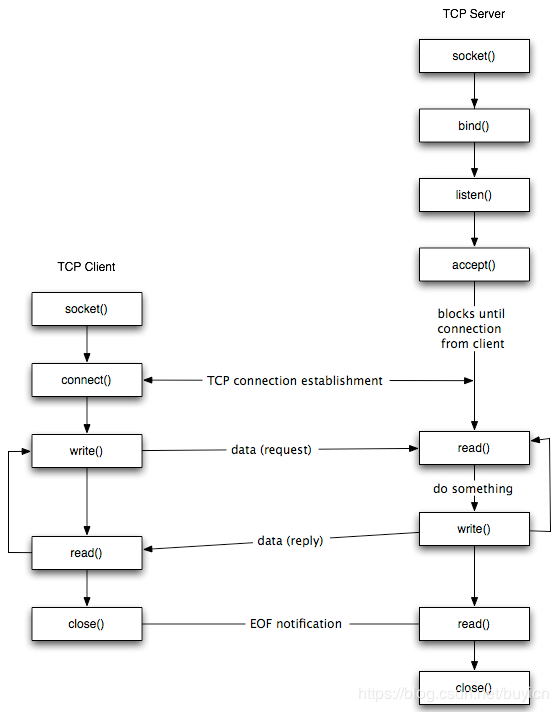
TCP和UDP的Client-Server交互流程

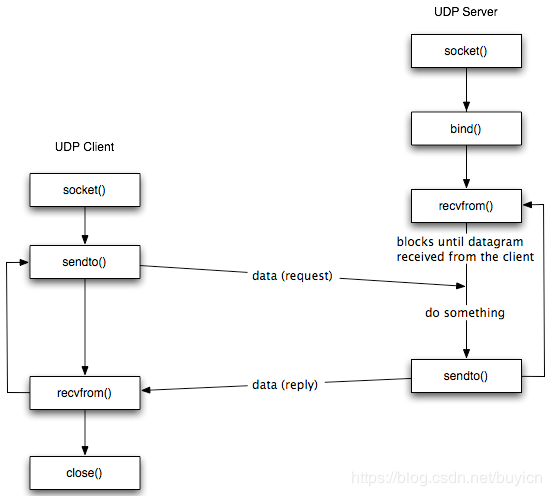
原创buyicn 最后发布于2019-07-03 09:04:37 阅读数 160 收藏

展开

1.TCP Client-Server交互流程



2.UDP Client-Server交互流程



注：图片来源（https://www.cs.dartmouth.edu/~campbell/cs60/socketprogramming.html#x1-100013）。

3.一个形象的比喻

建立TCP连接就像是一个电话系统。

socket()：你有了一部手机。

bind()：激活SIM卡，手机有了一个电话号码。（注：相当于把手机绑定到了一个电话号码）

listen()：开机或关掉飞行模式，手机处于可接听电话的状态。

connect()：一个人知道你的电话号码，开始拨打你的电话。

accept()：你听到手机振铃，按下了接听按钮。之后，你有了一个私有的通信链路。

write()/read()：通过私有的通信链路，你和对方开始通话。

close()：一方挂断了电话，关闭了你们之间的私有通信链路。

4.深入挖掘一

TCP是面向连接的协议，也就是说server和client之间发送/接收数据之前，需先建立连接。

因此，TCP的server需依次调用以下函数：

listen() ：使一个绑定的TCP套接字的状态由CLOSE转换成监听（LISTEN），这样来自客户的外来连接就可在该套接字上由内核接受。

accept()：它在侦听套接字sockfd的挂起连接队列上提取第一个连接请求，创建一个新的连接套接字，并返回一个引用该套接字的新文件描述符。 通常情况下，服务器进程在accept调用中进入休眠（sleep）状态，等待某个客户连接的到达并被内核接受。TCP连接使用三次握手来建立连接。握手完成时accept返回，其返回值是一个称为“已连接描述符（connected descriptor）”的新描述符。该描述符用于与新连接的那个客户端通信。accept为每个连接到本服务器的客户返回一个新描述符。

TCP的client需调用以下函数：

connect() ：为一个套接字分配一个自由的本地端口号，并试图获得一个新的TCP连接。

而作为无连接的UDP协议，则不需要上述三个函数。

注：参考文章（https://en.wikipedia.org/wiki/Berkeley\_sockets）。

5.深入挖掘二

关于accept()函数，微软的API指南中有非常详细的一段描述：

The accept function extracts the first connection on the queue of pending connections on socket s. It then creates and returns a handle to the new socket. The newly created socket is the socket that will handle the actual connection; it has the same properties as socket s, including the asynchronous events registered with the WSAAsyncSelect or WSAEventSelect functions.

The accept function can block the caller until a connection is present if no pending connections are present on the queue, and the socket is marked as blocking. If the socket is marked as nonblocking and no pending connections are present on the queue, accept returns an error as described in the following. After the successful completion of accept returns a new socket handle, the accepted socket cannot be used to accept more connections. The original socket remains open and listens for new connection requests.

The parameter addr is a result parameter that is filled in with the address of the connecting entity, as known to the communications layer. The exact format of the addr parameter is determined by the address family in which the communication is occurring. The addrlen is a value-result parameter; it should initially contain the amount of space pointed to by addr; on return it will contain the actual length (in bytes) of the address returned.

The accept function is used with connection-oriented socket types such as SOCK\_STREAM. If addr and/or addrlen are equal to NULL, then no information about the remote address of the accepted socket is returned.

6.深入挖掘三

下面两个讨论，是非常有意思的参考。

（1）https://stackoverflow.com/questions/489036/how-does-the-socket-api-accept-function-work?rq=1

（2）https://stackoverflow.com/questions/774571/server-vs-client-socket-low-level-details

7.深入挖掘四

TCP三次握手图。

注：参考《UNIX网络编程卷1：套接字联网API》。

8.深入挖掘五

关于accept()函数：

int accept (int sockfd, struct sockaddr \*cliaddr, socklen\_t \*addrlen);

我们称它的第一个参数（sockfd）为监听套接字（listening socket）描述符——它由socket()函数创建，随后用作bind()和listen()函数的第一个参数。称它的返回值为已连接套接字（connected socket）描述符。

（1）一个服务器通常仅仅创建一个监听套接字，它在该服务器的生命周期内一直存在。

（2）内核为每个由服务器进程接受的客户连接创建一个已连接套接字（它的TCP三次握手过程已经完成）。当服务器完成对某个给定客户端的服务时，调用close()函数，相应的已连接套接字就被关闭。如果已连接套接字被关闭，之后就需要重新调用accept()函数来获得新的已连接套接字。

————————————————

版权声明：本文为CSDN博主「buyicn」的原创文章，遵循 CC 4.0 BY-SA 版权协议，转载请附上原文出处链接及本声明。

原文链接：https://blog.csdn.net/buyicn/article/details/94546026