TCP在运输层

TCP三次握手发生在客户端与服务端之间。

1. 第一次握手：A的TCP客户进程首先创建传输控制块TCB（传输控制模块，存储每一个连接中的重要信息，如TCP连接表，指针等）,然后向B发出连接请求报文段，（首部的同部位SYN=1，初始序号seq=x）,SYN=1的报文段不能携带数据，但是要消耗掉一个序号，此时TCP客户进程进入SYN-SENT状态。
2. 第二次握手：B收到连接请求报文段后，如同意建立连接，则向A发送确认，在确认报文段中（SYN=1，ACK=1，确认号ack=x+1，初始序号seq=y），测试TCP服务器进程进入SYN-RCVD（同步收到）状态；
3. 第三次握手：TCP客户端收到B的确认后，要向B给出确认报文段（ACK=1，确认号ack=y+1，序号seq=x+1）（初始为seq=x,第二个报文段所以要+1），ACK报文段可以携带数据，不携带数据则不消耗序号。TCP建立连接，A进入ESTABLISHED。

问：

为什么A还要发送一次确认呢？

-主要是为了防止已失效的连接请求报文段突然又传送到了B,因而产生错误。

Server端易受到SYN攻击

-SYN洪泛攻击，就是Client在短时间内伪造大量不存在的IP地址，并向Server不断地发送SYN包，而服务器回复确认包，因为ip地址是虚假的，所以服务器就要不停地重发，导致正常的SYN请求因为队列满而被丢弃。

四次握手协议：

起初A和B处于建立连接状态，A发出连接释放报文段FIN-WAIT-1状态，B发出确认报文段并进入CLOSE-WAIT状态，A收到确认后进入FIN-WAIT-2状态，等待B的连接释放报文段，B的消息都发完了，所以想A发出连接释放报文段，A发出确认报文段且进入TIME-WAIT状态，B收到确认报文段后进入CLOSED状态，A经过一段时间后自动进入CLOSED状态。

问：为什么A在TIME-WAIT状态必须等待2MSL的时间？

MSL最长报文段寿命Maximum Segment Lifetime，MSL=2

答：　　两个理由：**1）保证A发送的最后一个ACK报文段能够到达B。2）防止“已失效的连接请求报文段”出现在本连接中。**

1）这个ACK报文段有可能丢失，使得处于LAST-ACK状态的B收不到对已发送的FIN+ACK报文段的确认，B超时重传FIN+ACK报文段，而A能在2MSL时间内收到这个重传的FIN+ACK报文段，接着A重传一次确认，重新启动2MSL计时器，最后A和B都进入到CLOSED状态，**若A在TIME-WAIT状态不等待一段时间，而是发送完ACK报文段后立即释放连接，则无法收到B重传的FIN+ACK报文段，所以不会再发送一次确认报文段，则B无法正常进入到CLOSED状态。**

2）A在发送完最后一个ACK报文段后，再经过2MSL，就可以使本连接持续的时间内所产生的所有报文段都从网络中消失，使下一个新的连接中不会出现这种旧的连接请求报文段。

问：为什么连接的时候是三次握手，关闭的时候却是四次握手？

答：因为当Server端收到Client端的SYN连接请求报文后，可以直接发送SYN+ACK报文。其中ACK报文是用来应答的，SYN报文是用来同步的。但是关闭连接时，当Server端收到FIN报文时，很可能并不会立即关闭SOCKET，所以只能先回复一个ACK报文，告诉Client端，"你发的FIN报文我收到了"。只有等到我Server端所有的报文都发送完了，我才能发送FIN报文，因此不能一起发送。故需要四步握手。

问：为什么TIME\_WAIT状态需要经过2MSL(最大报文段生存时间)才能返回到CLOSE状态？

答：虽然按道理，四个报文都发送完毕，我们可以直接进入CLOSE状态了，但是我们必须假象网络是不可靠的，有可以最后一个ACK丢失。所以TIME\_WAIT状态就是用来重发可能丢失的ACK报文。