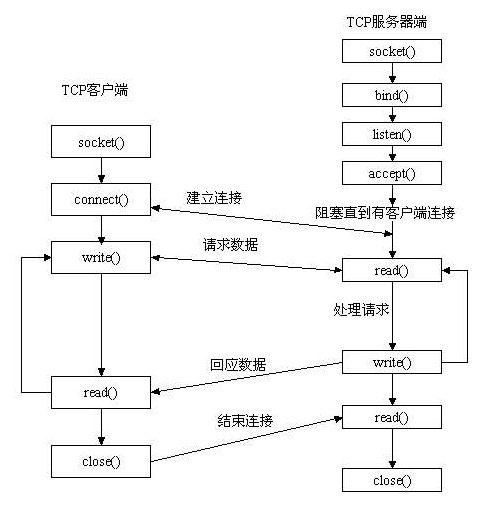
**Socket简单介绍**

1. **Socket是什么？**

socket起源于Unix，而Unix/Linux基本哲学之一就是“一切皆文件”，都可以用“打开open –>读写write/read–>关闭close”关闭close”模式来操作。我的理解就是Socket就是该模式的一个实现，socket即是一种特殊的文件，一些socket函数就是对其进行的操作（读/写IO、打开、关闭）。

1. **流程图**

一个生活中的场景。你要打电话给一个朋友，先拨号，朋友听到电话铃声后提起电话，这时你和你的朋友就建立起了连接，就可以讲话了。等交流结束，挂断电话结束此次交谈。（下图是TCP协议的原理图）



先从服务器端说起。服务器端先初始化Socket，然后与端口绑定(bind)，对端口进行监听(listen)，调用accept阻塞，等待客户端连接。在这时如果有个客户端初始化一个Socket，然后连接服务器(connect)，如果连接成功，这时客户端与服务器端的连接就建立了。客户端发送数据请求，服务器端接收请求并处理请求，然后把回应数据发送给客户端，客户端读取数据，最后关闭连接，一次交互结束。

1. **socket的基本操作**

3.1、socket()函数

3.2、bind()函数

3.3、listen()、connect()函数

3.4、accept()函数

3.5、read()、write()或者是recv()和send()函数等

3.6、close()函数

具体函数解释请参考以下网址：

http://learn.tsinghua.edu.cn/kejian/data/9281/54841/tcpip/TCP/SocketHSSM.htm

1. TCP三次握手/四次挥手详解

三次握手：

第一次握手：主机A发送位码为syn＝1,随机产生seq number=1234567的数据包到服务器，主机B由SYN=1知道，A要求建立联机；

第二次握手：主机B收到请求后要确认联机信息，向A发送ack number=(主机A的seq+1),syn=1,ack=1,随机产生seq=7654321的包

第三次握手：主机A收到后检查ack number是否正确，即第一次发送的seq number+1,以及位码ack是否为1，若正确，主机A会再发送ack number=(主机B的seq+1),ack=1，主机B收到后确认seq值与ack=1则连接建立成功。

完成三次握手，主机A与主机B开始传送数据。

四次挥手：

（1）客户端A发送一个FIN，用来关闭客户A到服务器B的数据传送。

（2）服务器B收到这个FIN，它发回一个ACK，确认序号为收到的序号加1。和SYN一样，一个FIN将占用一个序号。

（3）服务器B关闭与客户端A的连接，发送一个FIN给客户端A。

（4）客户端A发回ACK报文确认，并将确认序号设置为收到序号加1。

1. TCP和UDP区别



1. 基于连接（三次握手）与无连接
2. 对系统资源的要求（TCP较多，UDP少）
3. UDP程序结构较简单
4. 流模式与数据报模式
5. TCP保证数据正确性，UDP可能丢包
6. TCP保证数据顺序，UDP不保证

具体编程时的区别

1. socket()的参数不同
2. UDP Server不需要调用listen和accept
3. UDP收发数据用sendto/recvfrom函数
4. TCP：地址信息在connect/accept时确定  
   UDP：在sendto/recvfrom函数中每次均 需指定地址信息
5. UDP：shutdown函数无效