计算机网络

Computer Network

复习资料

柳 洋

Spring 2021

目录

目录
一、概述
    1. 协议的作用？协议分层的优点？
    2. 五层协议的划分及各自功能及相关协议：
    3. 通过网络链路和交换机转移数据的方法
    4. IP地址、MAC地址、PORT地址在跨网段传递分组时各自的作用？
    5. 为什么实际带宽小于链路带宽（为什么网速慢）？
二、应用层
    1. HTTP连接
    2. 发送方和接收方都设置邮件服务器（用户代理）的原因？
    3. DNS层次化查询？
    4. DNS服务层次结构中各层之间的相互作用？DNS层次化的优点？
    5. 如何使用DNS查询尽快获取能够解析www.baidu.com的DNS服务器IP地址？
三、运输层
    1. TCP和UDP为应用层提供的服务
    2. TCP和UDP多路分解和多路复用的区别？
    3. 可靠数据传输协议rdt
    4. 流水线差错恢复方法GBN和SR的区别？
    5. TCP可靠数据传输的发送方和接收方各自的特点？接收方收到错误或者乱序的数据包如何处理？
    6. TCP如何保证可靠数据传输？
    7. TCP拥塞控制的机制
四、网络层
    1. 虚电路网络和数据报网络的区别
    2. IP地址划分的三种方式
    3. **路由分组转发算法**
    4. LS算法、DV算法求路由表（LS算法必须有拓扑生成、洪泛控制的过程）
    5. 因特网采取层次路由的原因是什么？AS是如何划分的？各包含什么常用的选路协议？这些协议的特点是什么？
    6. 层次OSPF的优点及各级路由器的作用？
    7. BGP路由选择
    8. DHCP+NAT协议解决学生上网问题
五、数据链路层
    1. 多路访问链路协议
    2. 以太网中CSMA/CD算法的实现过程
    3. **随机时间量的选择——二进制指数后退算法**
六、访问Google网页的全过程
七、其他

一、概述

1. 协议的作用？协议分层的优点？

协议作用：定义多个通信实体之间交换的报文**格式**和**次序**以及采取的**动作**。为互联网各层次进行数据交换而建立的**规则标准**。

协议分层的优点：层次分明、灵活性好；有利于构建通信标准；易于管理和维护。

2. 五层协议的划分及各自功能及相关协议：

- 应用层：为应用层程序和应用层协议驻留。涉及HTTP、DNS、SMTP、FTP等协议。
- 运输层：供不同主机的进程间的端到端通信。涉及TCP、UDP等协议。
- 网络层：负责跨越多个网络的数据包从源端到目的端的数据传送。涉及IP等协议。
- 数据链路层：提供一段链路两端节点的数据传递。涉及ARP等协议。
- 物理层：将比特流在物理信道从一个结点移动到另一个结点。

3. 通过网络链路和交换机转移数据的方法

	电路交换	**报文交换**	**分组交换**
定义	端到端系统之间建立专用链接，传输数据时独占信道资源。对信道资源划分进行多路复用的方式可分为时分多路复用、频分多路复用、波分多路复用、码分多址等。	端到端系统之间传输数据时，需要由上级系统一次性发送完成后，下级系统一次性接收。	需要进行报文拆分和重组，按需分配链路使用（包括数据报网络和虚电路网络）。
优点	传输时延小；数据传输保证较高正确性；物理链路一旦建立即可随时通信、**实时性强**	不存在连接建立时延，可以随时发送报文；存储转发机制可以提高传输可靠性	无需建立连接，加快了传输速度，适用于**猝发式传输**；按需分配链路，资源共享能力高
缺点	物理通路被通信双方独占，信道利用率低；链路发生故障会导致整体崩溃	不能保证缓存容量以支持用户一次性上传的数据；容错率极低，一旦数据包出错将会导致全部重传，导致信道资源浪费	存在分组丢失和时延，需要可靠的传输控制和拥塞控制

4. IP地址、MAC地址、PORT地址在跨网段传递分组时各自的作用？

1. IP地址定义在网络层并存放在IP数据报头部。用于识别网络中互联的主机和路由器；根据子网掩码确认IP地址所在网段；利用IP地址进行路由选择。
2. MAC地址定义在数据链路层并存放在帧头部。用于识别同一链路中的不同主机，并且在同一网段下传递分组。
3. PORT地址定义在传输层并存放在TCP/UDP头部。用于识别主机上不同的应用程序，进行主机间的进程寻址。

5. 为什么实际带宽小于链路带宽（为什么网速慢）？

1. 网络协议（CSMA/CD、TCP可靠数据传输）限制了物理带宽的使用。
2. 信道利用率低。
3. 分组时延和丢失
    - 传输时延：发送数据时，将分组推入到信道中的时间。

- 传播时延：分组在信道中的传播所需要的时间。
- 排队时延：结点缓存队列中分组排队所经历的时延。最需要考虑的时延。
- 处理时延：检查分组首部和决定将改分组导向何处所需要的时间。

4. 流量控制：发送方速率应匹配接收方应用进程的接收速率，从而抑制了发送方的发送速率。
5. 拥塞控制：数据报长度应小于拥塞窗口，也应小于流量窗口，当出现拥塞时，发送方应减小发送速率。

提高网速的方法：增加网络带宽、使用信道利用率高的数据传输方法、安装缓存器等。

二、应用层

1. HTTP连接

例：请求一个含有10个jpeg图片html文件的web页面。

1. 非持久连接的串行工作方式：$2*RTT+10*2*RTT=22RTT$
2. 非持久连接的并行工作方式：$2*RTT+\lceil 10/3\rceil *2*RTT=10RTT$（假设同时最多可有3个并行的TCP连接）
3. 持久连接的不带流水的工作方式：$2*RTT+10*RTT=12RTT$
4. 持久连接的带流水的工作方式：$2*RTT+1*RTT=3RTT$

2. 发送方和接收方都设置邮件服务器（用户代理）的原因?

对于接受方而言，为了能使其及时接收在任何时候到达的新邮件，他的PC必须总是不间断的运行着并一直保持在线，这对许多因特网用户而言是不现实的，因此需要设置接收方邮件服务器来缓存收到的邮件，在用户方便时可以调用用户代理阅读邮件。

对于发送方而言，若接收方邮件服务器未打开，发送方将无法成功发送该邮件，只能通过一次次尝试发送，无疑是耗时耗力的，因此需要设置发送方邮件服务器定时对未成功发送的邮件再次尝试发送直至发送成功为止。

3. DNS层次化查询?

例如查询www.baidu.com的IP地址

1. 在根DNS服务器使用DNS层次化查询
    1. 到根DNS服务器上查询能够解析com的顶级域DNS服务器

       nslookup -qt=ns com 198.41.0.4 => 得到一个顶级域DNS服务器IP 192.5.6.30
    2. 到顶级域DNS服务器上查询能够解析baidu.com的权威DNS服务器

       nslookup -qt=ns baidu.com 192.5.6.30 => 得到一个权威DNS服务器IP 110.242.68.134
    3. 到权威DNS服务器上查询www.baidu.com的IP

       nslookup -qt=a www.baidu.com 110.242.68.134 => 得到该网站的IP地址39.156.66.14
2. 在本地DNS服务器使用DNS层次化查询

   使用nslookup命令查询时，不指定查询服务器，则默认向本地名称服务器查询。或者通过命令ipconfig/all查询本地DNS服务器的IP地址替换上述过程中的根DNS服务器。

4. DNS服务层次结构中各层之间的相互作用？DNS层次化的优点?

作用：为解析某域名或其前缀提供下一层DNS服务器域名和IP地址。

优点：分散用户访问，降低负载；便于分区域分层次管理，提升访问速度；减轻发生故障的影响。

5. 如何使用DNS查询尽快获取能够解析www.baidu.com的DNS服务器IP地址?

首先主机向本地DNS服务器发送请求，发现本地DNS服务器并没有该网址解析，然后查看是否缓存了baidu.com的权威DNS服务器。

1. 若有该缓存直接向该权威DNS服务器发送请求，获取www.baidu.com的IP地址，然后回复主机该IP地址；
2. 若没有该缓存，查看是否缓存了com 顶级域DNS服务器。
    1. 若缓存了com顶级域DNS服务器，则向该顶级域DNS服务器查询权威DNS服务器的IP地址；
    2. 若没有com 顶级域名服务器的缓存记录，则向根DNS服务器获取com顶级域DNS服务器的地址，然后再进行后续查询。

三、运输层

1. TCP和UDP为应用层提供的服务

1. TCP
    - 面向连接服务：客户端和服务器进程之间需要建立通信，即通信前的三次握手确认机制
    - 提供：**可靠数据传输、流量控制、拥塞控制**
    - 不提供：实时性，最小链路带宽保证（因为和流量控制和拥塞控制情况有关）
    - 适用于对数据传输要求完全可靠的请求，即要求传输的数据分组不丢失、不重复、不乱序
2. UDP
    - 无连接服务：客户端和服务器进程之间无需建立通信即可进行数据传输
    - 不提供：可靠数据传输、流量控制、拥塞控制、瞬时实时性、最小链路带宽保证
    - 适用于应对瞬时的大量请求，和大量数据传输（因为 UDP 协议要求少， 建立通信简单，能够非常快的进行传输）

2. TCP和UDP多路分解和多路复用的区别?

一个 UDP 套接字是由一个二元组（目的 IP，目的端口号）唯一标识，因此具有相同目的IP和目的端口号的UDP报文段，会通过相同的目的套接字定向到相同的目的进程。

TCP 套接字是由一个四元组（源 IP 地址，源端口号，目的 IP 地址，目的端口号）唯一标识，因为不同TCP报文段请求同一主机同一应用程序时，目的IP和目的端口号均为相同的“欢迎套接字”的IP和端口号，因此需要四元组才能区分。

3. 可靠数据传输协议rdt

1. rdt1.0：经完全可靠信道的可靠数据传输。
    - 假设数据单向传输，并且底层信道完全可靠。
2. rdt2.0：经具有比特差错信道的可靠数据传输。
    - 假设数据单向传输，但是在底层信道中分组的比特可能会出错。
    - 接收方进行差错检测，根据是否正确接收分组向发送方返回确认信息ACK或者NAK。若接收方收到NAK时需要重传当前分组（要求发送方有数据包缓存机制）。
3. rdt2.1：考虑ACK/NAK差错
    - 假设数据单向传输，但是在底层信道中，不仅分组的比特可能会出错，确认信息ACK/NAK比特也会出错。
    - 在数据分组增加新字段“序号”，取值为0/1。例如接收方收到0号坏包返回NAK0，正确接收到0号包返回ACK0，收到冗余0号包返回ACK0；发送方收到NAK或者坏包会重发最近发出的包。
4. rdt2.2：取消接收方的NAK回复
    - 假设数据单向传输，但是在底层信道中，不仅分组的比特可能会出错，确认信息ACK/NAK比特也会出错，但只需传ACK即可完成数据包确认。

- 取消接收方的NAK回复，对上次正确接受的分组发送冗余ACK暗示接收方需要重传当前包。例如接收方收到0号坏包返回ACK1，发送方发出0号包收到ACK1或坏包则需重传0号包，接收方接收到0号包返回ACK0又接收到0号包则返回ACK0。

5. rdt3.0：经具有比特差错的丢包信道的可靠数据传输
   - 在rdt2.2假设的基础上，假设在信道中分组或者ACK确认包会发生丢失。
   - 发送方设置计时器对已发出但是在规定时间内没有收到ACK的包进行重传。

注：rdt协议是典型的**停等协议**，即当发送方处于等待应答状态时，不能从上层接收更多数据。因此rdt协议具有很低的信道利用率。解决方案是允许发送方发送多个分组而无需等待确认。

4. 流水线差错恢复方法GBN和SR的区别?

GBN和SR都属于滑动窗口协议。

	出错全部重传GBN	**选择重传SR**
定义	GBN允许发送方发送多个分组（当有多个分组可用时）而无需等待确认，但是也受限于流水线中未确认分组数不能超过某个最大允许数N。	SR通过让发送方仅重传那些丢失或出错的分组而避免了不必要的重传。
发送方：**计时器**	设置**单一计时器**，对最早的已发送但是未被确认的分组进行计时。	**为每一个分组设置计时器。**
发送方：分组**超时**处理	若计时器超时，则将所有已发送但未被确认的分组进行**全部重传**（序号位于send_base与nextseqnum-1之间的分组）。	若某分组计时器超时，则只**选择重传**引起计时器超时的那个分组。
接收方：**缓存**	接收方**不设置缓存**，只记录下一个按序到达的分组的序号，遇到乱序或者出错分组直接丢弃并发送冗余ACK。	接收方**设置缓存**，缓存乱序到达的分组，并且在缺失的分组到达并且使得缓存分组有序后，一起递交上层。
接收方：**应答**	采用**累积应答**，即对于按序到达并成功接收的最大序号的分组进行肯定应答，以表示接收方已经按序接收到该分组及之前的分组。若出现乱序或出错分组会产生对最后一次按序接收的分组的应答（冗余ACK）。	对每一个分组进行**逐一应答**，对冗余数据包会进行再次应答。若出现出错分组或丢失分组则会引起计时器超时等待重发。
特点	未正确传输分组会产生大量冗余ACK以及数据包；接收方不设置缓存。	避免了不必要的重传；发送方窗口长度应小于等于序号空间的一半。

5. TCP可靠数据传输的发送方和接收方各自的特点？接收方收到错误或者乱序的数据包如何处理?

1. 发送方：
   1. 使用**单一计时器**，只为最早发送还没有得到ACK响应的分组计时，超时后重发该分组。
   2. 发送方设置发送缓存、多个发送方窗口
   3. 当收到三个冗余ACK之后会**快速重传**该分组（快速重传机制：在计时器超时之前就重传该分组）
   4. 维护接受窗口来进行流量控制
   5. 可以感知网络的拥塞程度来限制发送速率

2. 接收方：
    1. 接收方设置接收缓存、多个接收方窗口
    2. 采用**累积确认**，保证确认号之前的分组已经正确收到。
    3. 收到错误的数据包则对最近一次已经确认过的分组进行再次确认
    4. 收到乱序的数据包则根据上层应用的设置一般是进行缓存，当完整的接收到按序的分组后再向上递交。

6. TCP如何保证可靠数据传输?

1. **校验和**：在TCP报文首部封装一个校验和，接收方通过检测校验和判断分组是否出现差错，若出现差错则丢弃该分组并要求发送方重传该分组。
2. **序列号和累积确认**：发送方按顺序给每个要发送的数据包编号，接收方在接收到后进行累计确认，接收方通过ACK应答数据包决定发送新数据包还是重发。
3. **超时重传**：发送方在等待接收方回传ACK应答数据包超时后，重传该数据包并将超时间隔加倍。
4. **滑动窗口**：按照一定的窗口大小传输分组，窗口始终滑动到最先发送但尚未被确认的分组。
5. **流量控制**：TCP首部有一个接收窗口字段，接收方通过设置该字段来主动控制传输流量。
6. **拥塞控制**：发送方通过设置拥塞窗口大小主动控制传输流量。包括慢启动、拥塞避免和快速恢复机制。

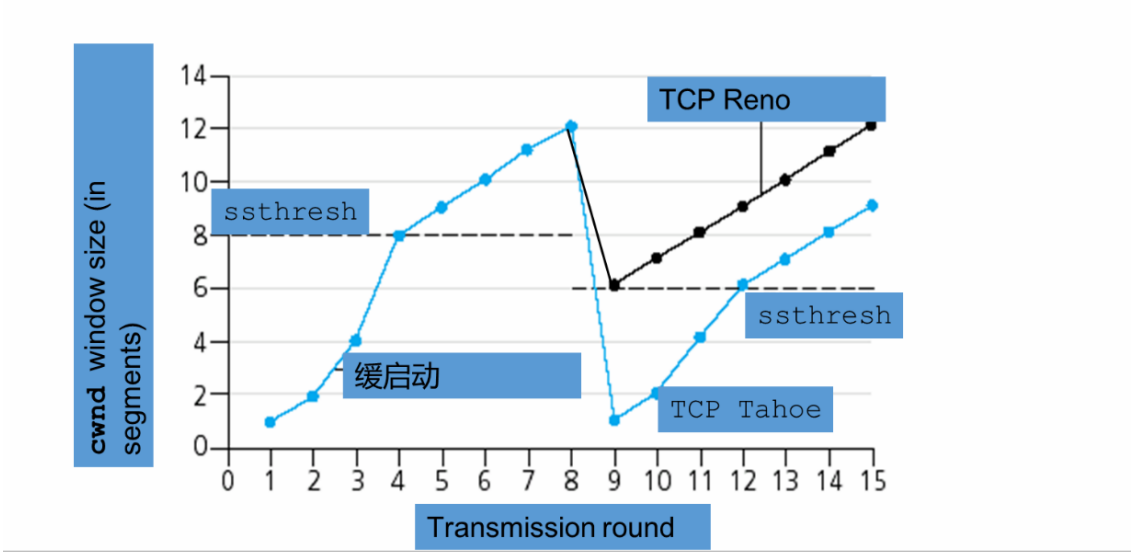
7. TCP拥塞控制的机制

（cwnd是拥塞窗口，ssthresh是慢启动阈值）

1. **慢启动**：cwnd的值以1个MSS开始，并且每当传输的报文段首次被确认就增加一个MSS，因此TCP起始发送速率慢，但是随着RTT增加呈指数增长。
2. **拥塞避免**：当cwnd增长至ssthresh时，进入拥塞避免状态，即每个RTT只将cwnd的值增加一个MSS，进入线性增长的探测模式。
3. **快速恢复**：

    Reno机制：当接收到3个冗余的ACK时，ssthresh设置为cwnd的一半，并从ssthresh开始进行拥塞避免的线性增长阶段。当发生超时事件时，ssthresh设置为cwnd的一半，并将cwnd设置为1个MSS进入慢启动阶段。

    Tahoe机制：当接收到3个冗余ACK或发生超时事件时，ssthresh设置为cwnd的一半，并将cwnd设置为1个MSS进入慢启动阶段。



四、网络层

1. 虚电路网络和数据报网络的区别

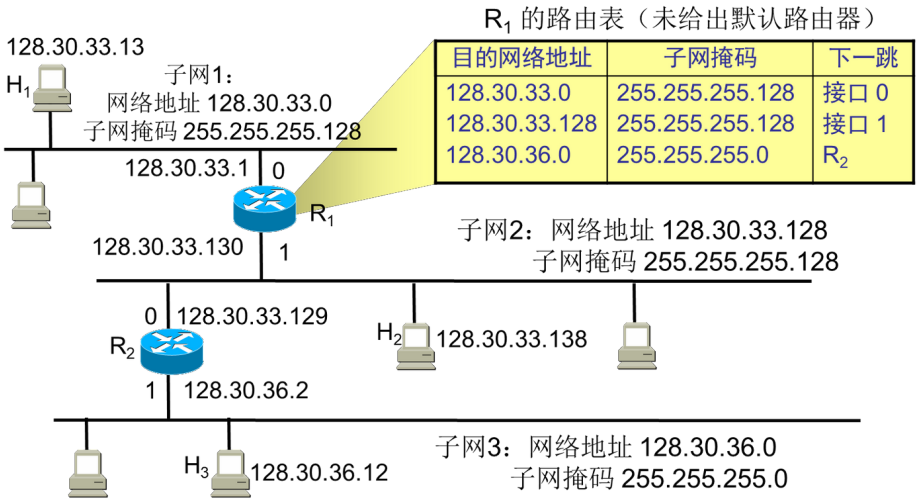
1. 传输方式：在源主机与目的主机通信前，虚电路网络需要先建立虚电路连接；而数据报网络发送信息报文，不需要建立连接。
2. 分组到达的顺序：虚电路网络分组按序到达；数据报网络分组可能乱序到达。
3. 路由转发：虚电路网络分组根据虚电路表示选择转发路径；数据报网络根据报文携带的目的IP地址进行路由选择。
4. 结点故障：虚电路网络中通过该故障结点的分组无法到达目的地；数据报网络中分组可以通过其他路由器到达目的地。
5. 可靠性与适应性：虚电路网络可靠性高；数据报网络适应性强。

2. IP地址划分的三种方式

1. **IP分类编址**：
    1. IP地址格式：{<网络号>，<主机号>}
    2. 不同的网络号和主机号的设置决定了IP地址的分类，包括A、B、C、D、E五类IP地址。
    3. 优缺点：根据不同需求划分IP地址类别，但是分类不够灵活，会造成IP地址的浪费。
2. **域间子网划分**：
    1. IP地址格式：{<网络号>，<子网号>，<主机号>}
    2. 从主机号中借用几位作为子网号，对同一网络下进行子网划分
    3. 优缺点：灵活性较高，但是存在IP地址浪费
3. **无类别域间路由选择(CIDR)**：
    1. IP地址格式：{<网络前缀>，<主机号>}
    2. CIDR 将子网寻址的概念一般化。形式为$a.b.c.d/x$的地址的 x 最高比特构成了 IP 地址的网络部分，并且经常被称为该地址的前缀。
    3. 优缺点：子网划分更加灵活，但是存在IP地址浪费

3. 路由分组转发算法

已知互联网和路由器 R1 中的路由表。主机 H1:128.30.33.13 向 H2:128.30.33.138 发送分组。试讨论 R1 收到 H1 向 H2 发送的分组后查找路由表的过程。



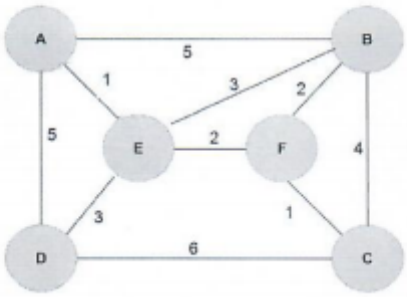
- H1 首先检查目的主机 128.30.33.138 是否连接在本网络上：
    - 将本机 IP 128.30.33.13 和目的 IP 128.30.33.138 分别和本网段的子网掩码 255.255.255.128 按位与，得出两个 IP 的网络地址为：128.30.33.0 和 128.30.33.128，说明目的主机不在该网段，需要路由器转发；
    - 若计算后得出的网络地址一致，则可以直接点对点交付；
- 数据报经主机 H1 中 ARP 表中存储的默认网络 MAC 地址，投递到网关路由器 R1：
    - 若 H1 未知 R1 的 MAC，则应该使用 ARP 广播包的情况进行查询；

- 路由器 R1 接收到数据报后，获取**目的 IP** ，将其**与**R1 路由表中的所有子网掩码相与，得到的所有网络地址按照**最长前缀匹配原则**在路由表中进行匹配，查询该数据报对应的路由器出口；

- 数据报经路由器 R1 的接口 1 发出，判断接口 1 所处的子网是否为目的子网：
  - 将端口 1 的 IP 128.30.33.130 和目的 IP 128.30.33.138 分别和端口1 的子网掩码 255.255.255.128 按位与，得出两个 IP 的网络地址为：128.30.33.128 和 128.30.33.128，说明目的主机在本网段中，可以直接点对点递交；
  - 若计算后得出的网络地址不一致，则继续发送到下一跳路由器进行转发；
- 路由器 R1 向主机 H2 点对点递交的时候，需要知道 H2 的 MAC 地址：
  - 若未知，则应使用 ARP 广播查询；
  - 若已知，封装 IP 数据报的源 MAC 为端口 1 的MAC 地址，目的 MAC为主机 H2 的 MAC 地址；

4. LS算法、DV算法求路由表（LS算法必须有拓扑生成、洪泛控制的过程）

4、请用分别用 LS 及 DV 算法求 B 节点的路由表（LS 算法必须有拓扑生成、防止洪泛的过程；DV 算法中 B 节点必须有邻接节点的距离向量表）（20 分）



LS：

N	D(A)P(A)	D(C)P(C)	D(D)P(D)	D(E)P(E)	D(F)P(F)

B 节点路由表：

目的	路由
A	
B	
C	
D	
E	
F	

DV:

目的	A 的向量表	C 的向量表	E 的向量表	F 的向量表
邻接距离				

B 的向量表

目的	A 的向量表	C 的向量表	E 的向量表	F 的向量表
A				
C				
D				
E				
F				

B 节点路由表

目的	路由
A	
C	
D	
E	
F	

B 节点广播的向量表

目的	跳步
A	
C	
D	
E	
F	

对于LS算法：

1. 拓扑生成过程：
    1. **发现邻居结点**，并学习它们的网络地址；（发送HELLO包发现邻居节点）
    2. **测量**到每个邻居结点的延迟或开销；(一种直接的方法是：发送一个要对方立即响应的ECHO包，来回时间除以2即为延迟。)
    3. 每个节点都向网络中其他所有节点**广播链路状态分组**，链路状态分组包含它所连接的链路的特征和链路代价值，这样每个节点就拥有了网络中所有其他节点的链路状态信息。
2. 洪泛控制过程：在扩散信息的过程中，进行洪泛控制主要有三种方法
    1. 为每个广播包包头设置一个**站点计数器**，每经过一个路由器则计数器减1，减为0时则丢弃该包。（仅能限制广播包能走多远，但是不能保证重复发送）
    2. 记录广播包的**转发路径。**某路由器结点接收到来自另一个路由器标号为K的广播包，则记录下广播包来源路由器与标号，当再次接收到相同的广播包时则丢弃。
    3. 使用**序号控制机制**，在每个路由器中记录一个最大序号，若到来的广播包比最大序号小，说明已转发过该包，并丢弃该包；若比最大序号大，则转发，并修改最大序号的值。

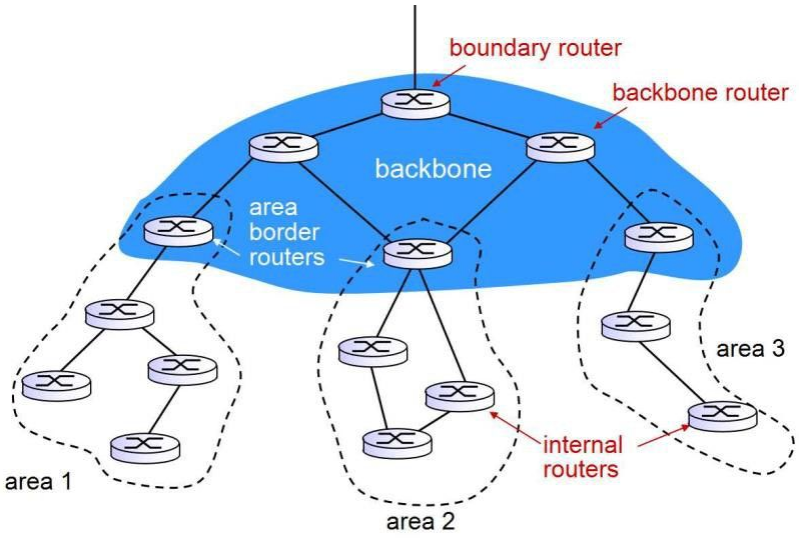
5. 因特网采取层次路由的原因是什么？AS是如何划分的？各包含什么常用的选路协议？这些协议的特点是什么？

1. 因特网采用层次路由的原因：首先是**规模**，随着路由器数目变得很大，涉及路由选择算法的计算、存储及通信的开销将高得不可实现；其次是**管理自治**，一个组织应当能够按照自己的愿望运行和管理网络，并且还能够和外部网络连接。
2. AS是根据相同的管理与技术控制进行的划分，同一个AS内具有相同的路由选择协议，不同的AS的路由选择协议不必相同，AS间的路由选择协议必须相同。
3. 常用的选路协议：
    1. AS内部路由选择协议：
        - RIP（路由选择信息协议）：是一种距离向量协议，采用DV算法，为信息在自治系统内的转发提供路由选择表。其健壮性较差。
        - OSPF（开放路径最短优先协议）：一个使用洪泛链路状态信息的链路状态协议和一个 Dijkstra 最低费用路径算法。安全性高，具有按层次结构构造一个自治系统的能力。
    2. AS间路由选择协议：
        - BGP（边界网关协议）：规定从相邻AS处获得子网的可达性信息，向AS内部所有的路由器传播这些可达性信息。

协议	RIP	OSPF	BGP	
类型	内部	内部	外部	
路由算法	距离—向量	链路状态	路径-向量	
传递协议	UDP	IP	TCP	
路径选择	跳数最少	代价最低	较好，非最佳	
交换结点	和本结点相邻的路由器	网络中的所有路由器	和本结点相邻的路由器	
交换内容	当前本路由器知道的全部信息，即自己的路由表	与本路由器相邻的所有路由器的链路状态	首次	整个路由表
			非首次	有变化的部分

6. 层次OSPF的优点及各级路由器的作用?

1. 层次OSPF优点：减少需要传递的路由信息数量，降低开销控制，缩短信息在自治系统内的传递时间。
2. 各级路由器的作用：
    1. 网关路由器：为传递到AS外部的分组进行路由转发
    2. 骨干路由器：为AS内部区域之间的分组转发提供路由选择
    3. 区域边界路由器：为传递到区域外的分组进行路由转发
    4. 区域内路由器：为区域内的分组进行路由转发



7. BGP路由选择

路由器根据 AS 内路由选择和 AS 间路由选择协议，可能知道到达任何一条前缀的多条路由，在这种情况下路由器必须在可能的路由中选择一条，BGP 顺序地调用下列规则，直到留下一条路由：

- 在所有路由中，优先选择具有最高本地偏好值的路由，通常由 AS 的网络管理员设定（包括默认路由、特定主机路由）。
- 在余下的路由中，选择具有最短 AS-PATH的路由。BGP 将使用一种距离向量算法来决定路径，其中距离测度使用 AS 跳的数目而不是路由器跳的数目；

- 在余下的路由中，将选择具有最靠近 NEXT-HOP 路由器的路由（热土豆路由）；
- 如果仍留下多条路由，该路由器使用 BGP 标识符来选择路由；

8. DHCP+NAT协议解决学生上网问题

以我校校园网为例（一个教育网ip为A，三个外网IP为B、C、D）
首先明确校园网的默认网关为A。

该名同学要连接校园网，首先发送一个DHCP DISCOVER请求，源ip为全0，目的ip全1，然后学校的DHCP服务器收到后发送广播包，提供给该主机分配的ip（内网ip），子网掩码，默认网关，本地DNS服务器的地址，可使用时间等信息。该同学可能收到多个DHCP OFFER回应，可以从这写OFFER中选择一个，向选中的服务器提供DHCP请求报文，该选中的DHCP服务器发送ACK对请求报文进行响应，证实参数。

当请求三个外网ip时，到达默认网关后，该路由器将A代替源ip(内网ip)，分配一个未使用的源端口号代替原源端口号。并将这条记录放到NAT转换表中；当外界发送响应信息到该路由器后，提取出目的ip和目的端口号，通过查看NAT表，找到该目的ip和目的端口号对应的内网ip和端口，然后用内网的ip和端口代替原目的ip和原目的端口，然后发至目的主机。实现通信过程。

五、数据链路层

1. 多路访问链路协议

1. 信道划分协议
    1. 时分多路复用：将时间划分为时间帧，每个时间帧划分为N个时隙，并且分别分配给N个结点。
    2. 频分多路复用：将信道划分为N个频段，并把N个频段分配给N个结点。
    3. 波分多路复用：将信道划分为N个波段，并把N个波段分配给N个结点。
    4. 码分多址：对每个结点分配一种不同的编码，每个结点用它唯一的编码来对它发送的数据进行编码。
2. 随机接入协议
    1. 纯ALOHA：当一帧首次到达，节点立即将该帧完整传输进广播信道。传输过程中若发生碰撞，则在完全传输完该帧后以概率P立即重传该帧。
    2. 时隙ALOHA：当一帧首次到达，节点等到下一个时隙开始传输该帧。传输过程中若发生碰撞，则在在后续的每个时隙中以概率P重传该帧直到不发生碰撞。
    3. 载波监听多路访问（CSMA）：一个结点在传输前先监听信道。如果来自另一个结点的帧正向信道上发送，结点则等待直到检测到一小段时间没有传输，然后开始传输。
    4. 具有碰撞检测的载波监听多路访问（CSMA/CD）：传输介质的传输时延导致了载波监听节点监听网络传输的延迟。当某结点执行碰撞检测时，一旦它检测到碰撞将立即停止传输。在有限局域网中监听信道中信号强度即可实现。
    5. 具有碰撞避免的载波监听多路访问（CSMA/CA）：多用于无线网。
3. 轮流协议
    1. 轮询协议：主节点以循环的方式通知每个结点能够传输帧的最多数量，并决定结点何时完成了帧的发送。
    2. 令牌传递协议：没有主节点，节点之间以某种固定的次序进行交换令牌，持有令牌的结点可以发送帧。

2. 以太网中CSMA/CD算法的实现过程

1. 适配器从网络层获得一条数据报，准备链路层帧，并将其放入帧适配器缓存中；
2. 如果适配器监听到信道空闲（即无信号能量从信道进入适配器），它开始传输帧。如果适配器侦听到信道正在忙，它将等待，直到监听到没有信号能量时才开始传输帧；
3. 在传输过程中，适配器监视是否存在来自其他使用该广播信道的适配器信号能量；

4. 如果适配器传输整个帧而未检测到来自其他适配器的信号能量，则该适配器就完成了该帧。如果适配器在传输时检测到来自其他适配器的信号能量，则中止传输。中止传输后，适配器通过二进制指数后退算法得到一个随机等待时间，然后返回步骤 2。

3. 随机时间量的选择——二进制指数后退算法

- 总体思路：

  发生碰撞次数少时，时间间隔较短；发生碰撞次数多时，时间间隔较长。

- 具体算法：

  当传输一个给定帧时，在该帧经历了一连串的 n 次（n<10）碰撞后，结点随机地从{$0, 1, 2, \ldots, 2^n - 1$}中选择一个 K 值，适配器等待512*K个比特时间。（因此，一个帧经历的碰撞越多，K 可能选择的值就越大。对于以太网，一个结点等待的实际时间量是$K \cdot$ 512比特时间，即发送 512 比特进入以太网所需时间量的 K 倍。因为每个节点时延选择的随机性，重发的帧在时间上会大概率错开。）

六、访问Google网页的全过程

1. **获取本主机的动态IP地址**

   本主机需要通过DHCP协议**获取在内网中的动态IP地址**，共分为四个步骤。

   - **DHCP发现**：本主机向DHCP服务器发送 DHCP 发现报文，将其封装到 UDP 报文段并进一步封装到IP数据报中进行向所在网络发送广播包，源主机IP 是0.0.0.0，目的IP 是255.255.255.255，源 MAC 地址是主机的MAC 地址，目的 MAC 地址是 FF-FF-FF-FF-FF-FF。
   - **DHCP提供**：DHCP 服务器用一个 DHCP 提供报文作出响应，为客户端提供若干可用的动态 IP。报文的目的IP地址为255.255.255.255。
   - **DHCP请求**：本主机从DHCP提供报文中选择一个动态IP并且返回一个DHCP请求报文。
   - **DHCP ACK**：DHCP 服务器向客户端发送DHCP ACK报文，确定客户端的动态IP地址。

   经过以上步骤客户端同时还获得了**DNS服务器IP、子网掩码、第一跳路由器IP**。

2. **获取DNS服务器的MAC地址**

   首先判断本主机与DNS服务器是否在相同网段，将本地主机的 IP 地址和子网掩码相与，再将 DNS 服务器的 IP 地址和子网掩码相与，比较结果。

   1. 若结果相同，则说明在同一个网段中。本地主机通过发送 **ARP 广播包**获取DNS服务器的 MAC 地址，源 IP 为本地主机 IP，目的 IP 为 DNS 服务器 IP， 源 MAC 地址为本地主机 MAC 地址，目的 MAC 地址为 FF-FF-FF-FF-FF-FF。DNS 服务器收到 ARP 广播包后，发现目的 IP 是自己，即向本地主机返回其 MAC 地址，本地主机收到后，将记录加入到ARP表中，并向 DNS 服务器发送查询报文。
   2. 若结果不同， 则说明不在一个网段中，需要借助网关跨网段发送请求。本地主机通过发送 ARP 广播包获取第一跳路由器接收端口的 MAC 地址。接下来本主机发送ARP广播包获得DNS服务器的MAC地址，源 IP 为本地主机 IP，目的 IP 为 DNS 服务器 IP， 源 MAC 地址为本地主机 MAC 地址，目的 MAC 地址为第一跳路由器接收端口MAC地址。到达路由器后，需要将DNS服务器的IP地址和路由器路由表的所有转发端口所在的子网掩码依次相与，按照**最长前缀匹配原则**选择转发端口。接下来将转发端口IP地址与转发端口所在子网的子网掩码相与，再将DNS服务器IP地址和子网掩码相与，若结果相同，则在同一个网段，若不同，则说明不在同一个网段，重复以上过程，直到找到 DNS 服务器的 MAC 地址，本地主机收到后，将记录加入到ARP表中，并向 DNS 服务器发送查询报文。

3. **路由器决定报文的转发路径**

   在上一步中，第一跳路由器路由表按照以下顺序配置：选择具有最高本地偏好值的路由，根据**AS间路由选择协议**选择具有最短 AS-PATH的路由，根据**热土豆路由**选择具有最靠近 NEXT-HOP 路由器的路由，根据附加策略选择路由。

4. **获取Google网页的IP地址**

   本地主机向 DNS 服务器发送查询报文进行**DNS层次查询**，查找 Google 的 IP 地址。首先从根名称服务器查找顶级域名称服务器的 IP 地址，再从顶级域名称服务器中查找权威名称服务器的 IP 地址，最后从权威名称服务器中查找到 Google 域名和 IP 的对应记录，即向本地主机返回 Google 的 IP 地址。

5. **外网访问**

   获取到谷歌服务器IP地址后，还需要判断本主机IP和谷歌服务器IP是否在同一个网段当中。将本主机的 IP 地址和子网掩码相与，再将Google服务器的 IP 地址和子网掩码相与，比较结果。若结果相同，则可以直接点对点发送请求；若结果不同还需要借助路由器跨网段发送请求。由于本主机IP为动态IP，需要通过**NAT协议**转换为真实IP。则本主机发送的请求报文的源IP为该真实IP，目的IP为Google服务器IP。

6. **获取Google网页**

   本主机和Google服务器通过**三次握手**建立TCP连接，即本地主机发送SYN报文请求TCP连接，服务器发送SYNACK报文确认连接，本地主机发送ACK报文确认SYNACK报文，并且携带HTTP请求。

   本地主机向**套接字**中发送 **HTTP GET 报文**，报文通过 TCP 传输，Google 服务器从其套接字中接收报文，并向本地主机套接字发送一个 HTTP 响应报文，本地主机接收到 HTTP 响应报文，从中提取出网页信息，则从 Google 服务器上接收到了Web 页面。若在此期间TCP传输过程中发生分组丢失或出错问题还会触发**TCP可靠数据传输**的相关机制。

七、其他

1. SMTP发送邮件的过程
   - 发送方调用电子邮件代理程序并提供接收方的邮箱地址，撰写报文，然后指示用户代理发送该报文。
   - 发送方的邮件代理把报文发给他的邮件服务器，在那里该报文被放在报文队列中。
   - 运行在发送方的邮件服务器上的SMTP客户端发现了报文队列中的这个报文，他就要创建一个到运行在接受方邮件服务器上的SMTP服务器的TCP连接。
   - 在经过一些初始SMTP握手后，SMTP客户通过该TCP连接发送到报文
   - 在接收方邮件服务器上，SMTP的服务器端接收该报文。接收方的邮件服务器将该报文放入接收方的邮箱中。
   - 接收方在方便地时候，可以调用用户代理阅读该报文。
2. DNS记录的创建

   向注册登记机构注册域名networkutopia.com时，需要向其提供权威DNS服务器名称和IP地址。即向其提供一条NS记录与一条A记录。

   ```
   (networkutopia.com, dns1.networkutopia.com, NS)
   (dns1.networkutopia.com, 212.212.212.1, A)
   ```

3. TCP套接字与UDP套接字的区别?
   1. TCP服务器必须在客户端试图发起请求之前运行起来，而UDP服务器不需要。

2. TCP服务器程序必须创建欢迎套接字，并为每个请求创建一个连接套接字，而UDP只需为一个用户提供一个套接字。
3. TCP客户端在发送数据前需要先建立连接，而UDP不需要
4. TCP客户端套接字指定了服务器中创建的欢迎套接字的IP地址和端口号，并没有显式地创建分组并未分组附上目的IP地址和端口号，而UDP客户端明确地将目的IP地址和端口号附加到每个数据包上。

4. 拥塞控制和流量控制的区别及联系？

   区别：流量控制是为了让发送方发送速率与接收方接收速率进行匹配，避免接收方缓存溢出的情况。拥塞控制是针对整个网络而言，防止大量分组造成网络拥塞而抑制发送方的发送速率。

   联系：拥塞控制和流量控制协同工作，让发送方未被确认的数据量不超过cwnd（拥塞窗口）和rwnd（接收窗口）的最小值。

5. TCP三次握手

   1. 客户端TCP -> 服务器TCP：TCP报文段中SYN=1, seq=client_isn
   2. 服务器TCP -> 客户端TCP：TCP报文段中SYN=1, ACK=1, seq=server_isn, ACK=client_isn+1
   3. 客户端TCP -> 服务器TCP：TCP报文段中SYN=0, ACK=1, seq=client_isn+1, ACK=server_isn+1

6. 传输层连接与网络层连接对比：

   1. 传输层连接：**由传输层向应用层提供的进程到进程的服务。**
   2. 网络层连接：**由网络层向传输层提供的主机到主机的服务。**

7. 路由分组转发算法原理

   1. 从数据报的首部

      提取目的主机的 IP 地址 D，根据路由器中存储的 IP 地址对应的子网掩码和 D 进行按位与，得出目的网络地址为 N。可以在此做分组过滤，对一些特定的目的主机 IP 指定路由黑洞或直接销毁该访问不安全 IP 的分组；

   2. 若网络 N 与此路由器直接相连，则把数据报直接交付目的主机 D；否则是间接交付，执行(3)。查看是否是到达了目的网络 N 的最终的目的路由器；

   3. 若路由表中有目的地址为 D 的特定主机路由，则把数据报传送给路由表中所指明的下一跳路由器；否则，执行(4)。查看该路由器的系统管理员有没有指定特定主机路由；

   4. 若路由表中有到达网络 N 的路由，则把数据报传送给路由表指明的下一跳路由器；否则，执行(5)。若没有指定特定的主机路由，则使用路由器转发算法算出的下一跳；

   5. 若路由表中有一个默认路由，则把数据报传送给路由表中所指明的默认路由器；否则，执行(6)。若算不出来，则使用指定的默认路由器；

   6. 报告转发分组出错。若算不出来，且没有指定默认路由，则报错；

8. 保证路由表正确性的机制

   1. 最大度量值：路由器设置一个最大度量值，当达到该值时，路由器就会认为这条路由已经失效，将其清除出路由表；
   2. 水平分割：从一个方向上学来的路由信息，不能再放入发回那个方向的路由更新包并且又发回那个方向；
   3. 路由中毒：网络出现故障时，通知邻居该网段不可用；
   4. 毒性反转：当一条路径信息变为无效之后， 路由器并不立即将它从路由表中删除，而是用不可达的度量值将它广播出去，它可以立即清除相邻路由器之间的任何环路；
   5. 保持时间：该网段的路由变成“down”状态时，还要在路由器中保留一段时间；
   6. 触发更新：当路由器发现某个网段出现故障时，立即发送路由更新包来通知邻居，而不用等到下一次发送路由更新包的时间。

9. 虚电路网络和数据报网络

   - 仅在网络层**提供连接服务**的计算机网络称为**虚电路网络**
   - 仅在网络层**提供无连接服务**的计算机网络称为**数据报网络**

	虚电路网络	数据报网络
可靠通信实现方法	由网络保证	由用户主机保证
是否需要建立连接	是	否
分组转发	属于同一条虚电路的分组均按照同一路由进行转发	每个分组独立选择路由进行转发
分组顺序	总是按照发送顺序到达终点	不一定按照发送顺序到达终点
分组的目的地址	虚电路号	目的地址（MAC地址）
结点出现故障	通过该故障结点的虚电路不能工作	该故障结点可能会发生分组丢失
差错处理和流量控制	网络负责或用户主机负责	用户主机负责