

实验序号： 5



## 《UNIX/LINUX 编程环境》

### 实验报告

实验名称： Linux 网络编程之 TCP 网络编程

姓 名： 李昊唐

学 院： 工学院

专 业： 物联网工程

班 级： 1 班

学 号： 1995131017

指导教师： 彭凯

实验地址： 数学学院 416

实验日期： 2021 年 12 月 21 日

## 实验 5 Linux 网络编程之 TCP 网络编程

### 一、实验目的

- (1) 理解 Socket 概念
- (2) 掌握 Linux 平台下 TCP 编程方法

### 二、实验环境

#### 实验配置

本实验所需的软硬件配置如表 1 所示。

配置	2.6 GHz 六核 Intel Core i7, 16 GB 2400 MHz DDR4, Intel UHD Graphics 630 1536 MB
硬件	MacBook Pro (15-inch, 2019)
系统	macOS 12.0.1
应用 软件	clion

#### 实验环境

本实验的环境为 Macintosh 机，如图 1 所示。



图 1 操作实验环境

### 三、实验原理

- (1) 程序进行网络通信时，是通过 IP 地址和套接字来访问一个主机的。

**IP 地址：**IP 地址的作用是标识计算机的网卡地址，每一台计算机都有一个 IP 地址。在程序中是通过 IP 地址来访问一台计算机的。IP 地址是 32 位长度的二进制数值，存储空间是 4 个字

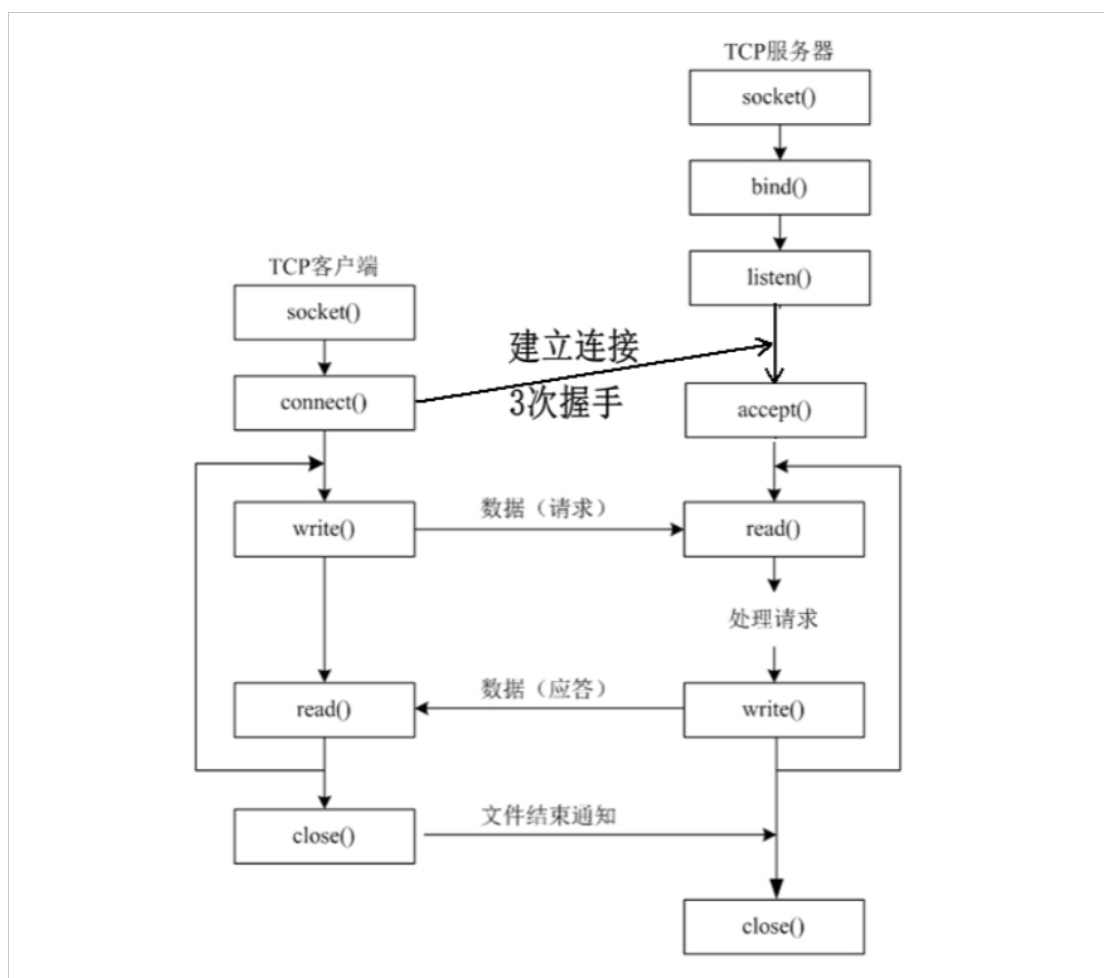
节。例如 1100000010101000 00000001 00000110 是一台计算机的 IP 地址。IP 地址可以使用点分十进制来表示，192.168.1.1。

端口 所谓端口，是指计算机中为了标识在计算机中访问网络的不同程序而设的编号。端口号是一个 16 位的无符号整数，对应的十进制取值范围是 0~65535。

## (2) TCP 与 UDP

TCP: TCP 是一种面向连接的网络传输方式。这种方式是可靠的，缺点是传过程复杂，需要占用较多的网络资源。

## (3) 套接字



TCP 编程流程图

## 四、实验步骤和实验结果

### 程序要求实现功能:

客户端和服务端通过 TCP 方式建立连接，客户端发送一个字母 A 给服务器端，服务器字母 A 加 1 变成字母 B 发送给客户端

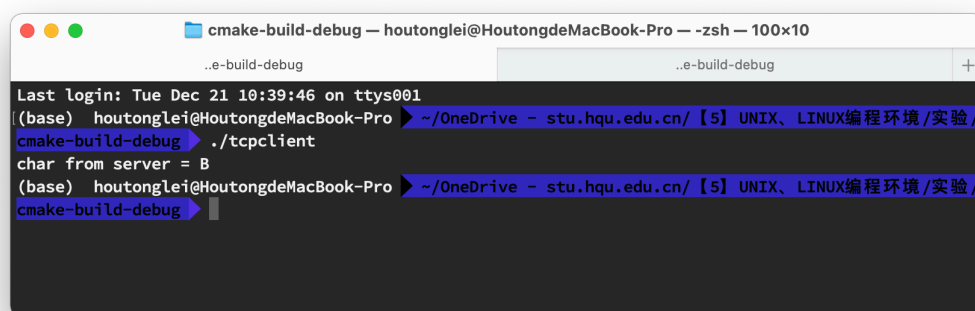
### 客户端:

```
#include <sys/socket.h>
#include <stdio.h>
#include <netinet/in.h>
#include <arpa/inet.h>
#include <unistd.h>
#include <stdlib.h>

int main() {
    int sockfd;
    int len;
    struct sockaddr_in address;
    int result;
    char ch = 'A';

    sockfd = socket(AF_INET, SOCK_STREAM, 0);
    address.sin_family = AF_INET;
    address.sin_addr.s_addr = inet_addr("127.0.0.1");
    address.sin_port = htons(9734);
    len = sizeof(address);

    result = connect(sockfd, (struct sockaddr *)&address, len);
    if (result == -1) {
        perror("oops: client1");
        exit(1);
    }
    write(sockfd, &ch, 1);
    read(sockfd, &ch, 1);
    printf("char from server = %c\n", ch);
    close(sockfd);
    exit(0);
}
```



```

cmake-build-debug - houtonglei@HoutongdeMacBook-Pro - zsh - 100x10
Last login: Tue Dec 21 10:39:46 on ttys001
(base) houtonglei@HoutongdeMacBook-Pro ~/OneDrive - stu.hqu.edu.cn/【5】UNIX、LINUX编程环境/实验/
cmake-build-debug ./tcpclient
char from server = B
(base) houtonglei@HoutongdeMacBook-Pro ~/OneDrive - stu.hqu.edu.cn/【5】UNIX、LINUX编程环境/实验/
cmake-build-debug

```

服务端:

```

#include <sys/socket.h>
#include <stdio.h>
#include <netinet/in.h>
#include <unistd.h>
#include <stdlib.h>

int main() {
    int server_sockfd, client_sockfd;
    int server_len;
    socklen_t client_len;
    struct sockaddr_in server_address, client_address;
    unlink("server_socket");

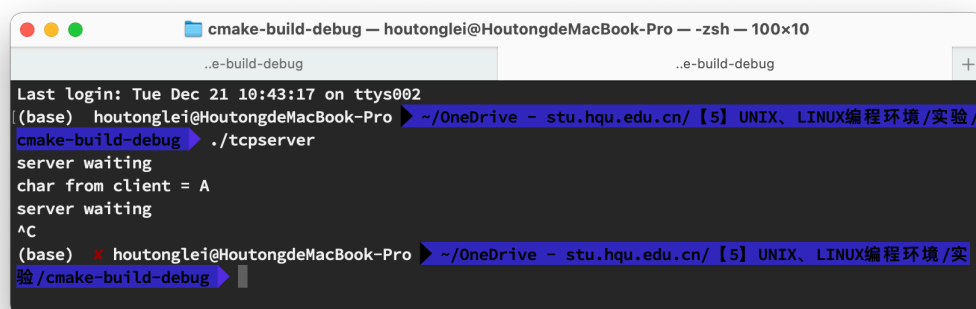
    server_sockfd = socket(AF_INET, SOCK_STREAM, 0);
    server_address.sin_family = AF_INET;
    server_address.sin_addr.s_addr = htonl(INADDR_ANY);
    server_address.sin_port = htons(9734);
    server_len = sizeof(server_address);
    bind(server_sockfd, (struct sockaddr *) &server_address,
server_len);
    listen(server_sockfd, 5);
    while(1) {
        char ch;
        puts("server waiting");
        client_len = sizeof(server_address);
        client_sockfd = accept(server_sockfd, (struct sockaddr
*)&client_address, &client_len);
        read(client_sockfd, &ch, 1);
        printf("char from client = %c\n", ch);
        ch++;
    }
}

```

```

        write(client_sockfd, &ch, 1);
        close(client_sockfd);
    }
    exit(1);
}

```



```

Last login: Tue Dec 21 10:43:17 on ttys002
(base) houtonglei@HoutongdeMacBook-Pro ~: ~/OneDrive - stu.hqu.edu.cn/【5】UNIX、LINUX编程环境/实验/
cmake-build-debug ➤ ./tcpserver
server waiting
char from client = A
server waiting
^C
(base) houtonglei@HoutongdeMacBook-Pro ~: ~/OneDrive - stu.hqu.edu.cn/【5】UNIX、LINUX编程环境/实
验/cmake-build-debug ➤

```

## 五、实验总结

在 TCP 编程中，每一个 Socket 有一个本地的唯一 Socket 号，由操作系统分配。流式套接字可以提供可靠的、面向连接的通信流。流式套接字可用于 HTTP 等需要使数据顺序传递的应用，它使用 TCP 协议保证数据传输的可靠性。服务器端使用 `bind()` 函数，来完成地址结构与 socket 套接字的绑定，这样 ip、port 就固定了，发送端即可发送数据给有明确地址的接收端。