

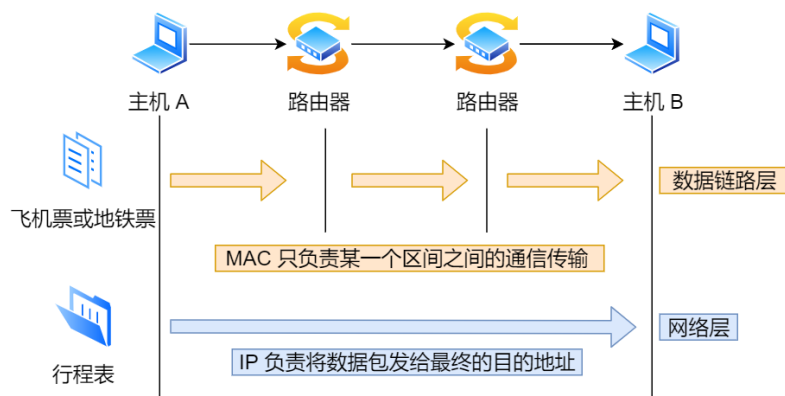
IP相关知识

• IP的基础

• IP的作用

- IP的位置：
 - IP，就是TCP/IP参考模型中的第三层——**网络层**
网络层的主要作用：主机和主机之间的通信，点对点的通信
- IP的作用：
 - 是在复杂网络的环境中，能够将数据包发送给最终目的的主机
- PS：TCP/IP模型：5层模型
 - 物理层 -> 数据链路层 -> 网络层 -> 传输层 -> 应用层
 - 对比OSI的7层模型
 - 物理层 -> 数据链路层 -> 网络层 -> 传输层 -> 会话层 -> 表示层 -> 应用层

• IP（网络层）与MAC（数据链路层）的关系



- IP的作用是：主机之间的通信用的，是负责在【没有直连】的两个网络之间通信
- MAC的作用是：实现【直连】的两个设备之间的通信
- eg：类比旅行：
 - 小A制定了一个旅行行程表，需要去杭州 - 云南苍山——一个是源IP，另一个是目的IP，整个行程表就是网络层
 - 小A根据行程表买了车票：地铁票，坐地铁到机场，飞机票，坐飞机到云南，公交车票，坐公交车到苍山。而在区间内移动就是数据链路层，该区间内出发点就是源MAC地址，目标地点就是目的MAC地址
 - 但是，两者必须同时都有，需要有行程表知道流程，也需要车票才能到达下一个点
 - ——计网中需要MAC层和IP层，才能实现向最终目标地址的通信
- 注意的是：

- 源IP地址和目的IP地址，在传输过程中不会发生变化——行程表的起点和终点
- 而源MAC地址和目的MAC地址不断发生变化——车票的起点和终点

• IP地址

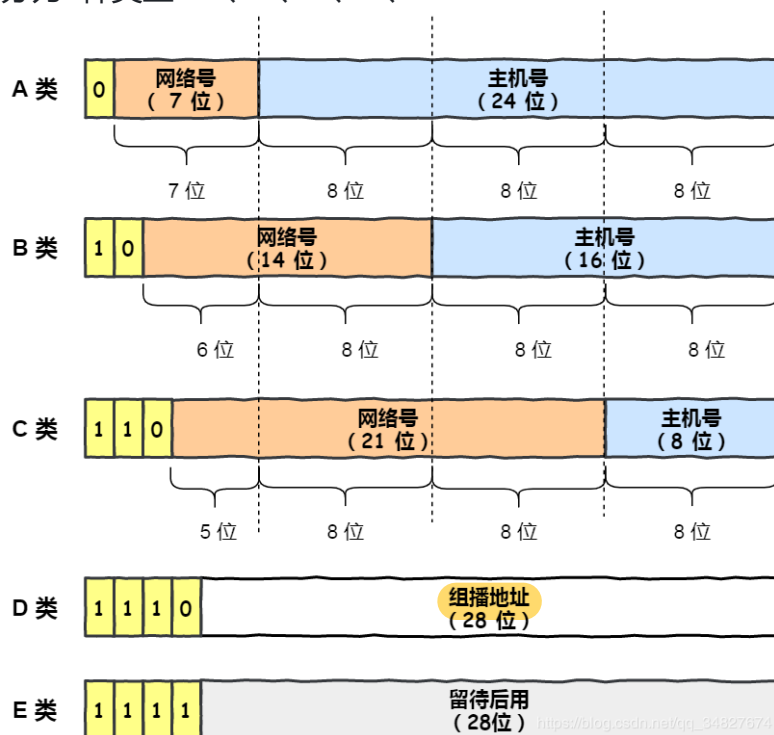
• IP地址的定义

- 为了实现网络通信，每个设备都需要配置有一个IP地址
- IP地址（IPv4），由32位的正整数表示，采用点分十进制表示，而32位最大能表示的范围为43亿，那么能够允许43亿台计算机连接到网络
- IP地址的分配，是根据网卡。eg：服务器、路由器都是有2个以上的网卡，那么就有2个以上的IP地址
- 实际上，全球的电子设备是远大于43亿，所以不可能用32位表示所有入网的电子设备，那么利用更换IP地址的技术——NAT，使得实现这个功能

• IP地址的分类

针对32位的IP地址，设计了分类地址

- 分为5种类型：A、B、C、D、E



- A类：由 网络号 + 主机号 组成

eg: 64.0.0.2/8

- 网络号：[0, 127]，即0000 0000 ~ 0111 1111，主机号就是后面的24位，即可存放16777214个网卡 + 2个特殊的IP地址

- B类：由 网络号 + 主机号 组成

eg: 172.20.0.1/16 (前16位表示网络号)

- 网络号：[128.0, 191.255]，即1000 0000, 0000 0000 ~ 1011 1111, 1111 1111，主机号就是后面的16位，即可存放65534个网卡 + 2个特殊的IP地址

- C类：由 **网络号 + 主机号** 组成

eg: 192.0.0.1/24

- 网络号：[192.0.0, 223.255.255.255]，即1100 0000, 0000 0000, 0000 0000 ~ 1101 1111, 1111 1111, 1111 1111，主机号就是后面的8位，即可存放254个网卡 + 2个特殊的IP地址

- D类、E类

- D类：是不包含主机号的，不可用于主机IP，常被用于多播（又称组播，穿透路由）

- 表示范围：224.0.0.0 ~ 239.255.255.255
- 多播：是用来**将包发送给特定组内的所有主机**，**因为广播无法穿透路由，如果想给其他网段发送同样的包，那么可以用穿透路由的多播**
- 32位中，前4位1110表示多播操作，而后28位是多播的组编号：
224.0.0.0 ~ 224.0.0.255为预留组播地址；
224.0.1.0~238.255.255.255为用户可用的组播地址；
239.0.0.0~239.255.255.255为本地管理组播地址，可在内部网中使用

- E类：是预留分类，暂时未使用

- 表示范围：240.0.0.0 ~ 255.255.255.255

- PS：特殊的IP地址

- 主机号全为1的地址，可以指定该**网络下的所有主机**，可以用来**广播**。

- 广播：在同一个链路种相互连接的主机之间发送数据包
- 分为本地广播和直接广播

- 本地广播：限制在本网络内的广播

eg: 在网络地址为192.168.0.0/24下，广播地址为192.168.0.255，这个就是本地广播，而该IP包会**被路由器屏蔽**，所以不会到达192.168.0.0/24范围外的网络中

- **直接广播：在不同网络之间的广播**

eg: 在网络地址为192.168.0.0/24的主机向192.168.1.255/24的目标地址发送IP包，那么路由器收到该IP包，将数据转发给192.168.1.0/24后，使得所有的主机192.168.1.1~192.168.1.254都能够收到该包——但是，直接广播**存在安全问题，一般路由器会判定为不转发**

- 主机号全为0的地址，**指定为该网络**（就是网络号）

- IP分类的优点

分类地址的优点就是简单明了、选路简单

- 可以不断判断前几位的值，从而得到是属于哪个类的
 - 第一位为0，那么为A类
 - 第一位为1，第二位为0，那么为B类
 - 第一位为1，第二位为1，第三位为0，那么为C类

- 第一位为1，第二位为1，第三位为1，第四位为0，那么为D类
- 第一位为1，第二位为1，第三位为1，第四位为1，那么为E类

- IP分类的缺点

- 同一个网络下没有地址层次

eg: 对于一个企业，使用B类网络，在该网络下没有层级，所有主机都是相同等级，不符合公司的多层分级

- A、B、C类无法与实际网络匹配：C类普通的不够用，而B类企业的太多

- 无分类的IP地址：CIDR

主要是针对，传统的IP分类地址存在的缺点：同一个网络下没有地址层次，与实际不匹配等，提出了CIDR

- 定义：CIDR，是无分类的地址，IP地址被划分为**前面为网络号，后面为主机号**

- 表示形式：**a.b.c.d/x**，x表示**前x位为网络号**，x的范围为[0, 32]——增加灵活性

eg: 10.100.122.2/24，那么前24位为网络号，后8位为主机号，该网络下最多可以有254个主机

- 子网掩码：用来划分网络号和主机号

就是掩盖掉主机号，剩下的就是网络号

eg: 10.100.122.2/24的子网掩码就是255.255.255.0

IP地址和子网掩码按位与，那么就能得到该IP地址的网络号

- why需要分离网络号和主机号？（用IP地址和子网掩码做相与）

- 两个计算机进行通信，先需要判断是否在同一个广播域内，即网络号是否相同，如果相同可直接发送给目标主机；如果不同，需要通过路由转发

- 路由器寻址，也是通过计算得到网络号，然后将数据包发送到对应的网络中

- 子网划分的方法

未做子网划分的 ip 地址：



做子网划分后的 ip 地址：



- 子网掩码还能划分子网：将主机地址划分为：子网网络地址 + 子网主机地址

eg: C类的网络地址：192.168.1.0，而子网掩码为255.255.192.0，那么根据网络地址得到前24位为网络号，后8位为主机号；而子网掩码又限定了后8位主机的前2位位子网网络地址——那么子网就有4个（00，01，10，11），那么第一个子网可以表示的范围为[192.168.1.1, 192.168.1.62]，第二个是[192.168.1.65, 192.168.1.126]，第三个是[192.168.1.129, 192.168.1.190]，第四个是[192.168.1.193, 192.168.1.254]

- 公有和私有的IP地址

类别	IP 地址范围	最大主机数	私有 IP 地址范围
A	0.0.0.0 ~ 127.255.255.255	16777214	10.0.0.0 ~ 10.255.255.255
B	128.0.0.0 ~ 191.255.255.255	65534	172.16.0.0 ~ 172.31.255.255
C	192.0.0.0 ~ 223.255.255.255	254	192.168.0.0 ~ 192.168.255.255

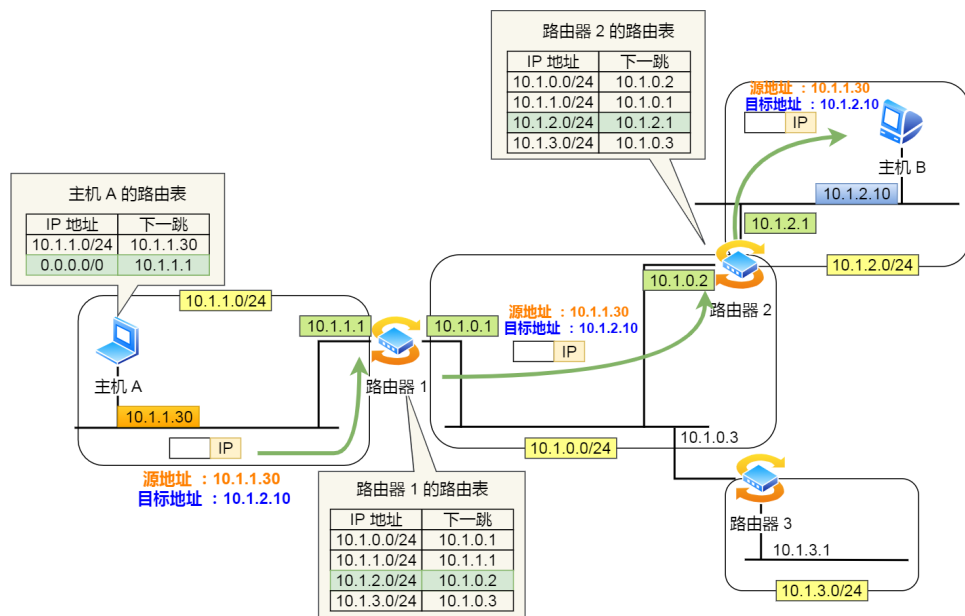
在A、B、C类的地址中，实际上会被分为：公有的IP地址 和 私有的IP地址
规定哪些IP地址是公有的，哪些IP地址是私有的

- 运行机制：

- 办公室、学校、家里用的都是**私有IP**，这些地址允许组织内部的人员进行IP分配和管理，在**该组织内私有IP不可重复**，但是可以和其他组织内的IP重复
- 但是，一旦访问组织外面的网络，就需要带上**公网IP**地址，这个地址是唯一的，不可重复
- **公网IP**：是有一个组织统一分配的，需要申请付费购买，那么全世界的人都可以访问到，它能够保证，该公网IP在**整个互联网范围内保持一致**
 - 管理部门：（私有IP地址，是内部的IT人员管理），**公网IP需要ICANN组织管理**（互联网名称与数字地址分配机构），IANA就是负责分配互联网IP地址的，按州的方式层层分配，在中国大陆是CNNIC的机构进行管理

- **IP地址与路由控制**

- IP地址的**网络地址这部分，来进行路由控制**
- 路由控制表（**主机和路由器都有各自的一张表**）
 - 各个网络地址的集合，和每个网络地址下一跳一个发送到的路由器的地址——就是查表，找到该数据包下一个要到的位置
- 路由控制表的作用
 - 发送IP包时，先确定IP包首部中的目标地址，通过路由控制表查找到与**该地址相同的网络地址的记录**，根据该记录将IP包转发给下一个路由器（表明通过该路径能正确传递到目标地址），如果路由控制表中存在多条相同网络地址的记录，那么选择相同位数最多的网络地址——**最长匹配**，而如果在路由表上没有找到匹配的，就转发到**默认路由**上
- 举例：



- 源地址为10.1.1.30，目标地址为10.1.2.10，
- 1. 从主机A开始发送，而在主机A的路由控制表中，未找到和目标地址匹配的，所以包被转发到**默认路由**
- 2. 路由器1收到后，需要转到出去查表得到，10.1.2.0/24最适合，到该网络地址需要发送到10.1.0.2的机器上（可能是路由器，也可能是主机），于是将包转发到路由器2
- 3. 路由2收到之后，查本地的路由控制表，发现10.1.2.0/24最适合，于是将包传送到下一跳的机器上10.1.2.1，而10.1.2.1就是路由器本身的一个接口的IP地址，与之直接相连的就是目标地址10.1.2.10
- **环回地址：不会流向网络**
 - **127.0.0.1**：环回地址，一般也称为**localhost**
 - 作用：在同一台计算上的**程序之间进行网络通信**时用到的一个默认地址，在浏览器中输入该地址，数据包不会流向网络

• IP分片与重组

- 背景：在传送数据时，肯定存在一个大小上限。且，每种数据链路的最大传输单元（MTU）是不同的——是因为，每个不同类型的数据链路使用目的不同，所以可以承载的MTU是不同的。最常见的**以太网的MTU是1500字节**，当数据包大小超过1500B，**IP数据包就会被分片**
- 分片规则：重组是由目标主机进行的，**路由器不会进行重组**
- 代价：在分片传输过程中，如果某个分片丢失，那么整个IP数据报就作废了，所以TCP引入**MSS**，使得TCP层（应用层）进行分片而不由IP层（网络层）分片（而**UDP没有一个上限控制**，所以就给IP层来分片，导致如果分片发生丢失，就需要重传整个数据包）

• IPv6的基本认识

- 背景：IPv4是32位，大概有42亿个地址，已经耗尽，所以需要更大范围的IP地址
- IPv6的优势：128位，数量很大，并且IPv6有更好的**安全性和扩展性**

- 1.可配置的地址变多
- 2. 可自动配置IPv6：即插即用
- 3. **包首部长度的固定，为40字节**，省去了包头的**校验和**，简化首部结构，减轻了路由器负荷，**提高传输性能**
- 4. **IPv6能够防止线路窃听**，也能够防止伪造IP的网络安全隐患，提升了**安全性**

- IPv6地址的标识方法：每16位为一组，每组用【:】隔开，如果中间的某些16位全为0，那么可以忽略，并且用【::】替换，但一个IP地址只允许出现一次双冒号

IPv6 用二进制数表示

1111111101101100 : 1011011001011000 : 0111011001011000 : 0000000000000000 : 0000000000000000 : 0000000000000000 : 0000000000000000 : 0011001000010000

IPv6 十六进制数表示

FEDC:BA98:7654::3210

- IPv6地址的分类——也是通过IP地址的前几位来标识IP地址的种类
 - 单播地址：一对一，下面还划分了3类
 - 链路本地单播地址：在同一链路中进行单播通信，不经过路由器（IPv6特有的）
 - 唯一本地地址：当在内网里单播通信，等同于IPv4的私有IP
 - 全局单播地址：互联网中通信时，等同于IPv4的公有IP
 - 组播地址：一对多
 - 任播地址，用于通信最近的节点，最近的节点是由路由协议决定的
 - 没有广播地址**
- 普及：IPv4和IPv6不能互相兼容，所以设备需要支持，还需要运营商升级设备，所以IPv6普及率较慢

IPv4首部和IPv6的区别

IPv4 首部

Version 版本	IHL 首部长度	TOS 服务区分	Total Len 总长度	
Identification 标识		Flags 标志	Fragment Offset 片偏移	
TTL 生存时间	Protocol 协议	Header Checksum 首部校验和		
Source Address 源地址				
Destination Address 目标地址				
Options 可选字段			Padding 填充	

IPv6 首部

Version 版本	Traffic Class 通信量号	Flow Label 流标号	
Payload Length 有效数据长度		Next Header 下一个首部	Hop Limit 跳数限制
Source Address 源地址			
Destination Address 目标地址			

保留字段

取消字段

名字位置变化

新增字段

- IPv6 和IPv4的首部改进
 - 取消首部校验和字段：

因为IPv6会在数据链路层和传输层进行校验，所以没有必要进行IP校验
 - 取消分片/重新组装相关字段：

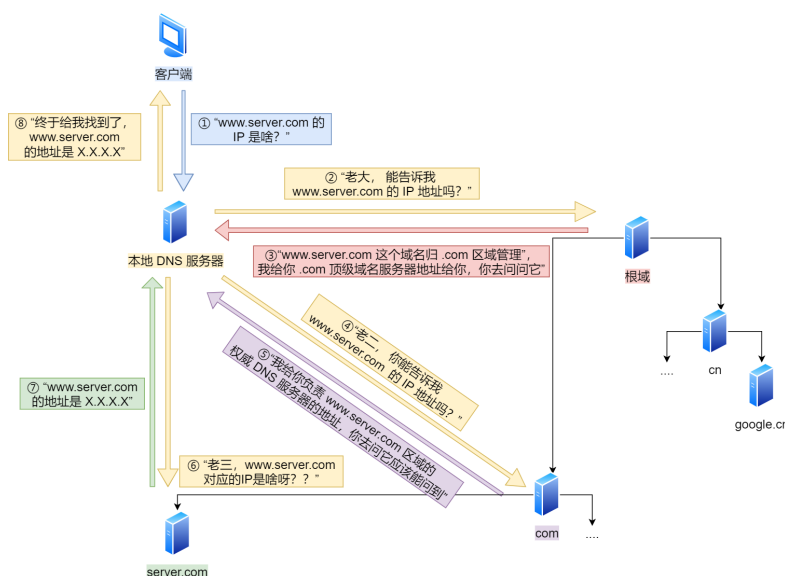
IPv6不允许在中间路由器中进行分片和重组，分片和重组只能在源与目标主机上，可以提高路由器转发的速度
 - 取消可选字段

而替换到了，IPv6的下一个首部的位置中，那么IPv6的首部大小是固定的，为40字节

IP协议的相关技术

• DNS的域名解析

- 我们上网时，常见使用方式是**域名**，而不是IP地址，主要是域名方便人记忆，例如 baidu.com
- 域名网址自动转换为IP地址：使用的就是DNS域名解析
- 举例：www.baidu.com 域名通过句点进行分隔，通过两个句点，分隔了不同层次，越靠右的位置层次越高
- 域名的层级关系
 - eg: www.baidu.com
 - 域名用点进行分隔，句点代表了不同层次之间的界限。域名越右的位置，层级越高——更外国人的习惯思维一致
 - **根域**是最顶层，下一层就是**顶级域**，即com,
 - 层级关系是一个树状结构：
 - 根域DNS服务器：该信息是保存在互联网中**所有的DNS服务器**中，那么所有的DNS服务器都可以找到并且访问根域DNS服务器，所以根据树状结构来看，客户端可以通过任何一台DNS服务器，那么一定能找到根域DNS服务器，然后通过树状结构一定能找到目标DNS服务器
 - 顶级域DNS服务器，常见的就是com,cn等
 - 权威DNS服务器，eg: baidu.com
 - ——客户端只要能找到任何一台DNS服务器，那么就能找到根域服务器，那么能顺藤摸瓜找到任何一个目标服务器
- 域名解析的工作流程



浏览器首先看一下浏览器的缓存中是否存在域名对应的IP，如果没有找OS的cache中要，还没有就去检查本机域名解析文件 `hosts`，如果没有就会找DNS服务器进行查询

1. 客户端首先发出一个DNS请求，请求查询指定域名的IP地址，eg: www.baidu.com，发给本地DNS服务器——客户端的TCP/IP设置中填写的DNS服务器地址（这个就是普通的IP通信）
2. 本地DNS服务器收到客户端的请求后，先看自己的缓存中是否存在该网址对应的IP，如果找到了那么直接返回IP地址；如果没有，那么本地DNS服务

器会去访问根DNS服务器，请求获得该网址对应的IP

- 3. 根DNS服务器**不直接用于域名解析，只是用来告知查询哪个顶级域DNS服务器**。根DNS服务器根据网址，查到哪个顶级域DNS服务器，即.com是所属哪个域名管理器，然后将该信息返回给本地DNS服务器
- 4. 本地DNS服务器收到该信息后，向该顶级域DNS服务器询问该网址对应的IP
- 5. 顶级域DNS查询自己下属的权威DNS服务器，然后将权威DNS服务器的信息告知本地DNS服务器
- 6. 本地DNS服务器收到该信息后，向权威DNS服务器查询网址
- 7. 权威DNS服务器，就是域名解析结果的原出处，然后将网址的IP地址返回给本地DNS服务器
- 8. 本地DNS收到IP地址后，返回客户端，客户端就拿着IP地址向目标地址传递数据包
- ——整个过程中，都是本地DNS在不断DFS的查询内容，但是三级DNS服务器，都是只**查询自己知道的，并不帮助向下查**
- ——域名解析，就是只指路不带路

• ARP与RARP协议

• ARP协议

- 背景：传输一个IP数据报时，知道源IP地址和目的IP地址，在传递过程中会通过当前机器的路由表确定传输的【下一跳】，但是路由表得到的是下一跳的IP地址，而到数据链路层中，需要将下一跳的IP地址转换为MAC地址，才能在数据链路层上进行传输
- 作用：通过ARP协议，**将IP转化为对应的MAC地址**
- ARP将IP地址转换为MAC地址的过程：
 - 协议概括：ARP是借助**ARP请求与ARP响应**两种类型的包确定MAC地址
 - 1. 主机会通过**广播发送ARP请求**，而包中包含了等待转换的IP地址
 - 2. 同一个链路中的所有设备都能收到ARP请求，然后会去解析该包的内容，如果包中需要转换的IP地址和自己的IP地址符合，那么会将自己的

MAC地址塞入ARP的响应包返回给主机

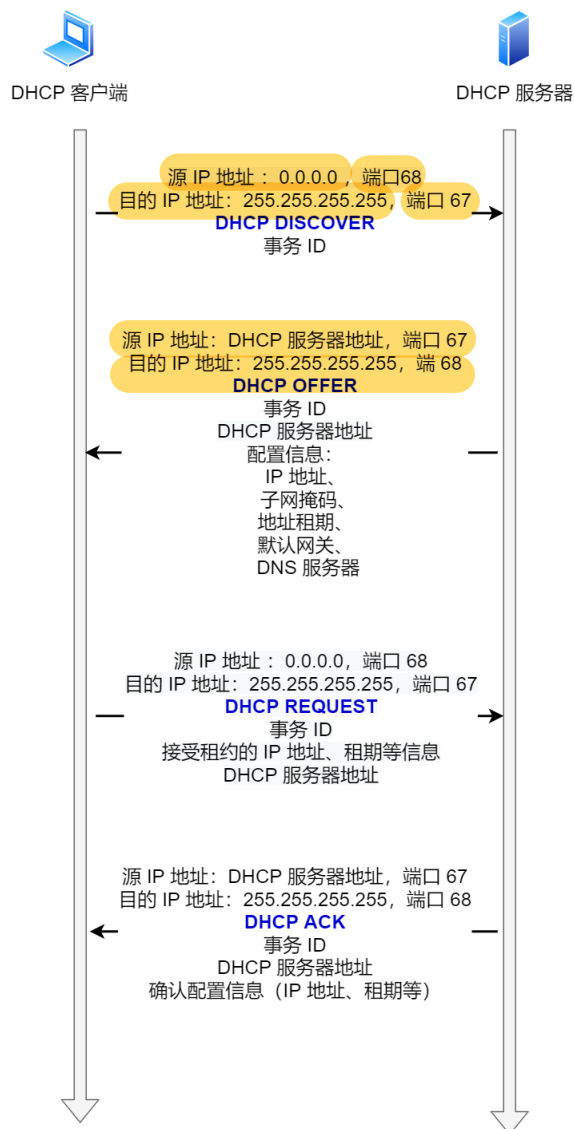
- 3. 如果是第一次通过ARP得到IP地址转换的MAC地址，那么OS会将对应的IP-MAC地址进行缓存，方便下次可以直接调用——但是，该缓存是有一定期限的，超过期限会被清除

RARP协议

- 作用：是根据MAC地址得到对应的IP地址
- 使用场景：将打印机服务器等嵌入式设备接入到网络时会用到
- 使用流程：
 - 需要配置一台 RARP 服务器，能够在该服务器上注册设备的MAC地址和IP地址，然后该设备就能接入网络：
 - 1. 该设备会发送一条请求信息到 RARP 服务器，格式如：我的MAC地址是....，请求得到IP地址
 - 2. RARP服务器收到该请求后，会给该设备分配一个IP地址，并且向该设备发送响应包，告知IP地址

DHCP动态获取IP地址

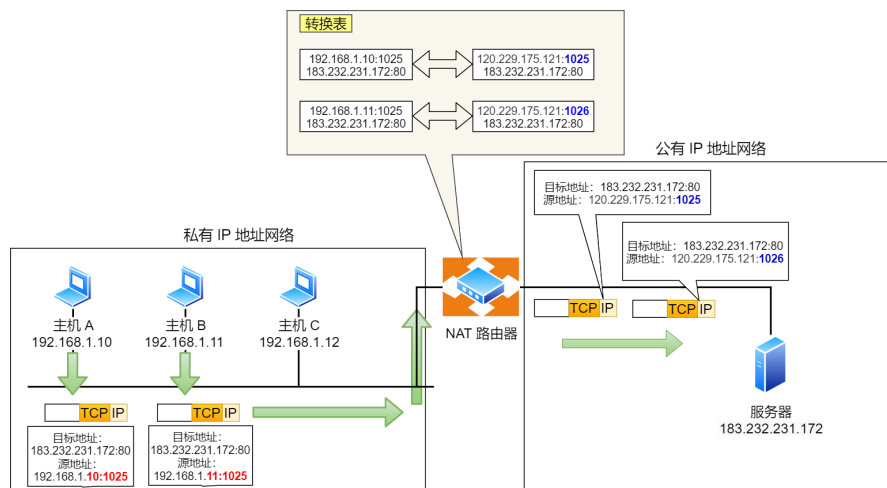
- 常见，电脑通过DHCP动态获取IP地址，能够省去配置IP信息的过程
- 4步获得IP



- ps: DHCP客户端进程监听的是68端口号, 服务端进程监听的是67端口号
- 1. 客户端先发起DHCP发现报文的IP数据报, 由于客户端没有IP地址, 也不知道DHCP服务器的地址, 那么使用的UDP广播通信, 使用的目的地址是255.255.255.255 (端口67), 而使用0.0.0.0 (端口68) 作为源地址, 客户端将IP数据报传递给链路层, 然后链路层将帧广播到所有的网络设备中
- 2. DHCP服务器收到DHCP发现报文, 服务器响应DHCP提供报文, 该报文仍然是广播, 目的地址是255.255.255.255。该报文信息携带了可租约的IP地址、子网掩码、默认网关、DNS服务器和IP地址的租用期
- 3. 客户端会收到1个或多个服务器的DHCP提供报文, 从中选择一个服务器, 并向选择的服务器发送DHCP请求报文作为响应, 回显配置参数
- 4. 服务端用DHCP ACK报文对收到的请求报文做出响应, 应答要求的参数
- ——客户端收到DHCP ACK报文后, 交互便完成了, 客户端能在限定的时间内使用分配的IP地址
- 快到期的操作: (而不是到期之后才发送)
 - 客户端会向该服务器发送DHCP请求报文, 服务器收到报文后:
 - 服务器如果同意继续租用, 那么会用DHCP ACK报文响应, 客户端可以延长租期
 - 如果不同意, 那么用DHCP NACK报文响应, 客户端就要停止使用该IP
- DHCP交互中, 全程使用UDP广播通信:
 - 存在的问题: 路由器不会转发广播包, 那如果DHCP服务器和DHCP客户端不在同一个局域网中, 那么该如何通信, 是否需要在每个网络中配置一个DHCP服务器
 - 解决方法: 采用DHCP中继代理, 那么不同网段的IP地址分配都可以使用一个DHCP服务器进行统一管理
 - 客户端会向DHCP中继代理发送DHCP请求包, 而DHCP中继代理收到该请求包后, 以单播形式发送给DHCP服务器
 - 服务器收到包后向DHCP中继代理发送响应, 然后中继代理广播该响应

• NAT网络地址转换

- 目的: 主要是为了缓解IPv4地址耗尽的问题, 主要就是在在一个局域网内的主机对外通信时, 将私有IP地址转换为公有的IP地址
- 普通的NAT: 一个私有IP, 就要对应一个公有IP, 没有解决问题
- 使用的技术: NAPT (网络地址与端口转换), 使用IP地址 + 端口号一起进行转换



- 如上图，多个私有IP地址可以转换成同一个共有地址，但是它们用不同的端口作为区分

- 至此会生成一个NAPT路由器转换表，可以将地址和端口组合转换，那么多个客户端可以通过同一个IP地址与其他网络的主机进行连接通信

- ps：转换表是自动生成的

在TCP的情况下，TCP连接首次握手时，SYN包发出后，就会在表中生成对应条目。然后在关闭连接时，发出FIN包会从表中删除

- 缺点：主要原因是，转换时需要查询转换表

- 1. 外部无法主动与NAT内部服务器建立连接，因为NAPT转换表没有转换记录。即，只能从内部向外部发出连接
- 2. 转换表的生成和转换操作都会影响性能
- 3. NAT路由器重启，那么所有TCP连接都会被重置

- 解决方法：

- 1. 使用IPv6，那么每个设备都能分配到一个公有IP，不需要再进行地址转换了
- 2. NAT穿透技术，主要是让客户端会主动获取NAT设备的公有IP，然后为自己建立端口映射条目——就是让应用程序自动完成，而不是让NAT设备建立映射

• ICMP互联网控制报文协议

- 功能：——主要是控制

- 确认IP包是否成功送达目标地址
- 报告发送过程中IP包被废弃的原因
- 改善网络设置
-
- ——在IP通信时，如果某个IP包因为某种原因无法正确到达时，ICMP会负责发现该原因并通知

- 故障通知方式：通知会使用IP进行发送

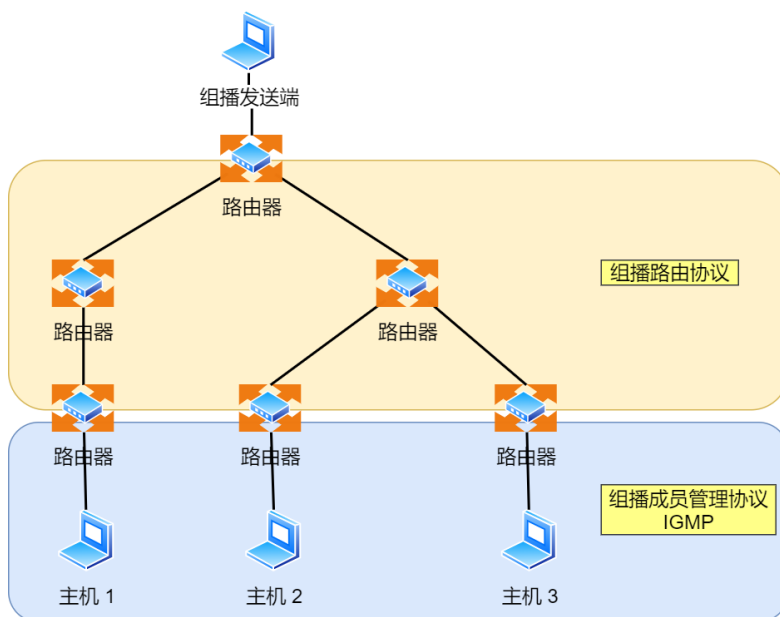
- ICMP类型：

ICMP 类型		
	内容	种类
0	回送应答 (Echo Reply)	查询报文类型
3	目标不可达 (Destination Unreachable)	差错报文类型
4	原点抑制 (Source Quench)	差错报文类型
5	重定向或改变路由 (Redirect)	差错报文类型
8	回送请求 (Echo Request)	查询报文类型
11	超时 (Time Exceeded)	差错报文类型

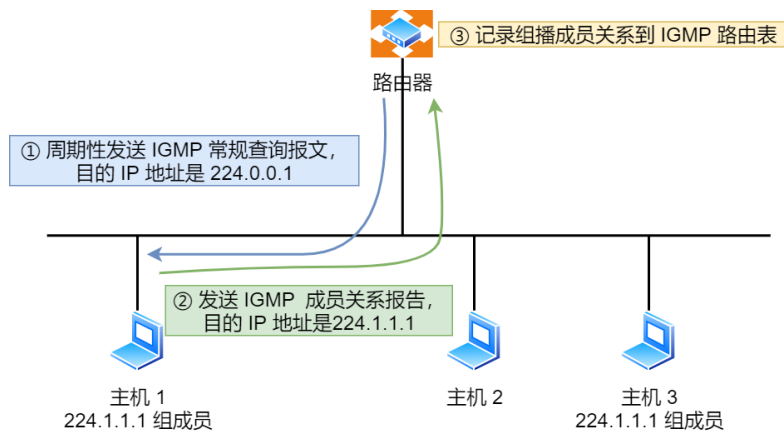
- 用来诊断的查询消息，**查询报文类型**
- 用来通知出错原因的错误消息，**差错报文类型**

• IGMP因特网组管理协议

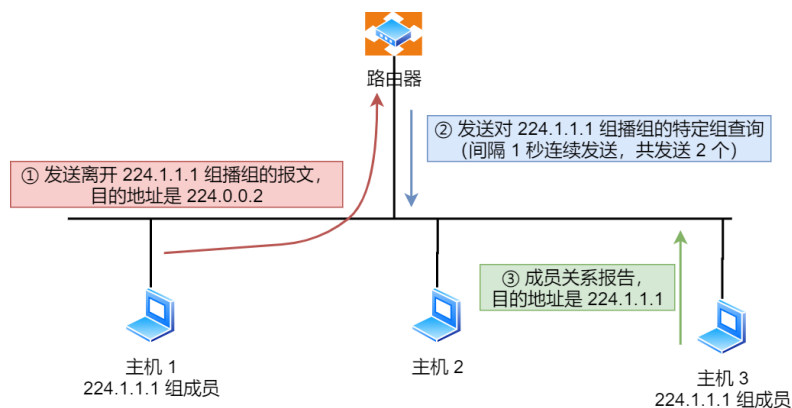
- 与ICMP毫无关系
- 背景：组播地址（D类地址），即只能有一组的主机能够收到数据包，而不在同一组的主机收不到——那么需要IGMP才能管理组
- 工作范围：在组播成员（主机）和最后一跳路由之间



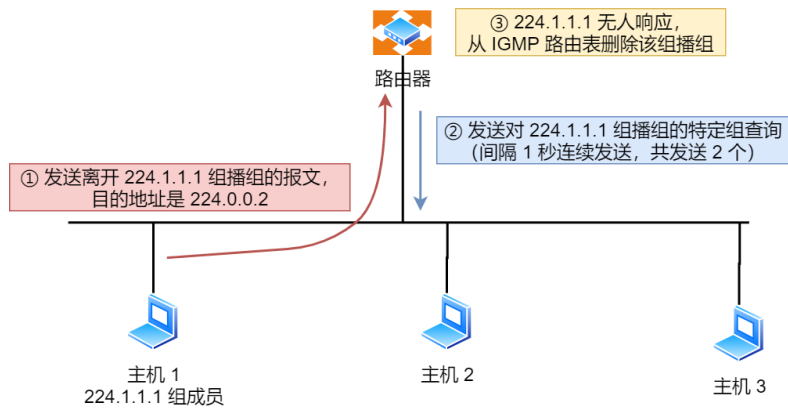
- IGMP报文向路由器申请加入和退出组播组
- IGMP报文会采用IP封装，IP首部的协议号为2，TTL字段值通常为1——**IGMP是工作在主机和直连的路由器之间**
- IGMP的版本：IGMPv1，IGMPv2，IGMPv3
- IGMP工作机制：以IGMPv2作为例子
 - 常规查询机制 和 响应工作机制：



- 1. 路由器会**周期性发送IGMP常规查询报文**到目的地址为**224.0.0.1**（表示为，**同一个网段内所有主机和路由器**）
- 2. 而组成员收到该查询后，eg：主机1，主机3，会启动报告延迟计时器，倒计时时间是随机的（0~10s），超时后，主机会发送**IGMP成员关系报告报文**，目的地址为组播地址。如果在倒计时内，收到同一个组内的其他主机发送的相同类型报文，那么自己不在发送——即，倒计时最短的最先发送报文，组内只发送一次。可以减少，网络中多余的IGMP报文数量
- 3. 路由器收到该成员关系报告报文后，会在IGMP路由表中加入该组播组，那么网络中报文的地址是该组播地址，路由器收到后会将数据包转发出去
- 离开组播组工作机制：
 - 如果删除之后，网段中仍然存在该组播组：



- 1. 主机1需要从该组中离开，会发送**IGMPv2离组报文**，报文的地址是：224.0.0.2——表示发向网段内的所有路由器
- 2. 路由器收到该报文后，1s为间隔连续发送**IGMP特定组查询报文**（共会发送2个），用来确认该网络中是否还有该组的其他成员
- 2. 主机3还在组内，会立即响应该特定组的查询。那么路由器知道该组内还存在其他组播成员，那么还会向该组内发送对应的组播包
- 如果删除之后，网段中不存在该组播组：



- 1. 同上面步骤
- 2. 路由器也会间隔1s发送2个特定组查询报文。但是组内没有其他组成员，所以没有得到响应
- 3. 超时等待后，路由器会认为224.1.1.1已经不存在成员了，不会向该网段转发该组播地址的数据包

• IP数据报

• IP的包头格式

版本 (4位)	首部长度 (4位)	服务类型 TOS (8位)
总长度 (16位)		
标识 (16位)		
标志 (3位)	片偏移 (13位)	
TTL (8位)		协议 (8位)
首部校验和 (16位)		
源IP地址 (32位)		
目标IP读者 (32位)		
选项		
数据		

- 选择合适网卡

- 背景：客户端有多个网卡，那么就对应多个IP地址，那么需要从中选择一个作为源地址——即选择一个合适的网卡
- 方式：通过路由表，选择合适的网卡（OS会自动选择，但是我们需要知道选择的方法）
- eg:

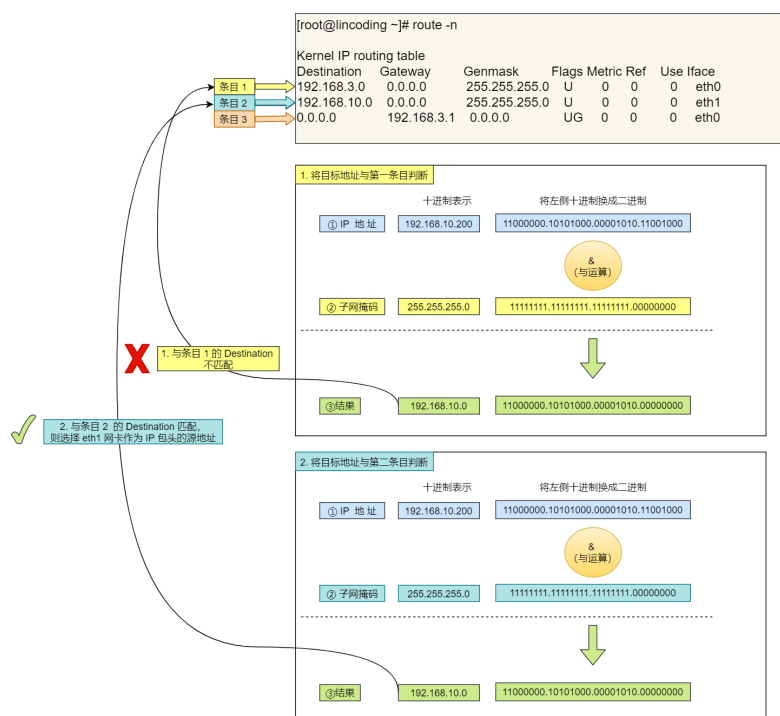
- 1. 首先查看Linux下的路由表

```
[root@lincoding ~]# route -n
```

Destination	Gateway	Genmask	Flags	Metric	Ref	Use	Iface
192.168.3.0	0.0.0.0	255.255.255.0	U	0	0	0	eth0
192.168.10.0	0.0.0.0	255.255.255.0	U	0	0	0	eth1
0.0.0.0	192.168.3.1	0.0.0.0	UG	0	0	0	eth0

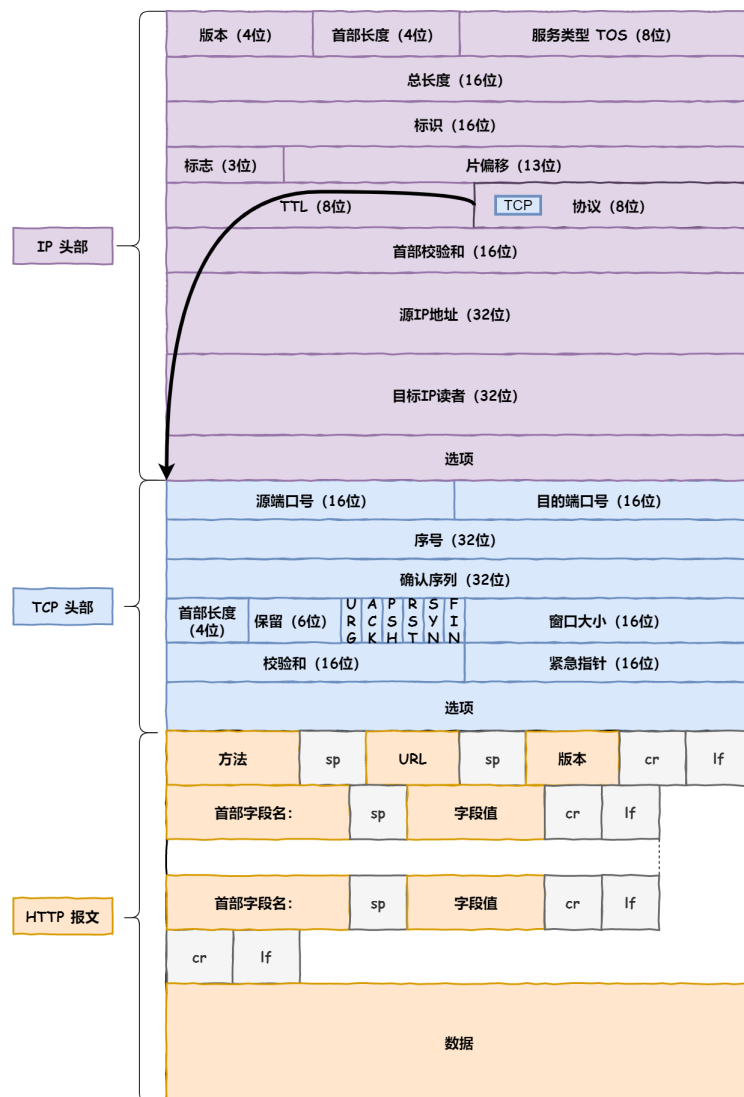
- 方式：route -n，可以看到路由表：即对之前发送出去的IP数据报目的IP的一个缓存，然后进行匹配，选择合适的，如果没有合适的就选择默认网关

- 2. 如果目的IP是：192.168.10.200



- 方式：目的IP & 子网掩码，destination & 子网掩码，如果两个结果相等，那么匹配成功，就通过该网卡发送；如果结果不等，就换一个；如果所有都不相等，那么选择默认网关。**gateway就是直接相连的路由器的IP地址**
 - ps：默认网关：destination和子网掩码都是0.0.0.0，和任意IP做&，都是一样的结果0.0.0.0，当所有结果都不匹配的时候，就选择该网关

- IP报文的生成



- = IP头部 + IP数据, IP数据 = TCP头部 + TCP 数据, TCP数据 = HTTP 头部 + HTTP数据, 因为TCP有MSS, 那么能够保证IP数据报不需要再分片了, 所以IP数据中一定都有TCP的头部, 而TCP的数据中不一定都有HTTP头部——因为TCP会对数据超过MSS的数据进行分片, 即对HTTP数据报进行分片