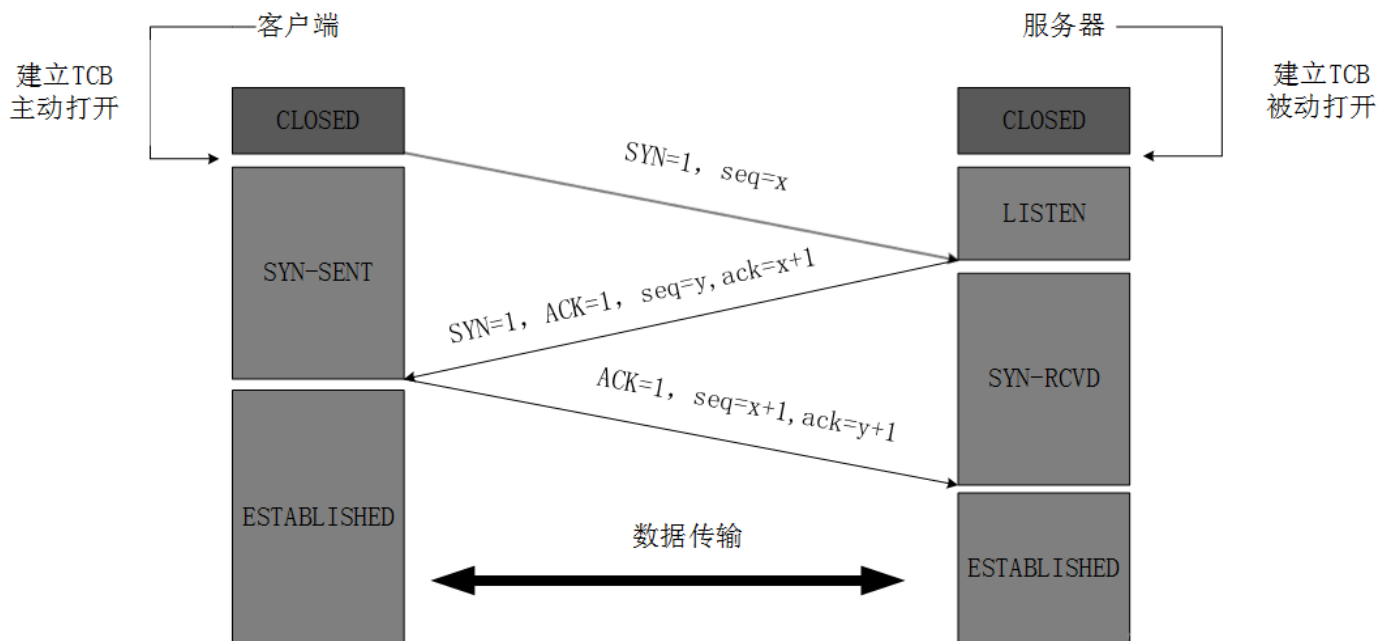


三次握手



- SYN=1表示这次请求是要与你建立连接
- ACK=1表示这次请求是确认你上次的请求

半连接和全连接

服务端发出第二次握手，等待第三次握手确认的状态称之为半连接；完成了三次握手称为全连接；SYN攻击的本质就是让伪造的客户端疯狂发送SYN报文，使服务端一直处于半连接状态

三次握手的目的

简而言之就是只有三次握手才能让客户端和服务端都确认对方发送数据和接收数据的能力都正常

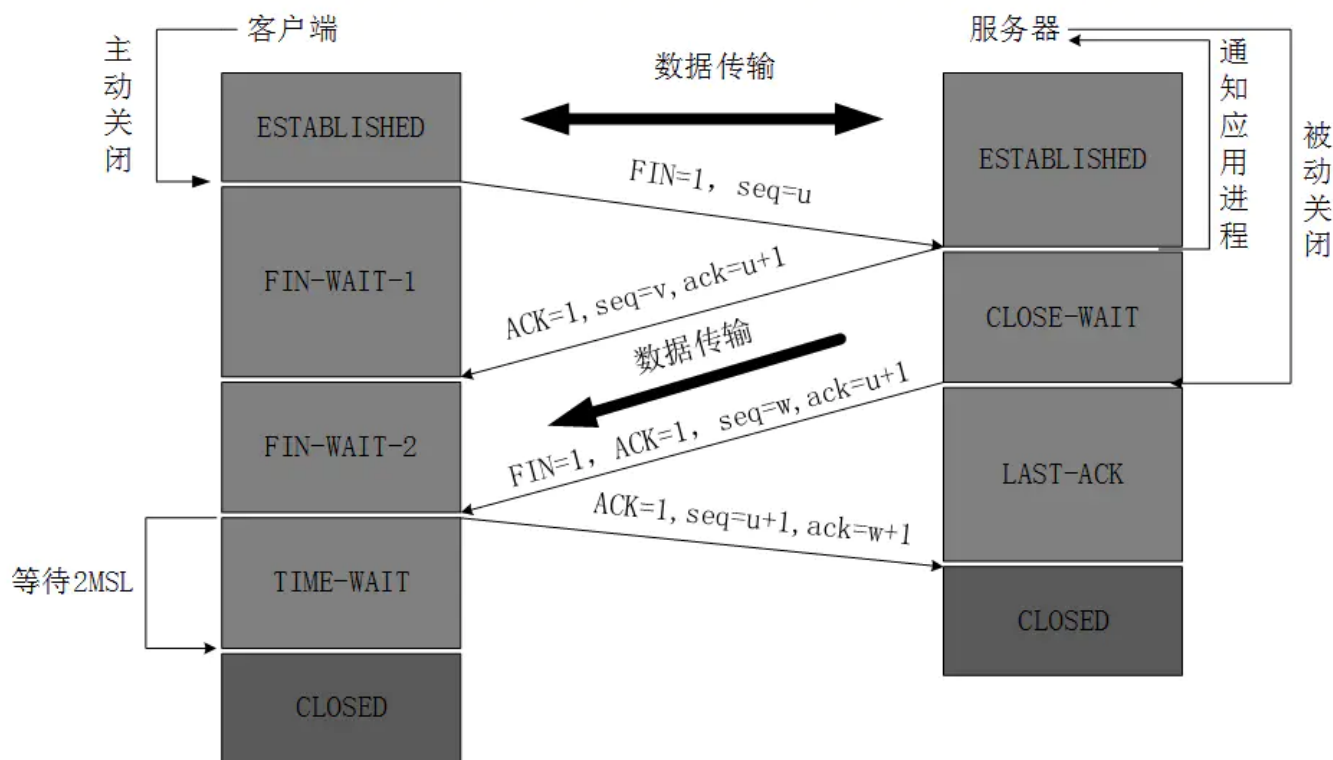
- 第一次握手：客户端向服务端请求建立连接 -> 服务端确认客户端的发送能力正常
- 第二次握手：服务端告知客户端已经确认收到了上次建立连接的请求，同意建连；同时向客户端发送建立连接的请求 -> 客户端确认服务端的接收和发送能力都正常
- 第三次握手：客户端告知服务端已经确认收到了上次建立连接的请求，同意建连 -> 服务端确认客户端的接收能力正常

前两次握手占用序列号，但是不能携带任何数据，因为要防止有人恶意发送大量SYN报文占用服务端资源

如果客户端发出连接请求，但因连接请求报文丢失而未收到确认，于是客户端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，客户端共发出了两个连接请

求报文段，其中第一个丢失，第二个到达了服务端，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服务端，此时服务端误认为客户端又发出一次新的连接请求，于是就向客户端发出确认报文段，同意建立连接，不采用三次握手，只要服务端发出确认，就建立新的连接了，此时客户端忽略服务端发来的确认，也不发送数据，则服务端一直等待客户端发送数据，浪费资源。

四次挥手



- **FIN=1**表示这次请求之后我就不再发数据了
- **ACK=1**表示这次请求是确认你上次的请求

为什么握手只要三次，而挥手要四次

第二次握手时，服务端可以直接发送**ACK+SYN**，既应答了客户端向服务端建立连接的请求，也同时服务端发起向客户端建立连接的请求。而第二次挥手时，服务端只能发送**ACK**，应答客户端不再向服务端发送数据的请求，然后等服务端该发的数据都发完之后，再通过第三次挥手主动告知客户端服务端也不再发数据了

客户端第四次挥手后等待**2MSL**的意义

MSL是**Maximum Segment Lifetime**的英文缩写，可译为“最长报文段寿命”，它是任何报文在网络上存在的最长时间，超过这个时间报文将被丢弃。

为了保证客户端发送的最后一个**ACK**报文段能够到达服务器。因为这个**ACK**有可能丢失，从而导致处在

LAST-ACK状态的服务器收不到对FIN-ACK的确认报文。服务器会超时重传这个FIN-ACK，接着客户端再重传一次确认，重新启动时间等待计时器。最后客户端和服务端都能正常的关闭。假设客户端不等待2MSL，而是在发送完ACK之后直接释放关闭，一但这个ACK丢失的话，服务器就无法正常的进入关闭连接状态。

tcp和udp比较

	udp	tcp
是否要建立连接	否	是
是否可靠	不可靠传输	可靠
连接对象个数	支持一对一、一对多、多对一、多对多	只能一对一
传输方式	面向报文	面向字节流
适用场景	实时应用（IP电话、视频会议、直播等）	高可靠传输，例如文件传输

- TCP向上层提供面向连接的可靠服务，UDP向上层提供无连接不可靠服务。
- 虽然UDP并没有TCP传输来的准确，但是也能在很多实时性要求高的地方有所作为
- 对数据准确性要求高，速度可以相对较慢的，可以选用TCP