三次握手

客户端：发送带有SYN标志的数据包到服务端，实现第一次握手；
服务端：发送带有SYN/ACK标志的数据包到客户端，实现第二次握手；
客户端：发送带有ACK标志的数据包到服务端，实现第三次握手。

为什么是三次？
　　要建立可信的通信通道，双方都需要确认自己和对方的发送和接收都是正常的。
　　第一次握手：客户端只发送消息，无法确认自己能否发送和接收；
　　第二次握手：客户端接收到服务端返回的消息，则客户端确认：发送、接收正常；服务端确认：接收正常；
　　第三次握手：服务端收到客户端的确认消息，客户端确认：发送、接收正常；服务端确认：发送、接收正常。

四次挥手

客户端想要释放连接

客户端：
　　发送带有FIN标志数据包到服务端，随后客户端进入FIN-WAIT-1阶段，也就是半关闭状态。停止往服务端发送数据，但是客户端仍能接收服务端数据----第一次握手；
服务端：
　　服务端接收到客户端传过来的FIN数据包后，进入CLOSE-WAIT阶段，并返回ACK确认报文----第二次握手；
服务端：
　　在CLOSE-WAIT阶段做好释放连接的准备后，服务端向客户端发出带有FIN标志的数据包，结束CLOSE-WAIT阶段，停止向客户端发送数据----第三次握手；
客户端：
　　客户端收到服务端的FIN数据包后，进入TIME-WAIT阶段，并向服务端发送一个ACK确认报文----第四次握手。

为什么第二次和第三次不一起发送？
　　因为服务端接收到客户端的连接释放请求后，并不能直接释放连接，因为还可能有数据需要处理，所以先返回一个ACK确认报文，表示收到了释放连接请求。待服务端在CLOSE-WAIT阶段将数据处理结束后，再发送FIN报文表示可以释放连接。

为什么第四次握手后，客户端会在TIME-WAIT阶段等待2MSL的时间？
　　这是为了确认服务器是否收到了客户端发送的ACK报文。MSL指的是一段报文在传输过程中的最大生命周期。2MSL相当于一个来回。如果服务端在发送FIN报文后的1MSL时间内没有收到客户端发送的ACK报文，则会再次向客户端发送FIN报文。
　　如果没有在2MSL内没有收到任何报文，则表示服务端正常接收了ACK确认报文，并关闭了连接。

TCP协议如何保证可靠传输

1.分割数据块传输；
2.每个数据包编号，按顺序传输；
3.校验和
4.流量控制：TCP接收端只允许发送端发送接收端缓冲区能容纳的数据。如果接收端来不及处理，则提示发送端降低发送速率，防止丢包。(滑动窗口实现流量控制)
5.拥塞控制：当网络拥塞时减少数据的发送；
6.ARQ协议
7.超时重传

ARQ协议

如果发送端在发送数据后一段时间内没有收到确认消息，则会重新发送该数据。

停止等待ARQ：
　　每发送完一个分组就停止发送，等到收到确认消息后再发送下一个分组。

优点：简单
缺点：等待时间长，效率低

连续ARQ：
　　发送方维持一个窗口，在该窗口内的数据可以发送，且无需等待确认。接收方一般采用累积确认，对于按顺序到达的最后一个分组进行确认。

优点：信道利用率高
缺点：如果中间数据丢失，则只能确认到该数据之前的数据已接收。后面的数据需要重传。

拥塞控制

某段时间，对网络中的某一资源的需求超过其资源所能提供的部分，则网络性能可能会变坏，导致网络拥塞。

为了实现拥塞控制，TCP发送方会维持一个拥塞窗口cwnd的状态变量。只要网络没有出现拥塞，cwnd就增大；只要出现拥塞，cwnd就减小。

判断是否发生网络拥塞的依据：是否按时收到确认报文。

维护一个慢开始极限值ssthresh的状态变量：
　　当cwnd < ssthresh：使用慢开始算法
　　当cwnd > ssthresh：将算法变为拥塞避免算法
　　当cwnd = ssthresh：即可以使用慢开始，又可以使用拥塞避免

4种拥塞控制算法：
慢开始：
　　假设当前发送方拥塞窗口cwnd的值为1，而发送窗口swnd等于拥塞窗口cwnd，因此发送方当前只能发送一个数据报文段（拥塞窗口cwnd的值是几，就能发送几个数据报文段），接收方收到该数据报文段后，给发送方回复一个确认报文段，发送方收到该确认报文后，将拥塞窗口的值变为2，发送方此时可以连续发送两个数据报文段，接收方收到该数据报文段后，给发送方一次发回2个确认报文段，发送方收到这两个确认报文后，将拥塞窗口的值加2变为4；

拥塞避免：
　　当达到初始设置的ssthresh值后，改为拥塞避免算法。该算法每一轮的窗口大小只会+1。
　　如果某个时刻发生了网络拥塞，则将ssthresh的值变为当前发生拥塞时的cwnd的一半，并且cwnd变为1.重新开始慢开始算法。

快重传和快恢复FRR：
　　可以快速恢复丢失的数据包。
　　一般情况，如果数据丢失了，则会无法收到确认信息，也不会发送新数据。有了FRR后，如果收到了一个不按顺序的数据段，则立即发送发一个确认信息，如果发送端连续接收到了三个这样的信息，则确认该数据段丢失，并重新发送。
　　快恢复是通过快重传判断发送拥塞的时候，将ssthresh和cwnd都降为一半，并开始执行拥塞避免算法。

输入www.qq.com到浏览器显示内容的过程

1. 输入地址后，浏览器会先查找域名的ip地址(依次通过浏览器缓存、路由器缓存、DNS缓存)；
2. 得到IP地址后，向该IP地址发起一个HTTP请求；
3. 服务器处理该请求；
4. 服务器返回HTTP报文；
5. 浏览器解析报文，渲染页面

Cookie和Session

HTTP是无状态协议。如果想要保存用户的状态则需要使用到Session。
Session主要是在服务端保存用户的状态，每个Session都有生存时间，超时就会被销毁。
服务端的Session有很多，要实现Session跟踪，一般是通过在Cookie中附加一个SessionID来实现用户的跟踪。

Cookie被禁用？
　　将SessionID写到URL上。

Cookie也是保存用户信息，但Cookie是保存在用户本地的。比如网站登录。

Http和Https的区别

1.端口：Http默认是80端口，Https默认是443端口
2.安全性：
　　HTTP传输的都是明文，客户端和服务端都无法验证对方身份。HTTPS所有传输的内容都经过加密。加密是对称加密。但是对称加密的密钥使用了服务器的证书进行非对称加密。