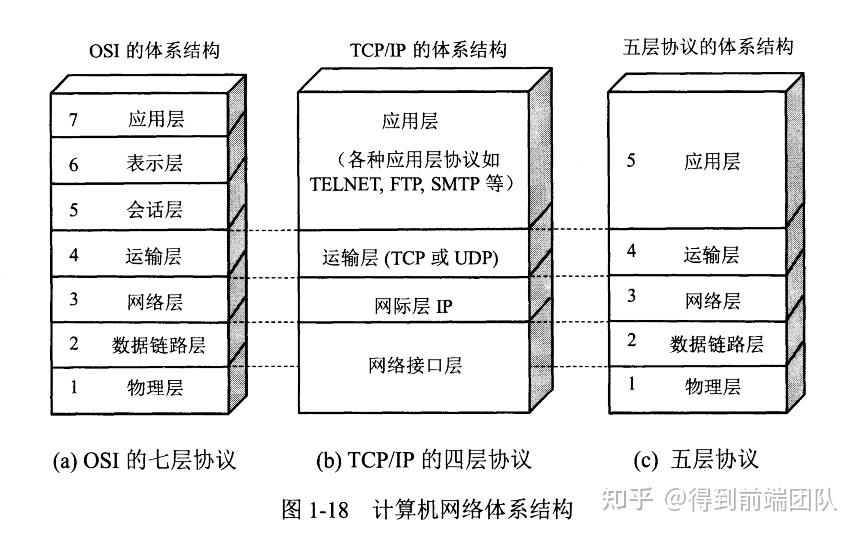
# **一、常见网络分层模型**



OSI体系结构概念清楚，理论也完整，但它既复杂又不实用。TCP/IP四层体系结构实际应用非常广泛，五层协议则是为了介绍网络原理设计的。

（5）**应用层**：通过应用进程间的交互来完成特定网络应用。定义了各种应用程序交互的规则。包括HTTP、 FTP、 TFTP、SMTP、SNMP、DNS、TELNET、HTTPS、POP3、DHCP等---报文

（4）**运输层**：为进程间的通信提供**通用的数据传输服务**。主要使用TCP（面向连接的、可靠的数据传输服务，报文段）、UDP（无连接的、尽最大努力的数据传输服务，用户数据报）两种协议

（3）**网络层**：为交换网上的不同主机提供**通信服务**，该层使用IP协议---分组/IP数据报/包。ICMP、IP、IGMP，主要负责网络中数据包的传送等

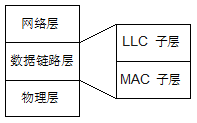
（2）**数据链路层**：建立逻辑连接、进行硬件地址寻址、差错校验等功能。（由底层网络定义协议）---帧，相关协议：ARP、PARP。

（1）**物理层**：建立、维护、断开物理连接--bit

# 以太网协议(802.3)

## 2.1 什么是以太网？

以太网是一种局域网协议(定义有线以太网的物理层和数据链路层相关标准)，它使用一组物理连接来连接多个计算机和网络设备。局域网不一定是以太网，只是由于目前大多数的局域网是以太网，所以一般说局域网，大家都默认为以太网。以太网是目前应用最普遍的局域网技术，取代了其他局域网技术如令牌环、FDDI和ARCNET2。更正式的讲，以太网是基于载波侦听多路访问/冲突检测（CSMA/CD）协议的IEEE 802.3标准的通用名称



## 2.2 以太网协议802.3x系列协议

以太网发展：经过长期的发展，以太网已包括标准以太网（10 Mbit/s）、快速以太网（100 Mbit/s）、千兆以太网（1000 Mbit/s）和万兆以太网（10 Gbit/s）。



以太网的线缆标准简介

以太网信息的传输介质包括同轴电缆，双绞线、光纤等。从以太网诞生到目前为止，成熟应用的以太网物理层标准主要有以下几种：



在这些标准中，前面的10、100、1000、10G分别代表运行速率；中间的BASE指传输的信号是基带方式；"-"后面的数字或字母，代表不同的物理传输方式，例如T表示使用的介质是同轴电缆，TX表明介质为双绞线，FX表明介质为光纤。

CSMA/CD协议（早期的共享以太网）：载波侦听多址/冲突检测，解决通信双方数据包冲突的问题

## 2.3 802.3 MAC帧格式以及帧字段含义

以太网传输的 PDU(Protocol Data Unit) 是以太帧(Ethernet frame)

IMG_256

在物理层上看，一个完整的以太网帧有7个字段，事实上，前两个字段并不能算是真正意义上的以太网数据帧，它们是以太网在物理层上发送以太网数据时添加上去的。为了实现底层数据的正确阐述，物理层使用7个字节前同步码（0和1交替的56位（55-55-55-55-55-55-55））实现物理层帧输入/输出同步；使用1个字节的SFD（帧首定界符，固定为10101011）标识帧的开始。上图中剩下的5个字段是真正的以太网数据，其中包含了目的地址和源地址，它们都是6字节长度（通常每个网卡都有1个6个字节MAC地址，以在以太网中唯一地标识自己）。网卡接收数据时，通过将目的地址字段和自身的MAC地址做比较，判断是否接收该数据包。通常，将这里的6字节目的地址按照下面的格式来书写，如：00-01-02-03-04-05。这6个字节在以太网中是按照从左到右的顺序发送的，同时对每个字节来说，最先发送的是最低位bit0，最后是最高位bit7。

上图中的长度/类型具有两个意义，当这两个字节的值小于1518时，那么它就代表其后数据字段的长度；如果这两个字节的值大于1518，则表示该以太网帧中的数据属于哪个上层协议（例如0x800，代表IP数据包；0x806，代表ARP数据包等。）

## 2.4 MAC地址的类型和bit位含义

在以太网帧中，目的地址可以分为三类：单播地址、多播地址和广播地址。单播地址通常与一个具体网卡的MAC地址相对应，它要求第一个字节的bit0（即最先发出去的位）必须是0；多播地址则要求第一个字节的bit0为1，这样，在网络中多播地址不会与任何网卡的MAC相同，多播数据可以被很多个网卡同时接收；广播地址的所有48位全为1（即FF-FF-FF-FF-FF-FF），同一局域网中的所有网卡可以接收广播数据包。

局域网虽然是个网络，但并不把局域网放在网络层讨论，因为网络层讨论的是多个网络互连的问题，即跨网段转发分组。而同一个局域网的分组转发并不经过路由器，因此整体来看，局域网仍属于数据链路层的范围。数据链路层主要使用两种信道：点对点信道和广播信道

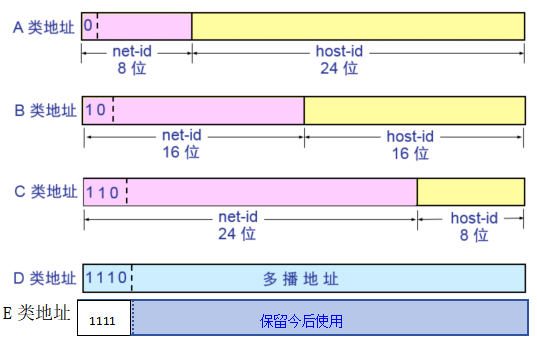
# IPv4基础系列协议之IP

## 3.1 IP地址的类型分类

**第一阶段：分类的IP地址**

IP地址::={<网络号>,<主机号>}

根据网络号的不同分为5种类型，A类地址、B类地址、C类地址、D类地址和E类地址。



**A类地址：1~126**

网络号127作为本地环回地址，用于本主机进程间的通信（目的地址为环回地址的IP数据报永远不会出现在任何网络上）

保留私网地址：10

**B类地址：128.1~191.255**

网络号128.0.0是不指派的，因此可以指派的B类最小网络地址为128.1.0

保留私网地址：172.16~172.31

**C类地址：192.0.1~223.255.255**

保留私网地址：192.168.0~192.168.255

转发器和网桥连接的若干个局域网仍为一个网络，具有不同网络号的局域网必须用路由器连接。

**第二阶段：子网的划分，从两级IP地址到三级IP**

IP地址::={<网络号>,<子网号>,<主机号>}

方法：从主机号借用若干位作为子网号，两级IP地址在**本单位内部**变成三级IP地址，但**对外仍然表现为一个网络**，子网号不能全为1或0

原因：分类的IP地址利用率太低，不够灵活

两级子网掩码：255.255.0.0 三级子网掩码：255.255.255.0

网络地址为：<网络号>,<子网号>

不管对于分类的IP地址还是划分子网的地址，只要让子网掩码与IP地址相“与”便可得出网络号

问题：在不划分子网的情况下，没有子网为什么还要子网掩码？

答：便于路由表查找，互联网标准规定，所有网络都必须使用子网掩码，路由表中也必须有，如果不划分子网，就使用默认子网掩码（255.0.0.0，255.255.0.0，255.255.255.0）

**第三阶段：无分类编址CIDR（构造超网）**

IP地址::={<网络前缀>,<主机号>}

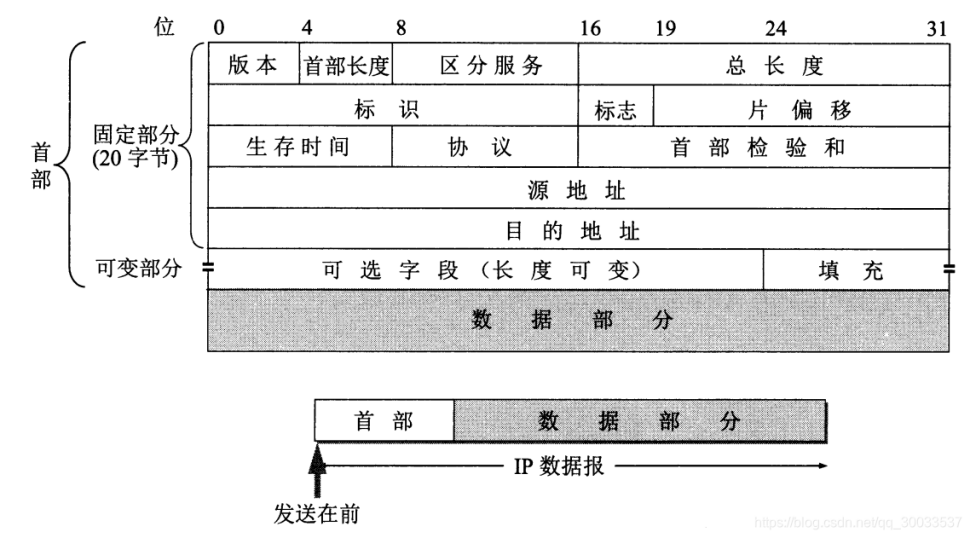
概念：使用变长子网掩码（VLSM），进一步提高IP地址利用率，并在VLSM的基础上进一步研究出无分类的编址方法，正式名字为“无分类域间路由选择CIDR”

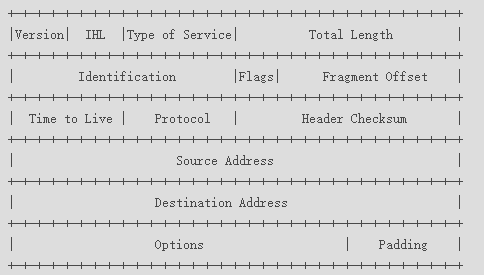
消除传统的地址分类和划分子网的概念。

方法：采用CIDR记法，即在IP地址后面加“/”，然后写上网络前缀所占的位数，为了方便路由，CIDR使用32位的地址掩码（01组成），查找路由表时，采用最长前缀匹配，即与路由表中的掩码相与，取网络号最长的那一项路由

优点：路由表利用CIDR地址块来查找目的网络，减少了路由表项数，能更加有效分配IPv4地址

## 3.2 IP的报文帧格式及其字段含义





**IP头字段解释**

|  |  |  |
| --- | --- | --- |
| 字段 | 长度 | 含义 |
| Version | 4bit | 4：表示为IPv4  6：表示为IPv6 |
| IHL | 4bit | 首部长度，如果不带Option字段，则为20，最长为60，该值限制了记录路由选项。以4字节为一个单位。 |
| Type of Service | 8bit | 服务类型。只有在有QoS差分服务要求时这个字段才起作用。 |
| Total Length | 16bit | 总长度，整个IP数据报的长度，包括首部和数据之和，单位为字节，最长65535，总长度必须不超过最大传输单元MTU。 |
| Identification | 16bit | 标识，主机每发一个报文，加1，分片重组时会用到该字段。 |
| Flags | 3bit | Bit 0: 保留位，必须为0。  Bit 1: DF（Don't Fragment），能否分片位，0表示可以分片，1表示不能分片。  Bit 2: MF（More Fragment），表示是否该报文为最后一片，0表示最后一片，1代表后面还有。 |
| Fragment Offset | 13bit | 片偏移：分片重组时会用到该字段。表示较长的分组在分片后，某片在原分组中的相对位置。以8个字节为偏移单位。 |
| Time to Live | 8bit | 生存时间：可经过的最多路由数，即数据包在网络中可通过的路由器数的最大值。 |
| Protocol | 8bit | 协议：下一层协议。指出此数据包携带的数据使用何种协议，以便目的主机的IP层将数据部分上交给哪个进程处理。 |
| Header Checksum | 16bit | 首部检验和，只检验数据包的首部，不检验数据部分。这里不采用CRC检验码，而采用简单的计算方法 |
| Source Address | 32bit | 源IP地址 |
| Destination Address | 32bit | 目的IP地址 |
| Options | 可变 | 选项字段，用来支持排错，测量以及安全等措施，内容丰富（请参见表3-7）。选项字段长度可变，从1字节到40字节不等，取决于所选项的功能。 |
| Padding | 可变 | 填充字段，全填0 |

# IPv4基础系列协议之ARP

## 4.1 什么是ARP协议？

ARP（Address Resolution Protocol，地址解析协议）是用来将IP地址解析为MAC地址的协议。主机或三层网络设备上会维护一张ARP表，用于存储IP地址和MAC地址的映射关系，一般ARP表项包括动态ARP表项和静态ARP表项。

## 4.2 为什么需要ARP?

在局域网中，当主机或其它三层网络设备有数据要发送给另一台主机或三层网络设备时，需要知道对方的网络层地址（即IP地址）。但是仅有IP地址是不够的，因为IP报文必须封装成帧才能通过物理网络发送，因此发送方还需要知道接收方的物理地址（即MAC地址），这就需要一个通过IP地址获取物理地址的协议，以完成从IP地址到MAC地址的映射。地址解析协议ARP即可实现将IP地址解析为MAC地址。

## 4.3 ARP有哪些类型？

**动态ARP**

动态ARP表项由ARP协议通过ARP报文自动生成和维护，可以被老化，可以被新的ARP报文更新，也可以被静态ARP表项覆盖。

动态ARP适用于拓扑结构复杂、通信实时性要求高的网络。

**静态ARP**

静态ARP表项是由网络管理员手工建立的IP地址和MAC地址之间固定的映射关系。静态ARP表项不会被老化，不会被动态ARP表项覆盖。

正常情况下网络中设备可以通过ARP协议进行ARP表项的动态学习，生成的动态ARP表项可以被老化，可以被更新。但是当网络中存在ARP攻击时，设备中动态ARP表项可能会被更新成错误的ARP表项，或者被老化，造成合法用户通信异常。

静态ARP表项不会被老化，也不会被动态ARP表项覆盖，可以保证网络通信的安全性。静态ARP表项可以限制本端设备和指定IP地址的对端设备通信时只使用指定的MAC地址，此时攻击报文无法修改本端设备的ARP表中IP地址和MAC地址的映射关系，从而保护了本端设备和对端设备间的正常通信。一般在网关设备上配置静态ARP表项。

**免费ARP**

设备主动使用自己的IP地址作为目的IP地址发送ARP请求，此种方式称免费ARP。

免费ARP有如下作用：

IP地址冲突检测：当设备接口的协议状态变为Up时，设备主动对外发送免费ARP报文。正常情况下不会收到ARP应答，如果收到，则表明本网络中存在与自身IP地址重复的地址。如果检测到IP地址冲突，设备会周期性的广播发送免费ARP应答报文，直到冲突解除。

用于通告一个新的MAC地址：发送方更换了网卡，MAC地址变化了，为了能够在动态ARP表项老化前通告网络中其他设备，发送方可以发送一个免费ARP。

在VRRP备份组中用来通告主备发生变换：发生主备变换后，MASTER设备会广播发送一个免费ARP报文来通告发生了主备变换。

**Proxy ARP**

如果ARP请求是从一个网络的主机发往同一网段但不在同一物理网络上的另一台主机，那么连接这两个网络的设备就可以回答该ARP请求，这个过程称作ARP代理(Proxy ARP)。

Proxy ARP有以下特点：

Proxy ARP部署在网关上，网络中的主机不必做任何改动。

Proxy ARP可以隐藏物理网络细节，使两个物理网络可以使用同一个网络号。

Proxy ARP只影响主机的ARP表，对网关的ARP表和路由表没有影响。

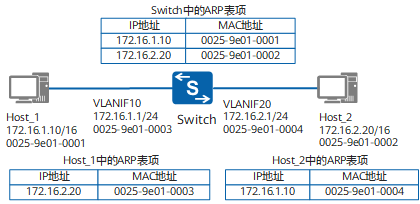
Proxy ARP分为路由式Proxy ARP、VLAN内Proxy ARP和VLAN间Proxy ARP。

**路由式Proxy ARP**

路由式Proxy ARP就是使那些在同一网段却不在同一物理网络上的网络设备能够相互通信的一种功能。

在实际应用中，如果连接设备的主机上没有配置缺省网关地址（即不知道如何到达本网络的中介系统），此时将无法进行数据转发。

如下图所示，Host\_1的IP地址为172.16.1.10/16，Host\_2的IP地址为172.16.2.20/16，Host\_1与Host\_2处于同一网段。Switch通过VLAN10和VLAN20连接两个网络，VLANIF10和VLANIF20的IP地址不在同一个网段。



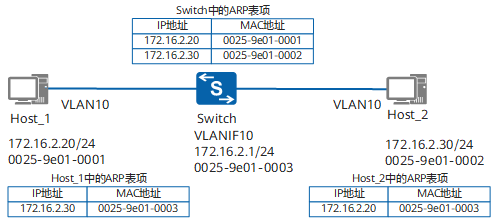
当Host\_1需要与Host\_2通信时，由于目的IP地址与本机的IP地址为同一网段，因此Host\_1以广播形式发送ARP请求报文，请求Host\_2的MAC地址。但是，由于两台主机处于不同的物理网络（不同广播域）中，Host\_2无法收到Host\_1的ARP请求报文，因此也就无法应答。

通过在Switch上启用路由式Proxy ARP功能，可以解决此问题。启用路由式Proxy ARP后，Switch收到ARP请求报文后，Switch会查找路由表。由于Host\_2与Switch直连，因此Switch上存在到Host\_2的路由表项。Switch使用自己的MAC地址给Host\_1发送ARP应答报文。Host\_1将以Switch的MAC地址进行数据转发。此时，Switch相当于Host\_2的代理。如上图所示，Host\_1上的ARP表项中到目的地址Host\_2的IP地址对应的MAC地址为Switch的VLANIF10接口的MAC地址。

**VLAN内Proxy ARP**

如果两个用户属于相同的VLAN，但VLAN内配置了端口隔离。此时用户间需要三层互通，可以在关联了VLAN的接口上启动VLAN内Proxy ARP功能。

如下图所示，Host\_1和Host\_2是Switch设备下的两个用户。连接Host\_1和Host\_2的两个接口在Switch属于同一个VLAN10。

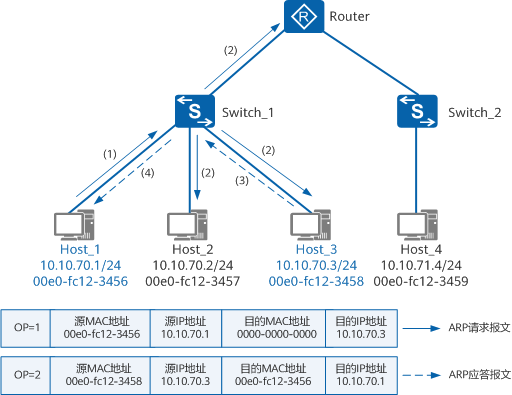


由于Host\_1和Host\_2属于不同的Sub-VLAN，Host\_1和Host\_2不能直接实现二层互通。

如果Switch上使能了VLAN间Proxy ARP功能，可以使Host\_1和Host\_2实现三层互通。Switch的接口在接收到目的地址不是自己的ARP请求报文后，并不立即丢弃该报文，而是查找ARP表项（包括动态学习的ARP表项和静态配置的ARP表项）。如果存在Host\_2的ARP表项，则将自己的MAC地址发送给Host\_1，并将Host\_1发送给Host\_2的报文代为转发。实际上此时Switch相当于Host\_2的代理。

## 4.4 ARP是如何进行地址解析的？

动态ARP通过广播ARP请求和单播ARP应答这两个过程完成地址解析。



ARP地址解析过程

当需要通信的两台主机处于同一网段时，如上图中的Host\_1和Host\_3，Host\_1要向Host\_3发送数据。

1. 首先，Host\_1会查找自己本地缓存的ARP表，确定是否包含Host\_3对应的ARP表项。如果Host\_1在ARP表中找到了Host\_3对应的MAC地址，则Host\_1直接利用ARP表中的MAC地址，对数据报文进行帧封装，并将数据报文发送给Host\_3。如果Host\_1在ARP表中找不到Host\_3对应的MAC地址，则先缓存该数据报文，并以广播方式发送一个ARP请求报文。如上图中所示，OP字段为1表示该报文为ARP请求报文，ARP请求报文中的源MAC地址和源IP地址为Host\_1的MAC地址和IP地址，目的MAC地址为全0的MAC地址，目的IP地址为Host\_3的IP地址。
2. Switch\_1收到ARP请求报文后，将该ARP请求报文在同一广播域内转发。
3. 同一广播域内的主机Host\_2和Host\_3都能接收到该ARP请求报文，但只有被请求的主机（即Host\_3）会对该ARP请求报文进行处理。Host\_3比较自己的IP地址和ARP请求报文中的目的IP地址，当两者相同时进行如下处理：将ARP请求报文中的源IP地址和源MAC地址（即Host\_1的IP地址和MAC地址）存入自己的ARP表中。之后以单播方式发送ARP应答报文给Host\_1，ARP应答报文内容如上图中所示，OP字段为2表示该报文为ARP应答报文，源MAC地址和源IP地址为Host\_3的MAC地址和IP地址，目的MAC地址和目的IP地址为Host\_1的MAC地址和IP地址。
4. Switch\_1收到ARP应答报文后，将该ARP应答报文转发给Host\_1。Host\_1收到ARP应答报文后，将Host\_3的MAC地址加入到自己的ARP表中以用于后续报文的转发，同时将数据报文进行帧封装，并将数据报文发送给Host\_3。

当需要通信的两台主机处于不同网段时，如上图中的Host\_1和Host\_4，Host\_1上已经配置缺省网关，Host\_1首先会发送ARP请求报文，请求网关Router的IP地址对应的MAC地址。Host\_1收到ARP应答报文后，将数据报文封装并发给网关，再由网关将数据报文发送给目的主机Host\_4。Host\_1学习网关IP地址对应的ARP表项的过程，以及网关设备学习Host\_4的IP地址对应的ARP表项的过程与上述同网段主机Host\_1和Host\_3之间进行ARP地址解析的过程类似。

1. **ARP老化机制**

如上图中所示，如果每次Host\_1和Host\_3通信前都要发送一个广播的ARP请求报文，会极大的增加网络负担。而且同广播域的所有设备都需要接收和处理这个广播的ARP请求报文，也极大的影响了网络中设备的运行效率。为了解决以上问题，每台主机或设备上都维护着一个高速缓存，这是ARP高效运行的一个关键。在这个高速缓存中，存放主机或设备最近学习到的IP地址到MAC地址的映射关系，即动态ARP表项。

主机或设备每次发送报文时，会先在本地高速缓存中查找目的IP地址所对应的MAC地址。如果高速缓存中有对应的MAC地址，主机或设备不会再发送ARP请求报文，而是直接将报文发至这个MAC地址；如果高速缓存中没有对应的MAC地址，主机或设备才会广播发送ARP请求报文，进行ARP地址解析。

一方面由于高速缓存的容量限制，另一方面为了保证高速缓存中ARP表项的准确性，设备会对动态ARP表项进行老化和更新。

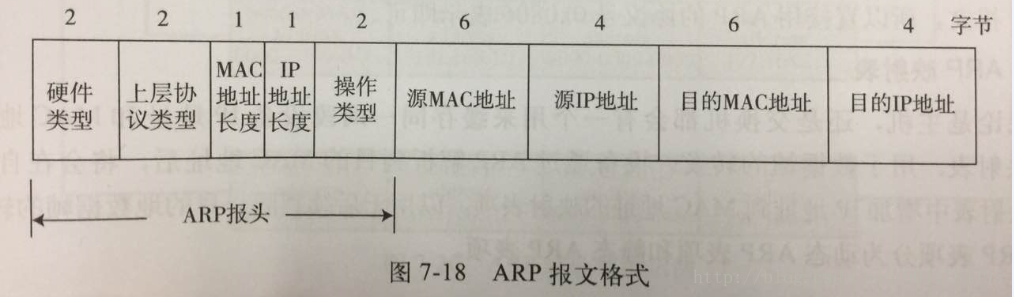
动态ARP表项的老化参数有：老化超时时间、老化探测次数和老化探测模式。设备上动态ARP表项到达老化超时时间后，设备会发送老化探测报文（即ARP请求报文），如果能收到ARP应答报文，则更新该动态ARP表项，本次老化探测结束；如果超过设置的老化探测次数后仍没有收到ARP应答报文，则删除该动态ARP表项，本次老化探测结束。

设备发送的老化探测报文可以是单播报文，也可以是广播报文。缺省情况下，设备只在最后一次发送ARP老化探测报文是广播模式，其余均为单播模式发送。当对端设备MAC地址不变时，可以配置接口以单播模式发送ARP老化探测报文。

当接口Down时设备会立即删除相应的动态ARP表项。

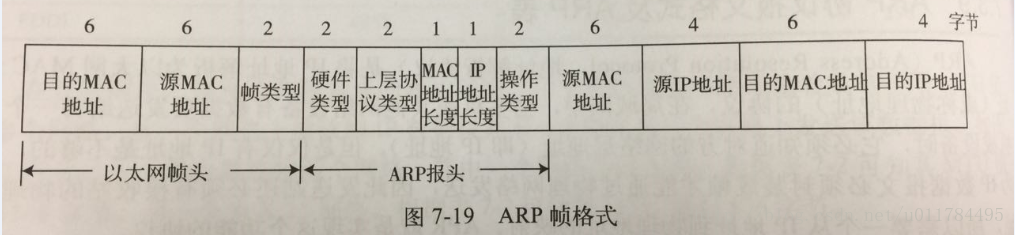
## 4.5 ARP报文的帧格式及其含义

ARP是一个独立的三层协议，所以ARP报文在向数据链路层传输时不需要经过IP协议的封装，而是直接生成自己的报文，其中包括ARP报头，到数据链路层后再由对应的数据链路层协议（如以太网协议）进行封装。ARP报文分为ARP请求和ARP应答报文两种，它们的报文格式可以统一为下图所示。



* 硬件类型：占两字节，表示ARP报文可以在哪种类型的网络上传输，值为1时表示为以太网地址。
* 上层协议类型：占两字节，表示硬件地址要映射的协议地址类型，映射IP地址时的值为0x0800。
* MAC地址长度：占一字节，标识MAC地址长度，以字节为单位，此处为6。
* IP协议地址长度：占一字节，标识IP得知长度，以字节为单位，此处为4。
* 操作类型：占2字节，指定本次ARP报文类型。1标识ARP请求报文，2标识ARP应答报文。
* 源MAC地址：占6字节，标识发送设备的硬件地址。
* 源IP地址：占4字节，标识发送方设备的IP地址。
* 目的MAC地址：占6字节，表示接收方设备的硬件地址，在请求报文中该字段值全为0，即00-00-00-00-00-00，表示任意地址，因为现在不知道这个MAC地址。
* 目的IP地址：占4字节，表示接受方的IP地址。

ARP报文不是直接在网络层上发送的，它还是需要向下传输到数据链路层，所以当ARP报文传输到数据链路层之后，需要再次进行封装。以以太网为例，ARP报文传输到以太网数据链路层后会形成ARP帧。ARP帧如下图所示，他就是在ARP报文前面加了一个以太网帧头。



* 目的MAC地址：占6字节，如果是ARP请求帧，因为它是一个广播帧，所以要填上广播MAC地址（FF-FF-FF-FF-FF-FF），其目标主机是网络上的所有主机。
* 源MAC地址：占6字节，这是发送ARP帧的节点MAC地址。
* 帧类型：占两字节，这里用来标识帧封装的上层协议，因为本帧的数据部分是ARP报文，所以直接用ARP的协议号0x0806表示就可以了。

## 4.6 实操linux下ARP的相关命令，以及命令结果的具体含义

ip neigh添加ARP\NDP静态记录、删除记录、查看记录

**邻居管理存在状态**

1、NONE： 初始状态。当一个新的路由缓存条目被创建时，arp\_bind\_neighbour()函数被调用.如果找不到相匹配的ARP缓存条目, neigh\_alloc()将创建一个新的ARP缓存条目并设置状态为NUD\_NONE.

2、INCOMPLETE：未完成状态。当第一个包发送需要新的ARP缓存条目neigh\_resolve\_output()被调用.最后实际调用的是neigh\_event\_send()函数,它发送一个ARP请求并置传输状NUD\_INCOMPLETE.

3、REACHABLE：可达状态。当收到ARP响应时,邻居表进入NUD\_REACHABLE状态.

4、STALE：过期状态。当定时器触发neigh\_periodic\_timer()函数扫描到邻居表条目的最后时间超过可达时间参数时，它设置条目状态为过期状态.

5、DELAY：延迟状态。当在NUD\_STALE状态下有报文需要发送时，调用neigh\_resolve\_output()函数发送数据，在发送报文同时，将状态迁移到这个状态，同时启动一个定时器。

6、PROBE：探测状态。NUD\_DELAY的定时器到时调用neigh\_timer\_handler()迁移到这个状态，这个状态下，发送ARP请求并等待响应。

7、FAILED： 失败状态。连续重发多次都没有响应

8、NOARP：不需要解析，如PPP接入

9、PERMANENT：静态ARP

10、IN\_TIMER：定时器状态。表示邻居系统正在运行一个定时器

11、VALID：有效状态。表示邻居表有一个可用地址

12、CONNECTED：连接状态。表示可以直接根据邻居系统发送数据包

**查看、添加和修改**

1、查看：ip neigh show

既可以显示ipv4也可以显示ipv6

2、设置静态记录：

sudo ip neigh add fe80::cc47:88ff:e5fd:2ea lladdr 00:12:ac:3d:a1:11 dev eth0

3、删除记录:

ip neigh del 192.168.36.1 dev eth0

ip neigh help

Usage: ip neigh { add | del | change | replace } { ADDR [ lladdr LLADDR ]

[ nud { permanent | noarp | stale | reachable } ]

| proxy ADDR } [ dev DEV ]

ip neigh {show|flush} [ to PREFIX ] [ dev DEV ] [ nud STATE ]

# IPv4基础系列协议之DHCP

## 5.1 什么是DHCP？

动态主机配置协议，是一种网络管理协议，用于集中对用户IP地址进行动态管理和配置，工作在osi的应用层。采用客户端/服务器通信模式，由客户端（DHCP Client）向服务器（DHCP Server）提出配置申请，DHCP Server为网络上的每个设备动态分配IP地址、子网掩码、默认网关地址，域名服务器（DNS）地址和其他相关配置参数。

## 5.2 为什么要用DHCP？

* 准确的IP配置：IP地址配置参数必须准确，并且在处理“ 192.168.XXX.XXX”之类的输入时，很容易出错。另外印刷错误通常很难解决，使用DHCP服务器可以最大程度地降低这种风险。
* 减少IP地址冲突：每个连接的设备都必须有一个IP地址。但是，每个地址只能使用一次，重复的地址将导致无法连接一个或两个设备的冲突。当手动分配地址时，尤其是在存在大量仅定期连接的端点（例如移动设备）时，可能会发生这种情况。DHCP的使用可确保每个地址仅使用一次。
* IP地址管理的自动化：如果没有DHCP，网络管理员将需要手动分配和撤消地址。跟踪哪个设备具有什么地址可能是徒劳的，因为几乎无法理解设备何时需要访问网络以及何时需要离开网络。DHCP允许将其自动化和集中化，因此网络专业人员可以从一个位置管理所有位置。
* 高效的变更管理：DHCP的使用使更改地址，范围或端点变得非常简单。例如，组织可能希望将其IP寻址方案从一个范围更改为另一个范围。DHCP服务器配置有新信息，该信息将传播到新端点。同样，如果升级并更换了网络设备，则不需要网络配置。

## 5.3 DHCP是如何工作的？

DHCP协议采用UDP作为传输协议，DHCP客户端发送请求消息到DHCP服务器的67号端口，DHCP服务器回应应答消息给DHCP客户端的68号端口。

只有跟DHCP客户端在同一个网段的DHCP服务器才能收到DHCP客户端广播的DHCP DISCOVER报文。当DHCP客户端与DHCP服务器不在同一个网段时，必须部署DHCP中继来转发DHCP客户端和DHCP服务器之间的DHCP报文。在DHCP客户端看来，DHCP中继就像DHCP服务器；在DHCP服务器看来，DHCP中继就像DHCP客户端。

PS：(1)直接广播 (能够跨越不同网络的广播)；本地广播 (只在本地网络中的广播)；

(2)往往一个网段中，网关、DNS、Web、DHCP等服务器地址都是同一个；且在一个网段中，一些重要主机一般需配置静态ip地址，因此DHCP服务器的地址池需要预留一段以供静态配置使用。

## 5.4不同场景下的DHCP工作原理

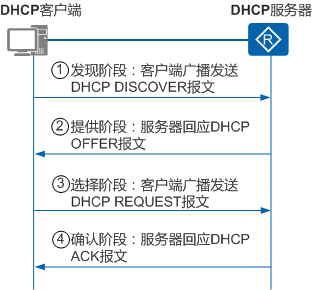
**无中继场景时DHCP客户端首次接入网络**

①**发现阶段：客户端广播发送DHCP DISCOVER报文（**0.0.0.0作为源IP地址，然后广播地址255.255.255.255作为目的地址+客户端MAC地址，发给同一网段的所有设备，包括DHCP服务器或中继；若无回复，则等待1s后继续发送3个DISCOVER报文，等待时延分别为9秒、13秒和16秒加上一个长度为0～1000毫秒之间的随机时延。如果仍然无法联系DHCP服务器，则认为自动获取IP地址失败，默认情况下将随机使用APIPA（自动专有IP地址，169.254.0.0/16）中定义的未被其他客户使用的IP地址，子网掩码为255.255.0.0，但是不会配置默认网关和其他TCP/IP选项，因此只能和同子网的使用APIPA地址的客户端进行通讯。**）**

**②提供阶段：服务器广播回应DHCP OFFER报文（**源IP地址为DHCP服务器的IP地址，而广播地址255.255.255.255作为目的地址--因为没客户端地址，此消息包括【客户端MAC地址；DHCP服务器提供的客户端IP地址、客户端子网掩码、DNS服务器、网关、租约期限等；】DHCP服务器选择跟接收DHCP DISCOVER报文接口的IP地址处于同一网段的地址池，并且从中选择一个可用的IP地址，然后通过DHCP OFFER报文发送给DHCP客户端，等待客户端回应时长为16秒；**）**

**③选择阶段：客户端广播发送 DHCP REQUEST报文（**0.0.0.0作为源IP地址，广播地址255.255.255.255作为目的地址，消息中包含了【DHCP客户端的MAC地址、接受的租约中的IP地址、提供此租约的DHCP服务器地址等】，若有多个DHCP服务器回应DHCP OFFER报文，则DHCP客户端一般只接收第一个收到的DHCP OFFER报文，然后以广播方式发送DHCP REQUEST报文，该报文中包含客户端想选择的DHCP服务器标识符（即Option50，填充了接收的DHCP OFFER报文中yiaddr字段的IP地址，DHCP客户端广播发送DHCP REQUEST报文通知所有的DHCP服务器，它将选择某个DHCP服务器提供的IP地址，其他DHCP服务器可以重新将曾经分配给客户端的IP地址分配给其他客户端。**）**

**④确认阶段：服务器回应DHCP ACK报文（**单播，当DHCP服务器收到DHCP客户端发送的DHCP REQUEST报文后，DHCP服务器回应DHCP ACK报文，表示DHCP REQUEST报文中请求的IP地址（Option50填充的）分配给客户端使用。如果DHCP客户端的操作系统为Windows版本，当DHCP客户端接收到DHCP ACK广播消息后，会向网络发出三个针对此IP地址的ARP解析请求以执行冲突检测，确认网络上没有其他主机使用DHCP服务器提供的IP地址，从而避免IP地址冲突。如果发现该IP已经被其他主机所使用（有其他主机应答此ARP解析请求），则DHCP客户端则会广播发送（因为它仍然没有有效的IP地址）DHCP DECLINE消息给DHCP服务器拒绝此IP地址租约，然后重新发起DHCP DISCOVER进程。此时，在DHCP服务器管理控制台中，会显示此IP地址为BAD\_ADDRESS。如果没有其他主机使用此IP地址，则DHCP客户端的TCP/IP使用租约中提供的IP地址完成初始化，从而可以和其他网络中的主机进行通讯。至于其他TCP/IP选项，如DNS服务器和WINS服务器等，本地手动配置将覆盖从DHCP服务器获得的值。**）**



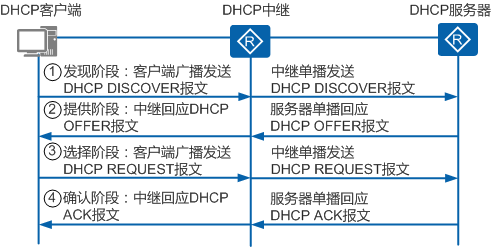
DHCP客户端收到DHCP ACK报文，会广播发送免费ARP报文，探测本网段是否有其他终端使用服务器分配的IP地址，如果在指定时间内没有收到回应，表示客户端可以使用此地址。如果收到了回应，说明有其他终端使用了此地址，客户端会向服务器发送DHCP DECLINE报文，并重新向服务器请求IP地址，同时，服务器会将此地址列为冲突地址。当服务器没有空闲地址可分配时，再选择冲突地址进行分配，尽量减少分配出去的地址冲突。

当DHCP服务器收到DHCP客户端发送的DHCP REQUEST报文后，如果DHCP服务器由于某些原因（例如协商出错或者由于发送REQUEST过慢导致服务器已经把此地址分配给其他客户端）无法分配DHCP REQUEST报文中Option50填充的IP地址，则发送DHCP NAK报文作为应答，通知DHCP客户端无法分配此IP地址。DHCP客户端需要重新发送DHCP DISCOVER报文来申请新的IP地址。

**有中继场景时DHCP客户端首次接入网络**

有DHCP中继的场景中，首次接入网络的DHCP客户端和DHCP服务器的工作原理与无中继场景时DHCP客户端首次接入网络的工作原理相同。主要差异是DHCP中继在DHCP服务器和DHCP客户端之间转发DHCP报文，以保证DHCP服务器和DHCP客户端可以正常交互。下面仅针对DHCP中继的工作原理进行介绍。

如下图所示，在部署DHCP中继的场景下，首次接入网络DHCP客户端与DHCP服务器的报文交互过程。



①**发现阶段**：DHCP中继接收到DHCP客户端广播发送的DHCP DISCOVER报文后，进行如下处理：

1. 检查DHCP报文中的hops字段，如果大于16，则丢弃DHCP报文；否则，将hops字段加1（表明经过一次DHCP中继），并继续下面的操作。
2. 检查DHCP报文中的giaddr字段。如果是0，将giaddr字段设置为接收DHCP DISCOVER报文的接口IP地址。如果不是0，则不修改该字段，继续下面的操作。
3. 将DHCP报文的目的IP地址改为DHCP服务器或下一跳中继的IP地址，源地址改为中继连接客户端的接口地址，通过路由转发将DHCP报文单播发送到DHCP服务器或下一跳中继。

如果DHCP客户端与DHCP服务器之间存在多个DHCP中继，后面的中继接收到DHCP DISCOVER报文的处理流程同前面所述。

②**提供阶段**：DHCP服务器接收到DHCP DISCOVER报文后，选择与报文中giaddr字段为同一网段的地址池，并为客户端分配IP地址等参数，然后向giaddr字段标识的DHCP中继单播发送DHCP OFFER报文。DHCP中继收到DHCP OFFER报文后，会进行如下处理：

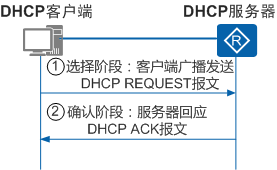
1. 检查报文中的giaddr字段，如果不是接口的地址，则丢弃该报文；否则，继续下面的操作。
2. DHCP中继检查报文的广播标志位。如果广播标志位为1，则将DHCP OFFER报文广播发送给DHCP客户端；否则将DHCP OFFER报文单播发送给DHCP客户端。

③**选择阶段：**中继接收到来自客户端的DHCP REQUEST报文的处理过程同无中继场景下的选择阶段。

④**确认阶段：**中继接收到来自服务器的DHCP ACK报文的处理过程同无中继场景下的确认阶段。

**DHCP客户端重用曾经使用过的地址**

DHCP客户端非首次接入网络时，可以重用曾经使用过的地址。如下图所示，DHCP客户端与DHCP服务器交互DHCP报文，以重新获取之前使用的IP地址等网络参数，该过程称为两步交互。

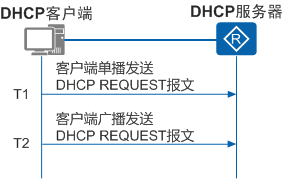


①**选择阶段：**客户端广播发送包含前一次分配的IP地址的DHCP REQUEST报文，报文中的Option50（请求的IP地址选项）字段填入曾经使用过的IP地址。

②**确认阶段：**DHCP服务器收到DHCP REQUEST报文后，根据DHCP REQUEST报文中携带的MAC地址来查找有没有相应的租约记录，如果有则返回DHCP ACK报文，通知DHCP客户端可以继续使用这个IP地址。否则，保持沉默，等待客户端重新发送DHCP DISCOVER报文请求新的IP地址。

**DHCP客户端更新租期**

DHCP服务器采用动态分配机制给客户端分配IP地址时，分配出去的IP地址有租期限制。DHCP客户端向服务器申请地址时可以携带期望租期，服务器在分配租期时把客户端期望租期和地址池中租期配置比较，分配其中一个较短的租期给客户端。租期到期或者客户端下线释放地址后，服务器会收回该IP地址，收回的IP地址可以继续分配给其他客户端使用。这种机制可以提高IP地址的利用率，避免客户端下线后IP地址继续被占用。如果DHCP客户端希望继续使用该地址，需要更新IP地址的租期（如延长IP地址租期）。



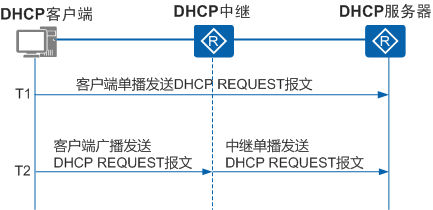
①**当租期达到50%（T1）时**，DHCP客户端会自动以单播的方式向DHCP服务器发送DHCP REQUEST报文，请求更新IP地址租期。如果收到DHCP服务器回应的DHCP ACK报文，则租期更新成功（即租期从0开始计算）；如果收到DHCP NAK报文，则重新发送DHCP DISCOVER报文请求新的IP地址。

②**当租期达到87.5%（T2）时**，如果仍未收到DHCP服务器的应答，DHCP客户端会自动以广播的方式向DHCP服务器发送DHCP REQUEST报文，请求更新IP地址租期。如果收到DHCP服务器回应的DHCP ACK报文，则租期更新成功（即租期从0开始计算）；如果收到DHCP NAK报文，则重新发送DHCP DISCOVER报文请求新的IP地址。

③**如果租期时间到**时都没有收到服务器的回应，客户端停止使用此IP地址，重新发送DHCP DISCOVER报文请求新的IP地址。

客户端在租期时间到之前，如果用户不想使用分配的IP地址（例如客户端网络位置需要变更），会触发DHCP客户端向DHCP服务器发送DHCP RELEASE报文，通知DHCP服务器释放IP地址的租期。DHCP服务器会保留这个DHCP客户端的配置信息，将IP地址列为曾经分配过的IP地址中，以便后续重新分配给该客户端或其他客户端。客户端可以通过发送DHCP INFORM报文向服务器请求更新配置信息。

如下图所示，部署DHCP中继时，更新租期的过程与上述过程相似。



**DHCP使用场景**

DHCP提供了两种地址分配机制，网络管理员可以根据网络需求为不同的主机选择不同的分配策略。

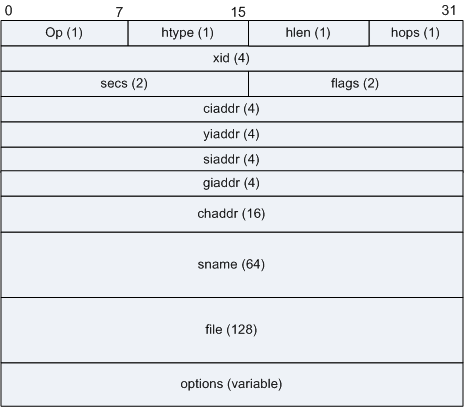
* 动态分配机制：通过DHCP为主机分配一个有使用期限的IP地址。

DHCP使用了租期的概念，或称为设备IP地址的有效期。租用时间是不定的，主要取决于用户在某地连接Internet需要多久，这种分配机制适用于主机需要临时接入网络或者空闲地址数小于网络主机总数且主机不需要永久连接网络的场景。

* 静态分配机制：网络管理员通过DHCP为指定的主机分配固定的IP地址。

相比手工静态配置IP地址，通过DHCP方式静态分配机制避免人工配置发生错误，方便管理员统一维护管理。

## 5.5 DHCP报文格式



DHCP报文中各字段的含义：

**Op**：表示报文的类型，取值为1或2，含义如下：

1：客户端请求报文

2：服务器响应报文

**Htype**：表示硬件类型。不同的硬件类型取值不同，最常见的值是1，表示以太网。

**Hlen**：表示硬件地址的长度。对于以太网，该值为6。

**Hops**：表示当前的DHCP报文经过的DHCP中继的数目。该字段由客户端设置为0，每经过一个DHCP中继时，该字段加1。

**Xid**：事务ID，由客户端选择的一个随机数，被服务器和客户端用来在它们之间交流请求和响应，客户端用它对请求和应答进行匹配。该ID由客户端设置并由服务器返回，为32位整数。

**Secs**：表示客户端从开始获取地址或地址续租更新后所用的时间，单位是秒。

**Flags**：只有标志字段的最高位才有意义，其余的位均被置为0。

最左边的字段被解释为广播响应标志位，内容如下所示：

0：客户端请求服务器以单播形式发送响应报文

1：客户端请求服务器以广播形式发送响应报文

**Ciaddr**：客户端的IPv4地址。只有客户端是Bound、Renew、Rebinding状态，并且能响应ARP请求时，才能被填充。

**Yiaddr**：表示服务器分配给客户端的IPv4地址。

**Siaddr**：表明DHCP协议流程的下一个阶段要使用的服务器的IPv4地址。

**Giaddr**：表示第一个DHCP中继的IPv4地址。当客户端发出DHCP请求时，如果服务器和客户端不在同一个网段，那么第一个DHCP中继在将DHCP请求报文转发给DHCP服务器时，会把自己的IPv4地址填入此字段，DHCP服务器会根据此字段来判断出客户端所在的网段地址，从而选择合适的地址池，为客户端分配该网段的IPv4地址。

服务器还会根据此地址将响应报文发送给此DHCP中继，再由DHCP中继将此报文转发给客户端。若在到达DHCP服务器前经过了多个DHCP中继，该字段作为客户端所在的网段的标记，填充了第一个DHCP中继的IPv4地址后不会再变更，只是每经过一个DHCP中继，hops字段的数值会加1。

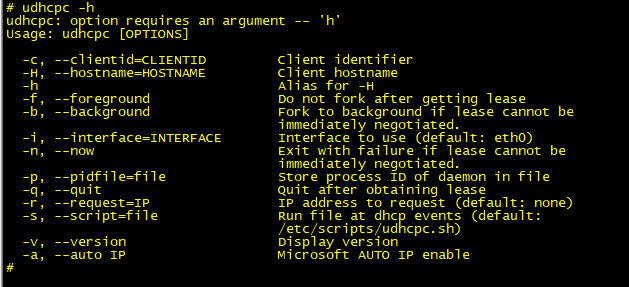
**Chaddr**：该字段表示客户端的MAC地址，此字段与前面的“Hardware Type”和“Hardware Length”保持一致。当客户端发出DHCP请求时，将自己的硬件地址填入此字段。对于以太网，当“Hardware Type”和“Hardware Length”分别为“1”和“6”时，此字段必须填入6字节的以太网MAC地址。

**Sname**：该字段表示客户端获取配置信息的服务器名字。此字段由DHCP Server填写，是可选的。如果填写，必须是一个以0结尾的字符串。

**File**：该字段表示客户端的启动配置文件名。此字段由DHCP Server填写，是可选的，如果填写，必须是一个以0结尾的字符串。

**Options**：该字段表示DHCP的选项字段，至少为312字节，格式为"代码+长度+数据"。DHCP通过此字段包含了服务器分配给终端的配置信息，如网关IPv4地址，DNS服务器的IPv4地址，客户端可以使用IPv4地址的有效租期等信息。

## 5.6路由器dhcpc命令的可选项参数的含义



1. \*\*-c, --clientid=CLIENTID\*\*

- 示例：`udhcpc -c MyClientID -i eth0`

- 效果：此命令会启动 `udhcpc` 并设置客户端标识符为 `MyClientID`，这有助于在 DHCP 服务器日志中追踪特定的客户端。

2. \*\*-H, --hostname=HOSTNAME\*\*

- 示例：`udhcpc -H MyHostName -i eth0`

- 效果：启动 `udhcpc` 并设置客户端的主机名为 `MyHostName`。当 DHCP 服务器分配 IP 时，它可能会使用该主机名。

3. \*\*-h\*\*

- 示例：`udhcpc -h MyHostName -i eth0`

- 效果：与 `-H` 相同，设置主机名为 `MyHostName`。

4. \*\*-f, --foreground\*\*

- 示例：`udhcpc -f -i eth0`

- 效果：使 `udhcpc` 在前台运行，它不会在获取到 IP 租约后转为后台进程。

5. \*\*-b, --background\*\*

- 示例：`udhcpc -b -i eth0`

- 效果：如果 `udhcpc` 无法立即获取 IP 租约，它将在后台继续尝试直到成功。对

6. \*\*-i, --interface=INTERFACE\*\*

- 示例：`udhcpc -i wlan0`

- 效果：指定 `udhcpc` 使用 `wlan0` 接口来获取 IP 配置信息。

7. \*\*-n, --now\*\*

- 示例：`udhcpc -n -i eth0`

- 效果：如果 `udhcpc` 无法立即获得 IP 租约，它将立即退出并不再尝试。

8. \*\*-p, --pidfile=file\*\*

- 示例：`udhcpc -p /var/run/udhcpc.pid -i eth0`

- 效果：将 `udhcpc` 的进程 ID 写入 `/var/run/udhcpc.pid` 文件中。

9. \*\*-q, --quit\*\*

- 示例：`udhcpc -q -i eth0`

- 效果：一旦 `udhcpc` 成功获取 IP 租约，就立即退出。

10. \*\*-r, --request=IP\*\*

- 示例：`udhcpc -r 192.168.1.10 -i eth0`

- 效果：请求特定的 IP 地址 `192.168.1.10`。DHCP 服务器可能会尝试分配这个地址给客户端，如果可用的话。

11. \*\*-s, --script=file\*\*

- 示例：`udhcpc -s /etc/scripts/my\_udhcpc.sh -i eth0`

- 效果：当 DHCP 事件发生时（如获取 IP 或租约更新），运行 `/etc/scripts/my\_udhcpc.sh` 脚本来执行自定义操作。

12. \*\*-v, --version\*\*

- 示例：`udhcpc -v`

- 效果：显示 `udhcpc` 的版本信息。

13. \*\*-a, --auto IP\*\*

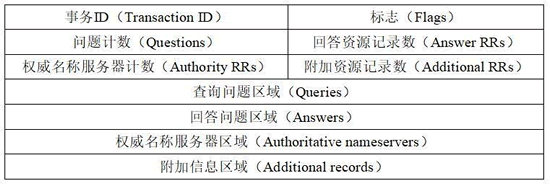
- 示例：`udhcpc -a 192.168.1.10 -i eth0`

- 效果：启用 Microsoft 自动私有 IP 寻址功能，并请求特定的 IP 地址 `192.168.1.10`。如果无法通过 DHCP 获取地址，则自动分配一个IP地址。

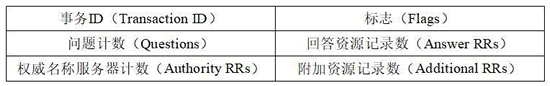
# IPv4基础系列协议之DNS（基于UDP）

## 6.1 DNS的报文帧格式

DNS 分为查询请求和查询响应，请求和响应的报文结构基本相同。DNS 报文格式如图所示



1. 报文首部



事务 ID：DNS 报文的 ID 标识。对于请求报文和其对应的应答报文，该字段的值是相同的。通过它可以区分 DNS 应答报文是对哪个请求进行响应的。

Flags：DNS 报文中的标志字段。

问题计数：DNS 查询请求的数目。

回答资源记录数：DNS 响应的数目。

权威名称服务器计数：权威名称服务器的数目。

附加资源记录数：额外的记录数目（权威名称服务器对应 IP 地址的数目）

基础结构部分中的标志字段又分为若干个字段，如图所示。

IMG_256

标志字段中每个字段的含义如下：

**QR（Response）**：查询请求/响应的标志信息。查询请求时，值为 0；响应时，值为 1。

**Opcode**：操作码。其中，0 表示标准查询；1 表示反向查询；2 表示服务器状态请求。

**AA（Authoritative）**：授权应答，该字段在响应报文中有效。值为 1 时，表示名称服务器是权威服务器；值为 0 时，表示不是权威服务器。

**TC（Truncated）**：表示是否被截断。值为 1 时，表示响应已超过 512 字节并已被截断，只返回前 512 个字节。

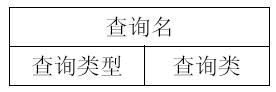
**RD（Recursion Desired）**：期望递归。该字段能在一个查询中设置，并在响应中返回。该标志告诉名称服务器必须处理这个查询，这种方式被称为一个递归查询。如果该位为 0，且被请求的名称服务器没有一个授权回答，它将返回一个能解答该查询的其他名称服务器列表。这种方式被称为迭代查询。

**RA（Recursion Available）**：可用递归。该字段只出现在响应报文中。当值为 1 时，表示服务器支持递归查询。

**Z**：保留字段，在所有的请求和应答报文中，它的值必须为 0。

**rcode（Reply code）**：返回码字段，表示响应的差错状态。当值为 0 时，表示没有错误；当值为 1 时，表示报文格式错误（Format error），服务器不能理解请求的报文；当值为 2 时，表示域名服务器失败（Server failure），因为服务器的原因导致没办法处理这个请求；当值为 3 时，表示名字错误（Name Error），只有对授权域名解析服务器有意义，指出解析的域名不存在；当值为 4 时，表示查询类型不支持（Not Implemented），即域名服务器不支持查询类型；当值为 5 时，表示拒绝（Refused），一般是服务器由于设置的策略拒绝给出应答，如服务器不希望对某些请求者给出应答。

1. 问题部分（问题查询区域Queries）



该部分中每个字段含义如下：

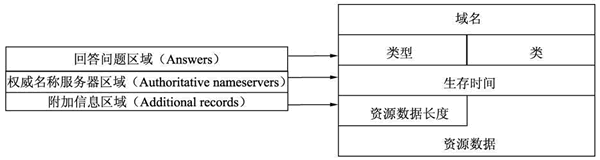
**查询名**：一般为要查询的域名，有时也会是 IP 地址，用于反向查询。

**查询类型**：DNS 查询请求的资源类型。通常查询类型为 A 类型，表示由域名获取对应的 IP 地址。

**查询类**：地址类型，通常为互联网地址，值为 1

1. 资源记录部分

资源记录部分是指 DNS 报文格式中的最后三个字段，包括回答问题区域字段、权威名称服务器区域字段、附加信息区域字段。这三个字段均采用一种称为资源记录的格式，格式如图所示。



资源记录格式中每个字段含义如下：

**域名**：DNS 请求的域名。

**类型**：资源记录的类型，与问题部分中的查询类型值是一样的。

**类**：地址类型，与问题部分中的查询类值是一样的。

**生存时间**：以秒为单位，表示资源记录的生命周期，一般用于当地址解析程序取出资源记录后决定保存及使用缓存数据的时间。它同时也可以表明该资源记录的稳定程度，稳定的信息会被分配一个很大的值。

**资源数据长度**：资源数据的长度。

**资源数据**：表示按查询段要求返回的相关资源记录的数据。

资源记录部分只有在 DNS 响应包中才会出现

## 6.2 DNS的原理

定义：域名系统DNS(Domain Name System)是因特网使用的命名系统，用来把便于人们使用的机器名字转换成为IP地址

方法：域名到IP地址的解析是由分布在因特网上的许多域名服务器程序共同完成的。域名服务器程序在专设的结点上运行，而人们也常把运行域名服务器程序的机器称为域名服务器

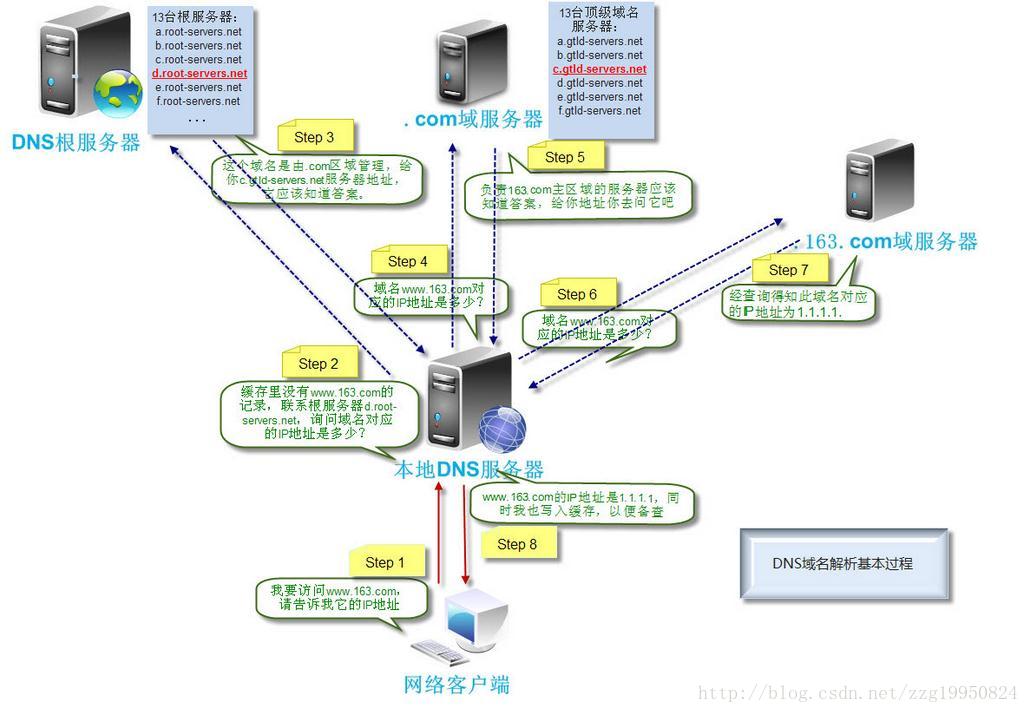
因为因特网规模很大，所以整个因特网只使用一个域名服务器是不可行的。因此，早在1983年因特网开始采用层次树状结构的命名方法，并使用分布式的域名系统DNS。并采用客户服务器方式。DNS使大多数名字都在本地解析(resolve)，仅有少量解析需要在因特网上通信，因此DNS系统的效率很高。由于DNS是分布式系统，即使单个计算机除了故障，也不会妨碍整个DNS系统的正常运行。

域名到IP地址的解析过程的要点如下：

当某一个应用需要把主机名解析为IP地址时，该应用进程就调用解析程序，并称为DNS的一个客户，把待解析的域名放在DNS请求报文中，以UDP用户数据报方式发给本地域名服务器。

本地域名服务器在查找域名后，把对应的IP地址放在回答报文中返回。应用程序获得目的主机的IP地址后即可进行通信。

若本地域名服务器不能回答该请求，则此域名服务器就暂时称为DNS的另一个客户，并向其他域名服务器发出查询请求。



# IPv4基础系列协议之ICMP（网络层）

## 7.1 什么是ICMP？

因特网控制报文协议ICMP（Internet Control Message Protocol）是一个差错报告机制，是TCP/IP协议簇中的一个重要子协议，通常被IP层或更高层协议（TCP或UDP）使用，属于网络层协议，主要用于在IP主机和路由器之间传递控制消息，用于报告主机是否可达、路由是否可用等。这些控制消息虽然并不传输用户数据，但是对于收集各种网络信息、诊断和排除各种网络故障以及用户数据的传递具有至关重要的作用。

## 7.2 为什么需要ICMP？

在数据传输的过程中，IP提供尽力而为的服务，指为了把数据包发送到目的地址尽最大努力。它并不对目的主机是否收到数据包进行验证，无法进行流量控制和差错控制。因此在数据包传输过程中，产生各种错误在所难免。为了更有效地转发IP数据包和提高数据包交付成功的机会，ICMP应运而生。使用ICMP，当网络中数据包传输出现问题时，主机或设备就会向上层协议报告差错情况和提供有关异常情况的报告，使得上层协议能够通过自己的差错控制程序来判断通信是否正确，以进行流量控制和差错控制，从而保证服务质量。

## 7.3 ICMP如何工作？

当数据包处理过程出现差错时，ICMP向数据包的源端设备报告这个差错，它既不会纠正这个差错，也不会通知中间的网络设备。因为ICMP报文被封装在IP数据包内部，作为IP数据包的数据部分通过互联网传递。IP数据包中的字段包含源端和最终的目的端，并没有记录报文在网络传递中的全部路径（除非IP数据包中设置了路由记录选项）。因此当设备检测到差错时，它无法通知中间的网络设备，只能向源端发送差错报告。

源端在收到差错报告后，它虽然不能判断差错是由中间哪个网络设备所引起的，但是却可以根据ICMP报文确定发生错误的类型，并确定如何才能更好地重发传递失败的数据包。

## 7.4 ICMP的典型应用？

IP数据报及其他应用程序通过ICMP报文可以实现多种应用，其中Ping程序和Tracert（Traceroute）程序最为常见。此外，在网络管理和监测中，网络质量分析NQA（Network Quality Analysis）技术更加充分应用了ICMP。

* Ping

Ping程序是最常见的用于检测IPv4和IPv6网络设备是否可达的调试手段，它使用ICMP的echo信息来确定：

远程设备是否可达；

与远程主机通信的来回旅程（round-trip）的延迟；

报文包的丢失情况。

* Tracert

Tracert程序主要用于查看数据包从源端到目的端的路径信息，从而检查网络连接是否可用。当网络出现故障时，用户可以使用该命令定位故障点。

Tracert利用ICMP超时信息和目的不可达信息来确定从一个主机到网络上其他主机的路由，并显示IP网络中每一跳的延迟（这里的延迟是指：分组从信息源发送到目的地所需的时间，延迟也分为许多的种类——传播延迟、传输延迟、处理延迟、排队延迟等）。

* NQA

网络质量分析NQA（Network Quality Analysis）是一种实时的网络性能探测和统计技术，可以对响应时间、网络抖动、丢包率等网络信息进行统计。NQA能够实时监视网络服务质量，在网络发生故障时进行有效的故障诊断和定位。

利用不同类型的ICMP报文，NQA实现了Ping和Tracert功能的扩展和增强，可以实现对网络运行状况的准确测试，输出统计信息。比如NQA的ICMP测试、ICMP Jitter测试和Trace测试等。

**ICMP测试**

ICMP测试是通过发送ICMP Echo请求报文和Echo响应报文来判断目的地的可达性、计算网络响应时间及丢包率，从而清晰地反映出网络性能及畅通情况。ICMP测试提供类似于命令行下的Ping命令功能，但输出信息更为丰富。

**ICMP Jitter测试**

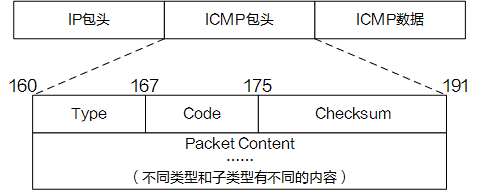
ICMP Jitter测试是以ICMP报文为承载，通过记录在报文中的时间戳信息来统计时延、抖动、丢包的一种测试方法。Jitter（抖动时间）是指相邻两个报文的接收时间间隔减去这两个报文的发送时间间隔。

**Trace测试**

NQA的Trace测试用于检测源端到目的端的转发路径，并沿该路径记录源设备到中间各个设备的时延等信息。Trace测试类似于Tracert命令功能，但输出信息更为丰富。每一跳信息中能够显示平均时延、丢包、最后一个包接收时间等信息。

1. 掌握ICMP报文的报文帧格式

ICMP报文格式如图所示，每一个ICMP消息都将包含引发这条ICMP消息的数据包的完全IP包头，ICMP报文则作为IP数据包的数据部分封装在IP数据包内部。ICMP包头中包含的三个固定字段就是源端设备确定发生错误的类型的主要依据。



**Type字段**：表示ICMP消息的类型；

**Code字段**：表示ICMP消息类型细分的子类型；

**Checksum字段**：表示ICMP报文的校验和。

不同的Type和Code值表示不同的ICMP报文类型，对应了数据包处理过程中可能出现的不同错误情况，不同类型的ICMP报文又分为差错报文和查询报文两种，如ICMP报文分类表所示。





此外，ICMP协议在以下情况下不会产生ICMP差错报文：

* ICMP差错报文不会产生ICMP差错报文（但ICMP查询报文可能会产生ICMP差错报文），此条规定主要为了防止ICMP消息的无限产生和传递；
* 目的地址是广播地址或多播地址的IP数据报文；
* 作为链路层广播的数据包；
* 不是IP分片的第一片；
* 源地址不是单个主机的数据包，也就是说，源地址不能为零地址、环回地址、广播地址或多播地址。

# PPPoE（Point-to-Point Protocol over Ethernet）

## 8.1 什么是PPPoE？

以太网上的点对点协议，是将点对点协议（PPP）封装在以太网（Ethernet）框架中的一种网络隧道协议。

## 8.2 为什么要用PPPoE？

PPPoE技术将以太网技术的经济性与PPP协议的可管理控制性结合在一起，提供接入互联网的功能。对于运营商来说，它能够最大限度地利用电信接入网现有的体系结构，利用现有的拨号网络资源，运营和管理的模式也不需要很大的改变；对于用户来说，使用感与原来的拨号上网没有太大区别，较容易接受。

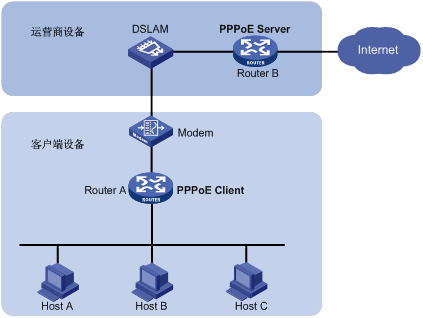
## 8.3 PPPoE是如何工作的？

应用在链路层，PPPoE使用Client/Server模型。PPPoE Client向PPPoE Server发起连接请求，两者之间会话协商通过后，就建立PPPoE会话，此后PPPoE Server向PPPoE Client提供接入控制、认证、计费等功能。PPPoe的最主要作用就是使得多台设备可以借助以太网多点接入多点访问的特点去和一台设备做上网认证请求，这样运营商一台认证设备就可以满足多个用户的认证请求。

根据PPPoE会话的起点所在位置的不同，PPPoE分为Router-Initiated和Host-Initiated两种组网结构。

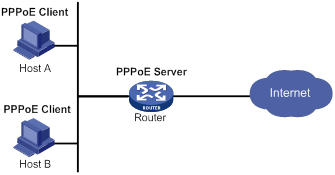
## 8.4 Router-Initiated组网结构

如下图所示，Router-Initiated组网结构是在两台路由器之间建立PPPoE会话，所有主机通过同一个PPPoE会话传送数据，主机上不用安装PPPoE客户端拨号软件，一般是一个企业共用一个账号接入网络（图中PPPoE Client位于企业/公司内部，PPPoE Server是运营商的设备）。

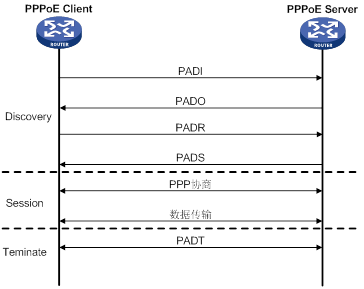


## 8.5 Host-Initiated组网结构

如图2所示，Host-Initiated组网结构是将PPPoE会话建立在Host和运营商的路由器之间，为每一个Host建立一个PPPoE会话，每个Host都是PPPoE Client，每个Host使用一个帐号，方便运营商对用户进行计费和控制。Host上必须安装PPPoE客户端拨号软件。



## 8.6 PPPoE的协商过程



**①Discovery阶段：**该阶段完成之后通信双方都会知道PPPoE的Session ID以及对方以太网地址，它们共同确定了唯一的PPPoE Session。

Discovery阶段由四个过程组成，具体如下：

1. PPPoE Client广播发送一个PADI报文，在此报文中包含PPPoE Client想要得到的服务类型信息。
2. 所有的PPPoE Server收到PADI报文之后，将其中请求的服务与自己能够提供的服务进行比较，如果可以提供，则单播回复一个PADO报文。
3. 根据网络的拓扑结构，PPPoE Client可能收到多个PPPoE Server发送的PADO报文，PPPoE Client选择最先收到的PADO报文对应的PPPoE Server做为自己的PPPoE Server，并单播发送一个PADR报文。
4. PPPoE Server产生一个唯一的会话ID（Session ID），标识和PPPoE Client的这个会话，通过发送一个PADS报文把会话ID发送给PPPoE Client，从而建立会话，并进入PPPoE Session阶段。

**②Session阶段：**PPPoE Session上的PPP协商和普通的PPP协商方式一致，分为LCP、认证、NCP三个阶段，具体如下：

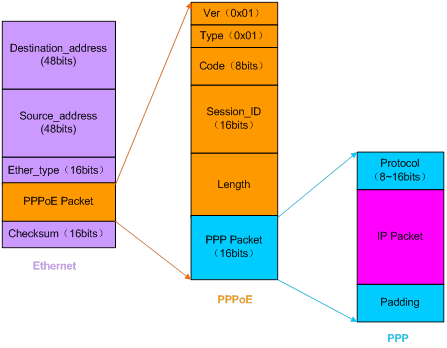
1. LCP阶段主要完成建立、配置和检测数据链路连接。
2. LCP协商成功后，开始进行认证工作，认证协议类型由LCP协商结果（CHAP或者PAP）决定。
3. 认证成功后，PPP进入NCP阶段。NCP是一个协议族，用于配置不同的网络层协议，常用的是IP控制协议（IPCP），它负责配置用户的IP和DNS等工作。

PPPoE Session的PPP协商成功后，其上就可以承载PPP数据报文。

在PPPoE Session阶段所有的以太网数据包都是单播发送的。

**③Terminate阶段：**进入PPPoE Session阶段后，PPPoE Client和PPPoE Server都可以通过发送PADT报文的方式来结束PPPoE连接。PADT报文可以在会话建立以后的任意时刻单播发送。在发送或接收到PADT后，就不允许再使用该会话发送PPP流量，即使是常规的PPP结束数据包也不允许发送。

⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐**PPPoE报文格式**⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐



各个字段解释如下：

**Destination\_address域**：一个以太网单播目的地址或者以太网广播地址（0xffffffff）。在Discovery阶段，该域的值可以是单播或者广播地址，PPPoE Client寻找PPPoE Server的过程使用广播地址，确认PPPoE Server后使用单播地址。在Session阶段，该域的值必须是Discovery阶段已确定的通信对方的单播地址。

**Source\_address域**：源设备的以太网MAC地址。

**Ether\_type**：取值为0x8863表示Discovery阶段或Terminate阶段，0x8864表示Session阶段。

**Ver域**：4bits，PPPoE版本号，值为0x1。

**Type域**：4bits，PPPoE类型，值为0x1。

**Code域**：8bits，PPPoE报文类型，取值包括：

* 0x00，表示会话数据。
* 0x09，表示PADI（PPPoE Active Discovery Initiation）报文。
* 0x07，表示PADO（PPPoE Active Discovery Offer）或PADT（PPPoE Active Discovery Terminate）报文。
* 0x19，表示PADR（PPPoE Active Discovery Request）报文。
* 0x65，表示PADS（PPPoE Active Discovery Session-confirmation）报文。

**Session\_ID域**：16bits，对于一个给定的PPP会话，该值是一个固定值，并且与以太网Source\_address和Destination\_address一起实际地定义了一个PPP会话。值0xffff为将来的使用保留，不允许使用。

**Length域：**16bits，定义PPPoE的Payload域长度。不包括以太网头部和PPPoE头部的长度。

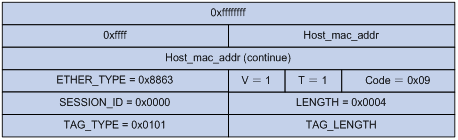
⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐**PADI报文**⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐

PADI报文的主要字段包括：

Code域为0x09。

Session\_ID域为0x0000。

TAG\_TYPE域为0x0101（Service-Name），表明后面紧跟的是服务的名称。有且只有一个TAG\_TYPE为Service-Name的TAG，其他类型的TAG可选。



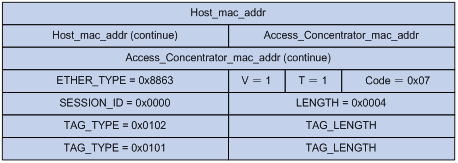
⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐**PADO报文**⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐

PADO报文的主要字段包括：

Code域为0x07。

SESSION\_ID域为0x0000。

TAG\_TYPE域为0x0101（Service-Name），表明后面紧跟的是服务的名称；为0x0102（AC-Name）表明后面紧跟的字符串唯一地表示了某个特定的访问集中器。有且只有一个TAG\_TYPE为AC-Name的TAG，至少一个TAG\_TYPE为Service-Name的TAG。



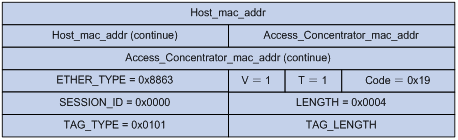
⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐**PADR报文**⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐

PADR报文的主要字段包括：

Code域为0x19。

SESSION\_ID为0x0000。

TAG\_TYPE域为0x0101（Service-Name），表明后面紧跟的是服务的名称。有且只有一个TAG\_TYPE为Service-Name的TAG，其他类型的TAG可选。



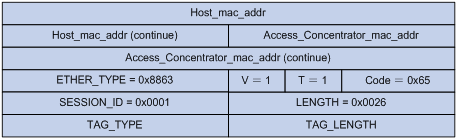
⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐PADS报文⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐

PADS报文的主要字段包括：

Code域为0x65。

SESSION\_ID为Discovery阶段分配的数值。

TAG为可选。



⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐PADT报文⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐⭐

PADT报文的主要字段包括：

Code域为0xa7。

SESSION\_ID域为Discovery阶段分配的数值。

无TAG。

