



链滴

socket 编程

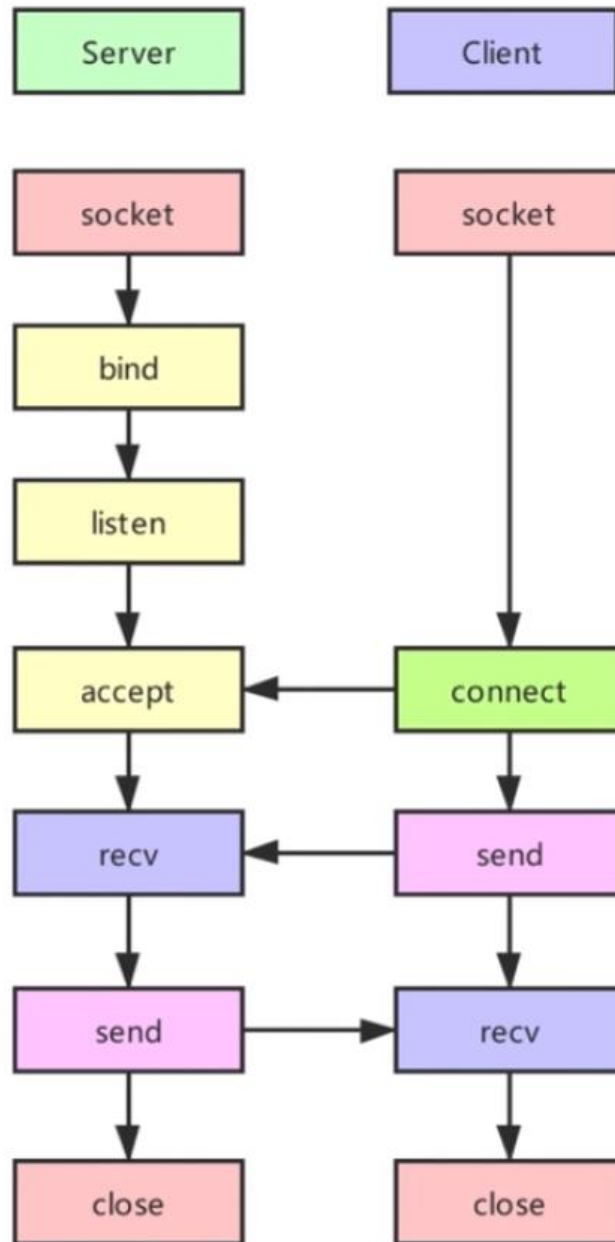
作者: [sailorlee](#)

原文链接: <https://ld246.com/article/1603327577835>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

1 实现一个简单的 TCP Server



服务端会相对麻烦一些代码如下：

```
#include <iostream>
#include <netinet/in.h>
#include <sys/socket.h>
#include <unistd.h>
#include <stdlib.h>

#define PORT 8111
#define MESSAGE_LEN 1024
using namespace std;
```

```

int main(int arc, char *argv[]) {
    int ret = -1;
    int socket_fd = -1, accept_fd = -1;
    int on = 1;
    int backlog = 10;

    struct sockaddr_in localaddr,remoteaddr;

    char in_buff[MESSAGE_LEN] = {0,};

    //创建一个socket
    socket_fd = socket(AF_INET, SOCK_STREAM, 0);
    if (socket_fd == -1) {
        cout << "Failed to create socket!" << endl;
        exit(-1);
    }
    // SO_REUSEADDR 允许套接口和一个已在使用中的地址捆绑
    ret = setsockopt(socket_fd,
                    SOL_SOCKET,
                    SO_REUSEADDR,
                    &on,
                    sizeof(on));
    if (ret == -1) {
        cout << "Failed to set socket options" << endl;
    }
    localaddr.sin_family = AF_INET;
    localaddr.sin_port = PORT;
    localaddr.sin_addr.s_addr = INADDR_ANY;
    // 绑定
    ret = bind(socket_fd, (struct sockaddr *) &localaddr, sizeof(struct sockaddr));
    if (ret == -1) {
        cout << "Failed to bind addr!" << endl;
        exit(-1);
    }
    // 侦听
    ret = listen(socket_fd, backlog);
    if (ret == -1) {
        cout << " Failed to listen socket!" << endl;
        exit(-1);
    }
    for (;;) {
        socklen_t addr_len = sizeof(struct sockaddr);
        accept_fd = accept(socket_fd,
                          (struct sockaddr *) &remoteaddr,
                          &addr_len);
    // 开始接受数据
    for (;;) {
    // 第一个空从accept_fd接受数据,
        ret = recv(accept_fd, (void *) in_buff, MESSAGE_LEN, 0);
        if (ret == 0)//没有发数据和我们的断开连接了
        {
            break;
        }
        cout << "rev:" << in_buff << endl;
    }
}

```

```
        send(accept_fd, (void *) in_buff, MESSAGE_LEN, 0);
    }
    close(accept_fd);
}
// 把socket关闭
close(socket_fd);
return 0;
}
```