

ELG5374期末考试押题

概念

1. IRP ERP

- IRP: **RIP OSPF**
- ERP: **BGP**

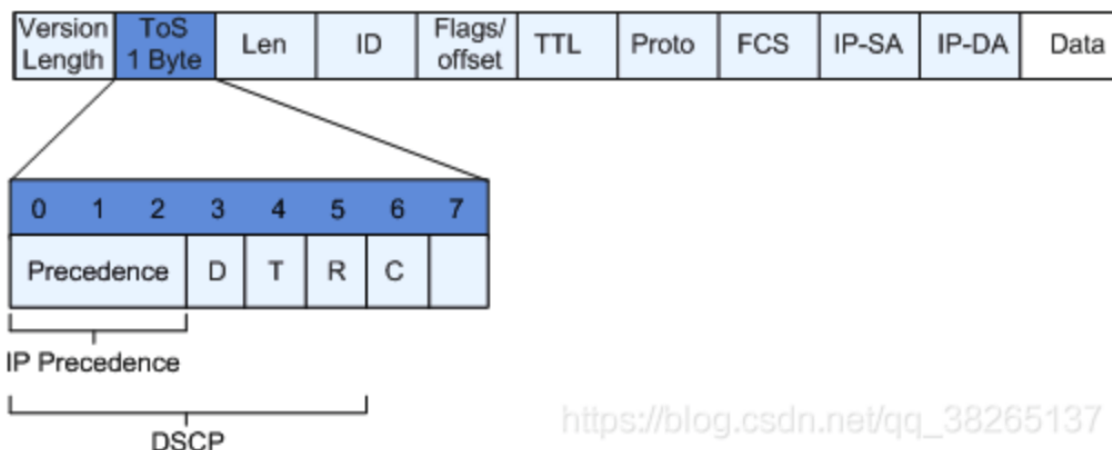
2. distance vector(RIP) v.s. link state(OSPF)

Criterion	Link State Algorithm	Distance Vector Algorithm
Message complexity	with n nodes, E links: $O(nE)$ messages sent	- exchange between neighbors only - convergence time varies
Speed of convergence	- $O(n^2)$ algorithm requires $O(nE)$ messages - may have oscillations	- convergence time varies - may create routing loops - count-to-infinity problem
Robustness (what happens if router malfunctions?)	- node can advertise incorrect link cost - each node computes only its own table	- Any node can advertise incorrect path cost - Each node's table is used by others. Therefore, errors propagate through the network!

- 路由协议类型：RIP是距离矢量协议，而OSPF是链路状态协议。距离矢量协议使用跳数来确定传输路径。链路状态协议分析不同的源，如速度，成本和路径拥塞，同时识别最短路径。
- 路由表构造：RIP使用周围的路由器请求路由表。然后合并该信息并构造自己的路由表。该表定期发送到相邻设备，同时更新路由器的合并表。在OSPF中，路由器通过仅从相邻设备获取所需信息来合并路由表。它永远不会获得设备的整个路由表，并且路由表构造非常简单。
- 跳数限制：RIP最多只允许15跳，而在OSPF中没有这样的限制。
- 使用的算法：RIP使用距离向量算法，而OSPF使用最短路径算法Dijkstra来确定传输路由。
- 网络分类：在RIP中，网络分为区域和表格。在OSPF中，网络被分类为区域，子区域，自治系统和骨干区域。
- 复杂性级别：RIP相对简单，而OSPF则要复杂得多。
- **RIP与OSPF应用**：RIP适用于较小的网络，因为它具有跳数限制。OSPF非常适合大型网络 [原文链接](#)

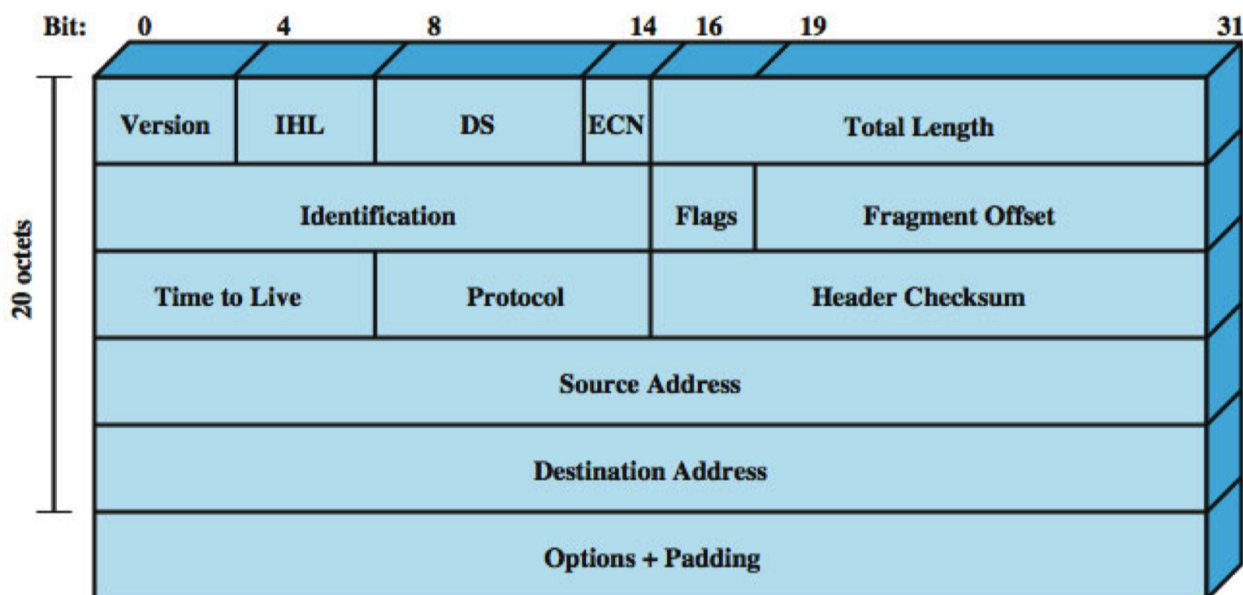
3. QOS

- IP报文头ToS (Type of Service) 域由8个比特组成，其中3个比特的Precedence字段标识了IP报文的优先级，Precedence在报文中的位置如下图所示：



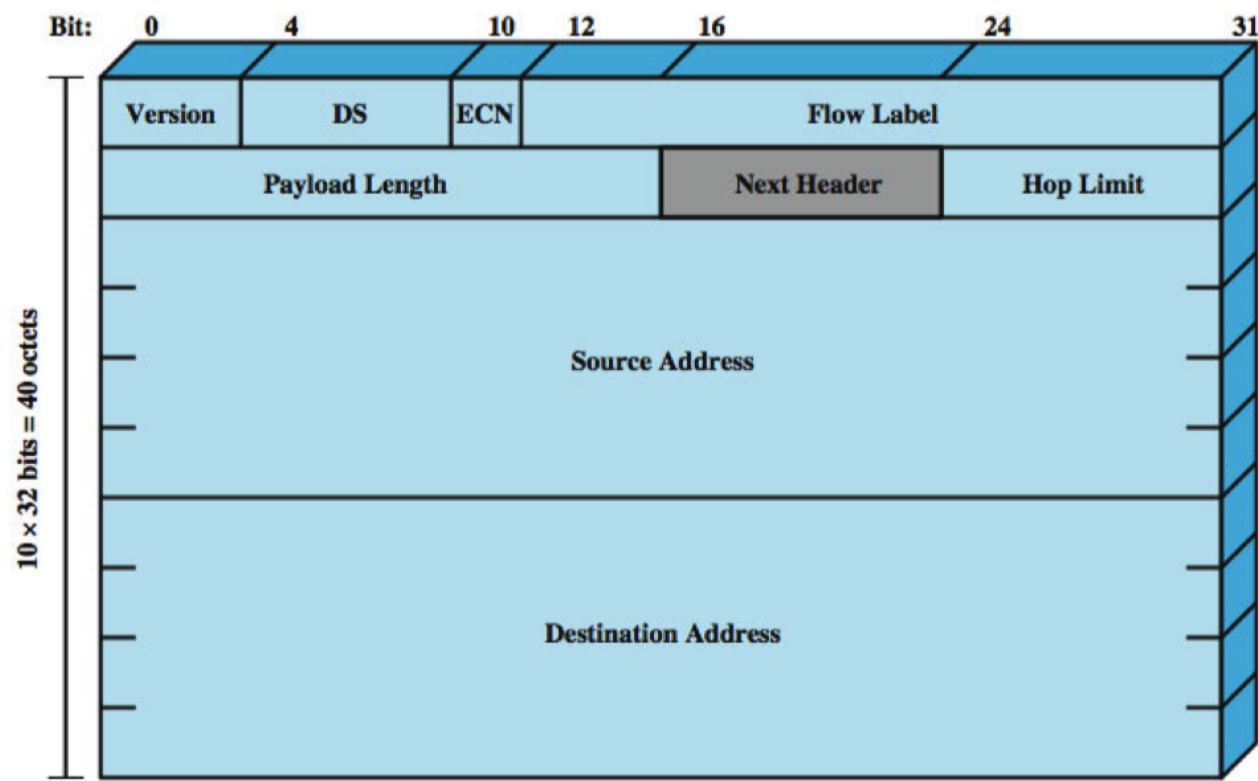
-
- D比特表示延迟要求 (**Delay**, 0代表正常延迟, 1代表低延迟)
- T比特表示吞吐量 (**Throughput**, 0代表正常吞吐量, 1代表高吞吐量)
- R比特表示可靠性 (**Reliability**, 0代表正常可靠性, 1代表高可靠性)
- C比特表示开销 (**Cost**)
- 影响网络质量的因素：
 - **bandwidth**
 - **delay**: 实时应用通信质量都比较关注时延大小, 如语音、视频等。以语音传输为例, 时延是指从说话者开始说话到对方听到所说内容的时间。若时延太大, 会引起通话声音不清晰、不连贯或破碎
 - **jitter**: 由于每个报文的端到端时延不一样, 就会导致这些报文不能等间隔到达目的端, 这种现象叫做抖动。一般来说, 时延越小则时延抖动的范围越小。语音和视频等实时业务是极其不能容忍抖动的。报文到达时间的差异将在语音或视频中造成断续; 另外, 抖动也会影响一些网络协议的处理, 有些协议是按固定的时间间隔发送交互性报文, 抖动过大就会导致协议震荡。利用缓存来减小抖动
 - **loss**: 语音传输中, 丢失一个比特或一个报文的信息, 通话双方往往注意不到; 在视频广播期间, 丢失一个比特或一个报文可能造成屏幕上瞬间的波形干扰, 但视频很快就会恢复正常。但文件传输中丢包问题就很严重了
- 服务类型
 - **Cell Transfer Delay (CTD)**: delay between start & finish of cell
 - **Peak to Peak Cell Delay Variation (CDV)**: difference between maximum CTD and minimum CTD 最大可接受信元抖动
 - **Peak Cell Rate (PCR)**
 - **Cell Loss Ratio (CLR)**: % of cells lost
 - **Sustained Cell Rate (SCR)**: average rate of transmitted cells
 - **Burst Tolerance (BT)**: maximum burst size at PCR
 - **Maximum Burst Size (MBS)**: maximum No. of cells sent at PCR

4. IPv4 header 20~60字节



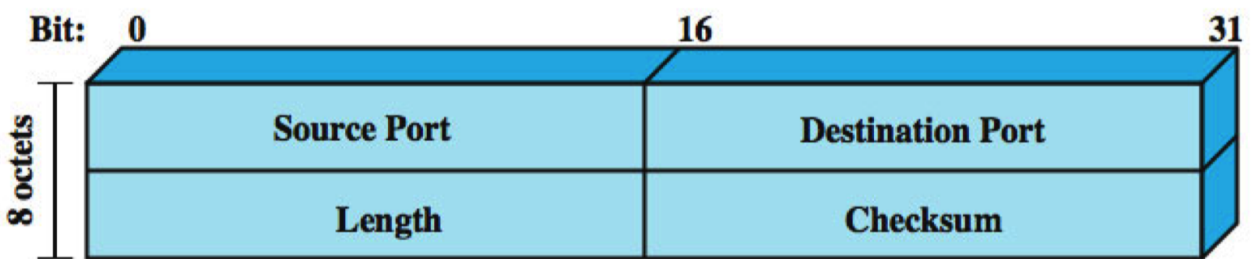
- **Version:** 4
- **IHL:** header长度，以4字节为单位
- **DS+ECN:** 服务类型，QOS相关
- **Total Length:** packet总长度
- **Identification:** 分段时，用来标注同一原片段
- **Flags:** More: 接下来还有无分片 DF: 该片段还能否继续细分
- **Fragment Offset:** 分片起始点偏移量
- **TTL:** 生存时间，到达一个router便先减一
- **Protocol:** 标注packet内容是什么protocol
- **Header Checksum:** 只检查header
- **Source Address:** 32位
- **Destination Address:** 32位
- **Options**
- **Padding:** 补0至4字节的整数倍

5. IPv6 header 固定长度40字节

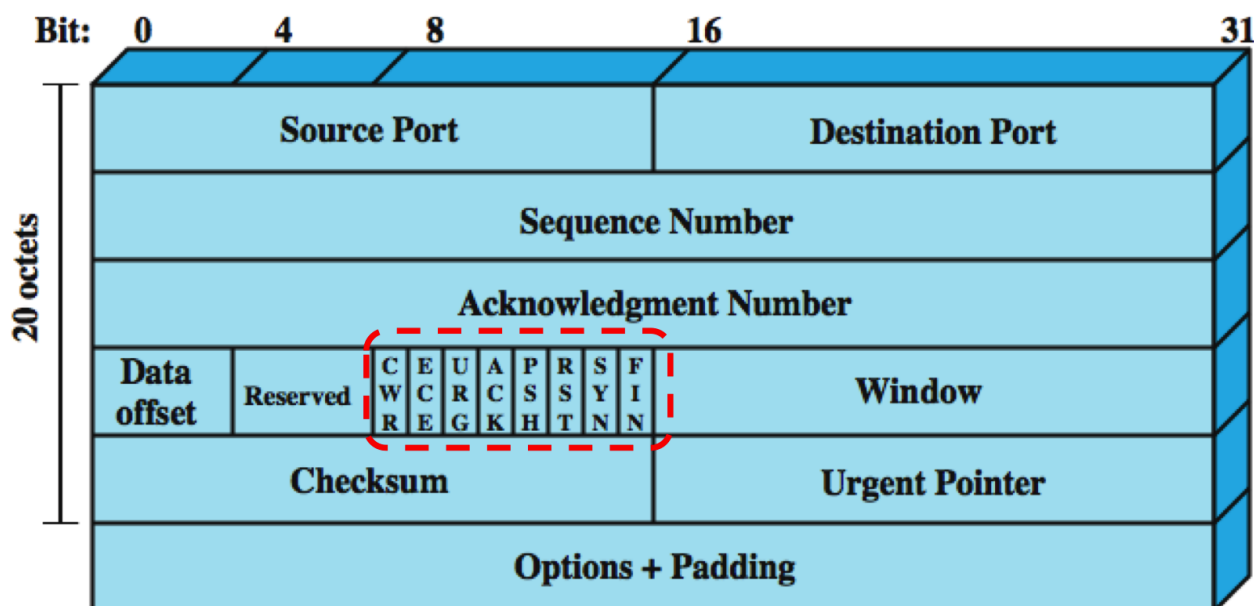


- **Version:** 6
- **DS+ECN:** 同理
- **Flow Label:** flow label+两个地址=唯一flow, 可用于QOS处理
- **Payload Length:** payload长度 (包括extension header)
- **Next Header:** 指向第一个extension header
- **Hop Limit:** 同IPv4的TTL
- **Source Address:** 128位
- **Destination Address:** 128位

6. UDP header



7. TCP header



- **Source Port**
- **Destination Port**
- **Seq Num**: 这个TCP segment是总数据的第几个Byte
- **Ack Num**: Ack num之前的数据已经接收成功，现在想收到Ack num开始的Byte
- **Data Offset**: 首部长度
- **CWR**: congestion window reduced，与ECE共同作用
- **ECE**: =1意味着通知对方已将接收窗口缩小
- **URG**: 紧急标志位，=1则urgent pointer有效
- **ACK**: =1则Ack num有效
- **PSH**: 不进行缓存直接推送到应用的标志位
- **RST**: 重新连接
- **SYN**: 同步序号以初始化连接的标志位
- **FIN**: 发送数据完毕的标志位（表明不会再发送数据过来）
- **Window**: 接收窗口大小
- **Checksum**: 首部检验
- **Urgent Pointer**
- **Options**
- **Padding**: 补0至4字节的整数倍

8. ATM

1	GFC		VPI	
2	VPI		VCI	
3	VCI			
4	VCI		PT	CLP
5	HEC			
6-53	User Information			

- **GFC**: 通常置为0。它的主要功能是控制由用户终端至网络方向ATM接续的业务流量。
- **VPI**: 该域在用户——网络接口处由8位组成, 在网络结点接口中由信头的前12位组成。VPI的主要功能是用于路由选择, 包含ATM路由地址的第一部分。
- **VCI**: VCI由16位组成。用于虚信道的路由选择, 包含ATM路由地址的第二部分。
- **PT**: 信息类型由3位组成, 用来区分信元中信息域的信息类, 即区分是否为用户信息信元。其中, 非用户信息信元包括运行维护和管理信元和资源管理信元。
- **CLP**: 用于表示信元丢失的优先级, 可由用户或业务提供者设置。当CLP为0时, 表示该信元具有较高的优先级别; 当CLP为1时, 该信元的优先级别较低。网络发生拥塞时, 将选择丢弃CLP为1的信元, 以缓解网络压力。
- **HEC**: 主要功能是用于ATM信元信头差错的检测(多个比特位)和纠正(单个比特位)以及信元定界。
- **服务类型**:
 - 实时服务, 对时延有严格要求
 - Constant Bit Rate (**CBR**): 恒定比特率业务。CBR业务在发送点与接收点之间提供持续的比特流。业务提供者确保在整个虚电路上向用户提供预定的固定比特的可用带宽。这种业务对于需要持续数字信息流的交互式数字语音和视频应用是非常理想的。常见的一些应用实例包括电话会议、电话业务、远端教学和付费电视。
 - Real-Time Variable Bit Rate (**rt-VBR**): 对时延有高要求的突发业务, 如分组化语音业务和多媒体检索
 - 非实时服务, 对时延没有约束
 - Non-Real-Time Variable Bit Rate (**nrt-VBR**): 对时延没有过高要求的突发业务, 如订票系统和银行交易业务
 - Available Bit Rate (**ABR**): 用流控机制动态调节流量提交的比特率

- Unspecified Bit Rate (**UBR**): 不保证用户的使用带宽, 用户只能使用尚未被占用的带宽。“尽力传输”。应用如: 电子邮件, 远程终端和文件传输
- Guaranteed Frame Rate (**GFR**): 能够保证最小速率并可以动态访问附加带宽

9. ICMP

- destination unreachable
- time exceeded
- parameter problem
- source quench
- redirect
- echo & echo reply
- timestamp & timestamp reply
- address mask request & reply

10. ARP/RARP

- ARP: 已知IP和port, 求mac地址。向整个以太网广播ARP报文。Request from source asking for hardware address. Then, reply from destination carrying hardware address
- RARP: 已知自己的mac地址, 向RARP server求自己的IP地址

11. Details about "Next Header"

- 每个**extension header**都包含一个next header part指向下一个header
- **Hop-by-hop options header**
- **Destination options header**
- **Fragment header** 8字节
 - fragmentation只能发生在source, 中间节点不能随便分片
 - 包含 fragment offset, more flag, identification
- **Generic routing header**
 - Contains list of one or more intermediate nodes to visit

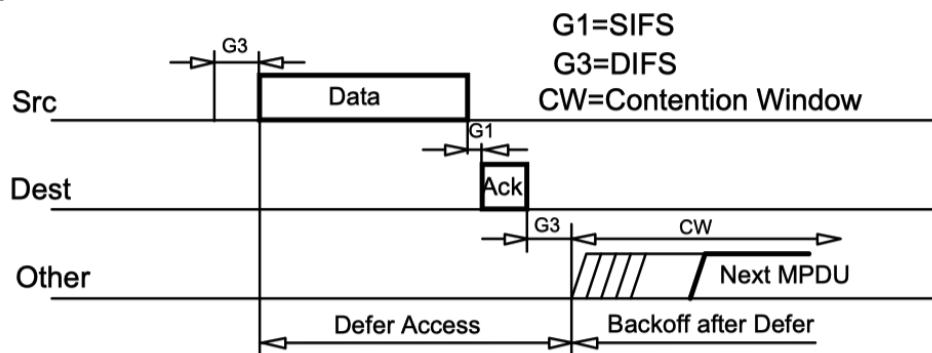
12. token bucket & leakage bucket

- 如果令牌桶中存在令牌, 则允许发送流量; 而如果令牌桶中不存在令牌, 则不允许发送流量。假如用户配置的平均发送速率为 r , 则每隔 $1/r$ 秒一个令牌被加入到桶中; 假设桶最多可以存发 b 个令牌。如果令牌到达时令牌桶已经满了, 那么这个令牌会被丢弃; 当一个 n 个字节的数据包到达时, 就从令牌桶中删除 n 个令牌, 并且数据包被发送到网络; 如果令牌桶中少于 n 个令牌, 那么不会删除令牌, 并且认为这个数据包在流量限制之外; 算法允许最长 b 个字节的突发, 但从长期运行结果看, 数据包的速率被限制成常量 r
- 漏桶算法思路很简单, 水(请求)先进入到漏桶里, 漏桶以一定的速度出水, 当水流入速度过大会直接溢出, 可以看出漏桶算法能强行限制数据的传输速率。
- 令牌桶算法不能与另外一种常见算法“漏桶算法(Leaky Bucket)”相混淆。这两种算法的主要区别在于“漏桶算法”能够强行限制数据的传输速率, 而“令牌桶算法”在能够限制数据的平均传输数据外, 还允许某种程度的突发传输。在“令牌桶算法”中, 只要令牌桶中存在令牌, 那么就允许突发地传输数据直到达到用户配置的门限, 因此它适合于具有突发特性的流量

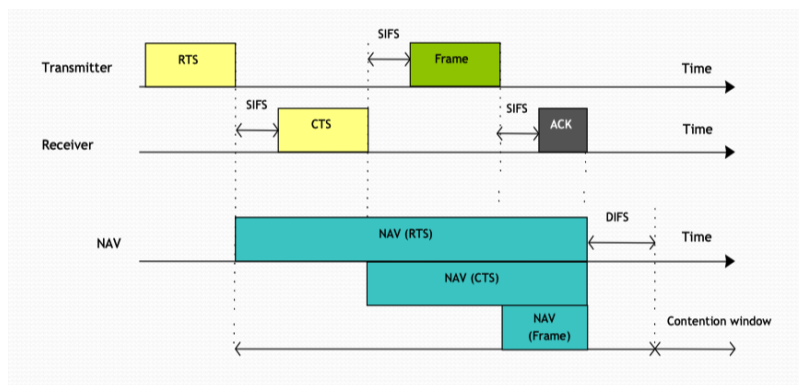
13. 介质访问控制办法

- 静态划分信道

- **FDMA**
- **TDMA**
- **CDMA**
- **WDMA**
- 动态分配信道
 - 轮询访问
 - **token ring**
 - 随机访问（所有用户随机发送信息，占据全部带宽）
 - **ALOHA** 不听就说
 - **CSMA** 先听再说
 - **1-persistent CSMA**
 - 监听到信道忙则持续监听，直到听到信道空闲
 - 监听到信道空闲立刻发送数据
 - 若有冲突则等待随机时间，继续监听
 - **0-persistent CSMA**
 - 监听到信道忙则退避随机时间，继续监听
 - 监听到信道空闲则立刻发送数据
 - **p-persistent CSMA**
 - 监听到信道忙着持续监听，直到听到信道空闲
 - 听到信道空闲，则p概率传输，1-p概率等到下一个time slot传输
 - **CSMA/CD** 先听后说，边说边听，冲突停止，随机再说
 - 以太网中使用
 - 这个CD就体现在边说（数据）边听（碰撞）
 - **CSMA/CA**



- 无线网中使用
- CA=碰撞避免
 - 因为无法监听信道：隐蔽站问题
 - 因为无法监听碰撞：信号强度动态范围变化大
- 帧间间隔
 - SIFS（短，高优先级帧）
 - DIFS（长，低优先级帧）
- 信道预约
 - **RTS/CTS**



简答（相似概念的对比理解和运用）

1. **TCP** v.s. **UDP**
2. **Service** v.s. **Protocol**
3. **distance vector(RIP)** v.s. **link state(OSPF)**
4. 各种有线无线**media**的性质对比
5. **circuit switching** v.s. **packet switching**
6. **datagram switching** v.s. **virtual circuit switching**
7. **ALOHA** v.s. **slot ALOHA** v.s. **CSMA**
8. **CSMA/CD** v.s. **CSMA/CA**
9. **Hub** v.s. **Bridge** v.s. **Switch** v.s. **Router**

计算

1. TCP的**flow control**计算
 - sliding window=min[cgwn, 对面发来的接收窗口]
2. TCP的**congestion control**计算
 - **slow start**: 2的指数次方递增
 - **linear increase**: 到达**sssthresh**后线性递增
 - **fast retransmission**: 连续收到4次针对同一segment的Ack时重传
 - **fast recovery**: 重传后**sssthresh**=cgwn/2, 从这里开始线性递增
 - **time out**: **sssthresh**=cgwn/2, cgwn减为1, 开始slow start
3. TCP的**RTO**计算
 - **RTT(k+1)**
 - **SRTT(k+1) = 7/8*SRTT(k) + 1/8*RTT(k+1)**
 - **SDEV:(k+1) = 3/4*SDEV(k) + 1/4*|RTT(k+1)-SRTT(k)|**
 - **RTO(k+1) = SRTT(k+1) + SDEV(k+1)**
 - **Karn算法**: 在计算平均往返时延RTT时, 只要报文段重传了, 就不采用其RTT。同时就将RTO翻倍

4. IP的网络号主机号应用

一个X类网络，分成多少个部门，每个部门有多少台机器，怎么划分subnet mask?

5. IP的分类编址和subnet mask

- **A类：1~127** subnet mask: 255.0.0.0
- **B类：128~191** subnet mask: 255.255.0.0
- **C类：192~223** subnet mask: 255.255.255.0
- subnet mask:

```
10000000->128
11000000->192
11100000->224
11110000->240
11111000->248
11111100->252
11111110->254
```

6. IP的分段：more flag, DF flag, fragment offset, total length

- total length计算的时候不要忘记加上段首的20B
- more flag=1: 还有分片 =0:后续无分片

7. Dijkstra算法和Bellman-Ford算法应用

8. CRC计算

9. 同步非同步的frame boundary

10. Hamming Distance -> 最大检错位数

11. 奇偶校验