

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 2021 级卓越班

姓 名 黄子安

学 号 22920212204396

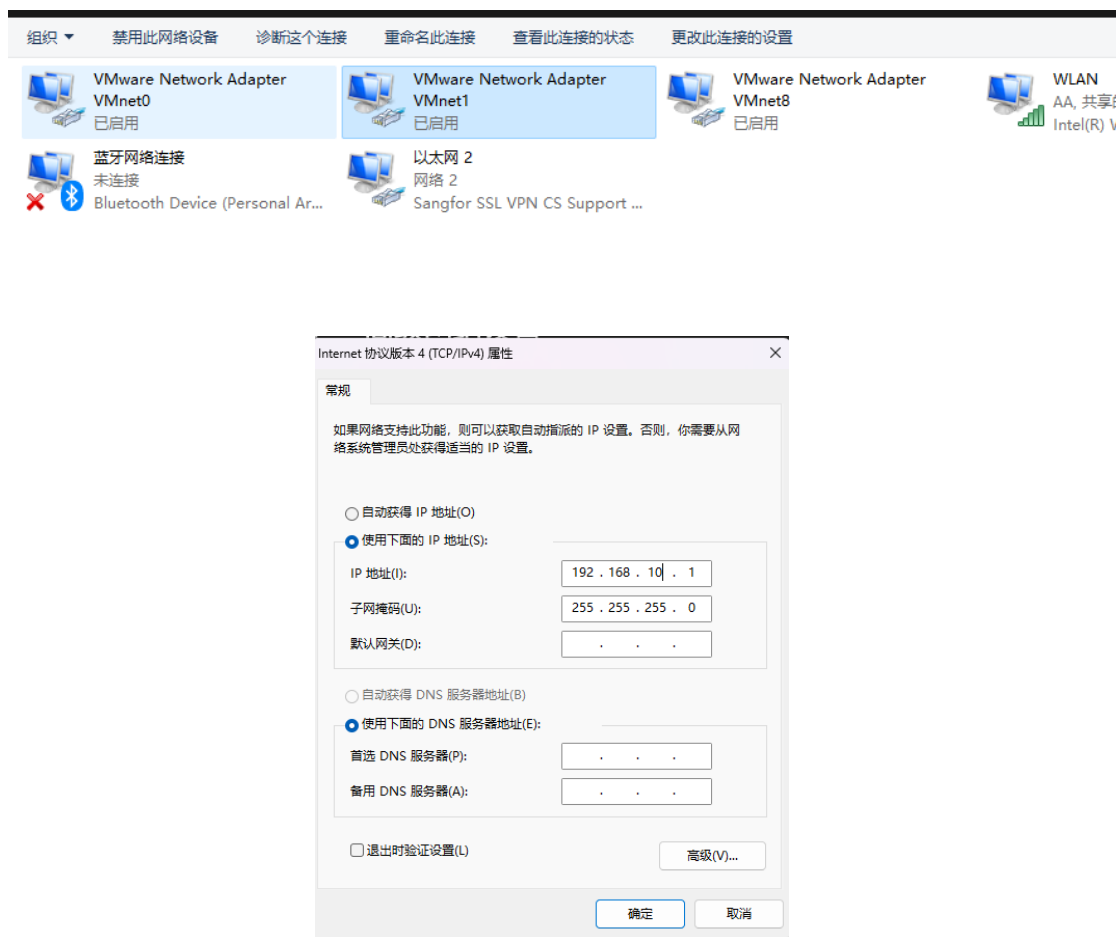
实验时间 2023 年 5 月 30 日

2023 年 5 月 30 日

填写说明

- 1、本文件为 Word 模板文件，建议使用 Microsoft Word 2021 打开，在可填写的区域中如实填写；
- 2、填表时勿改变字体字号，保持排版工整，打印为 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。

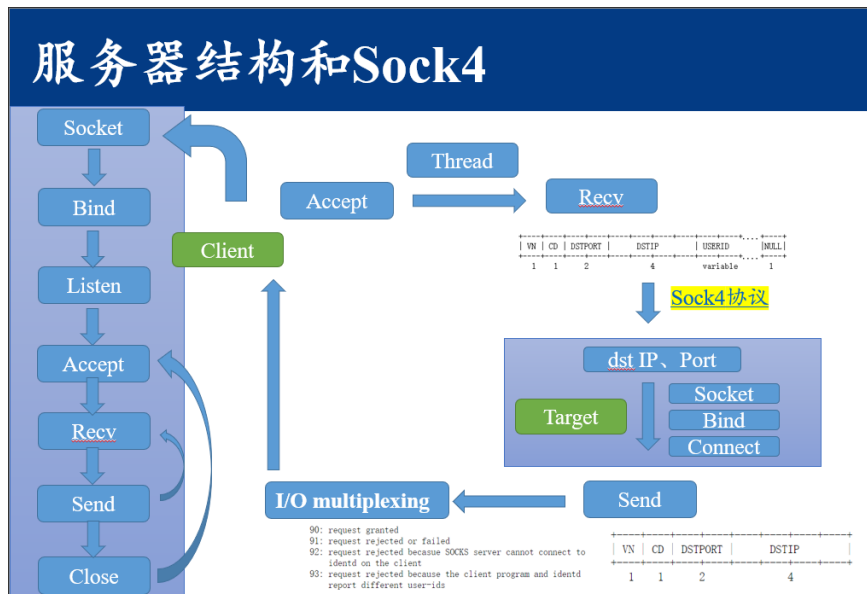
配置虚拟机的 IP 地址和物理机的 Vmnet 网络适配器，确保二者可以相互 ping 通



完成虚拟机配置之后，虚拟机无法直接访问外网



2.编写代理服务器软件，其核心的结构如下图所示



配置主机代理服务器的 Socket

```
serverAddress.sin_family = AF_INET;
serverAddress.sin_addr.s_addr = inet_addr("192.168.10.1");
serverAddress.sin_port = htons(1800);

if (bind(serverSocket, (struct sockaddr*)&serverAddress, sizeof(serverAddress)) == SOCKET_ERROR)
{
    std::cerr << "绑定套接字失败" << std::endl;
    return -1;
}

// 监听连接
if (listen(serverSocket, 5) == SOCKET_ERROR)
{
    std::cerr << "监听失败" << std::endl;
    return -1;
}

std::cout << "等待客户端连接..." << std::endl;
```

监听客户端接入，开启一个线程来处理客户端的请求

```
while (true)
{
    // 接受客户端连接
    clientSocket = accept(serverSocket, (struct sockaddr*)&clientAddress, &clientAddressLength);
    if (clientSocket == INVALID_SOCKET)
    {
        std::cerr << "接受连接失败" << std::endl;
        continue;
    }
    // 创建线程处理客户端请求
    std::thread clientThreadObj(clientThread, clientSocket);
    clientThreadObj.detach(); // 分离线程，使其在完成任务后自动释放资源
}
```

接受来自客户端的 Sock4 连接数据包，提取出客户端需要访问的目的 IP 和端口

```
//接收到来自客户端的连接请求
char buf[2048];
int len = recv(clientSocket, buf, sizeof(buf), 0);

uint32_t dst_ip = *(uint32_t*)(buf + 4);
uint16_t dst_port = ntohs(*(uint16_t*)(buf + 2));
struct in_addr inaddr;
inaddr.s_addr = dst_ip;
char* dst_ip_str = inet_ntoa(inaddr);
cout<< time_now() <<" 客户端接入并访问: " << dst_ip_str << ":" << dst_port << endl;
```

访问客户端需要访问的目的服务器的目的 IP 和端口

```
int targetSocket = socket(AF_INET, SOCK_STREAM, 0);
if (targetSocket < 0) {
    // 套接字创建失败, 要进行错误处理
}
struct sockaddr_in targetAddr;
memset(&targetAddr, 0, sizeof(targetAddr));
targetAddr.sin_family = AF_INET;
targetAddr.sin_port = htons(dst_port);
targetAddr.sin_addr.s_addr = inet_addr(dst_ip_str);

connect(targetSocket, (struct sockaddr*)&targetAddr, sizeof(targetAddr));
```

向客户端发送授权成功请求的响应数据包

```
char success_response[8] = { 0x00, 0x5A };
send(clientSocket, success_response, sizeof(success_response), 0);
```

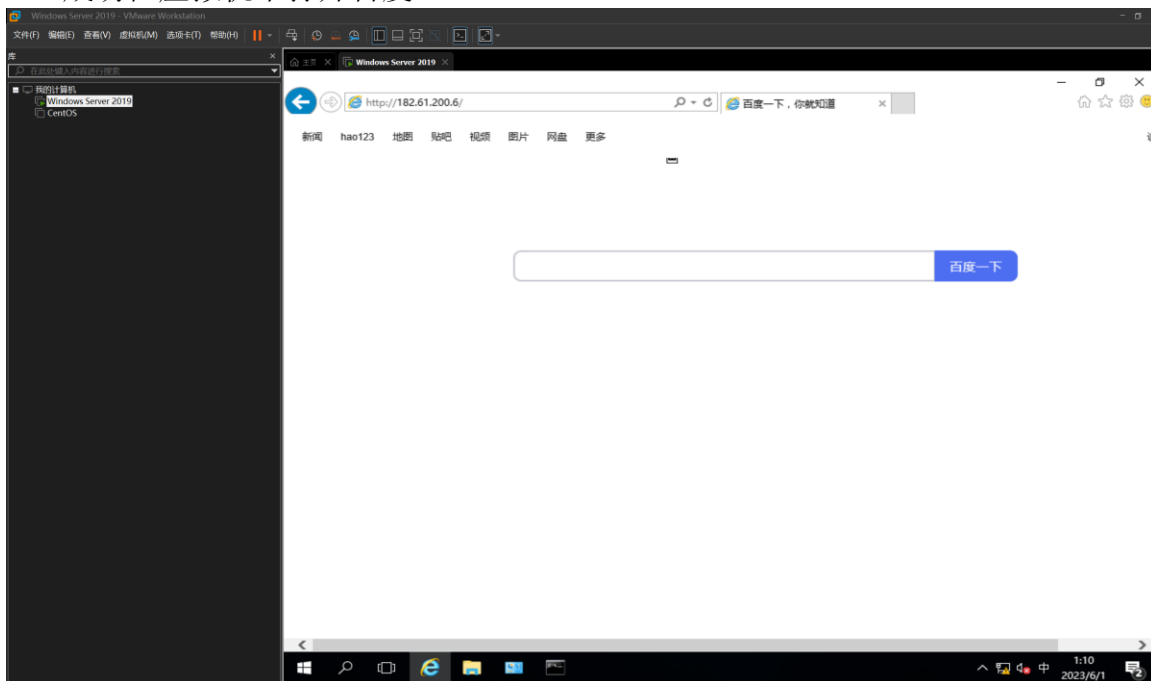
3.运行效果

通过日志打印的方式打印客户端发给代理服务器的 Http 请求以及代理服务器从目的服务器处获取的 Http 响应

```
D:\Desktop\计算机网络实验七\sock5 x + v
等待客户端连接...
2023-06-01 00:52:45 客户端接入并访问: 182.61.200.6:80
GET / HTTP/1.1
Accept: text/html, application/xhtml+xml, image/jxr, */*
Accept-Language: zh-CN
User-Agent: Mozilla/5.0 (Windows NT 10.0; WOW64; Trident/7.0; rv:11.0) like Gecko
Accept-Encoding: gzip, deflate
Host: 182.61.200.6
DNT: 1
Connection: Keep-Alive

HTTP/1.1 200 OK
Bdpagetype: 1
Bdqid: 0x89c7cbe1000b6224
Connection: keep-alive
Content-Encoding: gzip
Content-Security-Policy: frame-ancestors 'self' https://chat.baidu.com https://fj-chat.baidu.com https://hba-chat.baidu.com https://hbe-chat.baidu.com https://njjs-chat.baidu.com https://nj-chat.baidu.com https://hna-chat.baidu.com https://hnb-chat.baidu.com;
Content-Type: text/html; charset=utf-8
Date: Wed, 31 May 2023 16:52:45 GMT
P3p: CP=" OTI DSP COR IVA OUR IND COM "
P3p: CP=" OTI DSP COR IVA OUR IND COM "
Server: BWS/1.1
Set-Cookie: BAIDUID=C69472ACD1F236015DB76986B380B07F; expires=Thu, 31-Dec-37 23:55:55 GMT; max-age=2147483647; path=/; domain=.baidu.com
Set-Cookie: BIDUPSID=C69472ACD1F236015DB76986B380B07F; expires=Thu, 31-Dec-37 23:55:55 GMT; max-age=2147483647; path=/; domain=.baidu.com
Set-Cookie: BFTN=1605551065; expires=Thu, 31-Dec-37 23:55:55 GMT; max-age=2147483647; path=/; domain=.baidu.com
```

成功在虚拟机中打开百度



4 实验代码

本次实验的代码已上传于以下代码仓库：<https://gitee.com/aaaz718/experiment-7-proxy-server>

5 实验总结

通过代理服务器的配置对应用层的常用协议做了一些代码编写层面的了解，对应用层有了进一步的认识。

通过虚拟机的网络配置对计算机的网络环境有了进一步认知，包括桥接、单主机、NAT 模式等