

计算机网络第二次实验报告

PB20111686 黄瑞轩

一、实验内容

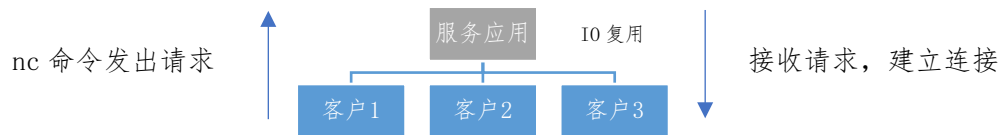
高级聊天程序：一个服务器，多个客户端，服务器负责消息中转，客户端之间可以互相聊天（广播）。

二、实验环境

实验平台：Linux 实验语言：C++

三、实验过程

程序流程图：



主要函数如下：

```
void epoll_add(int epollfd, int fd); // 在 EPOLL 中注册新事件的函数
void client_destroy(int epfd, Client* user); // 用户退出时执行的函数
void client_add(int epfd, int fd); // 用户加入时执行的函数
void send_all(Client* client, std::string s); // 将 client 发来的消息 s 广播
ssize_t receive(Client* client, void* buf, size_t n); // 接收消息至 buf 中
int main(int argc, char** argv); // 主函数，初始化 socket 服务后循环收-发
```

主要变量如下：

```
Client client_list[MAX_CLIENT_NUMBER]; // 目前的用户列表
std::map<int, int> client_map; // 根据 fd 找到其对应的用户编号 index
int connected_clients_num; // 当前已连接用户数
```

程序操作过程：

我编写的程序属于一个服务应用，在命令行上执行时，应提供端口名作为参数，如：

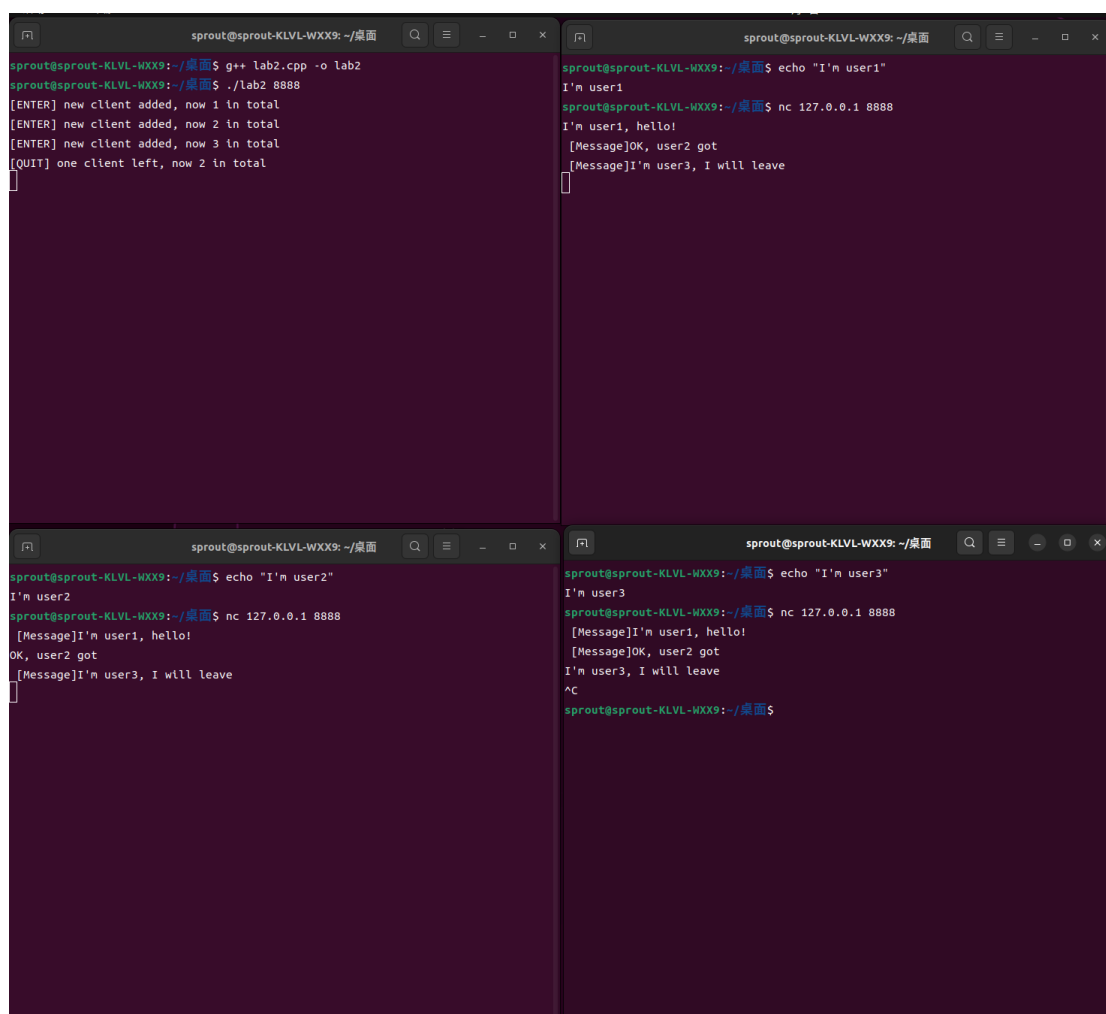
```
$ ./lab2 8888
```

这会在 8888 端口创建服务，各用户使用 netcat 包中的 nc 命令连接此端口即可，如：

```
$ nc 127.0.0.1 8888
```

此时服务应用建立连接，用户端发出的信息将被服务应用转发给每一个连接，用户端也会不断接收服务应用发来的信息。这里连接的建立采用的是 EPOLL 的 IO 复用技术，使用非阻塞 SOCKET 连接以提升 recv 的效率。

四、实验结果截图



```

sprout@sprout-KLVL-WXX9: ~/桌面
sprout@sprout-KLVL-WXX9:~/桌面$ g++ lab2.cpp -o lab2
sprout@sprout-KLVL-WXX9:~/桌面$ ./lab2 8888
[ENTER] new client added, now 1 in total
[ENTER] new client added, now 2 in total
[ENTER] new client added, now 3 in total
[QUIT] one client left, now 2 in total

sprout@sprout-KLVL-WXX9:~/桌面$ echo "I'm user1"
I'm user1
sprout@sprout-KLVL-WXX9:~/桌面$ nc 127.0.0.1 8888
I'm user1, hello!
[Message]OK, user2 got
[Message]I'm user3, I will leave

sprout@sprout-KLVL-WXX9:~/桌面$ echo "I'm user2"
I'm user2
sprout@sprout-KLVL-WXX9:~/桌面$ nc 127.0.0.1 8888
[Message]I'm user1, hello!
OK, user2 got
[Message]I'm user3, I will leave

sprout@sprout-KLVL-WXX9:~/桌面$ echo "I'm user3"
I'm user3
sprout@sprout-KLVL-WXX9:~/桌面$ nc 127.0.0.1 8888
[Message]I'm user1, hello!
[Message]OK, user2 got
I'm user3, I will leave
^C
sprout@sprout-KLVL-WXX9:~/桌面$

```

注：左上方是服务端，其余为用户端。

五、提交文件说明

- lab2.h // 头文件
- lab2.cpp // 源文件
- lab2 // 可执行文件
- lab2.docx // docx 格式的报告
- lab2.pdf // pdf 格式的报告

编译命令：

```
$ g++ lab2.cpp -o lab2
```