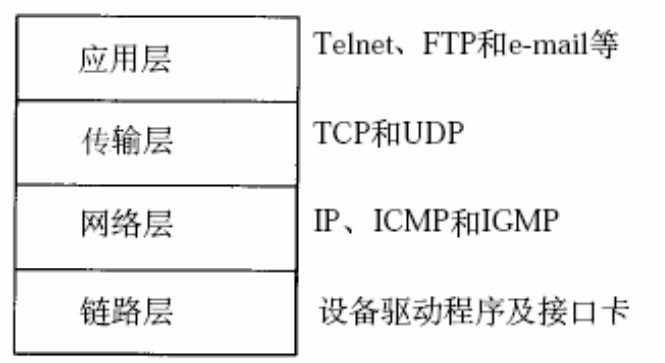
**以太网ip、tcp、udp数据包分析**

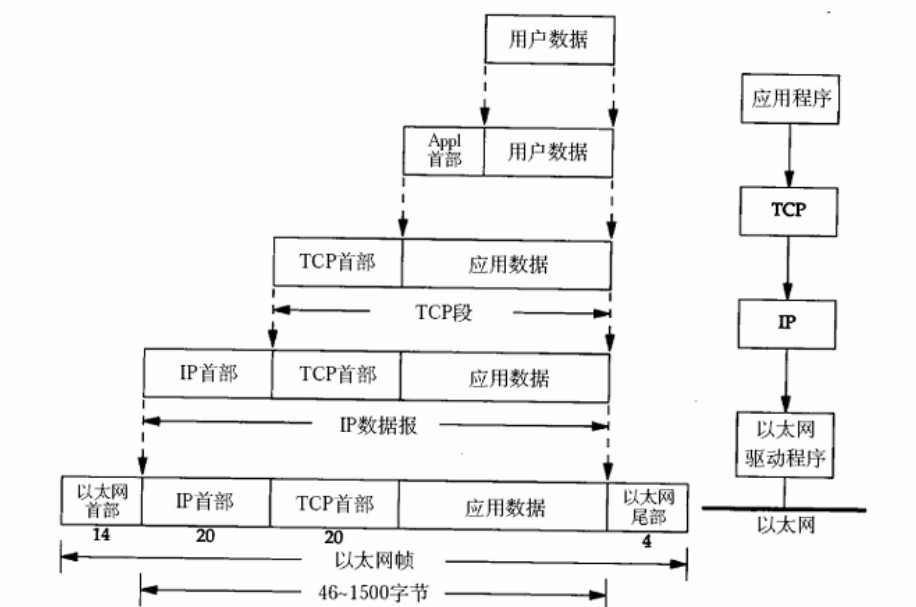
**1、ISO开放系统**有以下几层：

|  |  |
| --- | --- |
| 7 | 应用层 |
| 6 | 表示层 |
| 5 | 会话层 |
| 4 | 传输层 |
| 3 | 网络层 |
| 2 | 数据链路层 |
| 1 | 物理层 |

**2、**TCP/IP 网络协议栈分为应用层（Application）、传输层（Transport）、网络层（Network）和链路层（Link）四层。



通信过程中，每层协议都要加上一个数据首部（header），称为封装（Encapsulation），如下图所示



不同的协议层对数据包有不同的称谓，在传输层叫做段（segment），在网络层叫做数据报（datagram），在链路层叫做帧（frame）。数据封装成帧后发到传输介质上，到达目的主机后每层协议再剥掉相应的首部，最后将应用层数据交给应用程序处理。

其实在链路层之下还有物理层，指的是电信号的传递方式，比如现在以太网通用的网线（双绞线）、早期以太网采用的的同轴电缆（现在主要用于有线电视）、光纤等都属于物理层的概念。

**3、**集线器（Hub）是工作在物理层的网络设备，用于双绞线的连接和信号中继（将已衰减的信号再次放大使之传得更远）。

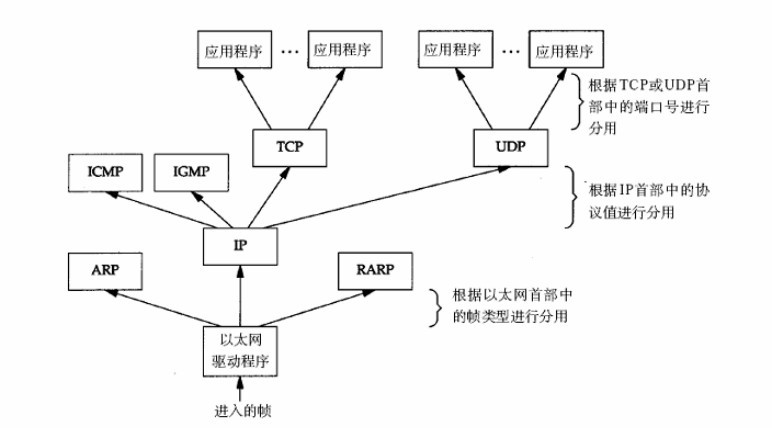
交换机是工作在链路层的网络设备，可以在不同的链路层网络之间转发数据帧（比如十兆以太网和百兆以太网之间、以太网和令牌环网之间），由于不同链路层的帧格式不同，交换机要将进来的数据包拆掉链路层首部重新封装之后再转发。

路由器是工作在第三层的网络设备，同时兼有交换机的功能，可以在不同的链路层接口之间转发数据包，因此路由器需要将进来的数据包拆掉网络层和链路层两层首部并重新封装。

**4、** 网络层的IP 协议是构成Internet 的基础。IP 协议不保证传输的可靠性，数据包在传输过程中可能丢失，可靠性可以在上层协议或应用程序中提供支持。

传输层可选择TCP 或UDP 协议。TCP 是一种面向连接的、可靠的协议，有点像打电话，双方拿起电话互通身份之后就建立了连接，然后说话就行了，这边说的话那边保证听得到，并且是按说话的顺序听到的，说完话挂机断开连接。也就是说TCP 传输的双方需要首先建立连接，之后由TCP 协议保证数据收发的可靠性，丢失的数据包自动重发，上层应用程序收到的总是可靠的数据流，通讯之后关闭连接。UDP 协议不面向连接，也不保证可靠性，有点像寄信，写好信放到邮筒里，既不能保证信件在邮递过程中不会丢失，也不能保证信件是按顺序寄到目的地的。使用UDP 协议的应用程序需要自己完成丢包重发、消息排序等工作。

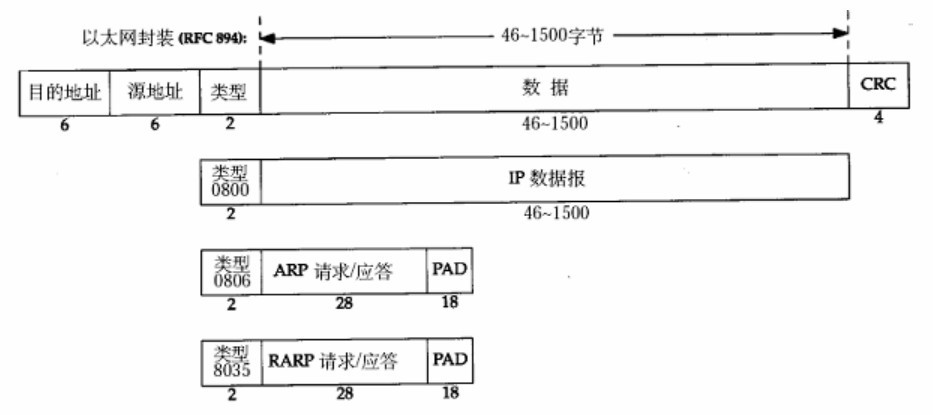
**5、 数据传输经过的各层协议过程如下**



以太网驱动程序首先根据以太网首部中的“上层协议”字段确定该数据帧的有效载荷（payload，指除去协议首部之外实际传输的数据）是IP、ARP 还是RARP 协议的数据报，然后交给相应的协议处理。假如是IP 数据报，IP 协议再根据IP 首部中的“上层协议”字段确定该数据报的有效载荷是TCP、UDP、ICMP 还是IGMP，然后交给相应的协议处理。假如是TCP 段或UDP段，TCP 或UDP 协议再根据TCP 首部或UDP 首部的“端口号”字段确定应该将应用层数据交给哪个用户进程。IP 地址是标识网络中不同主机的地址，而端口号就是同一台主机上标识不同进程的地址，IP 地址和端口号合起来标识网络中唯一的进程。

虽然IP、ARP 和RARP 数据报都需要以太网驱动程序来封装成帧，但是从功能上划分，ARP 和RARP 属于链路层，IP 属于网络层。虽然ICMP、IGMP、TCP、UDP 的数据都需要IP 协议来封装成数据报，但是从功能上划分，ICMP、IGMP 与IP 同属于网络层，TCP 和UDP属于传输层。

**6、以太网帧格式**



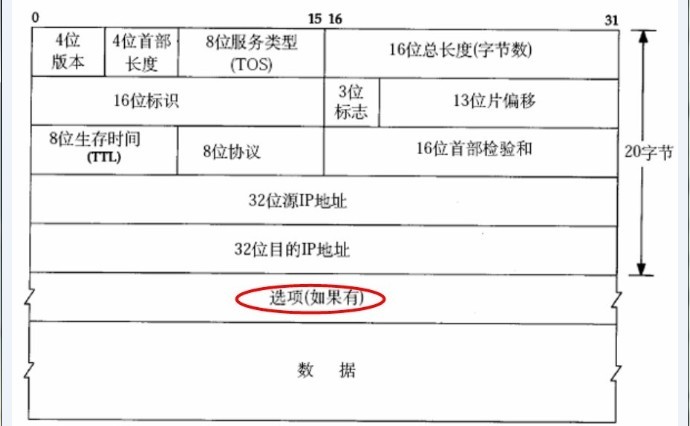
（1）其中的源地址和目的地址是指网卡的硬件地址（也叫MAC 地址），长度是48 位，是在网卡出厂时固化的。

（2）注意网卡芯片（例如DM9000A）收到的数据就是如上所示的一长串数据；其中包括以太网帧头、IP报报头、传输层协议段头、应用层所需数据。

（3）以太网帧中的数据长度规定最小46 字节，最大1500 字节，ARP 和RARP 数据包的长度不够46 字节，要在后面补填充位。最大值1500 称为以太网的最大传输单元（MTU），不同的网络类型有不同的MTU，如果一个数据包从以太网路由到拨号链路上，数据包度大于拨号链路的MTU了，则需要对数据包进行分片fragmentation）。ifconfig 命令的输出中也有“MTU:1500”。注意，MTU 个概念指数据帧中有效载荷的最大长度，不包括帧首部的长度。

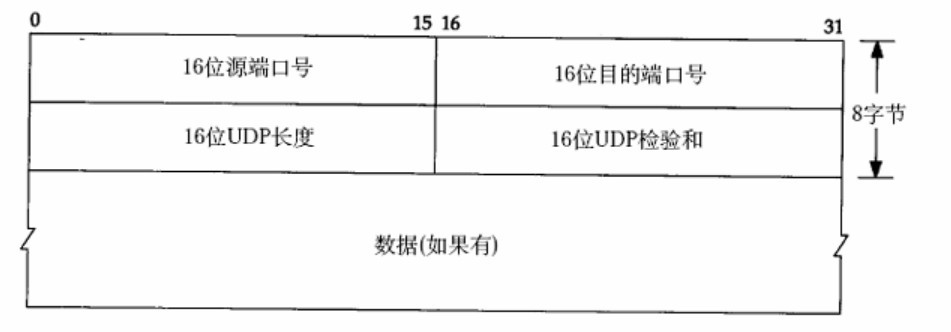
**7、IP数据报格式**

IPv4：



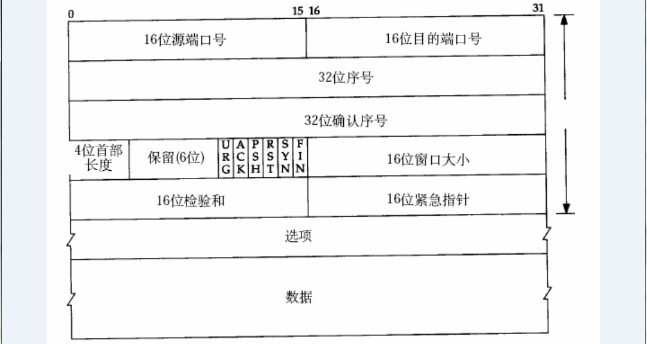
IP 数据报的首部长度和数据长度都是可变长的，但总是4 字节的整数倍。对于IPv4，4 位版本字段是4。4 位首部长度的数值是以4 字节为单位的，最小值为5，也就是说首部长度最小是4x5=20 字节，也就是不带任何选项的IP 首部，4 位能表示的最大值是15，也就是说首部长度最大是60 字节。8 位TOS 字段有3 个位用来指定IP 数据报的优先级（目前已经废弃不用），还有4 个位表示可选的服务类型（最小延迟、最大呑吐量、最大可靠性、最小成本），还有一个位总是0。总长度是整个数据报（包括IP 首部和IP 层payload）的字节数。每传一个IP 数据报，16 位的标识加1，可用于分片和重新组装数据报。3 位标志和13 位片偏移用于分片。TTL（Time to live)是这样用的：源主机为数据包设定一个生存时间，比如64，每过一个路由器就把该值减1，如果减到0 就表示路由已经太长了仍然找不到目的主机的网络，就丢弃该包，因此这个生存时间的单位不是秒，而是跳（hop）。协议字段指示上层协议是TCP、UDP、ICMP 还是IGMP。然后是校验和，只校验IP 首部，数据的校验由更高层协议负责。IPv4的IP 地址长度为32 位。

**8、UDP段格式**



UDP 协议不面向连接，也不保证传输的可靠性。

**9、TCP段格式**



**（1）序号：指出段中的数据部分在发送方数据流中的位置。**

确认号：指出接收方希望收到对方下次发送的数据的第一个字节的序号。

TCP段首部的定长部分为20个字节，即5个单位的长度。

URG位：紧急标志，和紧急指针配合使用，当其为1时表示，此报文要尽快传送。

ACK位：确认标志，和确认号字段配合使用，当ACK位置1时，确认号字段有效。

PSH位：为推送标志，置1时，发送方将立即发送缓冲区中的数据。

RST位：复位标志，置1时，表明有严重差错，必须释放连接。

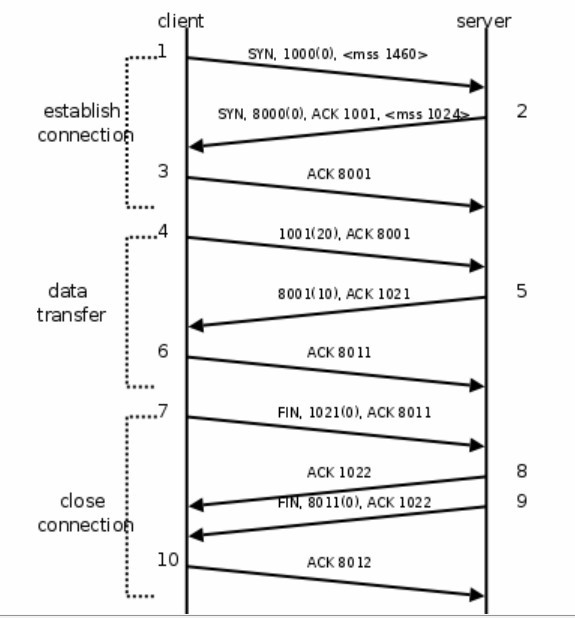
SYN位： 同步标志，置1时，表示请求建立连接。

FIN位：终止标志，置1时，表明数据已经发送完，请求释放连接。

窗口大小：32bit，用于向对方通告当前本机的接受缓冲区的大小。

校验和字段长度：16bit，校验范围包括段首部、数据以及伪首部。

**（2）TCP数据传输过程**



**（3）TCP连接的建立**

建立连接的过程：

【1】. 客户端发出段1，SYN 位表示连接请求。序号是1000，这个序号在网络通讯中用作临时的地址，每发一个数据字节，这个序号要加1，这样在接收端可以根据序号排出数据包的正确顺序，也可以发现丢包的情况，另外，规定SYN 位和FIN 位也要占一个序号，这次虽然没发数据，但是由于发了SYN 位，因此下次再发送应该用序号1001。mss表示最大段尺寸，如果一个段太大，封装成帧后超过了链路层的最大帧长度，就必须在IP 层分片，为了避免这种情况，客户端声明自己的最大段尺寸，建议服务器端发来的段不要超过这个长度。

【2】 服务器发出段2，也带有SYN 位，同时置ACK 位表示确认，确认序号是1001，表示“我接收到序号1000 及其以前所有的段，请你下次发送序号为1001 的段”，也就是应答了客户端的连接请求，同时也给客户端发出一个连接请求，同时声明最大尺寸为1024。

【3】 客户端发出段3，对服务器的连接请求进行应答，确认序号是8001。

在这个过程中，客户端和服务器分别给对方发了连接请求，也应答了对方的连接请求，其中服务器的请求和应答在一个段中发出，因此一共有三个段用于建立连接，称为'''三方握手（three-way-handshake）'''。在建立连接的同时，双方协商了一些信息，例如双方发送序号的初始值、最大段尺寸等。

**（4）TCP数据传输过程**

【1】  客户端发出段4，包含从序号1001 开始的20 个字节数据。

【2】  服务器发出段5，确认序号为1021，对序号为1001-1020 的数据表示确认收到，同时请求发送序号1021 开始的数据，服务器在应答的同时也向客户端发送从序号8001 开始的10 个字节数据，这称为piggyback。

【3】  客户端发出段6，对服务器发来的序号为8001-8010 的数据表示确认收到，请求发送序号8011 开始的数据。

在数据传输过程中，ACK 和确认序号是非常重要的，应用程序交给TCP 协议发送的数据会暂存在TCP 层的发送缓冲区中，发出数据包给对方之后，只有收到对方应答的ACK 段才知道该数据包确实发到了对方，可以从发送缓冲区中释放掉了，如果因为网络故障丢失了数据包或者丢失了对方发回的ACK 段，经过等待超时后TCP 协议自动将发送缓冲区中的数据包重发。

**（5）TCP连接的关闭**

【1】  客户端发出段7，FIN 位表示关闭连接的请求。

【2】  服务器发出段8，应答客户端的关闭连接请求。

【3】  服务器发出段9，其中也包含FIN 位，向客户端发送关闭连接请求。

【4】  客户端发出段10，应答服务器的关闭连接请求。

建立连接的过程是三方握手，而关闭连接通常需要4 个段（四次握手），服务器的应答和关闭连接请求通常不合并在一个段中，因为有连接半关闭的情况，这种情况下客户端关闭连接之后就不能再发送数据给服务器了，但是服务器还可以发送数据给客户端，直到服务器也关闭连接为止，稍后会看到这样的例子。

**10、DM9000A实验数据总结**

DM9000A收到上位机TCP的数据包packet[ ]如下

(0x01,0x60,0x6E,0x11,

0x02,0x0F,0xE0,0x05,

0xC5,0xF3,0x29,0x00,

0x08,0x00,0x45,0x00,

0x00,0x40,0x28,0x03,

0x40,0x00,0x40,0x06,

0x91,0x5D,0xC0,0xA8,

0x00,0x04,0xC0,0xA8,

0x00,0x03,0x0C,0x72,

0x04,0x00,0xA8,0x85,

0x77,0x60,0x00,0x00,

0x00,0x00,0xB0,0x02,

0xFF,0xFF,0x84,0x51,

0x00,0x00,0x02,0x04,

0x05,0xB4,0x01,0x03,

0x03,0x00,0x01,0x01,

0x08,0x0A,0x00,0x00,

0x00,0x00,0x00,0x00,

0x00,0x00,0x01,0x01,

0x04,0x02,0xD8,0x9C,

0x05,0xA6,)

具体分析如下：

这个包结构为：以太网帧头 + IP数据报 + TCP/UDP数据包；

0x01,0x60,0x6E,0x11,0x02,0x0F, 为目的物理地址；packet[0] ~ packet[5]。

0xE0,0x05,0xC5,0xF3,0x29,0x00,为源物理地址；packet[6] ~ packet[11]。

0x08,0x00：协议类型，0800为IP；packet[12] 、packet[13]。

0x45：版本号和首部长度；packet[14]；

0x00：TOS；packet[15]；

0x00,0x40:16位总长度；packet[16] packet[17]；；

0x28,0x03：16位标识；packet[18]~ packet[19]；

0x40,0x00 : 3位标志 + 13位片偏移；packet[20] packet[21]；；

 0x40,：8位的生存时间TTL；packet[22]；

0x06：8位的协议号；packet[23]；

（ICMP：1；IGMP：2；TCP：6；EGP：8；UDP：17；）

0x91,0x5D：首部校验和；packet[24]；packet[25]；

0xC0,0xA8,0x00,0x04：源IP地址；packet[26]~packet[29]；

0xC0,0xA8,0x00,0x04：目的IP地址；packet[30]~packet[33]；

选项（无）；

0x0c，0x72：源端口；packet[34],packet[35];

0x04,0x00；目的端口；pcket[36],packet[37];

0xa8,0x85,0x77,0x60：序号；packet[38] ~ packet[41]；

0x00，0x00，0x00，0x00：确认号；packet[42]~packet[45]；

0Xb0，0x02：（HLEN,保留6bit，URG,ACK,PSH,PST,SYN,FIN）；packet[46]~packet[47]；

0Xff,0xff：窗口大小；packet[48],packet[49]；

0x84,0x51：校验和；packet[50],packet[51];

0x00,0x00：紧急指针；packet[52],packet[53];

…………

最后还用4字节的CRC；属于以太网帧。