

## 五级项目 9：基于 TCP 的重复性服务器

### 1 学时

- 1 学时

### 2 实验目的

- 理解、掌握、应用 Linux 下基于 TCP 协议的重复性服务器的开发方法。

### 3 实验内容

- 本项目编写一个基于 TCP 协议的网络通信，要求客户端通过命令行参数向指定服务器发起连接，服务器向请求连接的客户端发送欢迎信息，客户端接收欢迎信息并显示。服务器在处理完一个客户端后还可以继续处理下一个客户端请求。在以下的例子中，假设服务器对每个客户端处理过程为 5 秒。

### 4 实验原理

基于TCP协议进行通信的方式是面向连接的通信过程，通信进程双方之间在通信之前必须要建立连接从而通过此连接进行数据通信，通信结束后释放连接，是可靠的传输。但是建立连接和释放连接需要一定过程，因此速度有所降低。流式套接字就是使用TCP协议进行网络进程间通信的套接字方式。

使用流式套接字建立连接及通信的步骤如图所示。

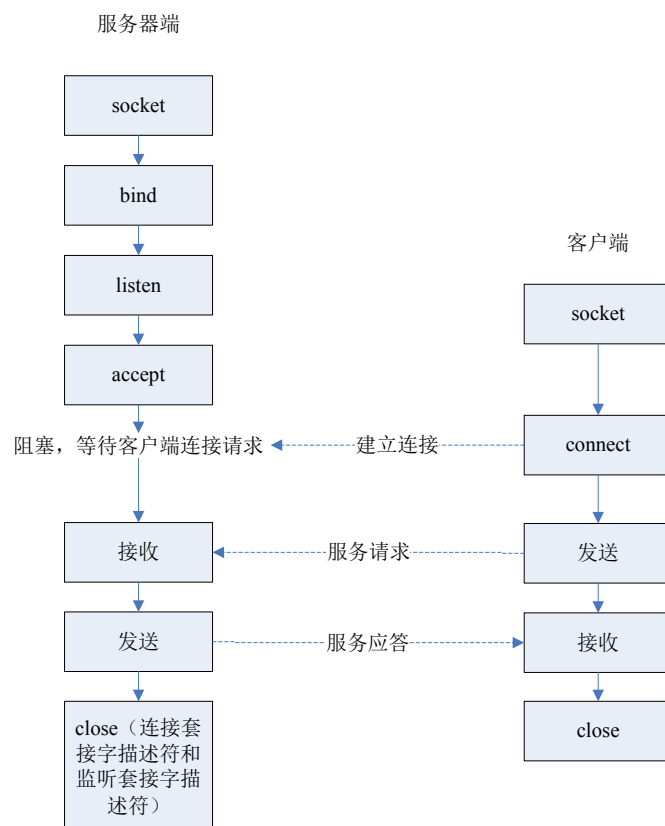


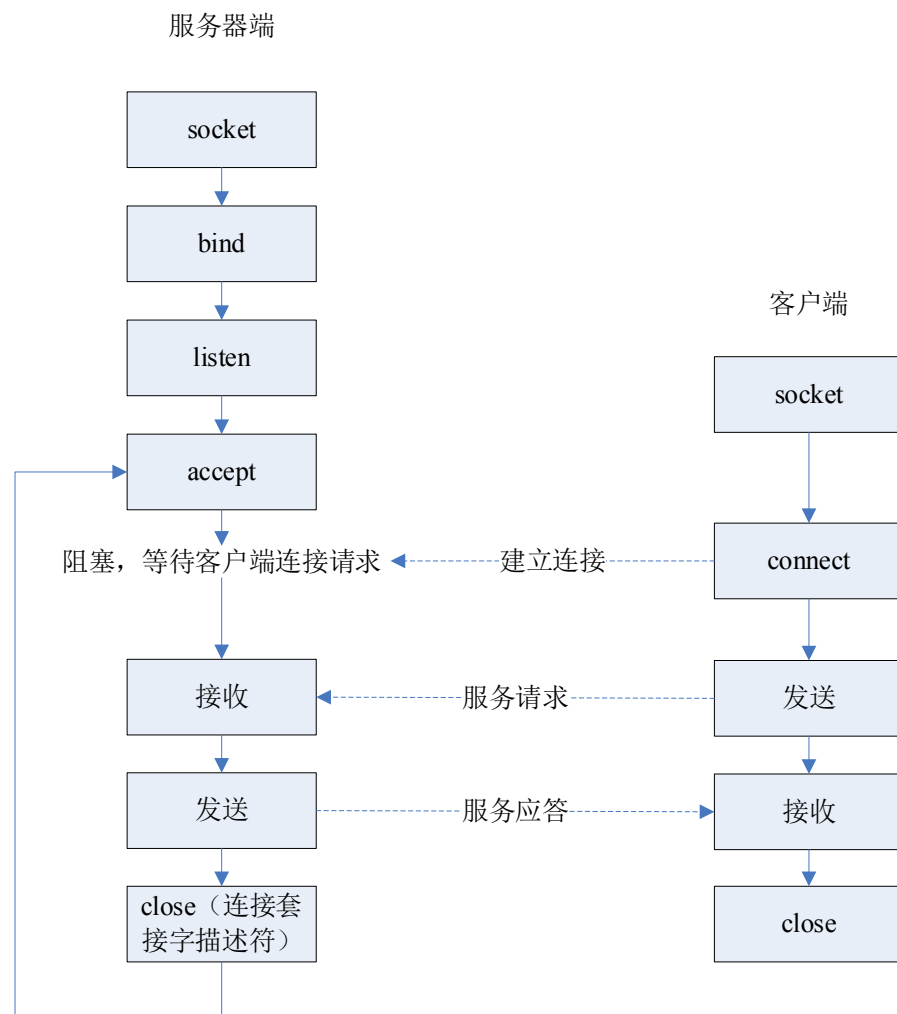
图 流式套接字通信过程

服务器端先启动，进行socket、bind、listen、accept一系列准备工作，然后如果没有客户端请求建立连接，则服务器端处于阻塞状态，等待客户端请求建立连接。

客户端启动后，首先向服务器端发起建立连接请求，得到服务器同意后，连接建立成功，客户端和服务端开始进行数据通信（即数据的发送和接收过程），客户机完成通信过程后释放连接，关闭套接字。

服务器关闭与某客户的连接，并根据情况决定是否结束整个服务器的服务，一般大型服务器是不间断服务的，除非进行系统维护和升级，所以一般服务器将会使用循环方式，不断接受客户请求以及与客户进行通信。另外服务器可以选择一次只处理一个客户请求（重复性服务器），也可以采用多进程或多线程等方式同时处理多个客户请求（并发服务器）。

重复性服务器的流程图如图所示。



## 5 预习要求和技术准备工作

- 掌握 Linux 基本操作
- 掌握 C 语言开发工具的使用
- 掌握基于 TCP 的套接字通信流程

## 6 实验环境

- PC 机
- 在 Windows 环境中的 VMware 虚拟机上运行 Ubuntu 操作系统或者独立的 Ubuntu 操作系统

- 基于 Linux 的 vi 编辑器和 gcc 编译器

## 7 实验设计及操作步骤

7.1 以 root 身份登录系统，在/home 目录中创建目录 exp59

```
cd /home
```

```
mkdir exp59
```

7.2 进入刚创建的目录

```
cd exp59
```

7.3 使用 vi 编辑文件，文件名分别是 nettcpserver.c 和 nettcpclient.c。

7.4 编写程序，实现要求的功能

7.5 编译可执行文件。

```
gcc -o nettcpserver nettcpserver.c
```

```
gcc -o nettcpclient nettcpclient.c
```

7.6 运行程序

在一个终端中运行服务器端nettcpserver。

```
#!/nettcpserver
```

在其他多个终端中分别运行客户端nettcpclient。

```
#!/nettcpclient
```

运行结果。

服务器端：

```
received a connection from 127.0.0.1
Simulation processing start(5s).....
Simulation processing stop.....
received a connection from 127.0.0.1
Simulation processing start(5s).....
Simulation processing stop.....
received a connection from 127.0.0.1
Simulation processing start(5s).....
Simulation processing stop.....
received a connection from 127.0.0.1
Simulation processing start(5s).....
Simulation processing stop.....
```

每个客户端都会显示：

```
Received: Hello, you are connected!
```

## 8 实验报告提交要求：

将实验操作每个步骤中的命令、源程序以及截图写入实验报告，实验报告命

名为“学号姓名-实验 59.doc”，交给指定人员。

## 9 项目思考

思考重复性服务器的优缺点。