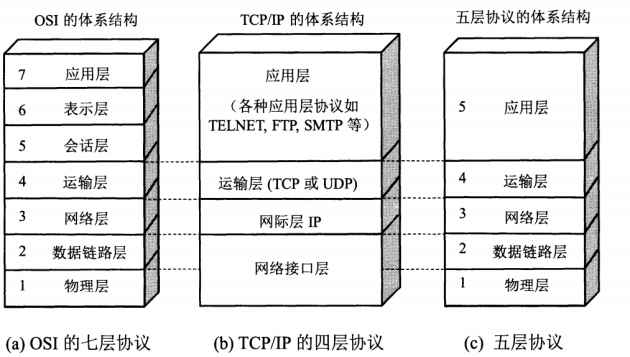
TCP相关

1 简述 TCP 三次握手以及四次挥手的流程。为什么需要三次握手以及四次挥手

- TCP协议是什么，位于哪一层
    - Transmission Control Protocol，传输控制协议，是一种面向连接的、可靠的、基于字节流的传输层通信协议
    - TCP协议位于OSI七层架构中第四层传输层，TCP/IP结构中第三层

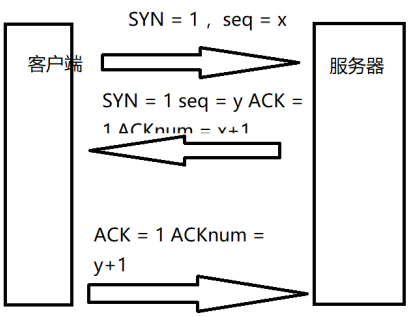


    - 其中网络层用来提供主机之间的逻辑通信，但不能确保可靠性，传输层在主机间的逻辑通信上，提供进程之间逻辑通信，TCP协议在不可靠的网络层协议上构建出可靠的，拥有拥塞机制的交付服务。

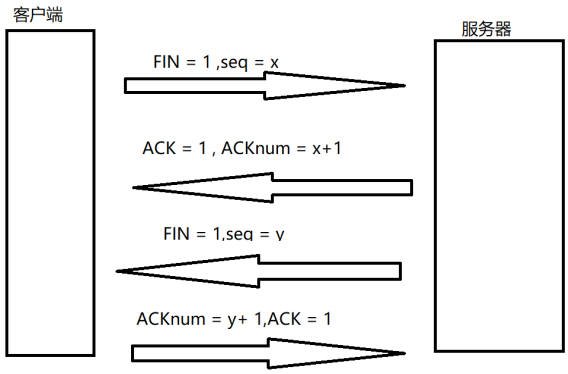
- 三次握手指的是什么，为什么要进行三次握手
    - 三次握手指的是TCP连接的建立需要服务器和客户端总共发送三个包
    - 在三次握手之后，双方确认了对方的起始序列号和窗口大小等
    - 只有经过三次握手才能确保服务器端客户端的收发功能都正常

- 四次挥手
    - TCP连接的拆除，当连接的一方想要关闭连接时，就会向对方发送FIN标识，对方回复一个确认包。过了一会，另一方也准备断开连接，发送FIN。接收方接收到回复确认包并等待一段时间在关闭连接。
    - 确保连接终止，不会出现多余的FIN标识在下一个包内

- 怎么进行操作



  - SYN标识表示该包请求建立连接
  - 客户端向服务器发送SYN = 1建立连接，并告诉服务器想要连接的端口号
  - 服务器发送SYN = 1 建立与客户端连接，发送ACK确认应答，并将自己的ISN号放在seq域中，将客户端的seq+1 放在ACKnum
  - 客户端向服务器发送ACK包，将ACKnum置为服务器的ISN（initial sequence number）号+1

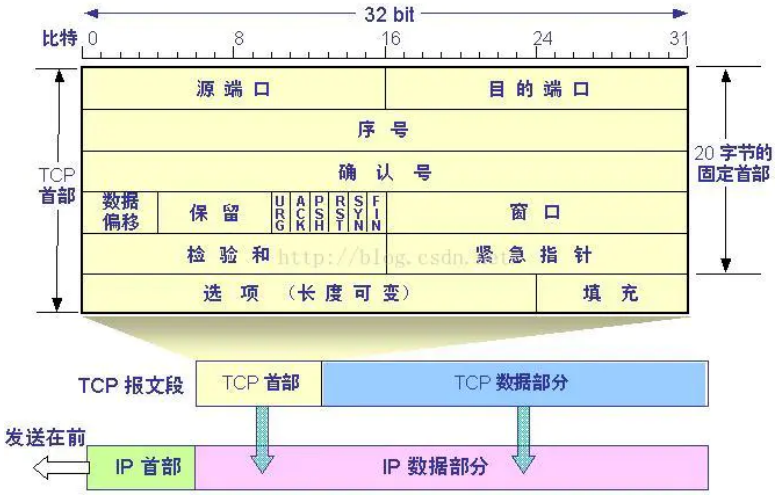


  - 通过close()指令
  - FIN标识表示该包请求断开连接
  - 客户端向服务器发送 FIN = 1 seq = x ,表示自己已经没有数据需要传输，但还可以接收数据，请求断开连接
  - 服务器向客户端发送一个确认包 ACK= 1 ， ACKnum = x+1表示收到客户端请求，但还没有准备好关闭连接
  - 服务器向客户端发送 FIN =1 seq = y 表示服务器可以断开连接了，等待客户端指示
    客户端向服务器发送确认包，ACK = 1 ACKnum = y+1，服务器收到后关闭连接，客户端等待一段时间后（两个生命周期）后自动关闭连接

2 TCP 怎么保证可靠传输?

- 是什么
- 为什么要提供可靠传输

- 怎样实现
  TCP报文头部



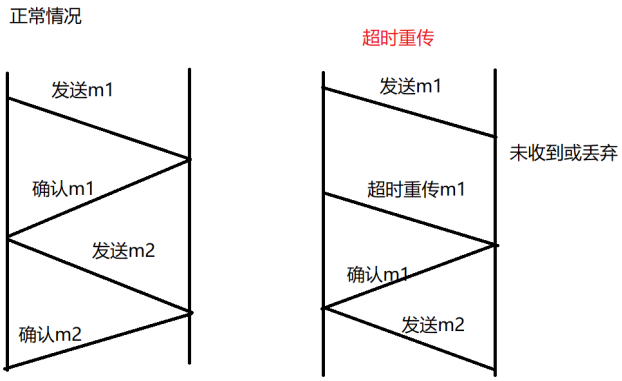
TCP 通过序列号，检验和，确认应答，重发控制，连接管理，窗口控制，流量控制，拥塞控制实现可靠性

校验和

- TCP将保持它首部和数据的检验和。检测数据在传输过程中的变化，如果检验和有差错，将丢弃报文不确认收到

停止等待协议

- 原理：发完一个分组停止发送等待确认，若接收方接收到重复分组，则丢弃该分组同时也要回复确认



- 每发送完一个分组设置一个超时计时器，重传时间比数据往返时间长一点。自动重传请求ARQ（优点，简单，缺点，信道利用率低）
- 如果确认迟到则发送方收到这个确认什么都不做，由接收方丢弃重复的分组。 **连续ARQ协议**
  - 位于发送窗口的分组可以连续发送而无需等待确认。接收方采取累计确认，对按序到达的最后一个分组发送确认，表明到这个分组为止所有的分组都已经正确收到
  - 优点：信道利用率高 缺点： 不能反应正确接收到所有分组的信息

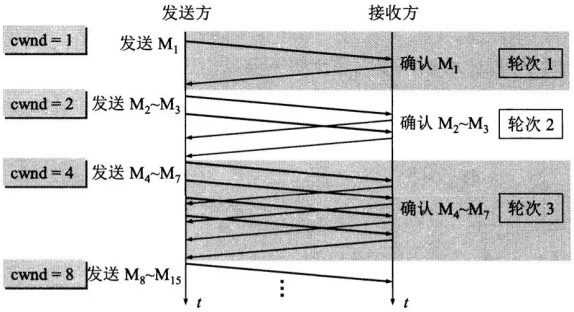
滑动窗口和流量控制

- 滑动窗口是一种流量控制技术，TCP中采用滑动窗口来进行传输控制。滑动窗口的大小意味着接收方还有多大的缓存区用来接受数据，当滑动窗口大小为0时，发送方不能在发送数据。（两种情况除外，一个是紧急数据，比如终端连接，另一种是发送1字节数据通知接收方声明希望接受的下一字节以及滑动窗口的大小）
- 接收方发送的报文段中的窗口字段可以控制发送方的滑动窗口大小

拥塞控制

- 如何产生拥塞
    - 某一时间段，对网络中某一资源的需求超过了该资源提供的可用部分

- 如何拥塞控制 -慢开始，拥塞避免，快重传和快恢复
    - **慢开始** 主机开始发送数据时由小到大（1，2，4，8）增加发送窗口，cwnd初始值为1，经过一个传播轮次加倍。



    - **快重传和恢复**（Fast Retransmit and Recovery, FRR）如果收到不按顺序的数据，接受方会立即给发送方一个重复确认。当发送方收到三个相同的重复确认，会迅速重传丢失的数据段。不会因为重传时要求的暂停被耽误。

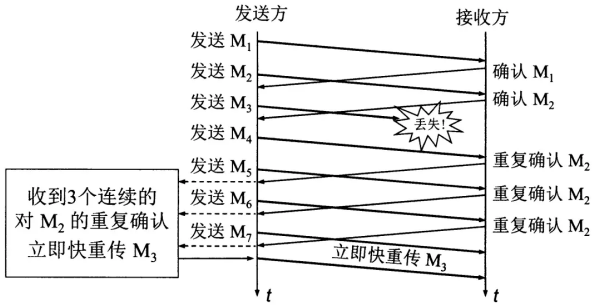


图 5-26　快重传的示意图