudo vim /etc/services, 添加以下内容: // 端口号和源码一致

```
myhttpd      12345/tcp      # tcp service
myhttpd      12345/udp      # udp service
```

- sudo service xinetd restart

- ./myhttpd // 如果端口号小于 1024，则为系统保留端口，改为 sudo ./myhttpd

http 服务程序主要函数实现功能

- send_headers: 发送 http 状态码及响应首部等信息
- send_error: 访问出错时，发送错误提示网页
- strencode: 将所要发送资源的名字编码以符合 URL 的编码规格
- file_infos: 获取文件信息
- strdecode: 将所接收的 URL 请求解码为可读的字符串以便于找到相应的请求资源
- get_mime_type: 获取资源类型
- hexit: 将 URL 中十六进制的数字转化为十进制数
- mylog: 日志记录函数

http 服务程序代码流程

- xinetd 为守护进程，一直在监听 12345 端口号；
- 当该端口号被访问时，xinetd 启动 myhttpd 进程，并传入相应的命令参数以及重定向相关的输入输出描述符；
- myhttpd 进程被启动后，则开始对相应的 http 请求进行处理；
- myhttpd 首先检测日志文件、传入参数等是否正常，如正常，则开始分析 http 请求头，取出相应的 GET 方法以及所请求的资源文件路径名字；
- 对所请求的资源进行检测，判断是否存在，若存在，检测其为文件类型还是目录类型，并分别进行处理；
- 如果为文件，则读取相应的文件并发送；
- 如果为目录，首先检测目录中是否含有 index.html，若有，则直接跳转到读取 index.html 文件并发送，若无，则显示目录文件。

http 下载器程序主要函数实现功能

- parse_url: 解析 url
- parse_header: HTTP 协议头部解析
- get_ip_addr: 获取 ip 地址
- progress_bar: 用于显示下载进度条
- get_file_size: 将所接收的 URL 请求解码为可读的字符串以便于找到相应的请求资源
- get_mime_type: 获取文件大小
- download: 下载文件函数

http 下载器程序代码流程

待补充。

项目结果

本地浏览器访问: <http://0.0.0.0:12345/>

命令行终端输入: `./a.out http://0.0.0.0:12345/`

或 `./a.out http://0.0.0.0:12345/3G 测试视频.mp4`