

廈門大學



信息学院软件工程系

《计算机网络》实验报告

题 目 实验七 代理服务器软件

班 级 软件工程 2020 级卓越班

姓 名 庾晓萍

学 号 20420192201952

实验时间 2022 年 5 月 12 日

2022 年 5 月 12 日

填写说明

- 1、本文件为 Word 模板文件，建议使用 Microsoft Word 2019 打开，在可填写的区域中如实填写；
- 2、填表时勿破坏排版，勿修改字体字号，打印成 PDF 文件提交；
- 3、文件总大小尽量控制在 1MB 以下，最大勿超过 5MB；
- 4、应将材料清单上传在代码托管平台上；
- 5、在实验课结束 14 天内，按原文件发送至课程 FTP 指定位置。

1 实验目的

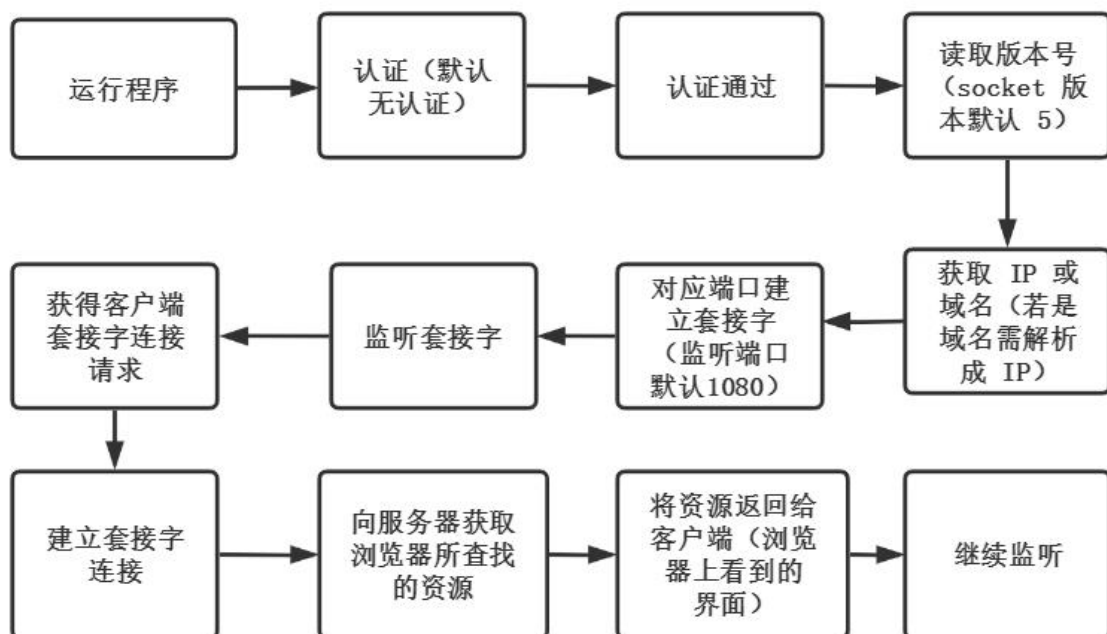
通过完成实验，掌握基于 RFC 应用层协议规约文档传输的原理，实现符合接口且能和已有知名软件协同运作的软件。

2 实验环境

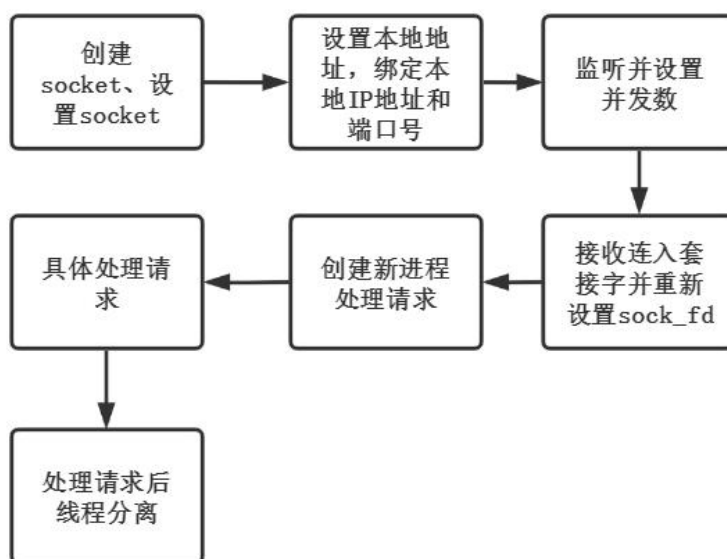
VMWare 虚拟机，Ubuntu 操作系统

3 实验结果

一、实验思路



二、核心代码



1、创建 socket、设置 socket。设置本地地址，绑定本地 IP 地址和端口号。

```

//创建流式套接字
//如果失败会返回-1，然后提示socket()，然后退出程序
if ((sock_fd = socket(AF_INET, SOCK_STREAM, 0)) < 0) {
    log_message("socket()");
    exit(1);
}
//setsockopt用于对套接字进行设置，这里是设置在closesocket后继续重用该socket
if (setsockopt(
    sock_fd, SOL_SOCKET, SO_REUSEADDR, (char *)&optval,
    sizeof(optval)) < 0) {
    log_message("setsockopt()");
    exit(1);
}
  
```

```

//给套接口分配一个本地名字来为套接口建立本地绑定
//即将本地地址和套接口绑定
if (bind(sock_fd, (struct sockaddr *)&local, sizeof(local)) < 0) {
    log_message("bind()");
    exit(1);
}
  
```

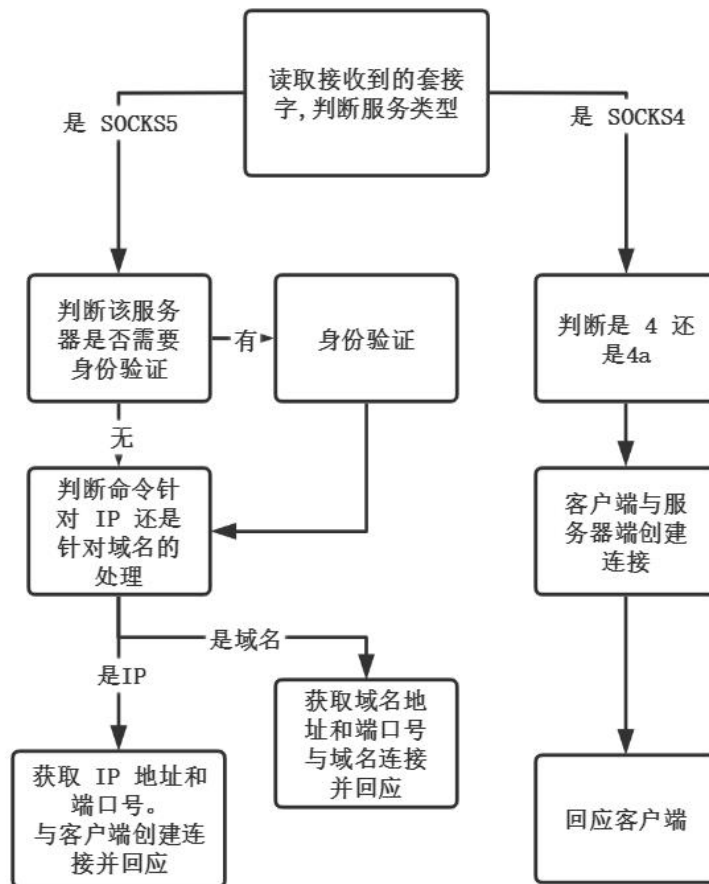
```

//创建一个套接口并监听申请的连接
//第二个参数是等待连接队列的最大长度
if (listen(sock_fd, 25) < 0) {
    log_message("listen()");
    exit(1);
}
  
```

2、接受连入套接字并重新设置 sock_fd，创建新进程处理请求，具体处理请求，处理请求后线程分离。

```
while (1)
{
    //本函数会阻塞等待直到有客户端请求到达,返回一个新的套接字,代表新的连接,相当于这个客户端的socket
    if ((net_fd =
        accept(sock_fd, (struct sockaddr *)&remote,
            &remotelen)) < 0) {
        log_message("accept()");
        exit(1);
    }
    int one = 1;
    //设置服务器socket
    setsockopt(sock_fd, SOL_TCP, TCP_NODELAY, &one, sizeof(one));
    //pthread_create是linux等操作系统创建线程的函数,功能是创建线程
    //net_fd是上面新创建的套接字
    if (pthread_create
        (&worker, NULL, &app_thread_process,
            (void *)&net_fd) == 0) {
        //将子线程从主线程分离出来,子线程结束后,资源自动回收
        pthread_detach(worker);
    } else {
        log_message("pthread_create()");
    }
}
```

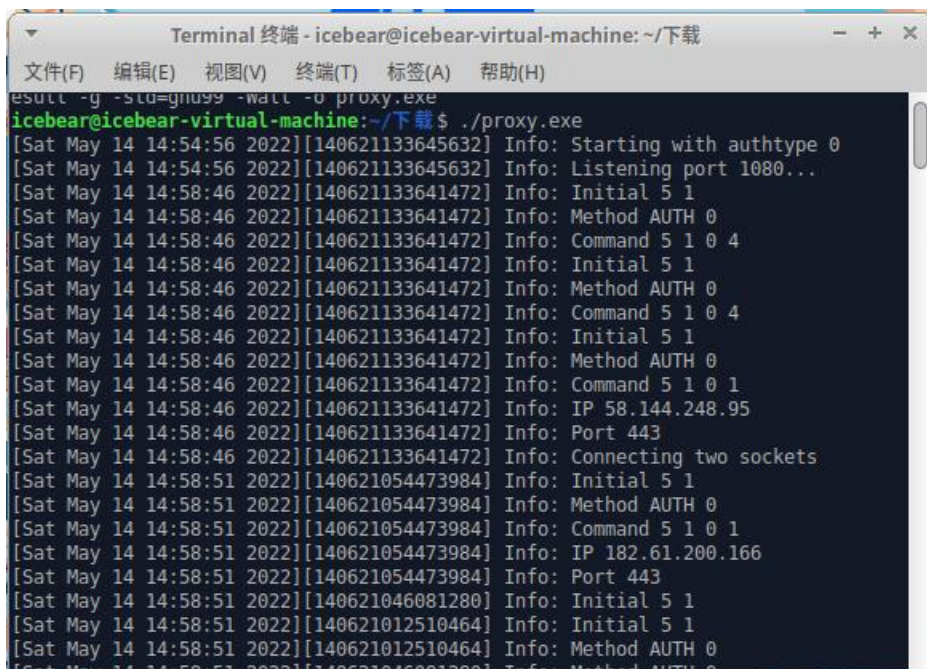
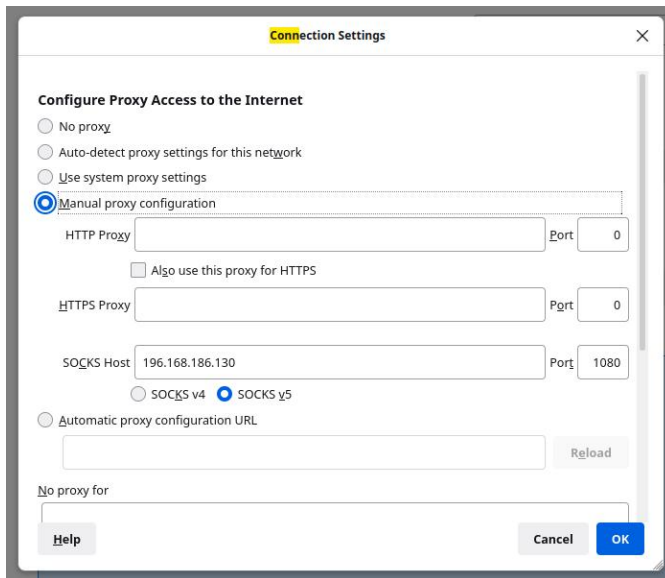
3、处理请求：app_thread_process。

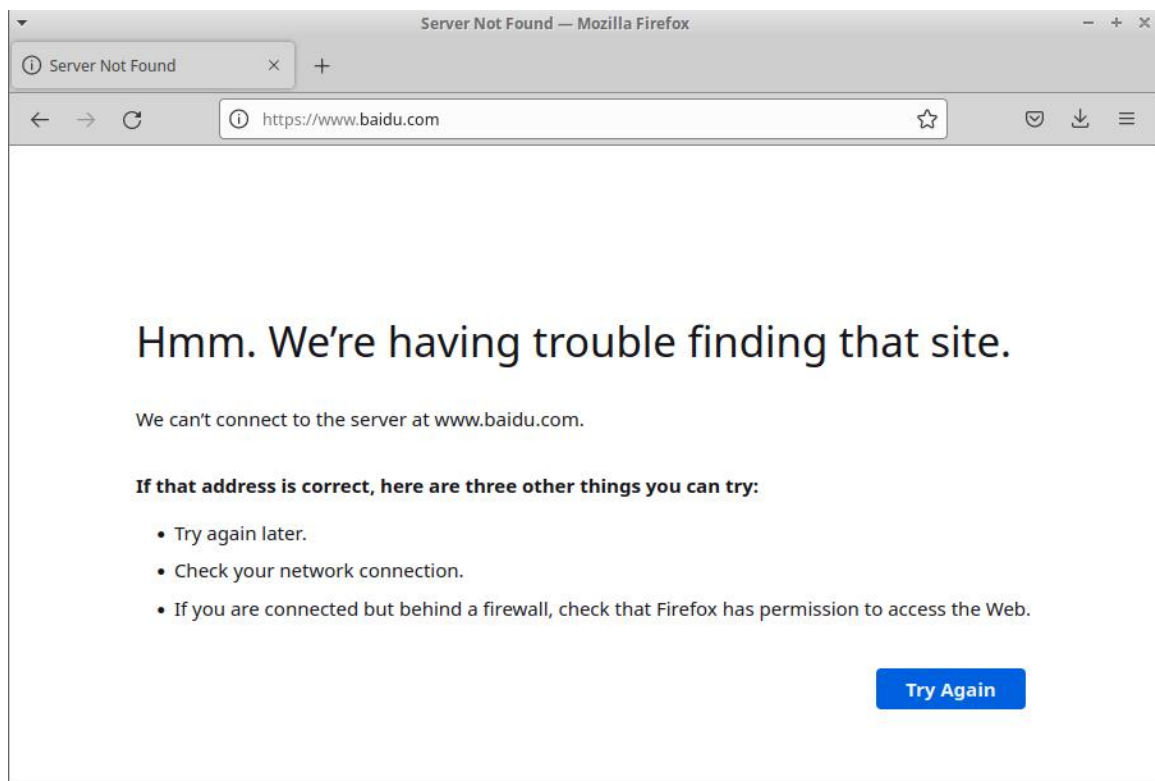
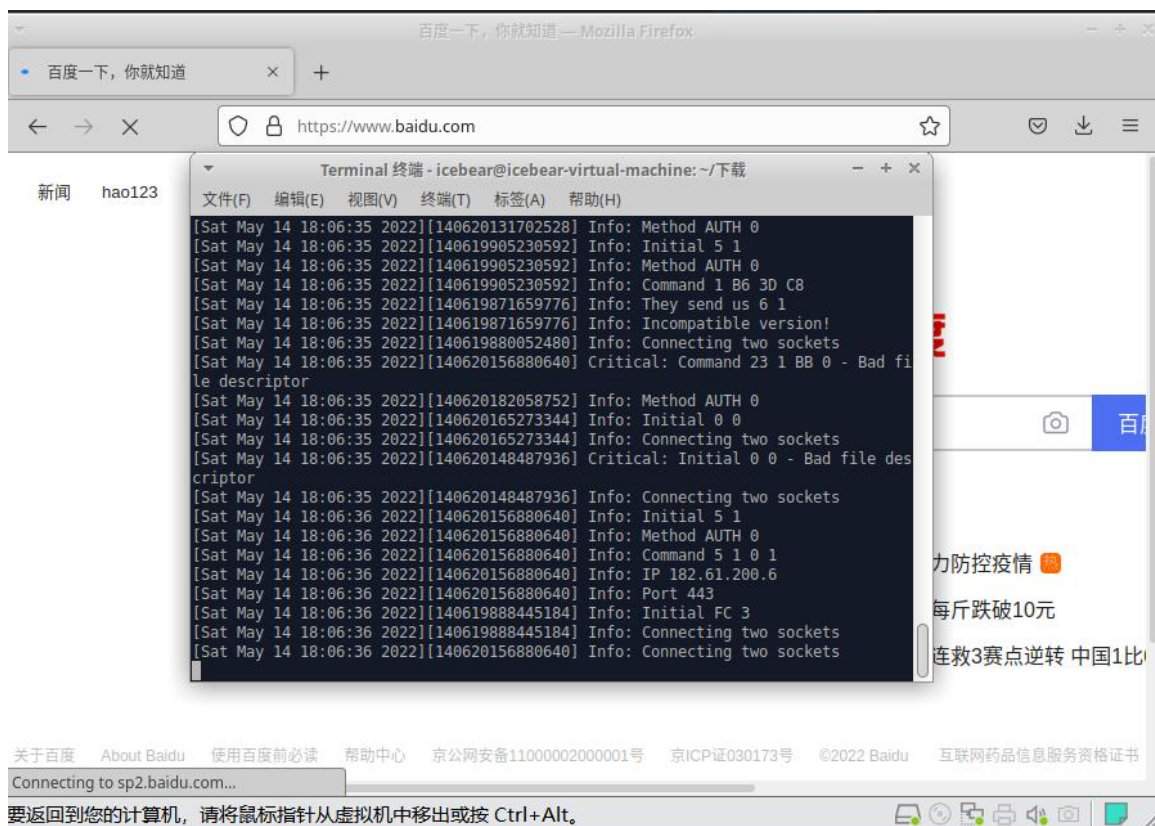


```
switch (version) {
case VERSION5: {
    //.....
    if (command == IP) {
        //.....
    } else if (command == DOMAIN) {
        //.....
    } else {
        app_thread_exit(1, net_fd);
    }
}
case VERSION4: {
    if (methods == 1) {
        //.....
        if (socks4_is_4a(ip)) {
            //.....
        } else {
            //.....
        }
        if (inet_fd != -1) {
            socks4_send_response(net_fd, 0x5a);
        } else {
            socks4_send_response(net_fd, 0x5b);
            free(ip);
            app_thread_exit(1, net_fd);
        }
        free(ip);
    } else {
        log_message("Unsupported mode");
    } break;
}
```

二、代码运行

在虚拟机中通过运行浏览器，通过输入外网 IP 地址访问网站，得到无法连接的错误提示。在“套接字”一行的标签中填入服务器端的 IP 地址与端口号，如：192.168.186.130 和 1080。刷新页面，如果成功则可以访问网站。将代理服务关闭则无法访问。





4 实验代码

本次实验的代码已上传于以下代码仓库：

https://github.com/ryanregal/Exp_ComputerNetwork

5 实验总结

经过本次实验，通过本次实验我学习了 Socks4/Socks5 协议的工作过程与原理，对 Socket 有更加深刻的理解，学习了代理服务器的搭建步骤，使得计算机网络的知识得到应用。