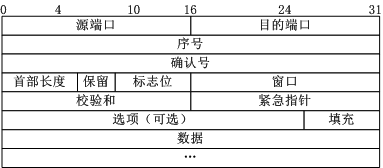
**TCP（Transmission Control Protocol ）传输控制协议**

TCP是主机对主机层的传输控制协议，提供可靠的链接服务，采用三次握手建立链接



TCP报文格式

(1)序号：Seq序号，占32位，用来标识从TCP源端向目的端发送的字节流，发起方发送数据时对此进行标记。

(2)确认序号：ack序号，占32位，只有ACK标志位为1时，确认序号字段才有效，Ack=Seq+1。

Sequence number (顺序号码)

SYN（synchronous ）建立联机

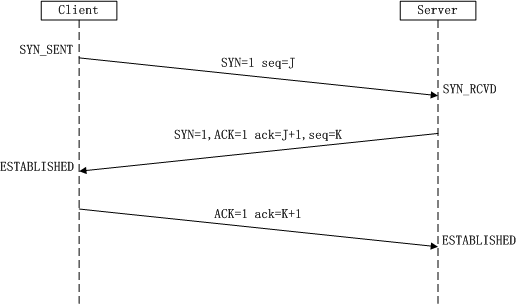
ACK（acknowledfement ）确认

PSH(push )传送

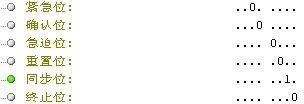
FIN（finish ）结束

RST(reset)重置

URG(urgent )紧急



第一次握手：Client将标志位SYN置为1,随机产生一个值seq=j，并将该数据包发送给Server，Clicent进入SYN\_SENT状态，等待Server确认。



第二次握手：Server收到数据包后由标志位SYN=1知道Client请求建立连接，Server将标志位SYN和ACK都置为1，ack=j+1,随机产生一个值seq=k，并将该数据包发送给Slient以确认连接请求，Server进入SYM\_RCVD状态。

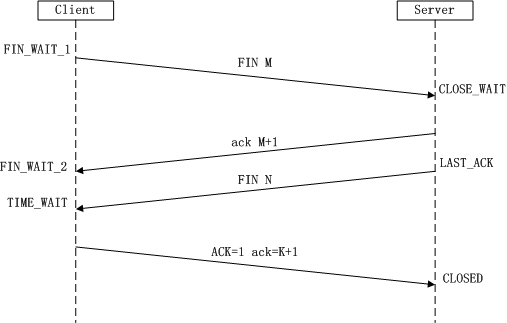


第三次握手：Client收到确认后，检查ack是否为j+1，ACK是否为1，如果正确则将标志位ACK置为1，ack=k+1，并将该数据包发送给Server，Server检查ack是否为k+1，ACK是否为1，如果正确则连接建立成功，Client和Server进入ESTABLISHED状态，完成三次握手，随后Client和Server之间可以开始传输数据了。



四次挥手

三次握手耳熟能详，所谓四次挥手(Four-Way Wavehand)即终止TCP连接，就是指断开一个TCP连接时，需要客户端和服务端总共发送4个包以确认连接的断开。在socket编程中，这一过程由客户端或服务端任一方执行close来触发



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。首先进行关闭的一方将执行主动关闭，而另一方则执行被动关闭，上图描述的即是如此。

(1)第一次挥手：Client发送一个FIN，用来关闭Client到Server的数据传送，Client进入FIN\_WAIT\_1状态。

(2)第二次挥手：Server收到FIN后，发送一个ACK给Client，确认序号为收到序号+1(与SYN相同，一个FIN占用一个序号)，Server进入CLOSE\_WAIT状态。

(3)第三次挥手：Server发送一个FIN，用来关闭Server到Client的数据传送，Server进入LAST\_ACK状态。

(4)第四次挥手：Client收到FIN后，Client进入TIME\_WAIT状态，接着发送一个ACK给Server，确认序号为收到序号+1，Server进入CLOSED状态，完成四次挥手。

为什么建立连接是三次握手，而关闭连接却是四次挥手呢?

这是因为服务端在LISTEN状态下，收到建立连接请求的SYN报文后，把ACK和SYN放在一个报文里发送给客户端。而关闭连接时，当收到对方的FIN报文时，仅仅表示对方不再发送数据了但是还能接收数据，己方也未必全部数据都发送给对方了，所以己方可以立即close，也可以发送一些数据给对方后，再发送FIN报文给对方来表示同意现在关闭连接，因此，己方ACK和FIN一般都会分开发送