TCP三次握手和四次挥手

一.三次握手

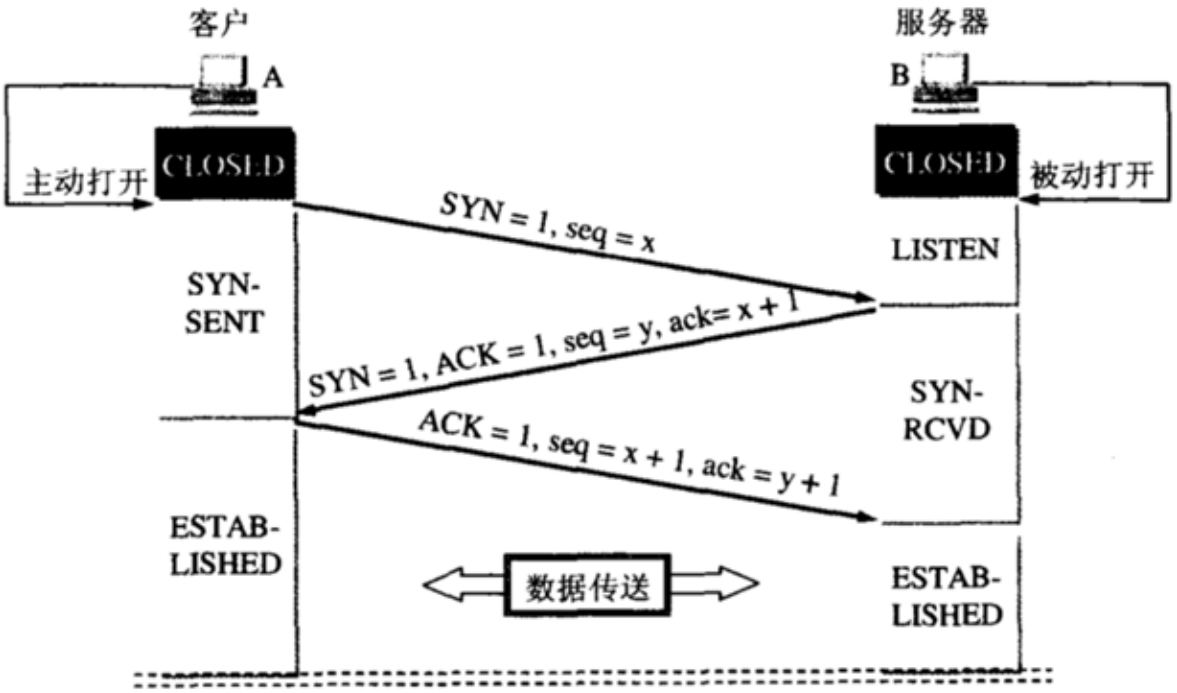

图 5-31　用三次握手建立 TCP 连接

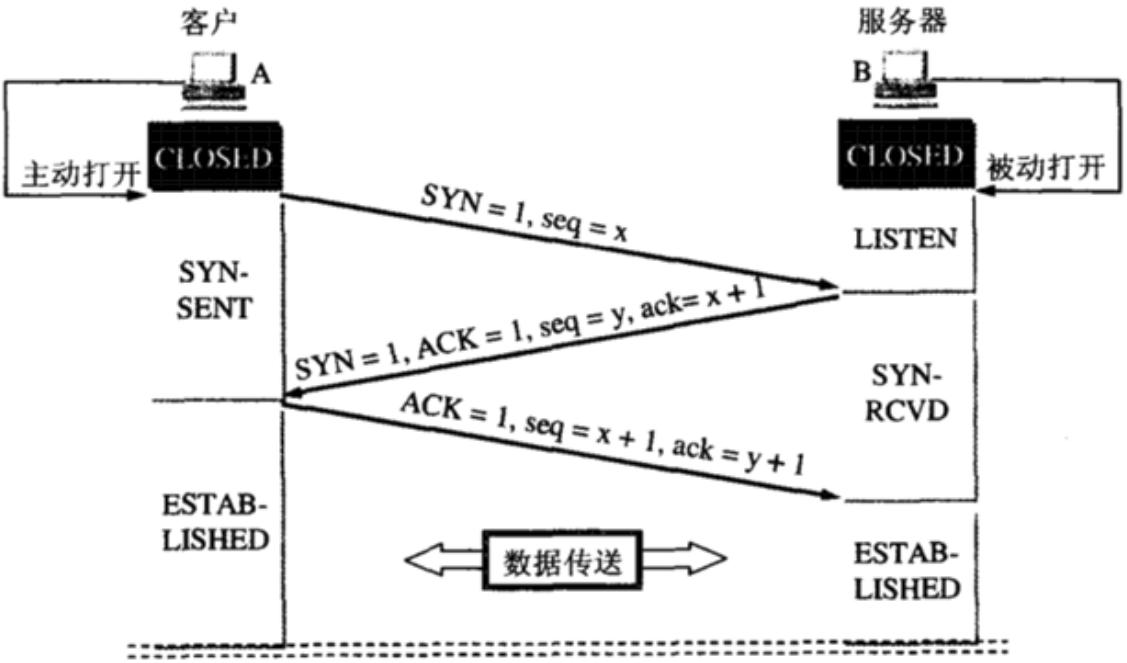

图 5-31　用三次握手建立 TCP 连接

最初两端的TCP进程都处于CLOSED关闭状态，A主动打开连接，而B被动打开连接（A、B关闭状态CLOSED——B收听状态LISTEN——A同步已发送状态

SYN-SENT——B同步收到状态SYN-RCVD——A、B连接已建立状态ESTABLISHED）

1.三次握手过程：

- **第一次握手：**Client端向Sever端发送连接请求报文段（首部的同步位SYN=1，初始序号seq=x）此时Client端进入SYN-SENT（同步已发送）状态。
- **第二次握手：**Sever端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），Sever端进入SYN-RCVD（同步收到）状态；
- **第三次握手**：Client端收到Sever端的确认后，要向Sever端给出确认报文段（ACK=1，确认号ack=y+1，序号seq=x+1）TCP连接已经建立，**Client端**进入ESTABLISHED状态，当Sever端收到Client端的确认后，也进入ESTABLISHED状态。

2.为什么客户端还要发送一次确认呢？可以二次握手吗？

答：主要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。如A发出连接请求，但因连接请求报文丢失而未收到确认，于是A再重传一次连接请求。后来收到了确认，建立了连接。数据传输完毕后，就释放了连接，A工发出了两个连接请求报文段，其中第一个丢失，第二个到达了B，但是第一个丢失的报文段只是在某些网络结点长时间滞留了，延误到连接释放以后的某个时间才到达B，此时B误认为A又发出一次新的连接请求，于是就向A发出确认报文段，同意建立连接，不采用三次握手，只要B发出确认，就建立新的连接了，此时A不理睬B的确认且不发送数据，则B一致等待A发送数据，浪费资源。

3.为什么Server端易受到SYN攻击?

服务器端的资源分配是在二次握手时分配的，而客户端的资源是在完成三次握手时分配的，所以服务器容易受到SYN洪泛攻击，SYN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，Server则回复确认包，并等待Client确认，由于源地址不存在，因此Server需要不断重发直至超时，这些伪造的SYN包将长时间占用未连接队列，导致正常的SYN请求因为队列满而被丢弃，从而引起网络拥塞甚至系统瘫痪。

防范SYN攻击措施：降低主机的等待时间使主机尽快的释放半连接的占用，短时间收到某IP的重复SYN则丢弃后续请求。

二.四次挥手

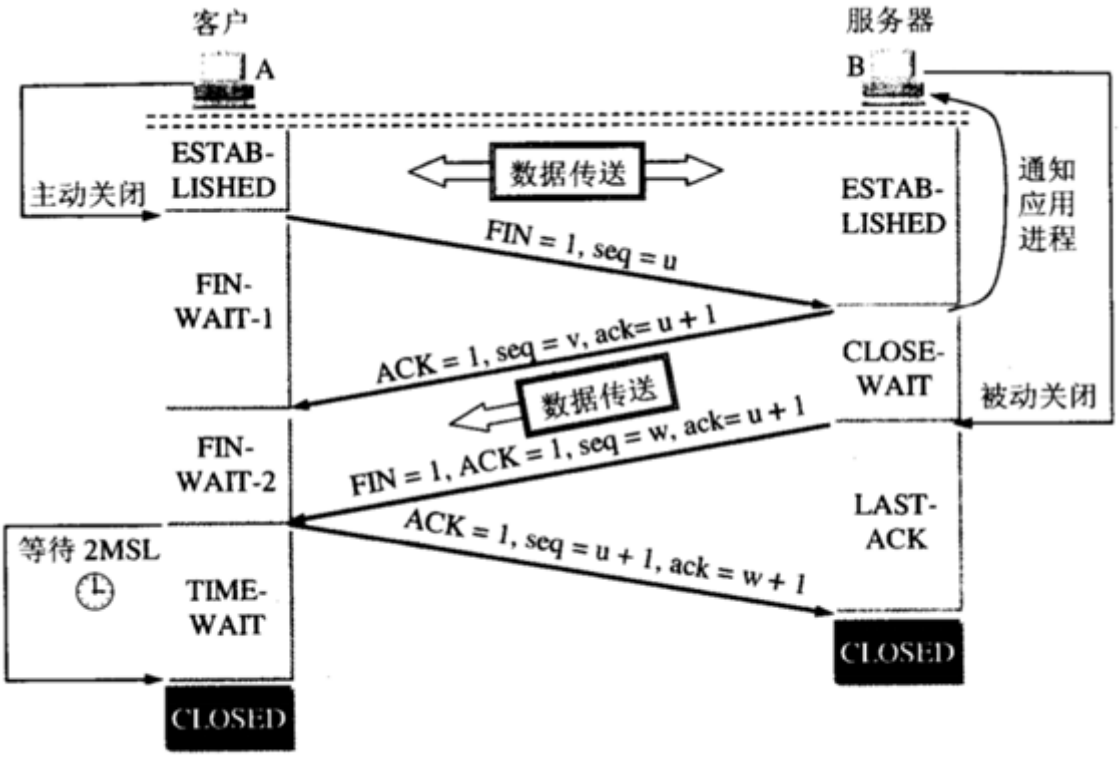

图 5-32　TCP 连接释放的过程

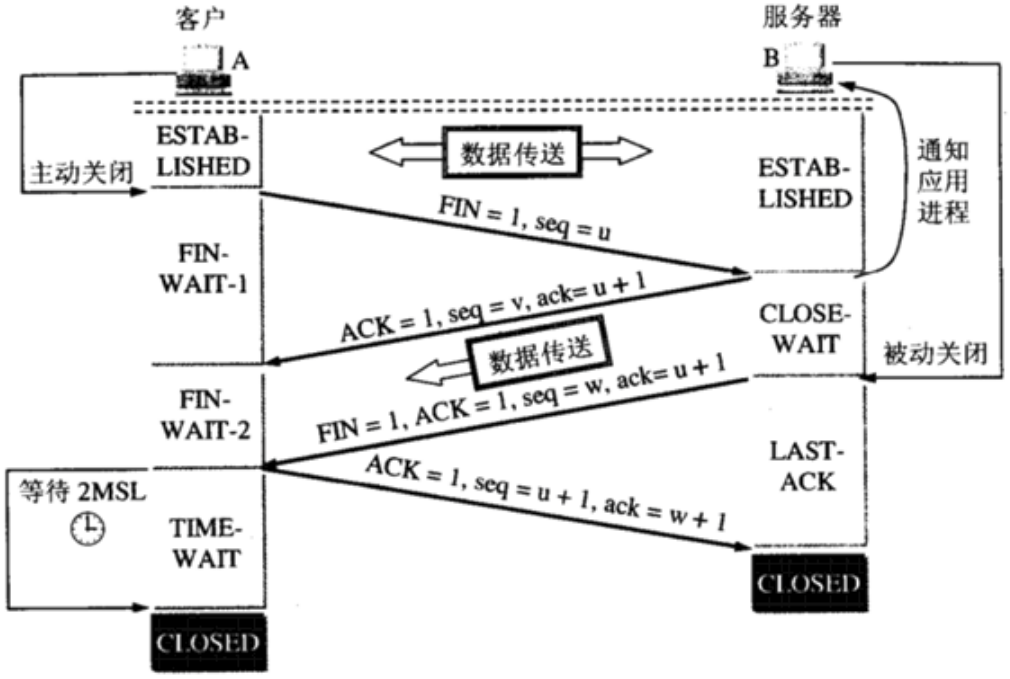

图 5-32　TCP 连接释放的过程

数据传输结束后，通信的双方都可释放连接，A和B都处于ESTABLISHED状态（A、B连接建立状态ESTABLISHED——A终止等待1状态FIN-WAIT-1——B关闭等待状态CLOSE-WAIT——A终止等待2状态FIN-WAIT-2——B最后确认状态LAST-ACK——A时间等待状态TIME-WAIT——B、A关闭状态CLOSED）

1.四次挥手过程：

- A的应用进程先向其发出连接释放报文段（FIN=1，序号seq=u）并停止再发送数据，主动关闭TCP连接，进入FIN-WAIT-1（终止等待1）状态
- B收到连接释放报文段后即发出确认报文段（ACK=1，确认号ack=u+1，序号seq=v），B进入CLOSE-WAIT（关闭等待）状态
- A收到B的确认后，进入FIN-WAIT-2（终止等待2）状态，等待B发出的连接释放报文段。
- B没有要向A发出的数据，B发出连接释放报文段（FIN=1，ACK=1，序号seq=w，确认号ack=u+1），B进入LAST-ACK（最后确认）状态，等待A的确认。
- A收到B的连接释放报文段后，对此发出确认报文段（ACK=1，seq=u+1，ack=w+1），A进入TIME-WAIT（时间等待）状态，此时TCP未释放掉，需要经过时间等待计时器设置的时间2MSL后，A才进入CLOSED状态。

2.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

1）保证A发送的最后一个ACK报文段能够到达B。2）本连接持续时间内所产生的所有报文段都从网络中消失，使得下一个新的连接不会出现旧的报文段。

- 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。

- 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报文段。

3.为什么连接的时候是三次握手，关闭的时候却是四次握手?

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。