

廈門大學



信息学院软件工程系

《计算机网络》实验报告

题 目 实验五 利用 Socket API 实现许可认证软件

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 4 月 25 日

2022 年 4 月 25 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

掌握应用层文件传输的原理；了解传输过程中传输层协议选用、应用层协议设计和协议开发等概念。

2 实验环境

Linux 操作系统，C 语言，Socket API

3 实验结果

一、手册的代码调试

1、接收数据报，发送数据报

dgsend.c 创建一个 socket，然后用它发送消息到命令行参数传入的特定的主机和端口号。程序 dgreceive.c 使用命令行传过来的端口号建立 socket，然后进入循环，接收和打印从客户端发来的数据报。recvfrom 函数阻塞直到数据报到达，当数据报到达时，消息内容、返回地址和其长度将被复制到缓存中。简单的基于数据报的服务器



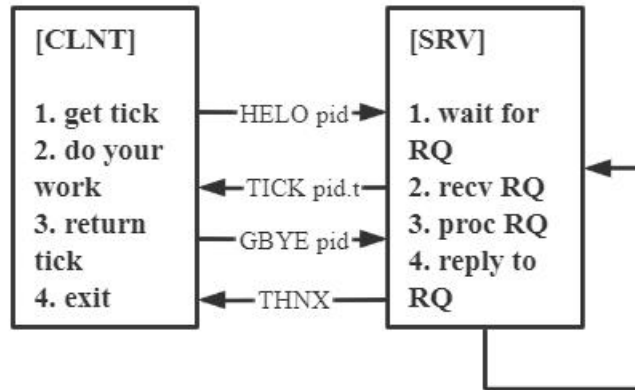
```
Terminal 终端 - ryanmilk@ryanmilk-VirtualBox: ~/program/my
文件(F) 编辑(E) 视图(V) 终端(T) 标签(A) 帮助(H)
ryanmilk@ryanmilk-VirtualBox:~$ cd program/my
ryanmilk@ryanmilk-VirtualBox:~/program/my$ ./dgrecv 4444 &
[1] 6650
ryanmilk@ryanmilk-VirtualBox:~/program/my$ ./dgsend ryanmilk-VirtualBox 4444 "test"
dgrecv: got a message: test
from: 127.0.0.1:41362
```

二、许可证服务器

① 思路简要

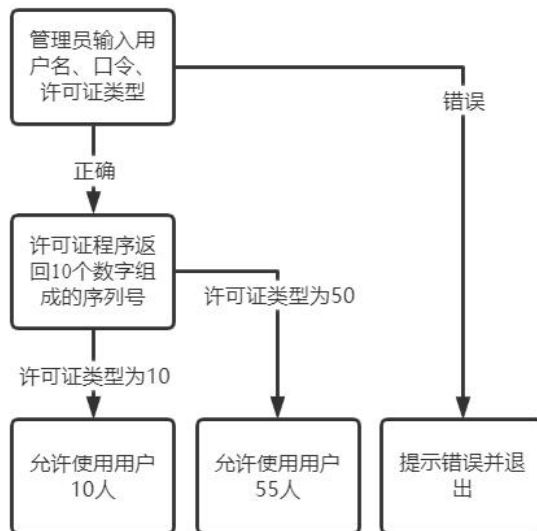
客户端：进行申请（验证、返还）后接受服务端返还消息决定下一步进行的操作。

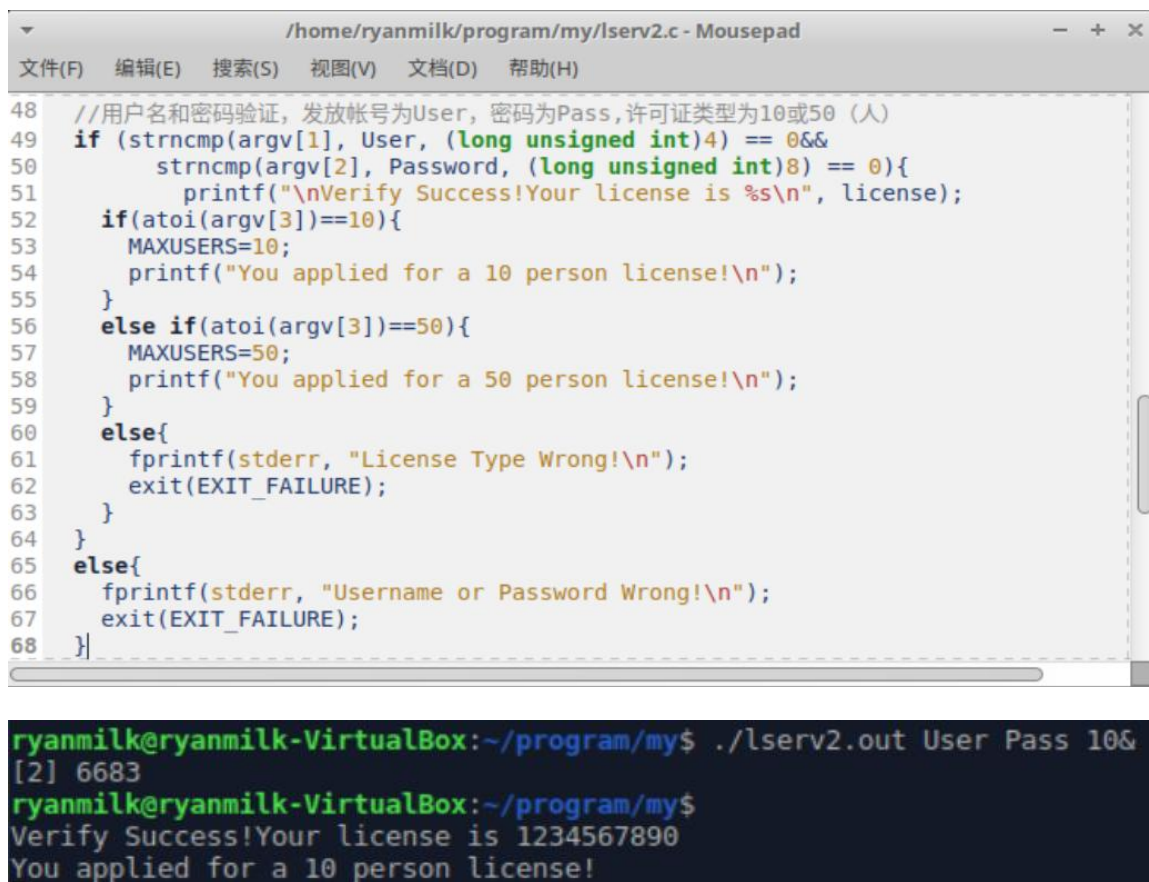
服务端：处理申请（验证、返还）并根据当前状态（剩余票据数，票据是否合法）进行回复。



② 具体分析

1、许可证序列号的获取





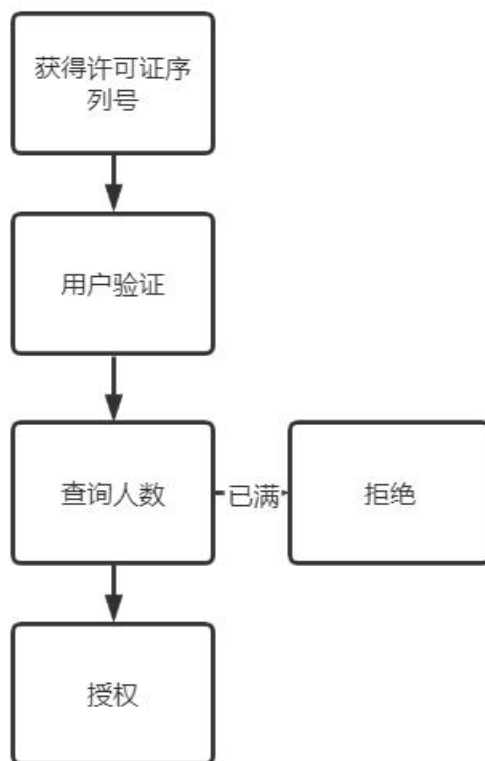
The image shows a code editor window titled "/home/ryanmilk/program/my/lserv2.c - Mousepad" with a menu bar (File, Edit, Search, View, Document, Help). The code is a C program for license verification. It checks if the username and password are correct. If they are, it checks the license type (10 or 50 users). If the license type is correct, it prints the license number and the number of users. If the license type is wrong, it prints an error message. If the username or password is wrong, it prints an error message.

```
48 //用户名和密码验证,发放帐号为User,密码为Pass,许可证类型为10或50 (人)
49 if (strncmp(argv[1], User, (long unsigned int)4) == 0&&
50     strncmp(argv[2], Password, (long unsigned int)8) == 0){
51     printf("\nVerify Success!Your license is %s\n", license);
52     if(atoi(argv[3])==10){
53         MAXUSERS=10;
54         printf("You applied for a 10 person license!\n");
55     }
56     else if(atoi(argv[3])==50){
57         MAXUSERS=50;
58         printf("You applied for a 50 person license!\n");
59     }
60     else{
61         fprintf(stderr, "License Type Wrong!\n");
62         exit(EXIT_FAILURE);
63     }
64 }
65 else{
66     fprintf(stderr, "Username or Password Wrong!\n");
67     exit(EXIT_FAILURE);
68 }
```

Below the code editor is a terminal window showing the execution of the program. The user runs the command `./lserv2.out User Pass 10& [2] 6683`. The output is:

```
ryanmilk@ryanmilk-VirtualBox:~/program/my$ ./lserv2.out User Pass 10&
[2] 6683
ryanmilk@ryanmilk-VirtualBox:~/program/my$
Verify Success!Your license is 1234567890
You applied for a 10 person license!
```

2、用户使用软件时,输入序列号。向许可证服务器发送验证,如果验证成功,输出 **Verification Succeeded**, 验证失败则输出 **wrong** 提示。许可证服务器查询得到该序列号的使用人数,如果未到达上限,则返回授权指令;否则,返回拒绝指令。



```
int main(int ac, char *av[]){
    if(ac!=2||strcmp(av[1], license, (long unsigned int)10) != 0){
        printf("\nwrong: please input correct license\n");
        exit(EXIT_FAILURE);
    }
    else{printf("Verification Succeeded!\n");};|
    set_up();
    if(get_ticket()!=0){
        exit(0);
    }
    do_regular_work();
    validate();
    do_regular_work();
    release_ticket();
    shut_down();
}
```

```
ryanmilk@ryanmilk-VirtualBox:~/program/my$ ./lclnt2.out 1234567890& ./lclnt2.out
1234567890& ./lclnt2.out 1234567890& ./lclnt2.out 1234567890& ./lclnt2.out 1234
567890& ./lclnt2.out 1234567890& ./lclnt2.out 1234567890& ./lclnt2.out 123456789
0& ./lclnt2.out 1234567890& ./lclnt2.out 1234567890& ./lclnt2.out 1234567890&
[2] 6841
[3] 6842
[4] 6843
[5] 6844
[6] 6845
[7] 6846
[8] 6847
[9] 6848
[10] 6849

Verification Succeeded!
[11] 6850
[12] 6851
SERVER: GOT: HELO 6842(127.0.0.1: 59139)
SERVER: SAID TICK 6842.0(127.0.0.1: 59139)
CLIENT[6842]:got ticket6842.0
SuperSleep version 1.0 Running-Licensed Software
ryanmilk@ryanmilk-VirtualBox:~/program/my$
```

```
SERVER: SAID FAIL no tickets availble(127.0.0.1: 44300)
CLIENT[6848]:Could not get ticketFAIL no tickets availble
```

3、处理异常崩溃：客户端崩溃，服务器崩溃。

(1) 客户端崩溃：思路：定期检查票据数组（调试方便，这里 20s 就报告一次，实际可以是 30min），确认其中的每个进程是否还活着，如果否则将进程从数组中去除，释放其占用的票据。

① 服务器两个操作：（1）等待客户请求（2）周期性回收数据。

注意：在处理请求时要关闭 alarm（周期性回收，避免冲突）


```
extern void ticket_reclaim(int a) {
    a += 1; // Unuseful param, just set this line to ignore warning
    char ticket[BUFSIZ];
    for (int i = 0; i < MAXUSERS; ++i) {
        //使用kill系统调用的特殊功能
        //可以通过给进程发送编号为0的信号以确定是否存在
        //如果不存在，内核将不会发送信号，而是返回错误，并设置错误为ESRCH
        if (ticket_array[i] != TICKET_AVAIL &&
            kill(ticket_array[i], 0) == -1 &&
            errno == ESRCH) {
            sprintf(ticket, "%d.%d", ticket_array[i], i);
            narrate("freeing", ticket, NULL);
            ticket_array[i] = TICKET_AVAIL;
            num_tickets_out--;
        }
    }
    alarm(RECLAIM_INTERVAL); //重设时钟
}
```

```
ryanmilk@ryanmilk-VirtualBox:~/program/my$
Verification Succeeded!
SERVER: GOT: HELO 7673(127.0.0.1: 44698)
SERVER: SAID TICK 7673.0(127.0.0.1: 44698)
CLIENT[7673]:got ticket7673.0
SuperSleep version 1.0 Running-Licensed Software

ryanmilk@ryanmilk-VirtualBox:~/program/my$ kill 7673
ryanmilk@ryanmilk-VirtualBox:~/program/my$ SERVER: freeing 7673.0
```

(2) 服务端崩溃：思路：票据验证

每个客户周期性向服务器发送票据的副本（含有数组编号和 PID）。服务器检查迁出列表，如果为空，则会把该票据加到列表中，逐步地，签出列表被重新填入。

```
//解决服务器崩溃：服务器增加函数验证票据
static char *do_validate(char *request) {
    int pid, x;
    if (sscanf(request + 5, "%d.%d", &pid, &x) == 2 && ticket_array[x] == pid)
        return "GOOD Valid ticket";

    narrate("Bogus ticket", request + 5, NULL);
    return "FAIL invalid tickert";
}
```

```
SuperSleep version 1.0 Running-Licensed Software
SERVER: GOT: VALD 6842.0(127.0.0.1: 59139)
SERVER: SAID GOOD Valid ticket(127.0.0.1: 59139)
SuperSleep version 1.0 Running-Licensed Software
```


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

这次编写网络应用程序需要遵循 UDP 协议，在客户端和服务端通信时，并没有建立连接，服务端并没有 `listen-accept` 函数。且本次通信时都是基于数据报的。