实验 4 网络程序设计

中国科学技术大学 曾凡平

4.1 实验目的

掌握网络程序的设计方法

4.2 实验内容

1.用数据报(UDP 协议)套接字实现 C/S 模式的通信；
2.用流式(TCP 协议)套接字实现 C/S 模式的可靠通信；

4.3 实验原理及步骤

套接字（socket）接口是 TCP/IP 网络的 API 接口函数，最先应用于 Unix 操作系统，目前已成为网络程序设计的标准接口。 socket 函数原型为：int socket(int domain, int type, int protocol)。

面向传输层的常用的 Socket 类型有两种：流式 Socket（SOCK_STREAM）和数据报式 Socket（SOCK_DGRAM）。流式 Socket 是一种面向连接的 Socket，针对于面向连接的 TCP 服务应用；数据报式 Socket 是一种无连接的 Socket，对应于无连接的 UDP 服务应用。

4.3.1 用数据报(UDP 协议)套接字实现 C/S 模式的通信

无连接的 UDP 网络通信过程如图 1 所示：

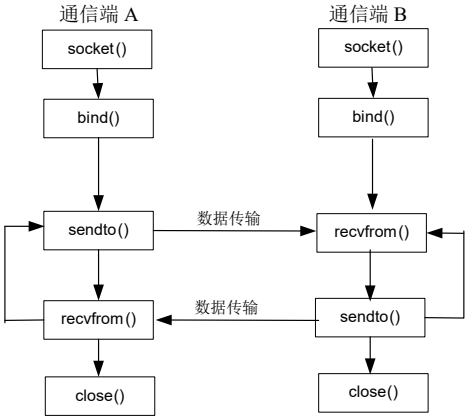

图 1 无连接的 UDP 网络通信

从课程主页下载 Windows 下的示例代码，启动 ClientA 虚拟机，并将代码拷贝到 ClientA。UDP\Sender.c 对应通信端 A，UDP\Receiver.c 对应通信端 B。在命令行下编译程序：

```
cl -o Receiver Receiver.c ws2_32.lib
cl -o Sender Sender.c ws2_32.lib
```

在一个命令行窗口启动 Receiver.exe：

Receiver.exe -p:8080 -i:0.0.0.0

在另一个命令行窗口启动 Sender.exe：

Sender.exe -p:8080 -r:192.168.11.202 -n:5 -b:5 -d:"H"

则实现了两个进程之间的 UPD 通信，如图 2 和图 3 所示：

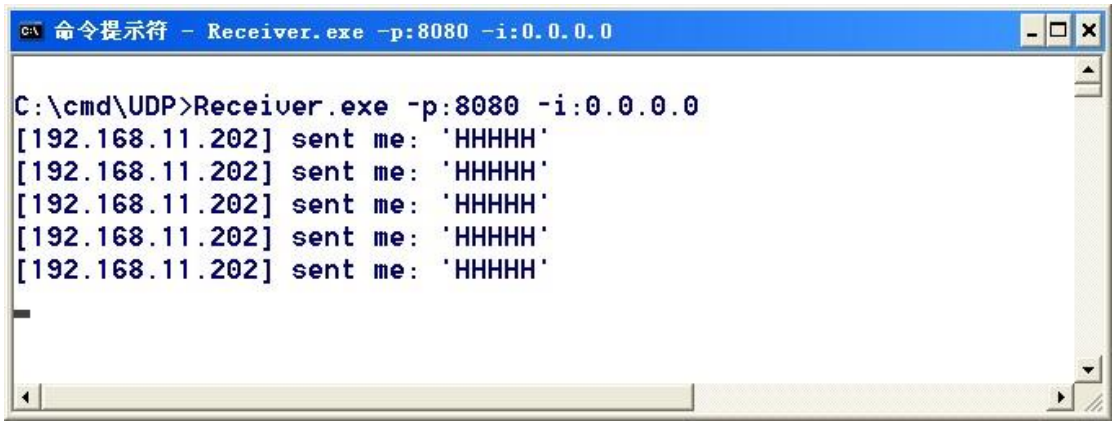

图 2 通信端 B

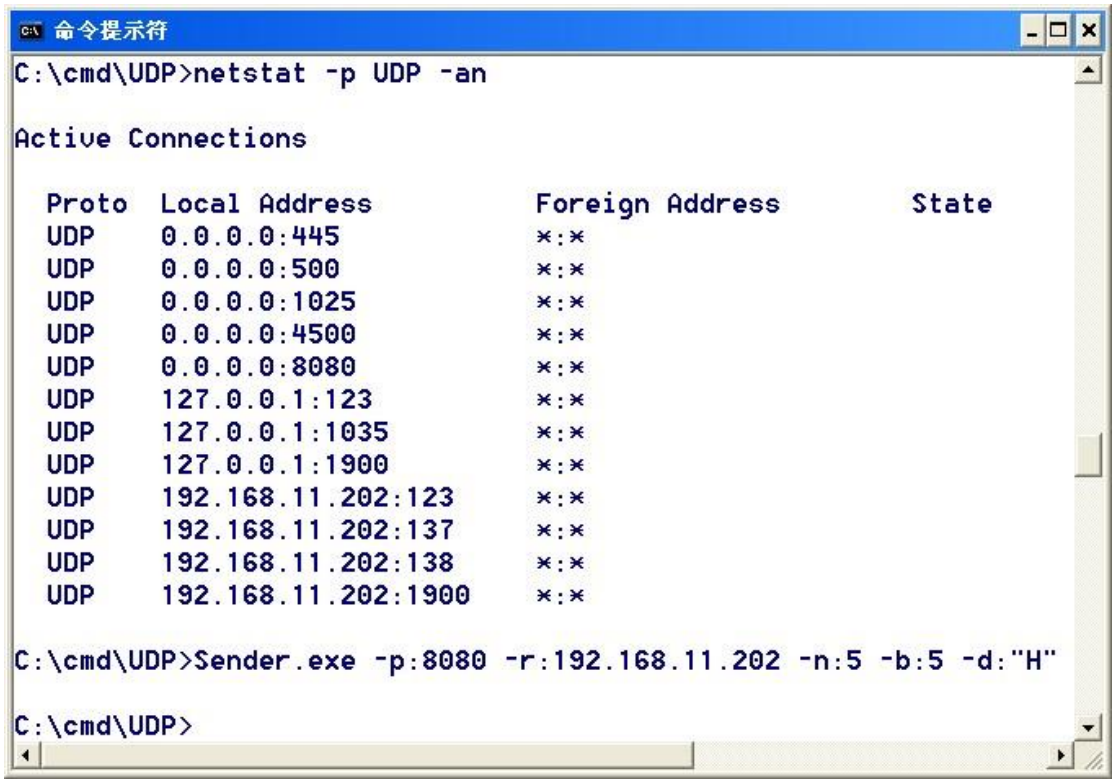

图 3 通信端 A

4.3.2 用流式(TCP 协议)套接字实现 C/S 模式的可靠通信

流式套接字提供可靠的传输服务，服务器在某个端口等待客户端的连接，连接成功后会生成一个**工作套接字**，然后利用**工作套接字**传输数据，传输完毕后关闭**工作套接字**。

- 服务器工作流程：
- 调用 socket 函数创建一个 Socket

- → 调用 bind 函数将其与本机地址以及一个本地端口号绑定
- → 调用 listen 在相应的 socket 上监听
- → 当 accpet 接收到一个连接服务请求时
- → 生成一个新的 socket
- → 生成一个子进程，传输数据
- → 关闭该 socket
- 客户程序工作流程：
- 创建套接口→与远程服务程序连接→读/写数据→终止连接。
- 创建套接口用 socket 函数，socket 函数成功时返回一个套接口描述字。
- 与远程服务程序连接使用 connect 函数，connect 函数调用成功后，套接口描述字就与远程服务程序建立好了连接，可以开始读/写了。
- 使用 read 和 write 函数(或 send 和 recv)读/写数据
- 读写完后可调用函数 close 关闭套接口。示例代码中的 TCP\Server.c 为服务器程序，TCP\Client.c 对应客户端程序。将例子代码拷贝到 C:\CMD 目录，在命令行下编译程序：

```
cl -o Server Server.c ws2_32.lib cl
-o Client Client.c ws2_32.lib 在一
个命令行窗口启动 Server.exe：
Server.exe 在另一个命令行窗口启动
Client.exe：
 Client.exe -p:5150 -s:192.168.11.202 -o -n:2
```

则实现了两个进程之间的 TCP 通信，如图 4 和图 5 所示：

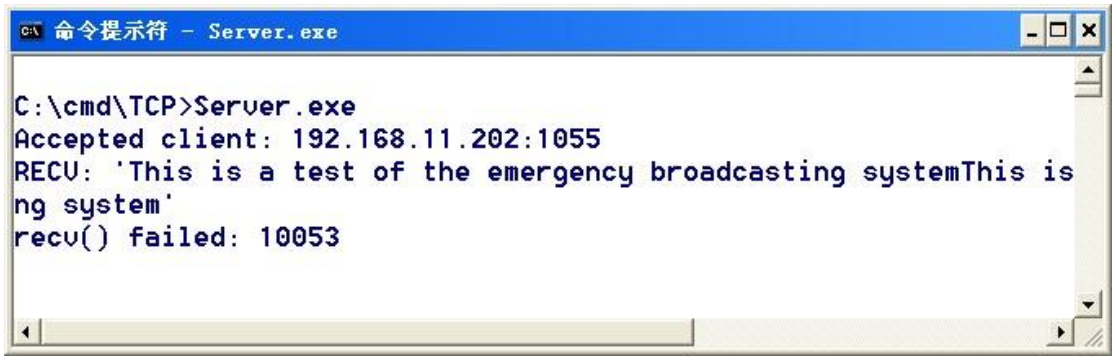

图 4 服务器进程

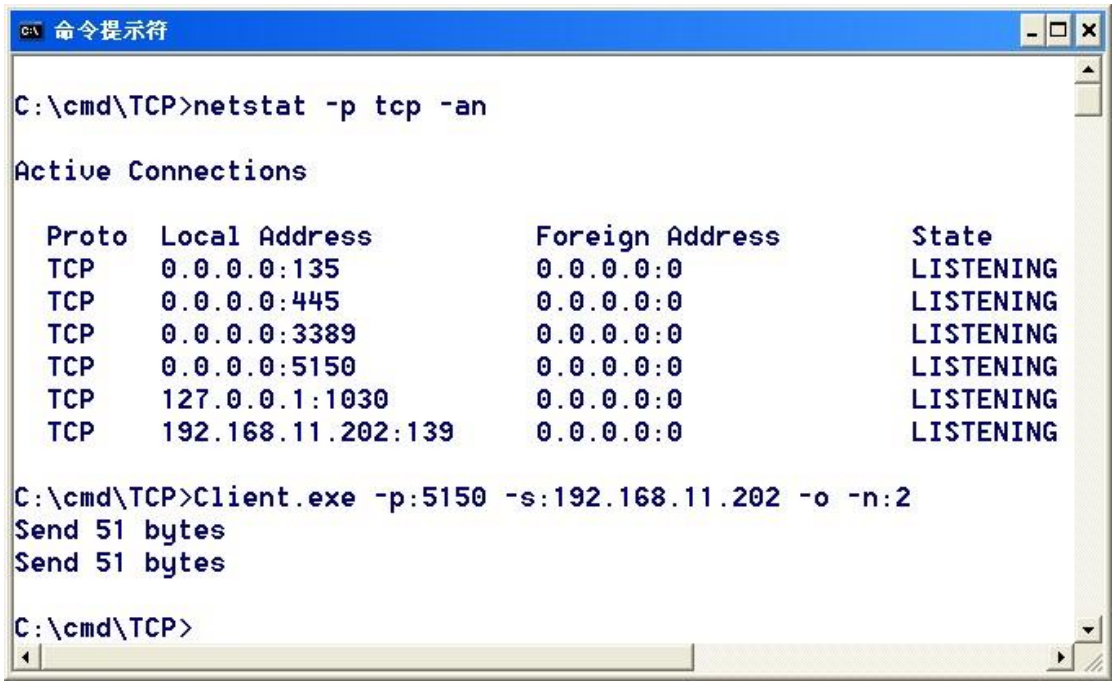

图 5 客户端进程

4.4 上机实践

修改 Client.c，向服务器进程发送其他信息，例如：“嘿！网络程序设计并不难！”。完成实验报告并提交。