|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 源端口（16） | | | | | | | | 目的端口（16） |
| 序号（32） | | | | | | | | |
| 确认号（32） | | | | | | | | |
| HLEN（4） | 保留（6） | URG | ACK | PSH | PST | SYN | FIN | 窗口大小（16） |
| 校验和（16） | | | | | | | | 紧急指针（16） |
| 选项与填充 | | | | | | | | |
| 数据（16的整数倍） | | | | | | | | |

TCP段首部格式

ARP请求应答格式

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 目的MAC（6） | 源MAC（6） | 帧类型（2） | 硬件类型（2） | 协议类型（2） | 物理地址长度 | 协议地址长度 | 操作（2） | 发送端MAC | 发送端IP（4） | 接收端MAC | 接收端IP（4） |

IP数据报格式

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本4 | 首部长度4 | 服务类型TOS8 | 总长度16 | |
| 标识符16 | | | 标志3 | 片偏移16 |
| 生存时间TTL8 | | 类型8 | 首部检验和16 | |
| 源IP32 | | | | |
| 目的IP32 | | | | |
| 选项 | | | | |
| 数据 | | | | |

1、TCP传输控制协议：

为两个任意处理速率的，使用不可靠IP连接机制的机器之间的通信提供了可靠的、具有流量控制的、端到端的数据流服务。

2、为什么我们要在IP层里面实现ICMP的差错报告？为什么不在MAC层呢？

ip层本身是不可靠传送，不能提供可达性、交付错误、路由错误报告以及控制信息，当路由或可达性问题阻止交付ip数据包时icmp可以提供返回出错信息的方法，这种能力很好地补充了ip的数据包交付服务。

3、这么多网络协议，为什么要分层？网络协议分层的好处有哪一些？TCP分层的功能？

网络协议之所以分层描述，是由于在实际的计算机网络中，两个实体之间的通信情况非常复为了降低通信协议实 现的复杂性，而将整个网络的通信功能划分为多个层次（分层描述），每层各自完一定的任务，而且功能相对独立，这样实现起来较容易。

分层的好处：

a.有利于将复杂的问题分解为多个简单的问题，分而治之

b.有利于网络的互联，进行协议转换时可能只涉及一个或几个层次而不是所有层次。

c.屏蔽下层的变化，新的底层技术的应用，不会对上层的协议产生影响。

分层功能：应用层：处理特定应用程序的细节

传输层：为两台主机上的应用程序提供端到端的通信

网络层：处理分组在网络中的活动

链路层：处理传输媒介的物理接口细节

4、ARP里面，ARP的范围通信，本质是什么？——APR的起因。

ARP是建立在网络中各个主机互相信任的基础上的，网络上的主机可以自主发送ARP应答消息，其他主机收到应答报文时不会检测该报文的真实性就会将其记住本机ARP缓存。

5、ARP的欺骗攻击的原理是什么？

ARP攻击就是通过伪造IP、地址和MAC地址实现ARP欺骗，能够在网络中产生大量的ARP通信量使网络阻塞，攻击者只要持续不断的发出伪造的ARP响应包就能更改目标主机ARP缓存中的IP-MAC条目，造成网络中断或中间人攻击。

TCP/IP:

应用层：协议：HTTP Telent FTP SMTP

功能：按照不同应用的特定要求和方式负责把数据传输到传输层或者接受从传输层返回的数据

传输层：协议：TCP UDP

功能：TCP为两台主机提供高可靠性的数据通信，UDP则为应用层提供一种非常简单的服务，它只是把数据报的分组从一台主机发送到另一台主机但并不保证该数据报能到达另一端

网络层：协议：IP ICMP IGMP

功能：为数据包选择路由，其中IP是TCP/IP协议族中最为核心的协议，所有的TCP UDP ICMP IGMP数据都是以IP数据报格式传输

链路层：协议：ARP RARP和设备驱动程序

功能：发送时将IP包作为帧发送，接收时把接收到的比特组装成帧，提供链路管理错误检测

6、TCP建立连接三次握手过程

①由要建立连接的客户向服务器发送连接请求段，该段首部的同步标志SYN被置为1，并在首部中填入本次连接的客户端初始段序号SEQ（如SEQ=16400）

②服务器收到请求后，发回连接确认（SYN+ACK）,该段首部中的同步标志SYN被置为1，表示认可连接，首部中的确认标志ACK被置为1，表示对所接收的段的确认，与ACK标志相配合的是准备接收的下一序号（ACK16401）,该段还给出了自己的初始序号（如SEQ=1300）,对请求段的确认完成了一个方向上的连接

③客户向服务器发出确认段，段首部中的确认标志ACK被置为1，表示对所接收到的段的确认，与ACK标志相配合的准备接收的下一序号被设置为收到的段序号加1（ACK1301）,完成了另一个方向上的连接。