**中国矿业大学计算机科学与技术学院**

**2023-2024(2)《计算机网络实验》课程报告(本科)**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 专业班级: | | 计算机科学与技术2022-2班 | | | | 姓名: | 杨晓琦 | | | 学号: | | 08222213 | |
|  | | | 指派的网络号： | | | 172.16.82~172.16.84 | | | | |  | | |
| **序号** | **报告题目** | **基础理论掌握程度** | | **综合知**  **识应用**  **能力** | **实验**  **内容** | **报告**  **格式** | **完成**  **状况** | **工作量** | **学习、**  **工作**  **态度** | **抄袭**  **现象** | | **其它** | **综合**  **成绩** |
|  |  |  | |  |  |  |  |  |  |  | |  |  |
|  |  |  | |  |  |  |  |  |  |  | |  |  |
|  |  |  | |  |  |  |  |  |  |  | |  |  |
|  |  |  | |  |  |  |  |  |  |  | |  |  |
|  |  |  | |  |  |  |  |  |  |  | |  |  |
|  |  |  | |  |  |  |  |  |  |  | |  |  |
|  |  |  | |  |  |  |  |  |  |  | |  |  |
|  |  |  | |  |  |  |  |  |  |  | |  |  |
| 任课教师： | | 杨东平 | | |  |  |  | |  | | |  |  |
| 批阅时间： | |  | | | |  |  | |  | | | | |

**目录**

[7 基于TCP的网络应用编程 1](#_Toc171599105)

[7.1 实验1：基于TCP的socket应用程序设计 1](#_Toc171599106)

[7.1.1 实验目的 1](#_Toc171599107)

[7.1.2 编程环境 1](#_Toc171599108)

[7.1.3 实验内容 1](#_Toc171599109)

[7.1.4 具体步骤（代码） 1](#_Toc171599110)

[7.1.4 运行结果 6](#_Toc171599111)

[7.1.5 运行抓包 7](#_Toc171599112)

[7.2 实验体会 7](#_Toc171599113)

# 7 基于TCP的网络应用编程

**实验内容：**

1. 基于TCP的socket应用程序设计

**实验报告要求(必须认真阅读)：**

1. 不得使用实验指导书中的图片(截图)及图片(截图)的相关说明，否则所涉及实验内容视作无效， 并按0分计入成绩；
2. 实验中所使用的IP地址必须按照统一分配给个人的IP地址完成，否则所涉及实验内容视作无效，并按0分计入成绩；
3. 实验内容需要使用的网站、数据文件或其它软件必须作明确的说明，并配有相应的截图(或运行截图或文件目录截图)，否则所涉及实验内容视作抄袭，并按0分计入成绩；
4. 使用仿真软件的实验内容必须同时提交仿真软件所形成的工程文件，否则所涉及的实验内容视作未完成，并按0分计入成绩；
5. 编程的实验内容必须在报告中指出编程环境及其版本号，并同时提交编程环境所形成的工作空间文件或工程文件，以及源代码文件，否则所涉及的实验内容视作未完成，并按0分计入成绩；
6. 实验步骤及实验步骤所涉及的参数必须明确，否则酌情扣除一定的分数；
7. 必须在规定的时间内完成每一次的实验任务，并提交相关文件，否则该次实验内容按0分计入成绩。

说明：如果撰写规范不符合《计算机学院考查类课程报告撰写规范》要求的，整体上酌情扣除1-10分。

## 7.1 实验1：基于TCP的socket应用程序设计

### 7.1.1 实验目的

熟悉基于 TCP 的 C/S 网络通信过程，掌握 Socket 的数据结构及常用的Socket 函数，用 C/C++语言编写一个完整的 C/S 模式的 TCP 网络应用程序并进行相应的运行、测试与抓包分析。

### 7.1.2 编程环境

Visual Studio 2022

### 7.1.3 实验内容

分别设计基于 TCP 的 client 和 server 程序，并在两个不同的主机上运行，实现文件的上传与下载功能；在两台主机上分别运行抓包软件，观察建立连接、传输数据与释放连接的过程。

### 7.1.4 具体步骤（代码）

说明：客户端和服务端都是在本地主机上运行，联网采用手机热点，主机ip地址为192.168.166.74。

服务器端代码

#pragma comment(lib, "ws2\_32.lib")

#include <WinSock2.h>

#include <WS2tcpip.h>

#include <stdio.h>

#include <malloc.h>

int main() {

char buf[1024], file[60];

SOCKET s, cli\_socket;

int k;

FILE\* source;

struct sockaddr\_in addr, client\_addr;

int len = sizeof(addr);

char\* pBuf = (char\*)malloc(1025);

printf("服务器启动...\n");

WSAData wsaData;

if (WSAStartup(WINSOCK\_VERSION, &wsaData) != 0) return 0;

s = socket(AF\_INET, SOCK\_STREAM, 0);

if (s < 0) {

printf("获取SOCKET号失败!!!");

return 0;

}

memset(&addr, 0, len);

addr.sin\_addr.S\_un.S\_addr = INADDR\_ANY;

addr.sin\_family = AF\_INET;

addr.sin\_port = htons(5100);

if (bind(s, (struct sockaddr\*)&addr, len) < 0) {

printf("不能将服务器地址捆绑到SOCKET号上！！！");

closesocket(s);

return 0;

}

if (listen(s, 5) < 0) {

printf("server:listen");

return 0;

}

while (TRUE) {

printf("\n--------------------------\n");

if ((cli\_socket = accept(s, (struct sockaddr\*)&client\_addr, &len)) < 0) {

printf("server:accept");

continue;

}

int ret = recv(cli\_socket, file, sizeof(file), 0);

if (ret == SOCKET\_ERROR) {

printf("recv：%d\n", WSAGetLastError());

closesocket(cli\_socket);

continue;

}

char client\_ip[INET\_ADDRSTRLEN];

inet\_ntop(AF\_INET, &(client\_addr.sin\_addr), client\_ip, INET\_ADDRSTRLEN);

printf("客户：%s \n下载文件%s\n", client\_ip, file);

if (fopen\_s(&source, file, "rb") != 0) {

printf("目标文件%s不存在或打开失败\n", file);

closesocket(cli\_socket);

continue;

}

while (!feof(source)) {

k = fread(pBuf, 1, 1024, source);

if (k <= 0)

break;

send(cli\_socket, pBuf, k, 0);

memset(pBuf, '\0', 1025);

}

printf("\n传输完毕!!!\n\n");

fclose(source);

closesocket(cli\_socket);

}

free(pBuf);

closesocket(s);

WSACleanup();

return 1;

}

客户端代码

#pragma comment(lib,"ws2\_32.lib")

#include <WinSock2.h>

#include <cstdio>

#include <cstring>

#include <ws2tcpip.h>

int main(int argc, char\* argv[]) {

char buffer[1024], filename[60], received\_filename[60];

SOCKET client\_socket;

int bytes\_read;

FILE\* file\_ptr = nullptr;

struct sockaddr\_in server\_address;

WSAData wsa\_data;

int error\_code = WSAStartup(WINSOCK\_VERSION, &wsa\_data);

if (error\_code != 0)

return 0;

printf("The client is start...\n\n");

client\_socket = socket(AF\_INET, SOCK\_STREAM, 0);

if (client\_socket == INVALID\_SOCKET) {

printf("获取SOCKET号失败\n");

WSACleanup();

return 0;

}

printf("请输入服务器IP地址: ");

char server\_ip[20];

if (scanf\_s("%19s", server\_ip, \_countof(server\_ip)) != 1) {

printf("输入IP地址错误！\n");

closesocket(client\_socket);

WSACleanup();

return 0;

}

int address\_length = sizeof(server\_address);

memset(&server\_address, 0, address\_length);

server\_address.sin\_family = AF\_INET;

server\_address.sin\_port = htons(5100);

if (inet\_pton(AF\_INET, server\_ip, &server\_address.sin\_addr) <= 0) {

printf("无效的IP地址格式！\n");

closesocket(client\_socket);

WSACleanup();

return 0;

}

printf("\n正在与服务器连接...\n\n");

if (connect(client\_socket, (struct sockaddr\*)&server\_address, address\_length) < 0) {

printf("无法与服务器连接！\n\n");

closesocket(client\_socket);

WSACleanup();

return 0;

}

printf("服务器连接成功\n\n");

printf("请输入要下载的文件名：");

if (scanf\_s("%59s", filename, \_countof(filename)) != 1) {

printf("输入文件名错误！\n");

closesocket(client\_socket);

WSACleanup();

return 0;

}

if (fopen\_s(&file\_ptr, filename, "wb+") != 0) {

printf("文件创建失败\n\n");

closesocket(client\_socket);

WSACleanup();

return 0;

}

send(client\_socket, filename, sizeof(filename), 0);

memset(received\_filename, 0, sizeof(received\_filename));

recv(client\_socket, received\_filename, sizeof(received\_filename), 0);

strncpy\_s(received\_filename, sizeof(received\_filename), received\_filename, strlen(received\_filename));

while ((bytes\_read = recv(client\_socket, buffer, sizeof(buffer), 0)) > 0) {

fwrite(buffer, sizeof(char), bytes\_read, file\_ptr);

}

fflush(file\_ptr);

printf("\n文件 %s 下载成功！\n\n", received\_filename);

printf("服务器断开连接\n\n");

if (file\_ptr)

fclose(file\_ptr);

closesocket(client\_socket);

WSACleanup();

return 1;

}

### 7.1.4 运行结果

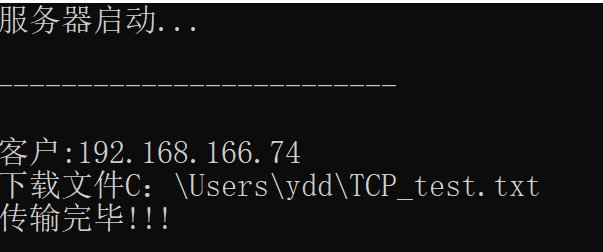
图7-1 客户端运行结果

图7-2 服务端运行结果

### 7.1.5 运行抓包

图7-3 抓包过滤

## 7.2 实验体会

我不仅掌握了TCP流式通信的基本原理和socket编程技术，还通过实践加深了对网络编程的理解。我相信这些知识和经验将对我未来的学习和工作产生积极的影响。同时，我也认识到自己在网络编程方面还有很多需要学习和提高的地方，我会继续努力，不断提升自己的专业技能。

----------------------------------以下表格用于评阅，不得擅自修改、删除--------------------------------------------

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **序号** | **报告内容** | **基础理论掌握程度** | **综合知**  **识应用**  **能力** | **报告**  **内容** | **报告**  **格式** | **完成**  **状况** | **工作量** | **学习、**  **工作**  **态度** | **抄袭**  **现象** | **其它** | **综合**  **成绩** |
| **8** | **应用层** | 请选择 | 请选择 | 请选择 | 请选择 | 请选择 | 请选择 | 请选择 | 学号:  姓名: | 目录、章节编目、错别字、乱码、截图、程序功能、程序源代码、源代码注释、体会、文献 |  |
| 批阅时间 | | 2024年5月22日 | | | | | | | | | |

----------------------------------以上表格用于评阅，不得擅自修改、删除--------------------------------------------