

三 TCP和UDP通信

▼ 1. tcp通信的基本步骤

- 服务端: socket-----bind-----listen-----accept-----recv-----send-----close。
- 客户端: socket-----connect---send---recv-----close。
- TCP通信像打电话

▼ 2. udp通信基本步骤

- 服务端: socket, bind, recvfrom, sendto, close。
- 客户端: socket, -----sendto, recvfrom, close。
- UDP像发短信

▼ 3. socket函数: 生成一个套接口描述符。

- ▼ int socket(int domain,int type,int protocol)。
 - domain:AF_INET表示ipv4,AF_INET6表示ipv6。
 - type :tcp:SOCK_STREAM, udp:SOCK_DGRAM。
 - int socketFd = socket(AF_INET,SOCK_STREAM,0);

▼ 4. bind函数:用来绑定一个端口号和ip地址, 使套接口与指定端口号和ip地址相关联。

- ▼ int bind(int sockfd,struct sockaddr* my_addr,socklen_t addrlen);
 - struct sockaddr_in serAddr;
 - serAddr.sin_family=AF_INET;
 - serAddr.sin_port=htons(port);
 - serAddr.sin_addr.s_addr=inet_addr(ip);
 - bind(socketFd,(struct sockaddr*)&serAddr,sizeof(serAddr));

▼ 5. listen:使ip和端口处于监听状态, 等待客户的连接请求。

- int listen(int socketFd, int backlog) ;backlog 是队列大小, 队列存放已完成三次握手的连接, accept()函数从这个队列中取走已完成握手的连接。

▼ 6. accept:接受远程计算机的连接请求, 建立与客户端之间的通信连接。

- int newFd=accept(socketFd,(struct sockaddr*)&client,&clientLen);
- 返回一个新的socket标识符, 跟客户端之间的数据传输就要通过这个新的socket来进行。

▼ 7. recv: 用socket套接字接受远端发来的数据。

- int recv_len = recv(newFd,buf,len,flags);
- recv是把数据从接收缓冲区读取到用户的buf中。

▼ 8. send: 用socket套接字发送数据给远端的主机。

- `int send_len = send(newFd,buf,len,flags);`
- send是把数据从用户态buf拷贝到发送缓冲区，并不代表发送到网络，缓冲区数据的发送时机，是由底层的网卡，不是我们能控制的。

▼ 9. close(sockfd);断开连接，关闭描述符（引用计数减一），释放资源。

- shutdown()半关闭。

▼ 10. connect(int sockfd,&serAddr,addrlen);

▼ 用来请求连接远程主机。

- `struct sockaddr_in addr;`
- `bzero(&serAddr,sizeof(serAddr));`
- `serAddr.sin_family=AF_INET;`
- `serAddr.sin_port=htons(port);`
- `serAddr.sin_addr.s_addr=inet_addr(ip);`
- `connect(sockfd,(struct sockaddr*)&serAddr,sizeof(serAddr));`

▼ 11.sendto 添加了对端信息，可以用于UDP通信。

- `ssize_t sendto(int sockfd, const void *buf, size_t len, int flags,const struct sockaddr *dest_addr, socklen_t addrlen)`

▼ 12. recvfrom 添加了对端信息，可以用于UDP通信。

- `ssize_t recvfrom(int sockfd, void *buf, size_t len, int flags,struct sockaddr *src_addr, socklen_t *addrlen);`
- TCP通信像管道，UDP通信像消息队列。