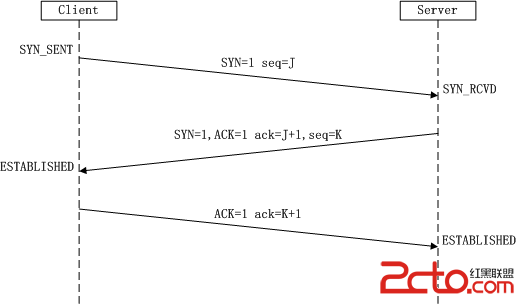
一．TCP报文格式



二．三次握手

所谓三次握手（Three-Way Handshake）即建立TCP连接，就是指建立一个TCP连接时，需要客户端和服务端总共发送3个包以确认连接的建立。在socket编程中，这一过程由客户端执行connect来触发，整个流程如下图所示：



1. 第一次握手

Client发送一个连接请求，其中标志位SYN=1, 一个随机序列SEQ=J，并且将请求发送给 Server 端， Client进入SYN\_SENT状态

1. 第二次握手

Server端收到一个连接建立请求，同意建立连接，返回一个SYN=1,ACK=1,ack=J+1,SEQ=k 的应答请求，并且等待回复

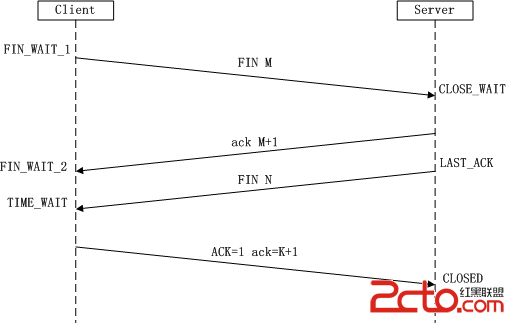
1. 第三次握手

Client收到握手回复，查看是否符合，然后回复给SERVER ACK=1, ack=k+1

为什么不采用两次握手？

采用两次握手，那么若Client向Server发起的包A1如果在传输链路上遇到的故障，导致传输到Server的时间相当滞后，在这个时间段由于Client没有收到Server的对于包A1的确认，那么就会重传一个包A2，假设服务器正常收到了A2的包，然后返回确认B2包。由于没有第[三次握手](https://www.baidu.com/s?wd=%E4%B8%89%E6%AC%A1%E6%8F%A1%E6%89%8B&tn=44039180_cpr&fenlei=mv6quAkxTZn0IZRqIHckPjm4nH00T1dhuyczmWD3PWKbuAcvuWm30ZwV5Hcvrjm3rH6sPfKWUMw85HfYnjn4nH6sgvPsT6KdThsqpZwYTjCEQLGCpyw9Uz4Bmy-bIi4WUvYETgN-TLwGUv3EPjc3PH6YPjbsPjTLrH01rjDz" \t "/home/jackie/Documents\\x/_blank)，这个时候Client和Server已经建立连接了。再假设A1包随后在链路中传到了Server，这个时候Server又会返回B1包确认，但是由于Client已经清除了A1包，所以Client会丢弃掉这个确认包，但是Server会保持这个相当于“僵尸”的连接。  
所以采用两次握手，有可能会浪费Server的网络资源。

三．四次挥手



由于TCP连接时全双工的，因此，每个方向都必须要单独进行关闭，这一原则是当一方完成数据发送任务后，发送一个FIN来终止这一方向的连接，收到一个FIN只是意味着这一方向上没有数据流动了，即不会再收到数据了，但是在这个TCP连接上仍然能够发送数据，直到这一方向也发送了FIN。

1. 第一个挥手：Client向Server端发送请求要求FIN 序列是M
2. 第二次挥手：Server端向 Client端发送ack信息 M+1，