TCP协议特点和三次握手／四次挥手

2017-07-215232浏览



DocMike

1实战 · 27手记 · 29推荐

[TA的实战](javascript:void(0);)

[一.TCP的特性](http://coding.imooc.com/class/101.html)

[TCP](http://coding.imooc.com/class/101.html)提供一种面向连接的、可靠的字节流服务  
在一个TCP连接中，仅有两方进行彼此通信。广播和多播不能用于TCP  
TCP使用校验和，确认和[重传机制](http://coding.imooc.com/class/101.html" \t "_blank)来保证可靠传输  
TCP给数据分节进行排序，并使用累积确认保证数据的顺序不变和非重复  
TCP使用滑动窗口机制来实现流量控制，通过动态改变窗口的大小进行拥塞控制  
注意：TCP 并不能保证数据一定会被对方接收到，因为这是不可能的。TCP 能够做到的是，如果有可能，就把数据递送到接收方，否则就（通过放弃重传并且中断连接这一手段）通知用户。因此准确说 TCP 也不是 100% 可靠的协议，它所能提供的是数据的可靠递送或故障的可靠通知。

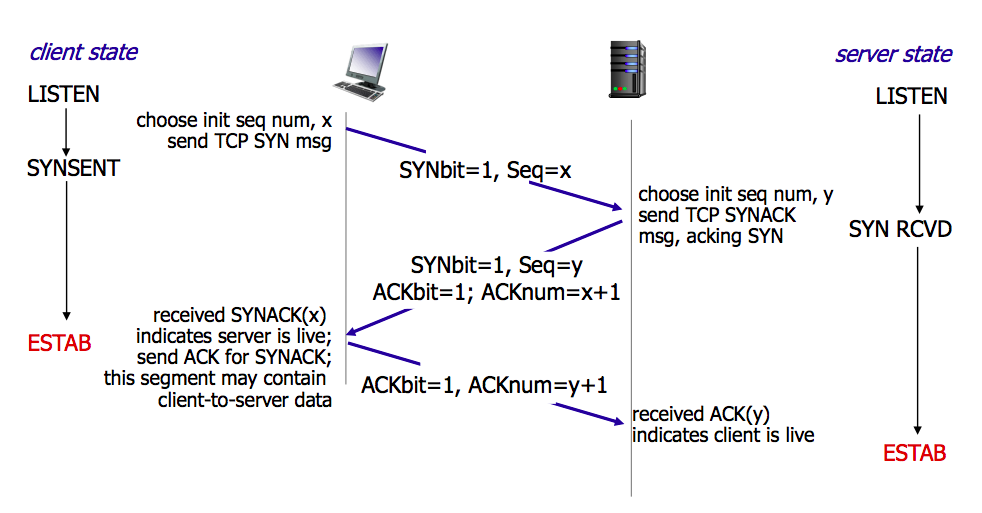
二.[三次握手与四次挥手](http://coding.imooc.com/class/101.html)

所谓三次握手(Three-way Handshake)，是指[建立一个 TCP 连接](http://coding.imooc.com/class/101.html" \t "_blank)时，需要客户端和服务器总共发送3个包。  
三次握手的目的是连接服务器指定[端口](http://coding.imooc.com/class/101.html)，建立 TCP 连接，并同步连接双方的序列号和确认号，交换 TCP 窗口大小信息。在 socket 编程中，客户端执行 connect() 时。将触发三次握手。

第一次握手(SYN=1, seq=x):  
客户端发送一个 TCP 的 SYN 标志位置1的包，指明客户端打算连接的服务器的端口，以及初始序号 X,保存在包头的序列号(Sequence Number)字段里。发送完毕后，客户端进入 SYN\_SEND 状态。

第二次握手(SYN=1, ACK=1, seq=y, ACKnum=x+1):  
服务器发回确认包(ACK)应答。即 SYN 标志位和 ACK 标志位均为1。服务器端选择自己 ISN 序列号，放到 Seq 域里，同时将确认序号(Acknowledgement Number)设置为客户的 ISN 加1，即X+1。 发送完毕后，服务器端进入 SYN\_RCVD 状态。

第三次握手(ACK=1，ACKnum=y+1)  
客户端再次发送确认包(ACK)，SYN 标志位为0，ACK 标志位为1，并且把服务器发来 ACK 的序号字段+1，放在确定字段中发送给对方，并且在数据段放写ISN的+1发送完毕后，客户端进入 ESTABLISHED 状态，当服务器端接收到这个包时，也进入 ESTABLISHED 状态，TCP 握手结束。

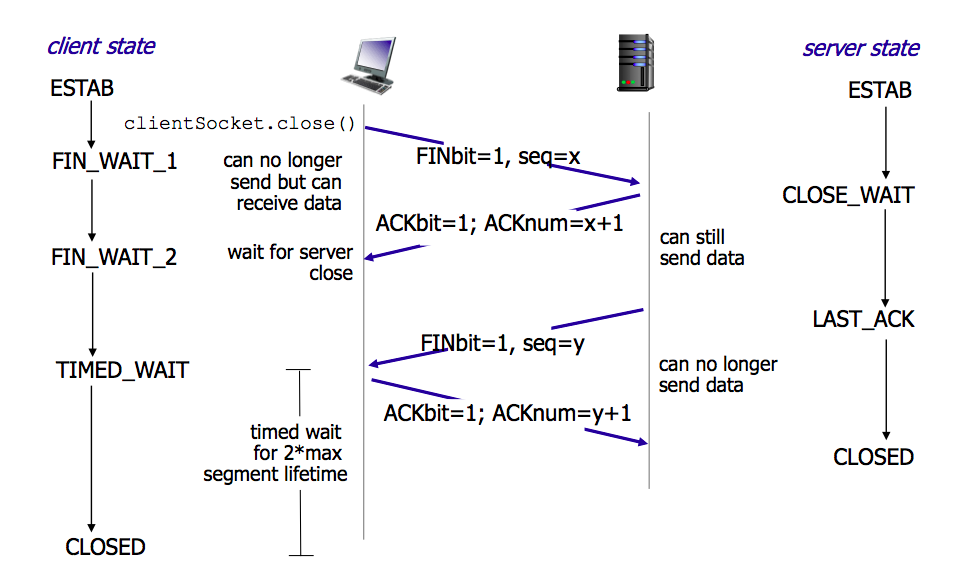


TCP 的连接的拆除需要发送四个包，因此称为[四次挥手](http://coding.imooc.com/class/101.html" \t "_blank)(Four-way handshake)，也叫做改进的三次握手。客户端或服务器均可主动发起挥手动作，在 socket 编程中，任何一方执行 close() 操作即可产生挥手操作。

第一次挥手(FIN=1，seq=x)  
假设客户端想要关闭连接，客户端发送一个 FIN 标志位置为1的包，表示自己已经没有数据可以发送了，但是仍然可以接受数据。发送完毕后，客户端进入 FIN\_WAIT\_1 状态。

第二次挥手(ACK=1，ACKnum=x+1)  
服务器端确认客户端的 FIN 包，发送一个确认包，表明自己接受到了客户端关闭连接的请求，但还没有准备好关闭连接。发送完毕后，服务器端进入 CLOSE\_WAIT 状态，客户端接收到这个确认包之后，进入 FIN\_WAIT\_2 状态，等待服务器端关闭连接。

第三次挥手(FIN=1，seq=y)  
服务器端准备好关闭连接时，向客户端发送结束连接请求，FIN 置为1。  
发送完毕后，服务器端进入 LAST\_ACK 状态，等待来自客户端的最后一个ACK。

第四次挥手(ACK=1，ACKnum=y+1)  
客户端接收到来自服务器端的关闭请求，发送一个确认包，并进入 TIME\_WAIT状态，等待可能出现的要求重传的 ACK 包。服务器端接收到这个确认包之后，关闭连接，进入 CLOSED 状态。客户端等待了某个固定时间（两个最大段生命周期，2MSL，2 Maximum Segment Lifetime）之后，没有收到服务器端的 ACK ，认为服务器端已经正常关闭连接，于是自己也关闭连接，进入 CLOSED 状态。  


这是我最近在做的java公众号，里面将会有bat java面试大量内部题目的解读  
公众号名字：攻克BAT之Java面试指南