# 计算机网络

## IP地址

* + 1. 在公网上全球唯一的IP地址
       1. IPV4（32位，4个字节）
       2. IPV6（128位，16个字节）
       3. 使用“.”分隔 也叫点分
       4. 每一个分段是8位
       5. IP地址分为 A B C D E 类

较为常见的是C类 家用，最为常见 192\*\*\*

* + - 1. 特殊IP地址

0.0.0.0 无意义，表示一个集合

255.255.255.255 广播地址

127.0.0.1 本地地址，搭建本地服务

169.254 路由器，DHCP 服务器分配无效的IP地址 不能访问外网

* + - 1. 子掩码

A类 255.0.0.0

B类 255.255.0.0

C类 255.255.255.0

* + - 1. 2

1. 计算机系统
2. 进程与线程

进程是系统资源分配的最小单位，线程是是程序执行的最小单位，线程只能依靠进程来执行。

进程使用独立的进程空间，而线程共享进程的共享空间

1. 线程调度

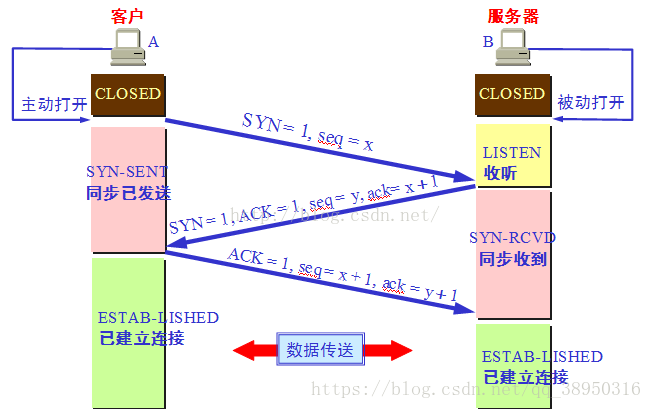
线程调度分为两种调度，分时调度、抢占式调度，分时调度是由cpu平均分配时间片，程序轮流占用CPU 。抢占式调度是通过系统给线程设置优先级来分配CPU的调度。

1. TCP协议

<https://blog.csdn.net/qq_38950316/article/details/81087809>

三次握手、四次挥手

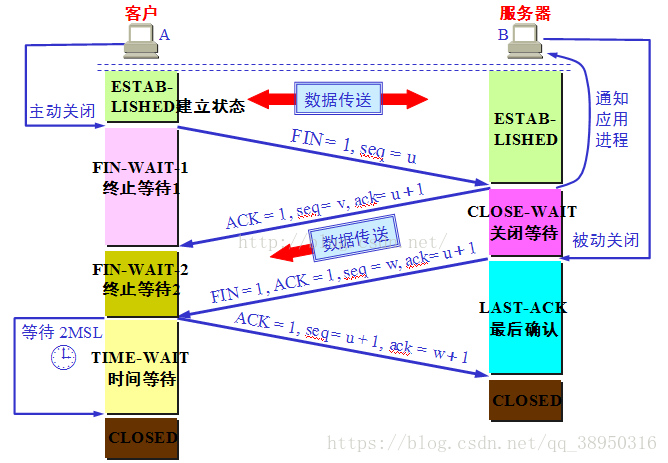




第一次握手：建立连接时，客户端发送syn包（syn=x）到服务器，并进入SYN\_SENT状态，等待服务器确认；SYN：同步序列编号（Synchronize Sequence Numbers）。

第二次握手：服务器收到syn包，必须确认客户的SYN（ack=x+1），同时自己也发送一个SYN包（syn=y），即SYN+ACK包，此时服务器进入SYN\_RECV状态；

第三次握手：客户端收到服务器的SYN+ACK包，向服务器发送确认包ACK(ack=y+1），此包发送完毕，客户端和服务器进入ESTABLISHED（TCP连接成功）状态，完成三次握手。



1）客户端进程发出连接释放报文，并且停止发送数据。释放数据报文首部，FIN=1，其序列号为seq=u（等于前面已经传送过来的数据的最后一个字节的序号加1），此时，客户端进入FIN-WAIT-1（终止等待1）状态。 TCP规定，FIN报文段即使不携带数据，也要消耗一个序号。

2）服务器收到连接释放报文，发出确认报文，ACK=1，ack=u+1，并且带上自己的序列号seq=v，此时，服务端就进入了CLOSE-WAIT（关闭等待）状态。TCP服务器通知高层的应用进程，客户端向服务器的方向就释放了，这时候处于半关闭状态，即客户端已经没有数据要发送了，但是服务器若发送数据，客户端依然要接受。这个状态还要持续一段时间，也就是整个CLOSE-WAIT状态持续的时间。

3）客户端收到服务器的确认请求后，此时，客户端就进入FIN-WAIT-2（终止等待2）状态，等待服务器发送连接释放报文（在这之前还需要接受服务器发送的最后的数据）。

4）服务器将最后的数据发送完毕后，就向客户端发送连接释放报文，FIN=1，ack=u+1，由于在半关闭状态，服务器很可能又发送了一些数据，假定此时的序列号为seq=w，此时，服务器就进入了LAST-ACK（最后确认）状态，等待客户端的确认。

5）客户端收到服务器的连接释放报文后，必须发出确认，ACK=1，ack=w+1，而自己的序列号是seq=u+1，此时，客户端就进入了TIME-WAIT（时间等待）状态。注意此时TCP连接还没有释放，必须经过2∗∗MSL（最长报文段寿命）的时间后，当客户端撤销相应的TCB后，才进入CLOSED状态。

6）服务器只要收到了客户端发出的确认，立即进入CLOSED状态。同样，撤销TCB后，就结束了这次的TCP连接。可以看到，服务器结束TCP连接的时间要比客户端早一些。

问题：为什么要等待2倍的MSL时间来关闭连接

1. 保证全双工连接能可靠的关闭
2. 防止重复的数据段从网络中消失，防止端口重用的时候可能会产生数据混淆
3. 1
4. 1