

## TCP/IP网络编程(六)

笔记本： 网络编程

创建时间： 2018/11/9 19:36

更新时间： 2018/11/9 20:51

作者： xiangkang94@outlook.com

标签： 第六章(基于UDP的服务器端/客户端)

---

### 1. UDP和TCP的主要差异

- 收发数据前后进行的链接设置及清除过程
- 收发数据过程中为保证可靠性而添加的流控制

### 2. UDP服务器端和客户端均只需要一个套接字

- 在UDP中，不管是服务器端还是客户端都只要一个套接字，不同于TCP，可能需要多个套接字

### 3. 基于UDP的数据I/O函数

- UDP套接字不会保持链接状态（可以想象成邮筒功能），因此每次传递数据的时候都要添加目标地址信息

```
#Include <sys/socket.h>
ssize_t sento(int sock,void *buff, size_t nbytes,int flags, struct sockaddr *to, socklen_t *addrlen )
//to 存有目标地址信息的sockaddr结构体变量的地址值
ssize_t recvfrom(int sock,void *buff, size_t nbytes,int flags, struct sockaddr *from, socklen_t
*addrlen )
// from 存有发送地址段信息的sockaddr结构体变量的地址值
```

### 4. UDP客户端套接字的地址分配

- TCP客户端通过connect函数自动完成此过程，而UDP中没有，调用sento函数传输数据前应完成对套接字的地址分配工作，因此调用bind函数。
- 如果调用sendto函数时发现尚未分配地址信息，则在首次调用sendto的时候给相应的套接字自动分配IP和端口

### 5. UDP的数据传输存在数据边界

- **输入函数的调用次数应和输出函数的调用次数完全一致**，这样才能保证接受全部已经发送的数据

### 6. 针对UDP套接字的connect函数

- 由于UDP通过sendto函数传输阶段需要注册目标IP和端口号，大致三阶段
  - 想UDP套接字注册目标IP和端口
  - 传输数据
  - 删除UDP套接字中注册的目标地址信息
- 由于每次调用sento都会重复上述过程，**所以与同一主机进行长时间通信时，将UDP套接字变成已链接的套接字会提高效率**，这样做之后，还可以使用write、read函数进行通信