TCP协议部分知识梳理

特点

- 面向连接，确认重传，流量控制，可靠交付的传输层协议。
- 可靠传输的定义：接收端能够按照正确的顺序接收到正确的内容。
- 可靠传输的实现：TCP的可靠性是通过顺序编号（seq）和确认（ACK/ack）来实现的。TCP在开始传送一个报文时，为准备重传而首先将该报文插入到发送队列之中，同时启动时钟。其后，如果收到了接受端对该报文的ACK信息，就将该报文从队列中删去。如果在时钟规定的时间内，ACK未返回，那么就从发送队列中再次发送这个报文。
- 一次完整的TCP通信过程：建立连接（三次握手）、数据传输、断开连接（四次挥手）。

TCP报文头部格式

- 源端口，目的端口
- 序列号 seq： 16位的二进制数，每个TCP报文都有，确保有序接收。
- 确认号 ack： 16位的二进制数，每个TCP报文都有，向对方确认收到了多少数据，同时也表示期望的对方（接收方）的下一次发送的报文的sequence number是多少。
- SYN同步标志位：代表是建立连接的请求报文，同步序列编号只有第一次和第二次握手时为1。
- ACK确认标志位：向对端确认收到的报文序号时为1。
- FIN结束标志位：四次挥手时第一次和第三次挥手时为1，代表通知对方，自己已经发送完了，准备断开。
- 窗口大小：滑动窗口的大小，滑动窗口用于流量控制。

TCP的“三次握手”

- 握手过程
  - 请求方发起第一次握手，发送SYN=1，seq=x；
  - 接收方收到请求后，发送SYN=1，ACK=1,seq=y，ack=x+1;
  - 请求方收到后，发送ACK=1，seq=x+1 ack=y+1；A收到B的ACK，是不会对ACK再做确认的
- 为什么需要三次握手？
  - 理解1：其实所谓的TCP三次握手请求连接，无非就是初始化一个序列号，保证后面的数据有序到达。
  - 理解2：“三次握手”的目的是“为了防止已失效的连接请求报文段突然又传送到了服务端，因而产生错误”。（不能使用两次握手的原因）。
  - 理解3：请求方发送同步信号，接收方收到后需要确认，TCP确认收到的包
    1. A 发送同步信号SYN+A的初始序列号seq=x
    2. B 确认收到A的同步信号，发送ACK=1，ack=x+1
    3. B发送同步信号SYN + B的初始化序列号seq=y
    4. A确认收到B的同步信号，发送ACK=1，ack=y+1

    为了减少时延，避免资源的浪费，2和3两步合并发送
- 握手时包丢了怎么办？
  - 第一个包，即A发给B的SYN 中途被丢，没有到达B
    - A会周期性超时重传，直到收到B的确认
  - 第二个包，即B发给A的SYN +ACK 中途被丢，没有到达A
    - B会周期性超时重传，直到收到A的确认
  - 第三个包，即A发给B的ACK 中途被丢，没有到达B
    - A发完ACK，单方面认为TCP为 Established状态，而B显然认为TCP为Active(SYN_REVD)状态
    - A会超时重传这个ACK吗？不会！TCP不会为没有数据的ACK超时重传
    - 1.双方都没有数据发送，B会周期性超时重传，直到收到A的确认，收到之后B的TCP 连接也为Established状态，双向可以发包
    - 2. 假定此时A有数据发送，B收到A的 Data + ACK，自然会切换为Established 状态，并接受A的Data
    - 3. 假定B有数据发送，数据发送不了，会一直周期性超时重传SYN + ACK，直到收到A的确认才可以发送数据

TCP的“四次挥手”

- 半关闭
- 在TIME_WAIT状态时的端口不能使用，要等到2MSL时间结束才可继续使用。当连接处于2MSL等待阶段时任何迟到的报文段都将被丢弃。
- 2MSL Maximum Segment Lifetime英文的缩写，中文可以译为“最大报文段生存时间”，他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。
- 为什么建立连接只需要三次握手？断开连接需要四次挥手？
  - 因为TCP是全双工模式，接收到FIN时意味将没有数据再发来，但是还是可以继续发送数据。
- 为什么TIME_WAIT状态需要经过2MSL(最大报文段生存时间)才能返回CLOSED状态?
  1. 保证TCP协议的全双工连接能够可靠关闭
  2. 保证这次连接的重复数据段从网络中消失