计算机网络期中论文之

网络协议简要Q&A

刘鹏

（云南大学 数学与统计学院信息与计算科学专业，昆明市 呈贡区 650500）

**Computer Network Midterm paper:  
Brief Analysis of Five Layer Model of Network Protocol Stack**

PENG Liu

(School of Mathematics and Statistics, Yunnan University, Chenggong District, Kunming 650500, China)

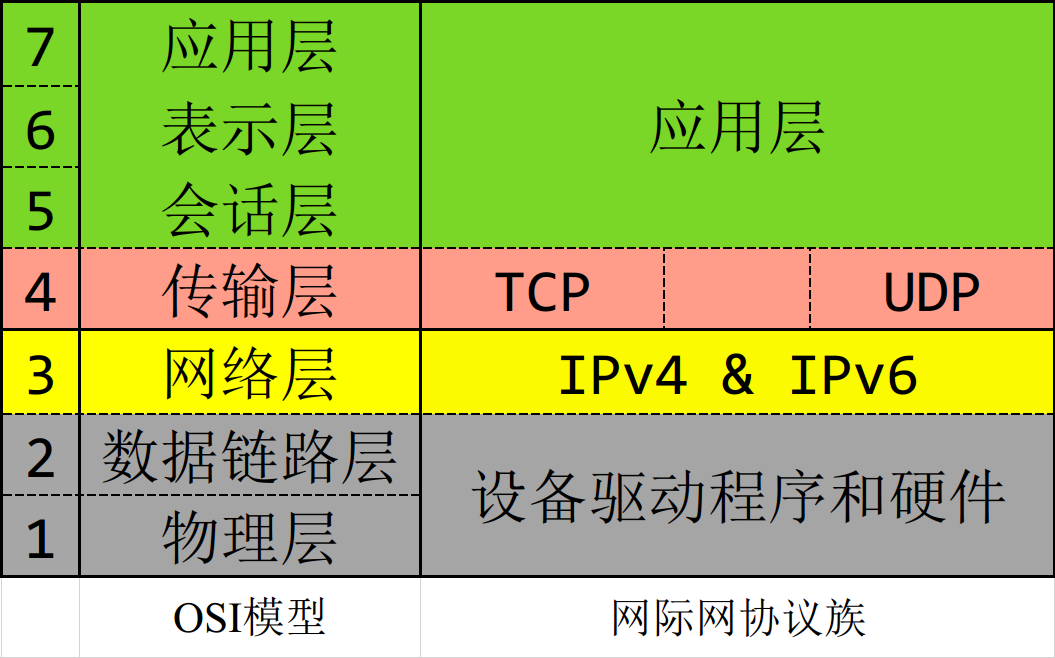
**ABSTRACT:** 描述一个网络中各个协议层的常用方法是使用国际标准化组织（International Organization for Stand-ardization, ISO）的计算机通信开放系统互连（Open Systems Interconnection, OSI）模型。这是一个七层模型，但是它与现实中的网际网（亦即互联网）有部分区别。两个模型均可实现网络功能，而且都是分层模型。本文着重探讨为何互联网协议栈选择了五层模型，而放弃了更为结构化、专一化的七层模型；探讨完现有体系的由来，本文还描述了五层协议是如何在应用层中实现通信加密的。

关键词：互联网；协议栈；五层模型

# 0 引言

互联网的直接始祖是美军最初为了全球作战的信息交换而开发的阿帕网。在世界计算机保有量到达一定规模的时候，让多台计算机之间互相通信成为了一个非常迫切的需要。让一台计算机控制另外一台计算机、让一台计算机传输另一台计算机需要的文件等需求，是网络产生的起源。在计算机网络诞生之初，已有电报网络、电话网络可以参考，甚至交通网络也可以提供一定的借鉴价值。然而无论是电报网络还是电话网络，其接入的边缘客户端往往是固定的，客户端之间互相通信依赖一条固定的、独占的信道。这种电话网络往往资源利用率非常低，而且容易被窃听，而利用率低、安全性低的主要原因在于其设计过于简单。现代计算机网络可以在应用层进行加密，然后通过一定的传输控制协议，在路由协议下进行分组的链路转发。这种实现之所以是可能的，主要在于现代计算机网络中存在着具有计算能力的路由器、交换机等设备，这些设备可以对分组所携带的一些附加信息进行解读，然后通过逻辑电路对信号进行条件转发，从而避免人工接线操作，大大提高了效率；而应用层的加密、传输控制等协议，可以大幅度提高信息的安全、完整性。

虽然相比先前的电报、电话网络，计算机网络的安全以及性能有了长足的进步，但是这种基于电路和分组的现代计算机网络与其诞生之前就有的电报、电话网络、邮政系统仍旧有着某些共性，例如，邮件网络中对信件进行格式化书写然后通过信封、胶水进行封装然后贴邮票、加盖时间戳、交付给邮政系统这一理念，在计算机网络中表现为通过应用层协议（如邮件进程间的跨网络通信、打印进程间的跨网络通信、Web客户端进程与Web服务器进程的通信、SMTP服务器进程与客户端进程间的通信）对信息进行格式化封装，然后推送给传输层或者直接给网络层；邮政系统对地区进行分级这一理念在IP协议中表现为对网络进行分段，通过IP地址和子网掩码进行计算，得到网段。



# 1 网络层中路由表的生成

路由表。物理层 链路层 网络层 传输层 应用层 物理层

bit 物理层解决如何在物理介质上传输bit的问题 说到底，任何电脑中需要传输的数据，都是一堆0和1，那么物理层的任务，就是传输0和1 最简单如以某个范围的高电压表示1、低电压表示0，或以电压由低变高表示1、由高变低表示0等等，也就是bit的编码方式

**调制/解调** 但是如此简单的方式，带来的后果就是无法进行远距离传输 于是调制/解调就发挥作用了。什么传输得远？当然是电磁波，载波就是一正弦波，而顾名思义，“载”波就是用来携带数据的波 调制的过程就是把01“写入”到载波中

比如振幅调制，就用振幅大代表1，振幅小代表0；频率调制，就是频率大代表1，频率小代表0，诸如此类 （也可以振幅小代表0，这些都是约定好的。这种所谓的约定也就是“协议”） 解调就是依据协议（比如双方都用某种振幅调制）来提取出01信息，接收方遇到振幅大就知道是1，小就是0

**设备** 工作在这一层的设备有中继器，其作用是放大信号来扩大传输距离、对信号进行整形来抵抗信号的衰减、变形等 链路层

**帧** 最关键的概念即“帧” 物理层传输bit，链路层传输帧 帧就是一块有相应格式的数据，通常可分为三部分：头部、数据部、尾部 受最大传输单元MTU的限制，链路层能传输的一个帧有长度限制

实现 不同的链路层实现具有不同的帧格式，而其相应的MTU也不同 最最最常见即以太网，以太网的帧也叫以太帧 以太网是局域网的一种实现，通信采用CSMA/CD的多点接入协议，其余的局域网实现还有IBM的令牌环网络、ATM网、无线网等等，它们使用的帧格式也是也均不相同 另一个很有用的即PPP协议，Point to Point Protocol，其帧为PPP帧，实现了点对点的通信，具有身份验证的功能。这个用来干什么呢。拨号上网。比如要用电信网就要有个电信账号和密码，拨号的时候用的就是PPP协议和电信的服务器通信，验证通过就把你的电脑接入到电信网中，于是就可以上网了。而现在一般用PPPoE协议，配置路由器也常会见到PPPoE。PPPoE即PPP over Ethernet。由于大多数局域网都是以太网，而位于以太网内的电脑采用的是以太网的帧格式，PPPoE协议则结合了PPP帧和以太网帧，实现了在以太网上传输PPP帧，实际上是把PPP帧作为数据部分封装成以太帧，在以太帧的头部的帧类型字段表明是PPP帧数据。于是大家在局域网里也就可以拨号上网了。

理论上的分层 再进一步，链路层通常可分为LLC层和MAC层，不过LLC似乎已经不怎么提及了，只要实现MAC层也就搞定了通信 什么是MAC层。Media Access Control，媒体接入控制，简单来说就是解决如何标识每一台主机的问题。想到了什么。mac地址呗，全球唯一的网卡地址，完美解决主机的标识问题。 以太网的MAC子层实现是基于网卡地址，由于以太网太过普及，干脆也就称为mac地址了 以太帧的头部有源mac地址（发送数据的主机）和目标mac地址（接受数据的主机），这个帧一发出去，同一个局域网的主机都能收到，然后对比一下是不是应该由我来接受（即帧中目标mac地址和我的mac地址一不一样）

设备 工作在这一层的设备有网桥、交换机，它们可以识别帧并进行转发。（不同的协议有不同的帧格式，所以要能识别出来必须配置相应的协议） 网络层 异构网络与IP地址

说的绕一点：网络层解决的是异构网络的通信问题 什么是异构网络。就是链路层有很多很多种不同的实现（也大致可以等同于局域网也不仅仅有以太网），各种不同的网络之间用来通信的帧格式不同，又如何能通信呢。 于是IP地址就出现了，网络层采用IP地址来标识每一个主机（更具体的说是每一个网络连接点）

以太网用的MAC地址，也就是网卡地址，不也是全球唯一的，但是这个地址只有以太网在用（虽然好像其实大家都在用网卡地址来标识主机，但最主要的是帧格式的不同），也就是异构网络的MAC子层不一样，尽管现在绝大部分都是以太网

IP数据报大家都用IP地址来通信，传输的是IP数据报。这个IP数据报即作为链路层的数据部分，加上帧头部和帧尾部，即封装成一个数据帧，但数据帧的数据部分不仅仅是IP数据报，还可以是其他链路层的协议所使用的数据，比如PPP协议中的LCP和NCP 而实际上，如果一个IP数据报要想传输成功，就必须封装成帧（具体怎样的帧格式要看在怎样的网络中，若在以太网中就封装成以太帧），然后送到目标主机，而网卡只认MAC地址，只认帧，网卡收到一个帧后，发现确实是发给自己的，于是把帧拆开取出数据部分（IP数据报），在向上提交给网络层 IP数据报也有格式，即IP首部和数据部。IP首部至少应当包括三个信息：发送方的IP地址，接收方的IP地址，总长度。 最终的效果就是，站在网络层的视角，我并不需要知道帧格式是怎样的、MAC地址是几，我只需要知道对方的IP地址，我就可以把数据传过去了。

问题1：IP地址与MAC地址 那么问题就来了:网卡可是只认MAC地址的啊，我如何实现IP地址到MAC地址的转换（映射）呢，即IP数据报最终一定是作为链路层帧的数据部分，那帧头部还需要填入MAC地址的信息，我如何得知呢——ARP协议 ARP协议工作在链路层，即直接使用帧来通信，但是其解决的问题是网络层的问题，所以有些地方把它按照目的归入网络层，有些地方按照工作方式归入链路层 ARP协议解决的是局域网内的通信问题（因为使用帧来通信嘛），好现在有一个主机host\_1，它的IP地址是ip\_1，MAC地址是mac\_1，同一局域网内有另一台主机host\_2、ip\_2、mac\_2。此时host\_1想向host\_2发送数据，而它只知道对方的IP地址是ip\_1,于是

\*1.\*host\_1广播（目标MAC地址全1）一个帧，内容是“我是ip\_1,mac\_1，我想知道ip\_2的MAC地址” 2.本局域网上的所有主机都收到了这个帧，但是只有host\_2会回答这个帧（因为它才是ip\_2），此时它知道了host\_1想要它的IP地址，也知道了host\_1的MAC地址是mac\_1 3.于是host\_2发送一个帧（目标MAC地址是mac\_1），内容是“我是ip\_2，我的MAC地址是mac\_2” 4.只有host\_1才能收到这个帧（因为只有它才是mac\_1），于是host\_1也就知道了ip\_2的MAC地址是mac\_2 每次都这样是不是很慢啊，慢就来个缓存呗——ARP高速缓存。好不容易厚着脸皮要到了mac地址，赶紧找个小本本记下来。

问题2：路由器如何工作 那么问题又来了：ARP解决的是局域网内的通信问题，那我平时上网都用路由器，主机位于我路由器内的局域网里，又怎么访问百度呢。 这就牵扯到路由器了 大致可以认为整个网络就是路由器在转发数据，从一个路由器到另一个路由器，最终到达目标主机 于是整个过程就大致可以这样理解：

1.首先你根据路由器的mac地址（它和你的主机在同一个局域网内），把数据发给路由器 2.路由器收到这个帧（因为mac地址写的是它），拆开帧，得知你要发送数据给ip\_baidu，于是路由器查找路由表，找到下一跳地址ip\_next，然后用ARP协议解析到ip\_next的mac地址（没错你的路由器和这个ip\_next的路由器在一个局域网），于是数据到达了另一个路由器 3.而这个ip\_next的路由器又和其他路由器在同一个局域网，但是你的路由器不和它们在同一个局域网，所以只能把你的数据先给你的路由器、你的路由器再给ip\_next这个路由器、这个路由器再给更远的路由器 4.最终到达一个路由器，这个路由器和ip\_baidu在同一个局域网里，然后你的数据就到达了目的地 命令行下“tracert”命令可以追踪从你的主机到目标IP地址的主机之间经过了哪些路由器，命令的原理是利用了ICMP协议

问题3：异构网络通信问题的解决 注意到没，一个路由器连接了好几个局域网，而着好几个局域网并不一定都是以太网，也就是说它们的帧格式不一定相同 所以路由器的任务很关键，这也就是解决异构网络通信的关键所在：路由器知道它连接了哪些局域网，也知道它们的帧格式分别是怎样的 而路由器在转发的时候，拆开一个帧（怎么拆。根据帧格式。如何得知帧格式。看它来自哪个局域网）后只拿到了数据部分，从数据部分（IP数据报）中得知它的目的IP，通过路由表得知我要转发给那一个路由器，而它又位于哪种局域网中，再按照这种局域网的帧格式把数据封装起来，发过去 路由器继续沉默着。“又一个帧来了”路由器撇了一眼，“又是以太帧，没劲”。说完便拆开了这个帧。以太帧对于路由器来说实在是太常见了，他甚至可以在半睡半醒的状态下闭着眼拆开它。看到这个帧要去的IP地址，路由器不厌其烦的翻阅着路由表，而找到的下一跳地址不由得让他眼前一亮。“很久没遇到一个要去这种局域网的帧了。IBM令牌环网络，有趣”。说罢路由器拿起了另一本厚厚的书，里面写满了IBM令牌环网络的知识。他飞快地翻着，略带生疏的把这个帧组装好，此时令牌恰好传到了他的手里。“下一个”

网络层的结果 解决了异构网络的通信问题：基于IP地址和路由器 向上提供基于IP地址的通信 尽最大努力交付：只发送一次，能不能到达看运气 传输层

端口概念 实际上的网络通信的两个主机之间的应用（进程）之间的通信 端口用来区分同一主机上的不同应用 简单的理解就是一个进程占用一个端口号

两个协议 UDP协议

面向无连接：不需要对方主机在线，当然只有在线才能收到 面向报文：一次发送一段数据 不可靠：发出去就不管了（不管能不能到，不管能不能有序到达） TCP协议

面向连接：需要对方主机在线，并建立连接 面向字节流：你给我一堆字节的数据，我给你发过去，但一次发多少我说了算 可靠：我有机制可以保证你的数据一定会有序的到达对方主机

TCP 一次发送一个TCP报文段，由TCP首部和数据部分组成 TCP首部至少包括三个信息：源端口，目标端口，序号，数据偏移

序号是用来编号字节的。TCP是面向字节流，也就是你给我一段有序的字节，我对其编号，比如从x到x+n，TCP内部机制不一定全部发送出去，而会选择一部分，比如x到x+m这m个字节作为数据部分，加上TCP首部组成一个报文段，这个报文段又作为数据部分交给网络层 数据偏移用来指出TCP的数据部分的起始位置在整个TCP报文段（首部+数据）的位置（也就是首部长度） 什么。不知道数据部分多长。还记得网络层IP数据报的首部吗，长度可以根据这里面的总长度算出来，因为整个TCP报文段是作为IP数据报的数据部分。 连接

TCP建立连接的两个点是(ip1,port1）和(ip2,port2） ip1==ip2时，即在本主机的不同进程间通信，此时port1和port2一定不同 ip1!=ip2时，即在两台主机之间通信，port1和port2不一定不同

TCP很棒 为什么。因为TCP可靠啊。 那么TCP是如何实现可靠。简单的理解就是——编号+重传。 重传保证数据一定能到

TCP每发一个数据，就会期待接收方的一个回复。如果在指定时间内收到了回复，就知道数据确实达到了 如果超过一定的时间没收到回复，就认为对方没有收到，所以再发送一遍 编号保证数据有序

TCP面向字节流，发送方每次选出一段字节发送的时候，都会带上一个序号，这个序号就是发送的这段字节中编号最小的字节的编号 接收方回复的时候，也会带上一个序号，这个序号是接收方期望接收到的下一个字节的编号 其他也很重要

流量控制 拥塞控制

传输层的结果 提供应用进行通信的socket编程接口，屏蔽了大量的传输细节，你只需要给它目的地和数据就好了，应用程序利用这个接口来进行网络

通信 有没有不基于socket的通信。有啊，在传输层之下的通信，比如ARP、ICMP等 应用层

# 参考文献

1. 竹下隆史, *图解 TCP/IP*. 5th ed. 图灵程序设计丛书. 2013, 北京: 人民邮电出版社.

2. Kurose, J.F. and K.W. Ross, *计算机网络：自顶向下方法*. 6th ed. 计算机科学丛书. 2014, 北京: 机械工业出版社.

3. https://yq.aliyun.com/ziliao/351916