计网总复习

重点内容

1、时延

- 传输时延
  - 数据不断从设备推送到网络上，需要花费一定的时间，推送的速度受到网络带宽的影响。(数据从设备传输到通信链路上的时间)
- 传播时延
  - 数据不断从链路的一端传送到链路的另一端，需要花费一定的时间，传送速度受到电子传送速度影响。(在通信链路上的时间)
- 节点处理时延
  - 设备接受数据以及决定往什么端口转发数据,处理这个的时延
- 排队时延
  - 分组在通信端口的排列队列上排队等待离开设备

2、HTTP持续和非持续传播

- HTTP1.0
  - 客户端启动TCP连接,请求与服务器上的80号端口建立TCP连接
  - 服务器的80端口一直监听是否有客户端请求,监听到了立刻与客户端建立TCP连接,并把成功信息传给客户端
  - 客户端获得了来自于服务端的TCP回应,知道TCP连接已经建立,开始发送请求报文
  - 服务器收到客户端请求报文,准备发送响应报文
  - 相应报文发送完毕,服务器关闭TCP请求
  - 时间:2RTTs+传输时间
    - TCP连接建立
    - HTTP请求报文和响应报文发送
- HTTP1.1
  - 持久连接,连接完毕了之后一直保持连接,直到过了一段时间后还没有响应才关闭连接
  - 非流水线方式(一个一个来)
  - 流水线方式(一股脑扔过去)
- 时延
  - 时延包括TCP握手和传输时间,TCP握手只考虑两次握手
  - 2RTT + 文件传输时间
  - 文件传输时间只包括传播时延

- TCP握手
  - 客户机发送SYN
  - 服务器接受SYN,返回SYN&ACK
  - 客户机接受服务器SYN和ACK,回应ACK(客户机发送ACK是不用管的,包括在数据传输这一部分)
  - 服务器接受ACK,TCP建立完成

3、可靠信息传输协议

- 假如说用k位来给报文段编号

    - 那么报文段编号的取值范围为$[0, 2^k - 1]$,也就是说报文段编号的可能取值是$[0, 2^k - 1]$
    - 序列号就代表报文段编号的取值数,是2^k,也就是说报文段的编号这个号码的数量为2^k
- 窗口
    - GBN发送:$2^k - 1$
    - 接受1
    - SR发送2^{k-1}
    - 接受2^{k-1}

rdt系列

- rdt1.0
- rdt2.0
- rdt2.1
- rdt2.2
- rdt3.0
    - 发送方
        - 在等待状态应用层数据状态,收到应用层数据,就向网络层转发,并启动定时器
        - 收到不对的ACK(期望为0却是1),重传
        - 超时,重传,并启动定时器.
    - 接收方
        - 收到了错的分组,就发送上一次正确收到分组的序号的ACK(期待0号元素,收到了1号元素就发送ACK1)
        - 收到了对的分组就发送正确的ACK

GBN

- 发送方
    - 数据结构
        - 滑动窗口
        - base:滑动窗口的第一个元素
        - Nextseq:滑动窗口中第一个“未发送”的序号,下一次发送就发送这个序号的分组
    - 收到应用层
        - 如果有位置,就放入滑动窗口,Nextseq+1
        - 没位置就拒绝
    - 收到ACK
        - 如果和base一样,base+1,定时器置0
            - 滑动窗口里面还有已经发送没确认的元素,继续计时
            - 滑动窗口里面已经没有已经发送没确认的元素,就不计时了
        - 如果不和base一样,不管
    - 超时
        - 重新传输滑动窗口内已经发送的但是没有确认的元素
- 接受方
    - 数据结构
        - 期待收到的元素Exc
        - 上一次收到的规律分组序号n
    - 收到期望的分组
        - 发送ACK(Exc),Exc++
    - 收到乱序分组

        - 发送ACK(n)

SR

- 发送方
    - 数据结构
        - 滑动窗口
        - base:滑动窗口的第一个元素
        - Nextseq:滑动窗口中第一个“未发送”的序号,下一次发送就发送这个序号的分组
        - 每个分组都有一个定时器
    - 收到应用层
        - 如果有位置,就放入滑动窗口,Nextseq+1
        - 没位置就拒绝
    - 收到ACK
        - 标记分组 n 为已接收
        - 如果n是发送窗口基序号base，则将窗口基序号前推到下一个未确认序号
    - 超时
        - 哪个分组的定时器超时了就重传谁
- 接受方
    - 数据结构
        - 接受滑动窗口
            - 标记收到还是没收到
    - 收到分组
        - 发送n的确认ACK(n)
        - 如果分组序号不连续(失序)：将其缓存
        - 按序分组: 将该分组以及以前缓存的序号连续的分组一起交付给上层, 将窗口前推到下一个未收到的分组

TCP

- 发送方
    - 收到应用层数据
        - 将数据封装入报文段中，每个报文段都包含一个序号
        - 序号是该报文段第一个数据字节的字节流编号
        - 启动定时器
        - 超时间隔: TimeOutInterval
        - 一个报文段一个定时器
    - 超时
        - 重传认为超时的报文段.就是重传滑动窗口里面第一个已发送未确认的元素
        - 重启定时器
    - 收到ACK
        - 如果是对以前的未确认报文段的确认
            - 更新SendBase
            - 如果当前有未被确认的报文段，TCP还要重启定时器
- 接受方
    - 收到一个一个正常的分组(按序分组)
        - 等待500ms,等待收取第二个按序分组
        - 如果500ms没收到,那就只发这个

        - 如果500ms内收到了,就按照第二个来的分组发送ACK.
    - 收到一个乱序分组
        - 立即发送冗余ACK，指明下一个期待字节的序号
- 快速重传
    - 收到三个冗余ACK
        - 重传滑动窗口里面第一个已发送未确认的元素

4、拥塞控制

- 拥塞窗口
    - 标记为CongWin,滑动窗口小于等于拥塞窗口
      取接受滑动窗口和拥塞窗口最小值
- 慢启动
    - 拥塞窗口是之前的两倍或者门限值中比较小的那个
    - 每收到一个ACK,拥塞窗口大小+一个MSS
- 拥塞避免
    - 拥塞窗口大小是之前的加一
    - 拥塞窗口的大小大于ssthresh(门限值)进入拥塞避免
    - 每收到一个ACK,拥塞窗口大小+一个MSS*(MSS/窗口大小)
- 三个冗余ACK
    - 门限值=原拥塞窗口一半
    - 拥塞窗口=原拥塞窗口一半+3
- 超时
    - 门限值=原拥塞窗口一半
    - 拥塞窗口=1
- 传输轮回:指发送方一次性发送拥塞窗口大小的数据,然后收到对应的全部ACK或出现异常就是一个传输轮回

5、CIDR

- 编码格式
    - IP地址 ::= {<网络前缀>, <主机号>}
    - 这样可以更加方便地划分子网,其中斜线后面的位数就是表示网络号的位数
    - 斜线记法：192.168.0.1/24
- 最长前缀匹配
    - 应当从匹配结果中选择具有最长网络前缀(n最大的)的路由：最长前缀匹配(longest-prefix mating)。
    - 匹配的意思就是对于一个表项xx:xx:xx:xx/n,IP包的目的地址的前n位与表项xx:xx:xx:xx的前n位相同就可以 目的地址的前n位和网络编号的前n位(前缀匹配)
        - 例子:206.0.71.142和206.0.71.128/25,这个地址和206.0.71.128的前25位相同,就称之为匹配
    - 可以代表这个网络的子网掩码是前n位是1,这个地址和子网掩码的并等于网络地址的话就匹配
    - 也就是说转发,只能转发匹配的表项,不匹配的是不能被转发的.
- 层次编址
    - 小的网络统一归大的网络管
- NAT
    - **本地网络只要使用一个IP地址就可以和外部网络相连**,本地网络中的所有主机(IP:10.0.0.x),在外部网络都有相同的IP地址.也就是说**本地网络的所有主机和外面的网络交流都是用同一个IP地址来进行通信**.所有离开本地网络的报文都拥有同一个源IP地址,有不同的源端口号

    - 下面的例子**本地网络的所有主机10.0.0.1,10.0.0.2和10.0.0.3和外面的网络交流都是用同一个IP地址138.76.29.7来进行通信**
    - 维护一个NAT路由,路由在本地网络有一个地址,在全球网络也有一个地址.还有一张NAT表,分别记录本地网络中的地址,进程和全球网络的地址,进程的对应关系
    - 做法
        - 发送数据报: 将每个外出报文的源IP地址,端口号替换为NAT IP地址以及新的端口号
        - 接收数据报:根据NAT转换表将每个进入报文的NAT IP地址,端口号替换为相应的目的IP地址以及端口号,目的地址变成NAT转换表里面的元素
    -

6、AS内部选路协议

- OSPF算法(Link State)
    - 所有路由器都知道整个网络拓扑图以及链路的费用信
    - 迪克斯特拉算法
        - 先计算每个点到源点的距离
        - 选择一个到源点距离最短的一个点w选一个进入K集合
        - 对于这个点w,看看所有和w邻接点v,看看是原来的D(v)短,还是经过w的新路径短:D(w)+c(v,w)
          D(v) = min(D(v), D(w) + c(w,v))
    - 向本自治系统中所有路由器发送信息，使用的方法是洪泛法
    - 发送的信息就是与本路由器相邻的所有路由器的链路状态
    - 只有当链路状态发生变化时，路由器才用洪泛法向所有路由器发送此信息,过了30分钟,就算没有发生变化,也要广播状态
    - 所有路由器会构建一个链路状态数据库,这个数据库就是全网络的拓扑结构图
- RIP算法(Distance Vector)
    - 每个路由器仅有与其相连链路的费用信息
    - dx(y)=minv{c(x,v)+dv(y)}
    - 路由器之间的链路消费永远是1
    - 更新算法
        - 路由器X得到相邻路由器Y的路由表，从而得知：Y到网络Z的最短距离为N
        - 如果路由器X没有到网络Z的路由条目，则添加一条经由路由器Y到网络Z距离N+1的路由条目
        - 如果路由器X已有到网络Z的路由条目，其距离为M，如果M>N+1，则更新该条目为经由路由器Y到网络Z距离N+1，否则不更新
    - 毒性逆转
        - 如果z通过y选路到达目的地x :z将通告y，它到x的距离是无穷大 (所以 y不会通过z到达x)
    - 选路更新消息每30s在邻居之间以RIP响应报文（RIP通告）的形式进行交换
    - 路由器经过180s没有收到来自某个邻居的RIP通告，则认为该邻居已离线，修改选路表，向其它邻居广播

7、CSMA/CD

- 当一个结点传输数据的时候收到了别的结点发来的数据就叫做碰撞.
- 结点发送数据,通过广播链路传输到别的节点所需要的时间是t(端到端时延)

- 强化碰撞:当发送数据的站一旦发现发生了碰撞时，除了立即停止发送数据外，还要再继续发送若干比特的人为干扰信号
- 最先发送数据帧的站，在发送数据帧后至多经过时间 2t （两倍的端到端时延）就可知道发送的数据帧是否遭受了碰撞。以太网的端到端往返时延 2t称为争用期，或碰撞窗口。
- 特点
    - 没有时隙
    - 当适配器侦听到其它适配器在传输，则它不传输帧, 即载波侦听
    - 正在传输的适配器若检测到其它适配器也在传输，则它中止自己的传输, 即碰撞检测
    - 在重新传输之前，适配器要等待一段随机时间，即随机回退
- 几个定义
    - 拥塞信号: 用来确保所有传输者都能检测到碰撞而传输的信号；48比特长
    - 比特时间: 传输1比特所需时间。在10Mbps的以太网中，当K=1023时，等待时间大约为50ms
- 指数回退
    - 中止传输后，适配器进入指数回退阶段，在经历第m次碰撞后，适配器随机从{0,1,2,…,2m-1}中选择K值。适配器在等待 K·512比特时间后在重传
- 争用期
    - 以太网取 51.2us 为争用期的长度。
    - 对于 10 Mb/s 以太网，在争用期内可发送512 bit，即 64 字节。
- 最短有效帧长
    - 以太网规定了最短有效帧长为 64 字节，凡长度小于 64 字节的帧都是由于冲突而异常中止的无效帧。
- 信号编码
    - 曼彻斯特编码
    - 差分曼彻斯特编码

8、链路层交换机

- 工作原理
    - 不断监听各接口是否有信号
    - 收到无差错的帧则缓存，反之将差错帧丢弃
    - 若所收帧的目的MAC地址属另一网段，则通过站表决定向何接口转发
    - 交换机不转发同一网段内通信的帧
    - 交换机不修改所转发的帧的源地址
- 交换机是透明的
    - 这里所谓“透明”是指局域网上的每个站并不知道所发送的帧将经过哪几个交换机，即交换机对各站来说是看不见的
- 选路原理
    - ① 从接口x收到帧，有差错则丢弃，否则在站表中查找目的站MAC地址；
    - ② 找到有，则取出相应的接口d，转③，否则转⑤；
    - ③ 如果所给MAC地址的接口d=x，则丢弃此帧（不需要转发），否则从接口d转发此帧；
    - ④ 转到⑥；
    - ⑤ 向除x以外的所有接口转发此帧（可保证找到目的站）
    - ⑥ 如源站不在站表中，则将源站MAC地址写入站表，登记该帧进入交换机的接口号和时间，设置计时器，然后转⑧。否则转⑦；
    - ⑦ 更新计时器（由于网络拓扑经常变化，因此，超时记录要删除，以反映最新状态）；
    - ⑧ 等待新的数据帧。转①
    - 支撑树---交换机互相知道各自的拓扑结构,构建一个最小生成树.

1计算机网络与互联网

1.1 什么是互联网

构成

- 硬件
    - 网络层路由器
    - 链路层交换机
    - 主机(端系统)
    - 链路
- 软件(协议)

ISP(互联网服务提供商)

- 主机接入接入ISP
- 接入ISP会继续接入更高级的ISP

应用程序角度

- 提供服务的基础设施
- 子主题 2

1.2 网络边缘

定义

- 可以运行网络应用程序的实体--主机

通信方式

- C/S模式
- P2P模式

网络接入

- 本质
    - 通过各种方式使主机连接到路由器
- 边缘路由器
    - 端系统到任何其它远程端系统的路径上的第一台路由器。
- 点对点方式
- 以太网
    - 有线以太网
    - WIFI
- 广域无线接入

1.3 网络核心

交换方式

- 电路交换
    - 特点
        - 数据交换前需建立起一条从发端到收端的物理通路
        - 在数据交换的全部时间内用户始终占用端到端的固定传输信道(在这个时间内,信道只能给它用)
        - 交换双方可实时进行数据交换而不会存在任何延迟

    - 分类
        - 频分复用
        - 时分复用
    - 做法
        - 先进行连接建立
        - 开始传送
        - 传送完成后释放链接

> 计算通过电路交换网络将一个640,000比特长的文件从主机A传送到主机B需要多长时间?
>
> p所有链路速率皆为1.536 Mbps
>
> p每条链路使用有24个时隙的TDM
>
> 建立端到端的电路需要500毫秒
>
> t=640000÷((1.536×1000×1000)÷24)+0.5 = 10.5s
>
> 0.5s+640k/（1536kbps/24）= 10.5s
>
> 重点：电路交换不能充分利用全部带宽

- 分组交换
    - 特点
        - 将要发送的报文分解层若干个小部分，称为分组
        - 存储转发
        - 路线不固定
        - 冗余路由
        - 动态分配带宽
    - 分类
        - 数据报
        - 虚电路
            - 建立虚电路链路
            - 在建立连接时决定链路的路由，在整个连接过程中保持不变
            - 在链路通过的每个节点，预留一定的资源
    - 做法
        - 要传输的数据分成小段
        - 加上首部,生成分组
        - 发送数据
        - 接收方接受数据并还原

概念

- 各种ISP互相连在一起
- 低级ISP可以连入高级ISP进行互通
- 同级ISP之间通过IXP和对等链路链接

1.4 分组网络的衡量

时延

- 传输时延
  - 数据不断从设备推送到网络上，需要花费一定的时间，推送的速度受到网络带宽的影响。(数据从设备传输到通信链路上的时间)
- 传播时延
  - 数据不断从链路的一端传送到链路的另一端，需要花费一定的时间，传送速度受到电子传送速度影响。(在通信链路上的时间)
- 节点处理时延
  - 设备接受数据以及决定往什么端口转发数据,处理这个的时延
- 排队时延
  - 分组在通信端口的排列队列上排队等待离开设备

丢包

- 分组在网络中环路传输
- 排列队列满

吞吐量

- 定义
  - 在发送方与接收方之间传输比特的速率（bps）
- 影响
  - 传播速率最慢的那段链路的传播速率

1.5 层次化的网络体系结构

分类

- 应用层
  - 报文
- 传输层
  - 报文段
- 网络层
  - 数据报
- 链路层
  - 帧
- 物理层

关系

- 发送数据:数据从高层往下层传递,每经过一层封装一层,直到物理层
- 接受数据:数据从下层往高层传递,每经过一层解封装一层.直到获得原始数据
- 路由器和链路层交换机都是收到数据传递到最高层,然后处理完毕之后再发送数据,到最低层

相关名词

- PDU
  - 协议数据单元是对等层次上传送数据的单位
- 对等体
  - 不同机器上包含对应层的实体称为对等体。
- 实体

- 实体是任何可以发送和接收信息的硬件和软件进程。通常是一个特定的软件模块。

CRC校验

公共除数 = 10011，则表示成 *x**4 + x + 1*

---

2 应用层

2.1 应用层协议原理

网络应用程序

- 功能
  - 可以向网络发送数据
  - 可以向网络接受数据
  - 对数据进行处理
- 体系结构
  - C/S模式
    - 服务器:一个可以想客户机提供服务的主机
    - 客户机:主动连接服务器,试图从服务器那里获取所需服务
    - 特点
      - 客户机不能通信
      - 通常采用服务器群的模式
  - P2P模式
    - 任何一方享受服务也提供服务
    - 结点之间可以互相通信
    - 结点之间的地址以及其连接会发生变化

应用层协议

- 交换的报文类型
- 报文的语法
- 字段的定义
- 进程何时、如何发送报文
- 报文的相应

需求的服务

- TCP
- UDP
- SSL
  - 介于应用层和传输层之间的协议
  - 数据完整性检查
  - 身份鉴权
  - 加密的TCP

Socket

- 标记每一个网络应用进程(给每个进程编号)
- 分成两部分
  - 32位主机地址:标记运行在哪个主机
  - 16位端口地址:在主机上标记运行在什么进程上
- 运行机理
  - 发送进程将报文发送到套接字
  - 套接字将这些报文传输到接受进程的套接字
  - 也就是说套接字像一个管道的两端,发送进程发送信息到管道里面(也就是传递给套接字),接受进程从管道里面接受信息(从套接字里面获取信息),具体管道怎么实现的进程不需要了解,双方在管道的两端,管道和双方之间就是套接字

2.2 Web和HTTP

构成

- 客户端
- 服务端
- 协议

内容表达

- Web 页面由一些对象组成
  - 对象可以是图片,文本,多媒体
  - 对象可以是HTML文件,HTML文件像是一个容器,装载着图片文本多媒体对象,是对象的容器
  - 任何一个对象都可以用 URL来定位

传输模式

- HTTP1.0（非持久连接）
  - 客户端启动TCP连接,请求与服务器上的80号端口建立TCP连接
  - 服务器的80端口一直监听是否有客户端请求,监听到了立刻与客户端建立TCP连接,并把成功信息传给客户端
  - 客户端获得了来自于服务端的TCP回应,知道TCP连接已经建立,开始发送请求报文
  - 服务器收到客户端请求报文,准备发送响应报文
  - 相应报文发送完毕,服务器关闭TCP请求
  - 时间:2RTTs+传输时间
    - TCP连接建立
    - HTTP请求报文和响应报文发送
- HTTP1.1（持久连接）
  - 持久连接,连接完毕了之后一直保持连接,知道过了一段时间后还没有响应才关闭连接
  - 非流水线方式(一个一个来)
  - 流水线方式(一股脑扔过去)

HTTP请求报文

(靠\r\n分割)

- 方法
  - GET(获取)
  - POST(发送)

    - etc...
- URL
- 浏览器版本
- 首部行(\r\n)
- (\r\n)一个空行
- 报文内容

HTTP响应报文

- URL
- 状态编码和短语
- 一个空行
- 报文内容

Cookies

Cookie是一个用户身份的识别码，WEB服务端的数据库用这个码找到对应的用户信息。用户的浏览器会储存用户在不同站点对应的码。

- 组成部分
    - 响应报文一个首部行
    - 请求报文一个首部行
    - 浏览器内部保存的Cookies
    - 服务器端的数据库
- 更改
    - 请求报文的Cookies行用来更改服务器数据库
    - 响应报文的Cookies行用来更改浏览器Cookies

DNS

因特网的目录服务：Diretory Service in the Internet

DNS是一个分布式数据库，由很多DNS服务器按照层次结构组织起来。（应用层协议，使用UDP传输，以C/S模式工作），不直接和用户打交道，而是因特网的核心功能

一次DNS解析过程：

- 在浏览器中输入 **www.hust.edu.cn/index.html** 链接，从该链接中取出 **www.hust.edu.cn** 部分，发送给DNS客户机
- DNS客户机向DNS服务器发送包含域名 **www.hust.edu.cn** 的查询请求报文
- DNS服务器向DNS客户机返回一个包含对应IP地址（202.114.0.245）的响应报文
- DNS客户机将获得的IP地址传送给浏览器
- 最后，浏览器向IP地址所在WEB服务器发起TCP链接

DNS服务器是按照区域分层的。

总之，DNS提供的服务：域名-IP地址的转换

3传输层

3.1 运输层概述

功能

- 为不同主机上运行的应用进程之间提供逻辑通信信道

工作内容

- 发送方：把应用数据划分成报文段(segments),交给网络层(从应用层获取,交付到网络层)
- 接收方：把报文段重组成应用数据，交付给应用层(从网络层获取,交付到应用层)

> 美国东西海岸两个家庭都有若干个孩子：他们是堂兄妹关系，互相写信。

每星期，大哥大姐回收集兄弟们的信件，交给邮递员，邮递公司负责底层送信。

应用层报文：信封上的字符

进程：堂兄弟姐妹

主机（端系统）：家庭

运输层协议：大哥大姐

网络层协议：邮政服务

协议簇

- 用户数据报协议UDP(数据报)
- 传输控制协议TCP(报文段)

功能(提供的Service)

- 进程间交付 3.2
- 差错检测 3.3
- 可靠数据传输 3.4 3.5
- 拥塞控制 3.6

3.2 复用和分解

端口

- 一个应用层和运输层的桥梁
    - 应用层可以通过端口获得传输层递交的数据
    - 传输层可以通过端口向应用层递交数据
- 一个主机应用进程的标记

套接字

- TCP的基础:“连接”,将连接用套接字标记
- UDP套接字
    - 目的端口
    - 目的地址
- TCP套接字
    - 目的端口
    - 目的地址
    - 源端口
    - 源地址
- 多路复用

    - 运输层从主机的不同套接字中收集数据,为数据加上首部信息转发到网络层
    - 多个来源一个目的,不同套接字的数据转发到一个出口(即网络层)
- 多路分解
    - 运输层的报文段根据首部交付到正确的套接字
    - 一个来源多个目的,一个网络层的数据可以根据套接字的不同转发到不同的端口

3.3 UDP和差错检验

流程

- 发送方
    - 从应用进程获得数据
    - 加上目的端口号,形成报文段
    - 递交给网络层,尽力而为地交付
- 接收方
    - 从网络层获得报文段,直接交付给应用层
    - 没有响应

特点

- 无连接的
- 分组开销小
- 在应用层实现可靠传递
- 容易击垮TCP连接

UDP报文段的首部

- 源端口号
- 目的端口号
- 长度
- 检验和

检验和

- 将报文段里面的内容按照16位为单位,每个16位元素进行相加构成检验和
- 如果出现了溢出,一定要做回卷

3.4 可靠传输原理

rdt系列

- rdt1.0
- rdt2.0
- rdt2.1
- rdt2.2
- rdt3.0
    - 发送方
        - 在等待状态应用层数据状态,收到应用层数据,就向网络层转发,并启动定时器
        - 收到不对的ACK(期望为0却是1),重传
        - 超时,重传,并启动定时器.
    - 接收方
        - 收到了错的分组,就发送上一次正确收到分组的序号的ACK(期待0号元素,收到了1号元素就发送ACK1)

        - 收到了对的分组就发送正确的ACK

GBN

- 发送方
    - 数据结构
        - 滑动窗口
        - base:滑动窗口的第一个元素
        - Nextseq:滑动窗口中第一个“未发送”的序号,下一次发送就发送这个序号的分组
    - 收到应用层
        - 如果有位置,就放入滑动窗口,Nextseq+1
        - 没位置就拒绝
    - 收到ACK
        - 如果和base一样,base+1,定时器置0
            - 滑动窗口里面还有已经发送没确认的元素,继续计时
            - 滑动窗口里面已经没有已经发送没确认的元素,就不计时了
        - 如果不和base一样,不管
    - 超时
        - 重新传输滑动窗口内已经发送的但是没有确认的元素
- 接受方
    - 数据结构
        - 期待收到的元素Exc
        - 上一次收到的规律分组序号n
    - 收到期望的分组
        - 发送ACK(Exc),Exc++
    - 收到乱序分组
        - 发送ACK(n)

SR

- 发送方
    - 数据结构
        - 滑动窗口
        - base:滑动窗口的第一个元素
        - Nextseq:滑动窗口中第一个“未发送”的序号,下一次发送就发送这个序号的分组
        - 每个分组都有一个定时器
    - 收到应用层
        - 如果有位置,就放入滑动窗口,Nextseq+1
        - 没位置就拒绝
    - 收到ACK
        - 标记分组 n 为已接收
        - 如果n是发送窗口基序号base，则将窗口基序号前推到下一个未确认序号
    - 超时
        - 哪个分组的定时器超时了就重传谁
- 接受方
    - 数据结构
        - 接受滑动窗口

        - 标记收到还是没收到
    - 收到分组
        - 发送n的确认ACK(n)
        - 如果分组序号不连续(失序)：将其缓存
        - 按序分组: 将该分组以及以前缓存的序号连续的分组一起交付给上层, 将窗口前推到下一个未收到的分组

3.5 TCP

报文首部

- 源端口号
- 目的端口号
- 长度
- 检验和
- 序列号
    - 在报文段数据中第一个字节在字节流中的编号
- 确认号
    - 期待得到的下一个字节的seq
      也就是说seq-1字节已经被确认,seq没有

超时设置

- SampleRTT
    - 对报文段被发出到收到该报文段的确认之间的时间进行测量
- EstimatedRTT
    - EstimatedRTT = (1- a) *EstimatedRTT + a* SampleRTT
    - 第一次:EstimatedRTT=SampleRTT
- DevRTT
    - DevRTT = (1-b) *DevRTT +b* |SampleRTT-EstimatedRTT|
    - 第一次:DevRTT=0.5*SampleRTT

可靠传输协议

- 发送方
    - 收到应用层数据
        - 将数据封装入报文段中，每个报文段都包含一个序号
        - 序号是该报文段第一个数据字节的字节流编号
        - 启动定时器
        - 超时间隔: TimeOutInterval
        - 一个报文段一个定时器
    - 超时
        - 重传认为超时的报文段
        - 重启定时器
    - 收到ACK
        - 如果是对以前的未确认报文段的确认
            - 更新SendBase
            - 如果当前有未被确认的报文段，TCP还要重启定时器
- 接受方

    - 收到一个一个正常的分组(按序分组)
        - 等待500ms,等待收取第二个按序分组
        - 如果500ms没收到,那就只发这个
        - 如果500ms内收到了,就按照第二个来的分组发送ACK.
    - 收到一个乱序分组
        - 立即发送冗余ACK，指明下一个期待字节的序号
- 快速重传
    - 收到三个冗余ACK
        - 确认数据之后的报文段丢失,重传

TCP建立的过程

- 客户机发送SYN
- 服务器接受SYN,返回SYN&ACK
- 客户机接受服务器SYN和ACK,回应ACK
- 服务器接受ACK,TCP建立完成

TCP断开的过程

- 客户机发送FIN
- 服务器接受客户机的FIN,回应ACK和FIN
- 客户机接受ACK和FIN,发送ACK
- 服务器接受ACK,连接断开
- 客户机不会向服务器发送消息,但是服务器还是可以向客户机发送消息(半关闭)

3.6&3.7 拥塞控制

拥塞控制方法

- 网络辅助的拥塞控制
- 端到端的控制

TCP Reno

先从1开始指数增长（*2），到达阈值后线性增长（+1）

若超时，则窗口设为1，阈值为当前拥塞窗口一半

若3个重复ACK，则阈值为当前拥塞窗口一半，当前窗口为这个新阈值+3

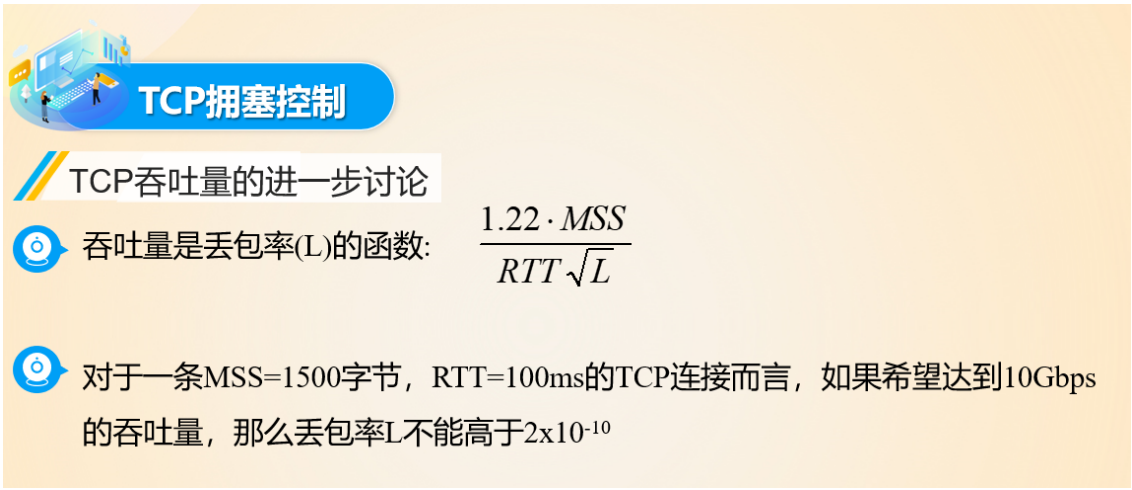
- 拥塞窗口
    - 标记为CongWin,滑动窗口小于等于拥塞窗口
      取接受滑动窗口和拥塞窗口最小值
- 慢启动
    - 拥塞窗口是之前的两倍或者门限值中比较小的那个
    - 每收到一个ACK,拥塞窗口大小+一个MSS
- 拥塞避免
    - 拥塞窗口大小是之前的加一
    - 拥塞窗口的大小大于ssthresh(门限值)进入拥塞避免
    - 每收到一个ACK,拥塞窗口大小+一个MSS*(MSS/窗口大小)
- 三个冗余ACK
    - 门限值=原拥塞窗口一半
    - 拥塞窗口=原拥塞窗口一半+3
- 超时

    - 门限值=原拥塞窗口一半
    - 拥塞窗口=1
- 传输轮回:指发送方一次性发送拥塞窗口大小的数据,然后收到对应的全部ACK或出现异常就是一个传输轮回

公平性

- 如果K个TCP连接共享同一个带宽为R的瓶颈链路, 每个连接的平均传输速率为 R/K

吞吐量

- 一个公式
- 

4 网络层

网络基础

网络层概述

- 目标
    - 实现主机到主机之间的通信
    - 为运输层提供支持
        - 为运输层传递数据,所有运向这个主机的信息就会先到网络层,网络层再转发到运输层,运输层再根据套接字编号进行转发
- 功能
    - 选路
    - 转发
    - 连接建立

虚电路和数据报网络

- 虚电路
    - 工作机制
        - 数据开始流动之前，呼叫建立；流动结束后要断开
        - 每一个分组携带虚电路的标识 (而不是目的主机的地址)
        - 路径上的每一个路由器必须为进行中的连接维持连接状态信息
        - 传输层的连接仅涉及到两个端系统（end system）
        - 链路, 路由器资源 (带宽、缓冲区) 可以分配给虚电路
        - 目的：为了达到类似线路交换的性能
    - 组成.
        - 从源到目的主机的路径

        - VC(Virtual Circuit)号, 沿着该路径的每段链路的一个号码
        - 沿着该路径的每台路由器中的转发表
            - 转发表由入接口,出接口以及各接口的VC号
            - 转发过程
                - 路由器之间或路由器和主机之间会建立许多链路
                - 在转发的时候,每个链路都会做一个标号
                - 根据进入的链路标号以及链路的结构来确定转发的端口和新的VC号(每一次转发都要更新VC号)
    - 用途
        - ATM网络
- 数据报
    - 特点
        - 在网络层没有连接建立过程
        - 路由器：在端到端的连接中不维护连接状态信息
        - 在网络层不存在“联接”的概念
        - 传输报文时使用目的主机地址信息
        - 同一对主机间的报文可能会走不同的路径
    - 用途
        - Internet

路由器工作原理

- 输入端口
    - 排队
        - 如果输入端口的处理速率超过了交换结构的速率，输入端口就可能产生排队
        - 如果若干个输入端口争用一个输出端口,也会造成排队
    - 按照给出的目的地址,使用输入端口的内存中存储的路由选择表，查找输出端口
- 转发结构
    - 功能
        - 选路算法(控制平面)
        - 转发表(数据平面)
    - 结构
        - 经内存交换
            - 分组从输入端口拷贝到内存中再拷贝到输出端口中
        - 经总线交换
            - 一根共享总线来进行交换
        - 经内联网络
- 输出端口
    - 排队
        - 当通过交换结构到达的分组速率超过了输出链路的速率时，需要对分组进行缓存
    - 分组调度策略
        - 先来先服务 FCFS
        - 加权公平排队 WFQ
    - 分组丢弃策略
        - 被动队列管理(丢弃尾部)

        - 主动队列管理
            - 随时计算平均队列长度avgth
            - 最小阈值minth、最大阈值maxth
            - avgth小于minth，允许分组入列
            - avgth大于maxth，分组被标记或丢弃
            - avgth在minth和maxth之间，按照概率标记或丢弃分组

数据平面

IP协议

- 报文格式
    - IP数据报首部
        - 20字节
        - 组成
            - 分片偏移
                - 该分片的第一个字节位于原来分片中的什么位置,假设该分片的第一个字节为原来分片中的第x个字节,那么就称其为x/8
            - 长度
                - 包括首部
            - ID
                - 分片之后ID一样
            - 源地址
            - 目的地址
            - 标志
                - DF
                - MF
                - FF
                    - 最后一个为0
                    - 不是最后一个就为1
    - IP数据报分片
        - 网络链路有MTU属性,就是一次性最大传输的帧长度
        - 大的IP数据报在网络中会被分成小的分片
        - 一个数据报变成了几个数据报,一个数据报的内容部分分成若干个小的数据报,然后加上头部元素
        - 重组只在目的主机进行
        - 数据报头部的标识、标志以及片偏移字段用于目的主机对接收的分片进行重组
- IP地址
    - 32位主机或路由器的接口标志符
    - 接口:连接主机,路由器之间的物理链路
    - IP地址只和接口有关,和路由器,主机没有关系
    - 结构
        - 网络号
        - 主机号
    - 在同一个局域网内的主机接口或者路由器接口IP地址中的网络号必须是一样的
    - 路由器的每一个接口都具有不同网络号的IP地址

- 子网和掩码
  - 从主机中借用一部分的位数作为子网号
  - 通过将网络号和子网号相应的位置全置1，主机号相应的位置全置0，即可得到子网掩码
  - 引入子网的转发
    - 比较目的地址和子网掩码的并和转发表中的网络号是不是一样的,如果不一样就是下一个,一样的话查找输出端口直接转发之.

CIDR无类域间路由

Classless InterDomain Routing

- 编码格式
  - IP地址 ::= {<网络前缀>, <主机号>}
  - 这样可以更加方便地划分子网,其中斜线后面的位数就是表示网络号的位数
  - 斜线记法：192.168.0.1/24
- 最长前缀匹配
  - 应当从匹配结果中选择具有最长网络前缀(n最大的)的路由：最长前缀匹配(longest-prefix mating)。
  - 匹配的意思就是对于一个表项xx:xx:xx:xx/n,IP包的目的地址的前n位与表项xx:xx:xx:xx的前n位相同就可以 目的地址的前n位和网络编号的前n位(前缀匹配)
    - 例子:206.0.71.142和206.0.71.128/25,这个地址和206.0.71.128的前25位相同,就称之为匹配
  - 可以代表这个网络的子网掩码是前n位是1,这个地址和子网掩码的并等于网络地址的话就匹配
  - 也就是说转发,只能转发匹配的表项,不匹配的是不能被转发的.
- 总结

  "/24"这种划分形式，相当于掩码的前n位为1，

  实现了：小的网络统一归大的网络管，即按照地址的前缀来划分

DHCP服务器

- DHCP: Dynamic Host Configuration Protocol 动态主机配置协议:从服务器上动态获取IP地址
- 做法
  - 首先客户端广播一个目的为255.255.255.255的DHCP发现报文,通过transaction ID来标记这个是我的请求.主机收到transaction ID和这个一样的DHCP提供报文就知道肯定是自己的了
  - DHCP服务器会广播一个DHCP提供报文,来响应DHCP发现,其中包括各种网络信息
  - 接着客户端会广播一个DHCP请求报文,请求使用DHCP服务器推荐的报文.
  - DHCP服务器响应一个ACK请求

NAT(内网-外网地址转换协议)

- 本地网络只要使用一个IP地址就可以和外部网络相连,本地网络中的所有主机(IP:10.0.0.x)在外面的人眼中就是一个特殊的外部地址.也就是说本地网络的所有主机和外面的网络交流都是用同一个IP地址来进行通信.所有离开本地网络的报文都拥有同一个源IP地址,有不同的源端口号
- 维护一个NAT路由,路由在本地网络有一个地址,在全球网络也有一个地址.还有一张NAT表,分别记录本地网络中的地址,进程和全球网络的地址,进程的对应关系
- 做法
  - 发送数据报: 将每个外出报文的源IP地址,端口号替换为NAT IP地址以及新的端口号
  - 接收数据报:根据NAT转换表将每个进入报文的NAT IP地址,端口号替换为相应的目的IP地址以及端口号,目的地址变成NAT转换表里面的元素

- 有服务器怎么办
  - 静态记录
    - 如(123.76.29.7,80) 总是指向(10.0.0.1,80)
  - 即用即插
    - 了解公共IP地址
    - 向路由器注册/移除映射记录
    - （内部IP地址，内部端口号）（公共IP地址，公共端口号）
    - 内部主机通过某种渠道向外部应用程序公开（公共IP地址，公共端口号）
  - 中继

ICMP(互联网控制报文协议)

- 用于主机、路由器、网关之间交换网络层信息
- ICMP 报文封装在IP分组(是IP报里面的数据部分)中

IPv6协议

- 无检查和，中间结点无需计算
- 中间结点不再负责分片和重组，由端结点负责
- 首部长度固定，加速中间结点转发速度
- 地址的长度是128位的
- 与IPv4的兼容
  - 隧道模式,IPv6报文段作为IPv4的数据放在分组
  - 隧道里面,如果有一部分网络不支持IPv6,那么就把IPv6的分组放到IPv4分组的数据部分
  - IPv4分组的目的地址是IPv4隧道的终点,原地址是IPv4隧道的起点

控制平面

选路算法

- AS内部选路协议
  - OSPF算法(Link State)
    - 所有路由器都知道整个网络拓扑图以及链路的费用信
    - 迪克斯特拉算法
      - 先计算每个点到源点的距离
      - 选择一个到源点距离最短的一个点w选一个进入K集合
      - 对于这个点w,看看所有和w邻接点v,看看是原来的D(v)短,还是经过w的新路径短:D(w)+c(v,w)

        D(v) = min(D(v), D(w) + c(w,v))
    - 向本自治系统中所有路由器发送信息，使用的方法是洪泛法
    - 发送的信息就是与本路由器相邻的所有路由器的链路状态
    - 只有当链路状态发生变化时，路由器才用洪泛法向所有路由器发送此信息,过了30分钟,就算没有发生变化,也要广播状态
    - 所有路由器会构建一个链路状态数据库,这个数据库就是全网络的拓扑结构图
  - RIP算法(Distance Vector)
    - 每个路由器仅有与其相连链路的费用信息
    - dx(y)=minv{c(x,v)+dv(y)}
    - 路由器之间的链路消费永远是1

        - 更新算法
            - 路由器X得到相邻路由器Y的路由表，从而得知：Y到网络Z的最短距离为N
            - 如果路由器X没有到网络Z的路由条目，则添加一条经由路由器Y到网络Z距离N+1的路由条目
            - 如果路由器X已有到网络Z的路由条目，其距离为M，如果M>N+1，则更新该条目为经由路由器Y到网络Z距离N+1，否则不更新
        - 毒性逆转
            - 如果z通过y选路到达目的地x :z将通告y，它到x的距离是无穷大 (所以 y不会通过z到达x)
        - 选路更新消息每30s在邻居之间以RIP响应报文（RIP通告）的形式进行交换
        - 路由器经过180s没有收到来自某个邻居的RIP通告，则认为该邻居已离线，修改选路表，向其它邻居广播
- AS间选路协议(AS自治系统)
    - BGP算法
        - AS路由需要做的事情
            - 向该AS内部的所有路由器传播这些可达性信息
            - 基于该可达性信息和AS策略，决定到达子网的“好”路由
        - BGP发言人
            - 和其他AS交换信息使用TCP连接
            - BGP发言人要向其他BGP发言人交换AS路由信息,可以是边界路由器
        - BGP路由通告
            - 其他AS可以通过BGP发言人传递通告
            - 通告分成两部分
                - 前缀
                    - 可以到达的网络号
                - 属性
                    - AS-PATH: 该属性包含了前缀的通告已经通过的那些AS(经过哪些AS了)
                    - NEXT-HOP: 指明到下一跳AS的具体的路由器 (因为可能从当前AS到下一跳AS之间可能有多条链路)
            - 然后BGP通过iBGP向本AS内所有路由器发送通告
        - BGP路由选路
            - 本地偏好值: 策略决定。具有最高本地偏好值的路由将被选择。
            - 最短AS-PATH： 在余下的路由中，具有最短AS-PATH的路由将被选择。
            - 从余下的路由中，选择具有最靠近NEXT-HOP路由器的路由:热土豆路由。
            - 如果仍余下多条路由，该路由器使用BGP标识以选择路由。
- 总的选路
    - 路由器知晓前缀的存在性
    - 确定此前缀的转发端口
        - 使用BGP路由选择确定最佳域间路由
            - 通过BGP算法算出去往哪一个AS合适
            - 然后根据表项中的NEXT-HOP找到路由
        - 使用IGP路由选择确定最佳域内路由
            - 根据AS内部的选路算法(OSPF/RIP)来选择去往本AS内的何处
        - 确定最佳路由的转发端口
            - 决定完之后就可以添加端口了

- 将（前缀，端口）表项放入转发表中

链路层

6.1 链路层概述

术语

- 节点：主机和路由器
- 链路：沿着通信路径连接相邻节点的通信信道
- 帧:数据链路层的分组单元

服务

- 成帧,链路访问
- 差错检测
- 可靠传递
- 流量控制
- 差错纠正

6.2 差错检测

CRC循环校验码

6.3 多路访问链路和协议

广播信道的特点

- 单个共享广播信道
- 两个或多个节点同时传输：相互干扰
- 碰撞：一个节点同时收到两个或多个信号

信道划分协议

- 将信道划分成小的“片”（时隙、频率、编码）
- 将“片”分配给节点使用
- 种类
    - TDMA
    - FDMA
    - CDMA

随机访问协议

- 信道没有被分割，允许碰撞
- 碰撞恢复
- ALOHA
    - 纯ALOHA
    - 时隙ALOHA
- 碰撞
    - 当一个结点传输数据的时候收到了别的结点发来的数据就叫做碰撞.
    - 强化碰撞:当发送数据的站一旦发现发生了碰撞时，除了立即停止发送数据外，还要再继续发送若干比特的人为干扰信号
    - 最先发送数据帧的站，在发送数据帧后至多经过时间 2t （两倍的端到端时延）就可知道发送的数据帧是否遭受了碰撞。以太网的端到端往返时延 2t称为争用期，或碰撞窗口。

轮流协议

- 节点轮流传送，但数据量大的节点轮流更长时间
- 主节点邀请从节点轮流传输
- 控制令牌依次通过各个结点

6.4 交换局域网

Mac地址

- 在数据链路层标识每块网络适配器，使得能够在广播信道上寻址目标节点,48位

地址解析协议（ARP）

- 根据目标的IP地址获取其MAC地址
- ARP缓存:局域网节点的IP/MAC地址映射
- ARP请求包
    - 目的为FF-FF-FF-FF-FF,包含自己源的Mac头
    - IP报头
    - ARP报文:你的Mac是啥啊
- ARP应答报
    - 源是自己Mac地址的报头
    - IP报头
    - ARP报文:我的Mac是...
- 注意:每一次经过路由器都更新一遍Mac头,但是不会更新IP头,Mac头的目的会是下一个路由器Mac,源就是自己路由器的Mac

以太网

- 帧
    - 结构
        - 数据
        - 首部
            - 同步码
            - 源Mac
            - 目的Mac
            - 类型
        - 尾部CRC

CSMA/CD

- 特点

  先听后发、边听边发、随机延迟后重发
    - 没有时隙
    - 当适配器侦听到其它适配器在传输，则它不传输帧, 即载波侦听
    - 正在传输的适配器若检测到其它适配器也在传输，则它中止自己的传输, 即碰撞检测
    - 在重新传输之前，适配器要等待一段随机时间，即随机回退
- 几个定义
    - 拥塞信号: 用来确保所有传输者都能检测到碰撞而传输的信号；48比特长
    - 比特时间: 传输1比特5所需时间。在10Mbps的以太网中，当K=1023时，等待时间大约为50ms
- 指数回退

    - 中止传输后，适配器进入指数回退阶段，在经历第m次碰撞后，适配器随机从{0,1,2,…,2m-1}中选择K值。适配器在等待 K·512比特时间后在重传
- 争用期
    - 以太网取 51.2us 为争用期的长度。
    - 对于 10 Mb/s 以太网，在争用期内可发送512 bit，即 64 字节。
- 最短有效帧长
    - 以太网规定了最短有效帧长为 64 字节，凡长度小于 64 字节的帧都是由于冲突而异常中止的无效帧。
- 信号编码
    - 曼彻斯特编码
    - 差分曼彻斯特编码
- 链路层交换机
    - 工作原理
        - 不断监听各接口是否有信号
        - 收到无差错的帧则缓存，反之将差错帧丢弃
        - 若所收帧的目的MAC地址属另一网段，则通过站表决定向何接口转发
        - 交换机不转发同一网段内通信的帧
        - 交换机不修改所转发的帧的源地址
    - 交换机是透明的
        - 这里所谓“透明”是指局域网上的每个站并不知道所发送的帧将经过哪几个交换机，即交换机对各站来说是看不见的
    - 选路原理
        - ① 从接口x收到帧，有差错则丢弃，否则在站表中查找目的站MAC地址；
        - ② 找到有，则取出相应的接口d，转③，否则转⑤；
        - ③ 如果所给MAC地址的接口d=x，则丢弃此帧（不需要转发），否则从接口d转发此帧；
        - ④ 转到⑥；
        - ⑤ 向除x以外的所有接口转发此帧（可保证找到目的站）
        - ⑥ 如源站不在站表中，则将源站MAC地址写入站表，登记该帧进入交换机的接口号和时间，设置计时器，然后转⑧。否则转⑦；
        - ⑦ 更新计时器（由于网络拓扑经常变化，因此，超时记录要删除，以反映最新状态）；
        - ⑧ 等待新的数据帧。转①
        - 支撑树---交换机互相知道各自的拓扑结构,构建一个最小生成树.

VLAN

- 局域网交换机是组建虚拟局域网的核心设备。
- 组成逻辑工作组的各结点不受物理位置的限制，换言之同一逻辑工作组的成员不一定要连接在同一个物理网段上。
- 当一个结点从一个逻辑工作组转移到另一个逻辑工作组时，只需要通过软件设定，而不需要改变它在网络中的物理位置。
- 不同的子网就是不同的VLAN,所以说VLAN具有流量隔离的作用,一个VLAN的内容只能传输到同一个VLAN或者trunk端口里面.
- 从一个VLAN转发到不同的VLAN里面,先转发到trunk端口,通过trunk端口进入路由器完成转发.

集线器--单纯把几个机器连接在一起,碰撞域变大,集线器左边的元素会影响集线器右边的元素传递数据,但是集线器左边和右边就互联了

7 无线网络和移动网络

7.1 概述

无线网络中的元素

- 无线主机
  - 可以是便携机, PDA, IP 电话;
  - 能运行程序;
  - 本身既可能是固定，也可能是移动的.
- 无线链路
  - 典型的作用是用于连接无线主机和基站;
  - 也可以用于骨干链路:就是基站与边缘路由器相连的链路
- 基站
  - 典型的作用是用于连接无线网络;
  - 负责向其覆盖范围内的主机发送和接收分组，在无线网络和无线主机之间起链路层中继作用。如：蜂窝塔、802.11 接入点
    - 基站向基站覆盖范围内的主机提供类似于链路层中继作用的通信服务
  - 无线主机连入基站,然后基站与更宽广的网络进行连接.主机->基站->局域网->因特网(宽广的网络)

两种模式

- 基础设施模式
  - 基础设施模式是指预先建立起来的、能够覆盖一定地理范围的一批固定基站。
  - 移动主机通过基站接入有线网络;
  - 切换:移动主机的移动可能会改变与之相关联的基站。
  - 关联
    - 无线主机位于某个基站的无线通信覆盖范围内
    - 该主机使用该基站中继它与更大网络之间的数据
  - 切换
    - 当一台移动主机移动范围超出一个基站的覆盖范围而到达另一个基站的覆盖范围后，它将改变其接入更大网络的连接点
- Ad hoc
  - 无基站;
  - 节点(移动主机)仅仅能够在其覆盖范围内向其他节点传送数据;
  - 节点之间相互通信组成的临时网络:在它们内部进行选路和地址分配。
  - 节点和节点之间传输数据,热点

7.2 无线链路特征

递减的信号强度

- SNR 信噪比
- BER 比特差错率
- 隐藏终端

来自其他源的干扰

多径传播

CDMA码分多址访问

- 每个用户都被指派一个m位长的码片序列
- 用户发送1的时候就发送原来的码片,如果发送0的化就是反码
- 习惯上，将码片序列中的“0”写成“-1”，“1”写成“+1”
- 任何两个站点的码片向量规格化内积为0,就是每个维度的值互相乘然后加起来的值为0
- 自己和自己相乘为1,自己和自己的反码
- 基站假如想获得来自于S的信号,就应该是(Sx+Tx)*S(S为码片向量)

7.3 802.11无线LAN

所有的无线终端通过基站AP通信

802.11b的信道划分

- 2.4GHz—2.485GHz，共85MHz
- 划分出11个信道
- 1,6,11是无重叠的,其他的都是有重叠

主机关联基站的过程

- 每个AP周期性发送信标帧，包括AP的SSID和MAC
- 主机对11个信道进行扫描，获取所有可用的AP的信标帧
- 主机选择其中一个AP进行关联，加入其所属子网
- 主机向关联AP发送DHCP发现报文，获取IP地址
- 可能需要身份鉴别

CSMA/CA

- 发送方工作流程
    - 如果侦听到信道闲置了DIFS 秒,发送方在发送数据帧之前首先使用 CSMA协议发送一个短的请求发送RTS(request-to-send)帧给AP：
        - RTS会被所有节点侦测到
            - 发送方发送
            - 其他站点让开
    - AP过SIFS后广播一个CTS给RTS,CTS是被广播的
    - 如果侦听到信道忙，则选择一个随机避退值作为定时器的定时时间，并在侦听
    - 信道闲置时递减该值。
    - 定时时间一到且信道空闲就发送数据
    - 过了SIFS秒后,主机开始发送数据
    - 如果收到确认，且站点要继续发送数据,过CSMA的定时器时间之后继续发送
    - 如果没有收到确认（ ACK）,则在更大范围内选取随机值作为定时器,过定时器时间后发送
- 接收方工作流程
    - 如果帧收到则OK，等待 SIFS秒后返回ACK (ACK是必须的因为隐蔽站问题)

802.11 Mac帧格式

- 地址1: 无线主机或 AP 接收该帧的MAC地址
- 地址2:无线主机或 AP 发送该帧的MAC地址
- 地址3: 与AP连接的路由器接口的MAC地址,注意,与AP连接的路由器,也就是边缘路由器的地址

7.5 移动管理不考

考试要点

时延：处理时延，等待时延，传播时延。

时延带宽积

应用层：无

传输层：

TCP：序号报文段，长度，RTT预估计算

CIDR：下一跳怎么计算，16进制表格数值变了，重新填值

差错检测：校验码，最短有效真长

交换机的转发原则，栈表更新的计算（），

最短路径树，(DIJIESTA)

拥塞控制：TCP Reno，TAHOLE折线图

TCP的可靠传输

无线网络帧校验和重传

CSMA/CD

1. ~~延时定义：处理延时，等待延时，传播时延，发生在哪些地方。时延，带宽积~~
2.

1. 给定链路费用的最短路径树(dijie)
2. ~~IP数据报，每个分类字段的具体含义~~
3. 拥塞控制 Reno Thole的计算方式，不给图，给表。
4. CIDR下一跳的计算方法，IP-》下一跳的端口
5. ~~延时定义：处理延时，等待延时，传播时延，发生在哪些地方。时延，带宽积~~
6. ~~TCP序号，，可靠传输重传，RTT的估算~~
7. 链路层差错检验，校验码，最小有效帧长
8. ~~交换机 栈，栈表更新原则~~
9. 无线CSMA/CD，无线帧的的校验和重传
10. WEB一次请求历程

1时延

延时定义：处理延时，等待延时，传播时延，发生在哪些地方。时延，带宽积

时延定义

- 传输时延
  - 数据不断从设备推送到网络上，需要花费一定的时间，推送的速度受到网络带宽的影响。(数据从设备传输到通信链路上的时间)
- 传播时延
  - 数据不断从链路的一端传送到链路的另一端，需要花费一定的时间，传送速度受到电子传送速度影响。(在通信链路上的时间)
- 节点处理时延
  - 设备接受数据以及决定往什么端口转发数据,处理这个的时延
- 排队时延
  - 分组在通信端口的排列队列上排队等待离开设备

时延带宽积

链路上可以存在的最大bit数

2传输层

GBN, SR, TCP

TCP可靠传输，RTT的估算，Reno，Thole

先从1开始指数增长（*2），到达阈值后线性增长（+1）

3个ACK：门限变为一半，窗口变为门限+3

超时：门限变为一半，窗口变为1

TCP Tahoe版本对上述两个事件并不区分，统一将CongWin降为1。

RTT的估算

样本RTT(SampleRTT): 对报文段被发出到收到该报文段的确认之间的时间进行测量

a是样本的系数，b是样本-估计差的系数

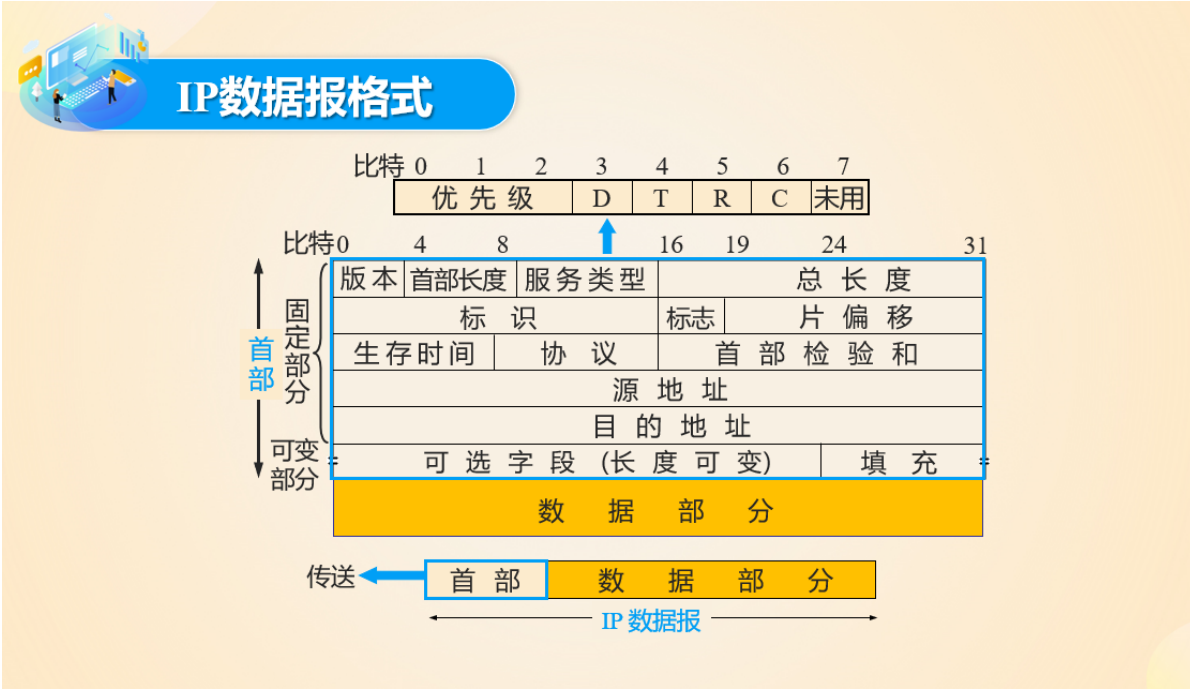
```
EstimatedRTT = (1- a)*EstimatedRTT + a *SampleRTT
第一次计算时：EstimatedRTT=SampleRTT

DevRTT = (1-b)*DevRTT + b*|SampleRTT-EstimatedRTT|
注意，第一次计算时，DevRTT=0.5*SampleRTT

最终：TimeoutInterval = EstimatedRTT + 4*DevRTT
```

3网络层

IP数据报格式



版本：IPv4，v6

服务类型：报文的处理方式，每一位分别代表最小延时、最大吞吐量、最高可靠性、最小成本，只选一个。其它保留。

生存时间：就是TTL跳数，每经过一个减1

首部校验和：只校验头部

总长度：占16bit，单位是字节，取值范围0-65535，实际1报文数据长度为总长度减去首部长度。

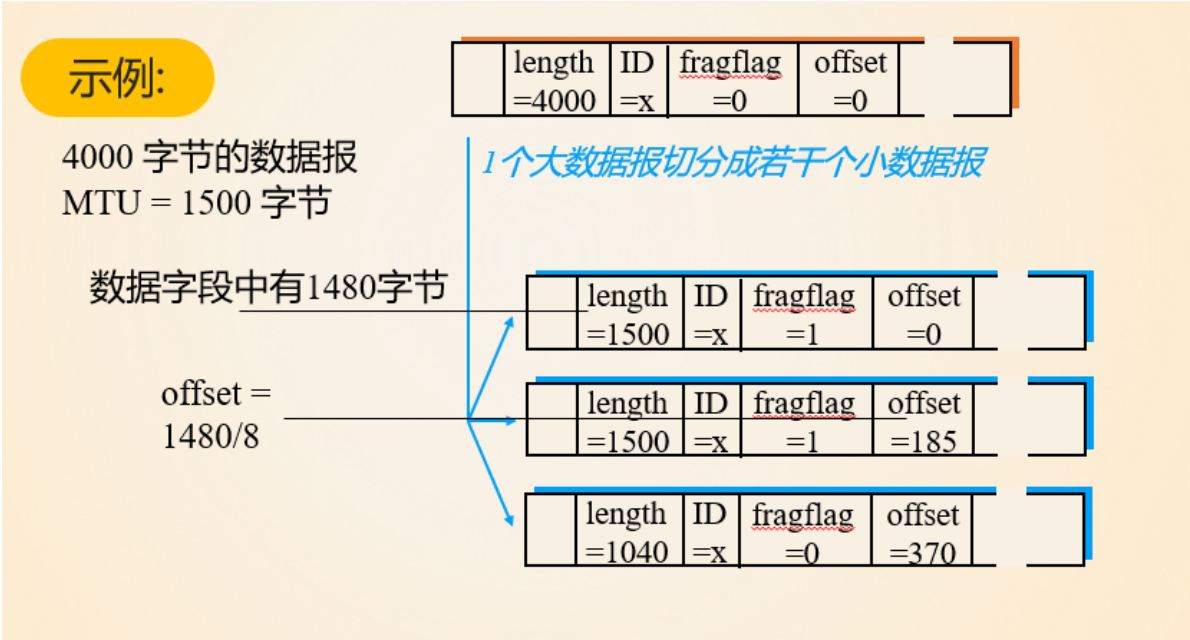
以太网链路报文只允许1500字节。这怎么办？网络链路 MTU (最大传输单位)

IP数据报需要分片和重组：

标识：传输层的同一次报文

标志：3bit，1位保留，2位表示是否能分片，3位表示分片是否结束。1为未结束，0位为结束。

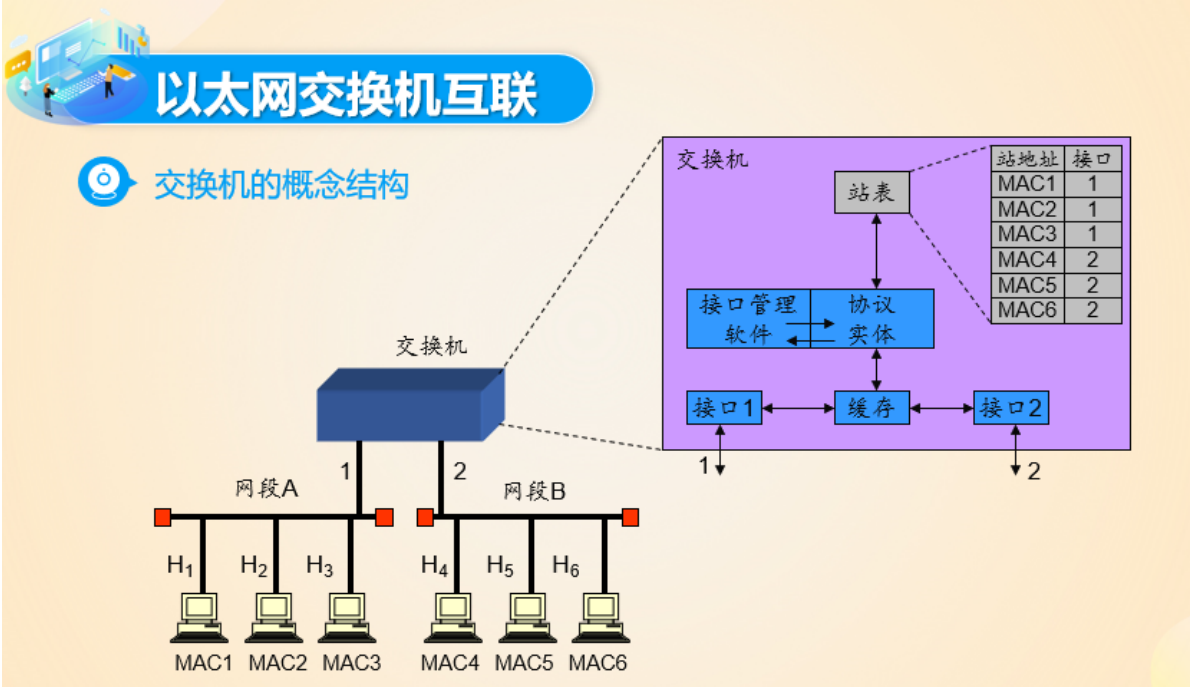
片偏移：每个分片在整个报文（分组）中的位置（8字节为度量单位）



CIDR下一跳

交换机

站表，站表更新原则



站表：（站MAC地址，接口号）

交换机站表，选录原理

1. 从接口x收到帧，有差错则丢弃，无差错则在站表中查找目的MAC地址
    1. 找到目的MAC地址，提取目的接口d
        1. 若d==x，丢弃（无需转发）
        2. 否则，转发
    2. 站表中无目的MAC地址，向除x外的所有接口转发（以寻找目的站）
    3. 如果源站不在站表中，写入站表，登记该帧进入的接口号和时间，设置计时器
    4. 如果在站表中，更新计时器

等待新的数据帧。

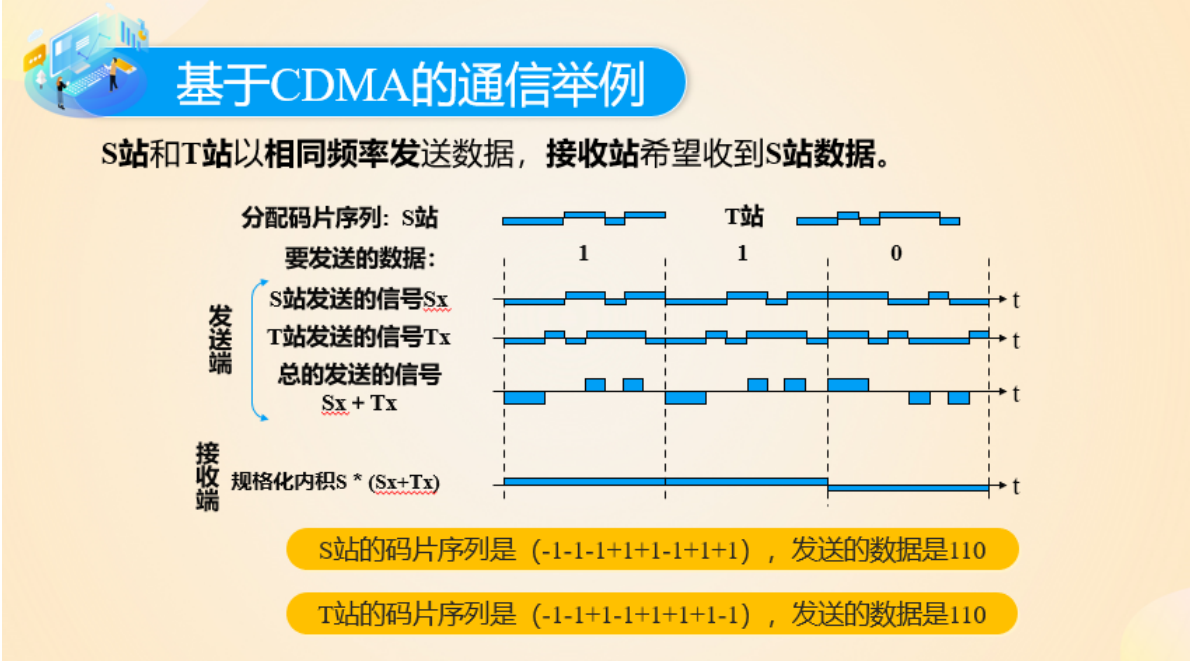
链路层

CRC校验

无线通讯

CDMA码片序列Code Division Multiple Access

- 每个用户都被指派一个m位长的码片序列，想表示1，就发送这个序列，表示0，就发送这个序列的反码。
- 不同用户间的码片序列乘积（规格化内积）为0
- 与自己的内积为1，与自己的反码内积为-1



如上图，想在信号中分离出S站发的帧，那就用S的码片序列S，与信号和作乘法。

CSMA/CA

适配器收到来自网络层的数据报，创建帧

若适配器检测到信道空闲，则开始传输帧；若检测到信道忙，就开始等待，直到信道空闲再开始传输该帧

若适配器传输了整个帧而没有检测到其它适配器的传输，则该适配器完成该帧的传输

若适配器在传输时检测到其它适配器也在传输，则停止传输，发送拥塞信号中止传输后，适配器进入指数回退阶段，在经历第m次碰撞后，适配器随机从{0,1,2,…,2^m-1}中选择K值。适配器在等待 K*512比特时间后，返回第2步

发送方"预约"，在发送数据帧之前先发送一个短的请求帧RTS(request-to-send)给AP；AP广播一个允许发送 **CTS**（clear-to-send）帧响应**RTS**；RTS 被所有节点侦听到，发送方发送数据，其他站点推迟发送。

- 发送方工作流程
    - 如果侦听到信道闲置了DIFS 秒,发送方在发送数据帧之前首先使用 CSMA协议发送一个短的请求发送RTS(request-to-send)帧给AP：
        - RTS会被所有节点侦测到
            - 发送方发送
            - 其他站点让开
    - AP过SIFS后广播一个CTS给RTS,CTS是被广播的
    - 如果侦听到信道忙，则选择一个随机避退值作为定时器的定时时间，并在侦听
    - 信道闲置时递减该值。
    - 定时时间一到且信道空闲就发送数据
    - 过了SIFS秒后,主机开始发送数据
    - 如果收到确认，且站点要继续发送数据,过CSMA的定时器时间之后继续发送
    - 如果没有收到确认（ ACK）,则在更大范围内选取随机值作为定时器,过定时器时间后发送
- 接收方工作流程
    - 如果帧收到则OK，等待 SIFS秒后返回ACK (ACK是必须的因为隐蔽站问题)

CRC校验

ARQ重传（ARQ好像包括停等，GBN，SR，并且有快传）

从开机到上网

正在连接的笔记本需要获得IP地址、网关、DNS服务器等信息：

DHCP 请求依次进行UDP封装，IP封装，802.3 以太网帧封装

以太帧向局域网发送广播 (目的：FFFFFFFFFFFF)， 由运行 DHCP Server的网关路由器收到

找DNS:

ARP广播，得到路由器的MAC地址

建立TCP连接

发送HTTP请求