**华中科技大学计算机学院**

**《计算机通信与网络》实验报告**

实验名称 分析Ethernet II帧、分析集线器和交换机工作原理

|  |  |  |  |
| --- | --- | --- | --- |
| 姓 名 | 班 级 | 学 号 | 得 分 |
| 潘翔 | IOT1601 | U201614898 |  |

教师评语：

# Lab3 分析Ethernet II帧、集线器和交换机工作原理

## 3.1 环境

操作系统： Manjaro-4.18.10-1 x86\_64 (Arch-Based Distribution)

网络平台： Wireshark 2.6.3

网络环境：

Link encap:Ethernet  HWaddr a0:8c:fd:24:5d:4c     
inet addr:222.20.100.153  Bcast:222.20.101.255  Mask:255.255.254.0   
inet6 addr: fe80::2476:27:cd9d:d75b/64 Scope:Link   
inet6 addr: 2001:250:4000:803c:e3c1:b69:d9f2:67b0/64 Scope:Global

## 3.2 实验目的

1. 分析Ethernet II帧
   1. 深入理解Ethernet II帧结构。
   2. 基本掌握使用Wireshark分析俘获的踪迹文件的基本技能。
2. 分析集线器和交换机工作机理
   1. 观察交换机处理广播和单播报文的过程。
   2. 比较交换机与集线器工作过程。
   3. 掌握使用Packet Tracer模拟网络场景的基本方法，加深对网络环境、 网络设备和网络协议交互过程等方面的理解。

## 3.3 实验内容及步骤

1. 分析Ethernet II帧
   1. 分析踪迹文件中的帧结构

用Wireshark俘获网络上收发分组或者打开踪迹文件，选取感兴趣的帧进行分析。

* 1. 分析以太帧结构

1. 分析集线器和交换机工作机理
   1. 在PacketTracer中配置网络拓扑
   2. 观察交换机如何处理广播和单播报文
   3. 观察交换机如何处理未知单播

## 3.4 实验结果

### 3.4.1 分析Ethernet II帧

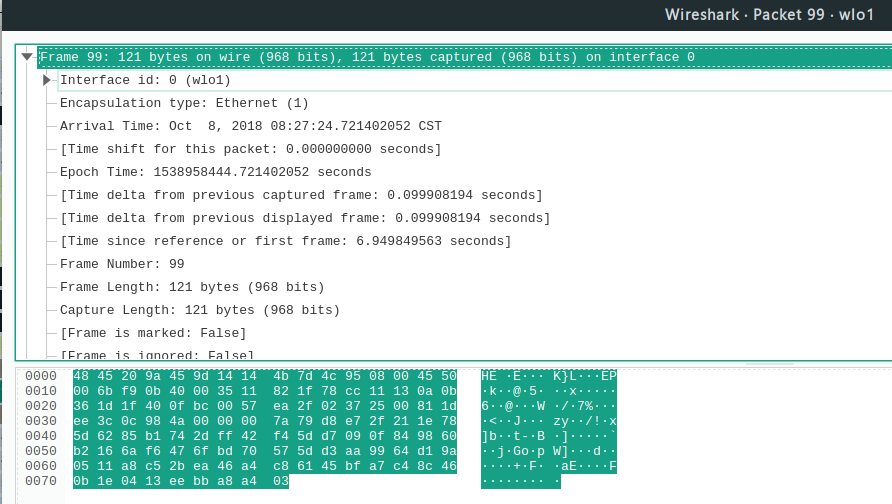


图3-1 物理层Frame结构

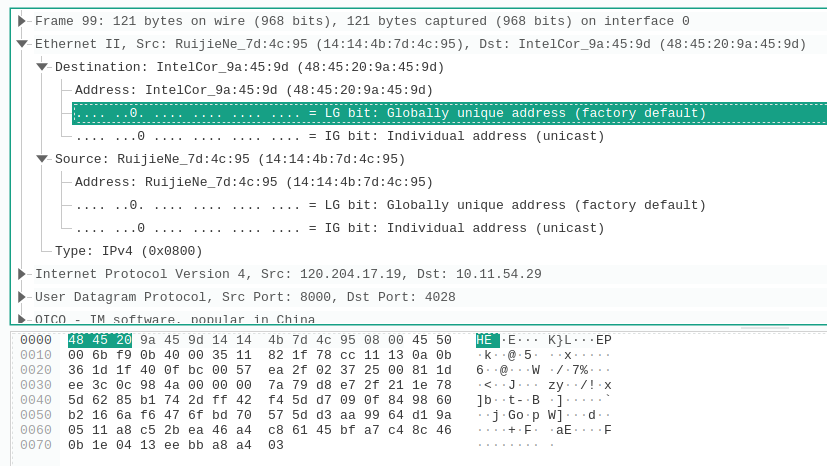


图3-1 链路层Ethernet II帧结构

可以看到，报文结构为，Frame结构为整个链路层传输的PDU，封装Ethernet II作为链路层PCI,此时传输层IP报文段作为链路层的SDU。

1. 目标MAC地址：

Destination: IntelCor\_9a:45:9d (48:45:20:9a:45:9d)

我们可以看到其目标MAC地址品牌为IntelCor。

此时需要对是否为同一子网进行区分，此时MAC地址涉及arp过程

A ping B

如果在同一网段：

* 1. 主机先会查看自己的ARP高速缓存中是否有B主机的MAC地址记录。
     1. 如果A主机的高速缓存中有B主机的记录，则直接通过这个MAC地址进行数据的传输。
     2. 如果A主机的高速缓存中没有B主机的记录，则会向局域网的所有主机广播一个ARP请求，寻找B主机  
        的MAC地址。
  2. 当B主机收到A主机广播的ARP请求后，就会直接给A主机回复一个ARP数据包。
  3. 当A主机收到B主机发送过来的请求后，将B的MAC地址写入高速缓存中，然后通过该MAC地址，A主机向B主机进行数据的传输。

如果不在同一网段：

网关把目标MAC改成下一跳路由器的MAC地址（通过ARP得到），而源端MAC改成发出端口的MAC地址，否则下一跳路由器收到目标MAC不是自己的数据包，会丢弃不予理睬，下一跳路由器再发给下一跳路由器同样要把目标MAC地址改为下一跳路由器的MAC地址再发出去。

故此处MAC地址不为实际的服务器MAC地址，而为网关地址。

1. 源目标MAC地址：

Source: RuijieNe\_7d:4c:95 (14:14:4b:7d:4c:95)

可以看到前缀反映了物理网卡的品牌信息，而这种信息对应了一定的前缀MAC地址编码。

在MAC地址中，使用LG Bit/IG Bit进行标识MAC地址类型

1. 网络协议类型：IPV4

如果为IPV6协议2字节以太类型字段中值为0x0800，它表示以太帧包含了 IP数据报，如果其中封装了IPv6协议，其值应为0x86DD。

1. 用户数据报文UDP
2. 应用层报文 OICP

### 3.4.2 分析集线器和交换机工作机理

1. 网络拓扑图及IP配置
   1. 集线器测试

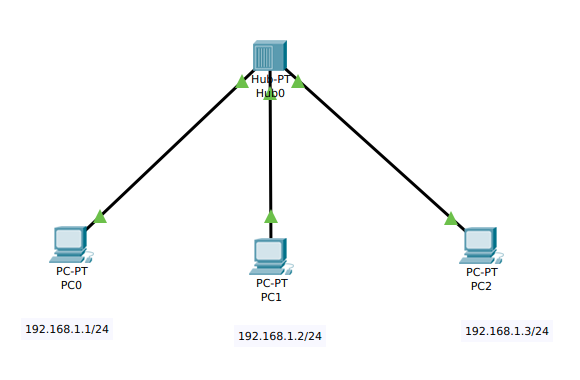


图3-1 集线器网络拓扑图

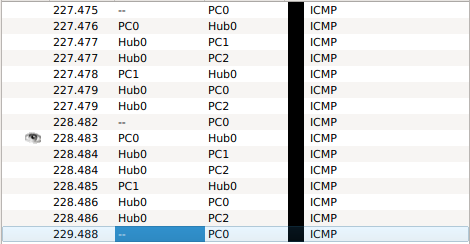


图3-1 PC0 PING PC1测试

从图中可以看出，集线器采用广播模式，对于报文对所有的端口进行转发。

* 1. 交换机环路测试

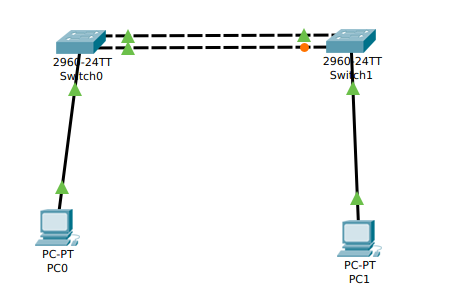


图3-1 交换机环路测

* + 1. 默认情况下STP协议是启用的。通过两台交换机之间传送BPDU协议数据单元。选出跟交换机、根端口等，以便确定端口的转发状态。图中标记为黄色的端口处于block堵塞状态。
    2. 设置RSTP。
    3. 查看交换机show spanning-tree状态，了解跟交换机和根端口情况。

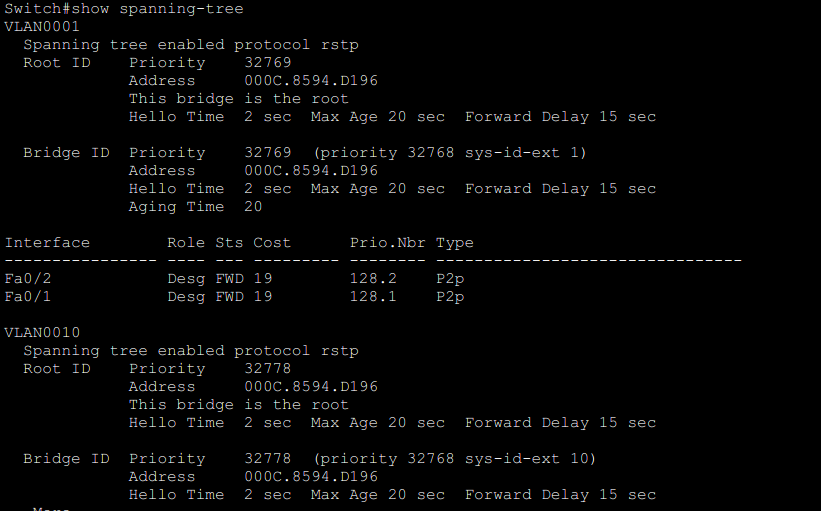


图3-1 查看生成树协议

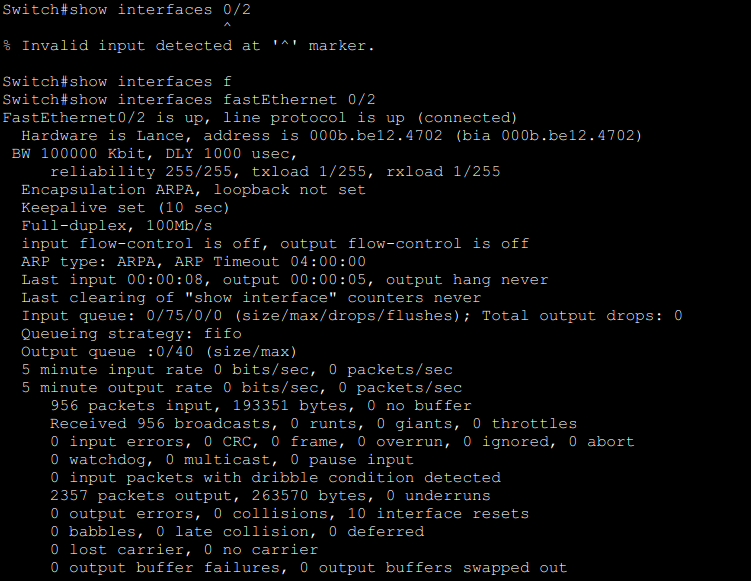


图3-2 查看端口流量

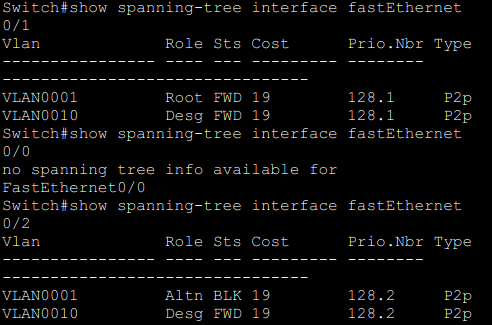


图3-3 查看端口状态

返回参数说明：

StpVersion : RSTP 生成树协议的版本

SysStpStatus : Enabled 生成树协议运行状态，disable 为关闭状态

Priority : 32768 交换机的优先级

RootCost : 200000 交换机到达根交换机的开销

RootPort : Fa0/1 交换机上的根端口

或：

RootCost: 0 交换机到达根交换机的开销，0 代表本交换机为根

RootPort: 0 交换机上的根端口，0 代表本交换机为根

可以看到0/1端口，即上端的端口具有较高的优先级(较小的Priority)，故为开启状态。而0/2端口为阻塞状态。

* + 1. 通过更改交换机生成树的优先级spanning-tree vlan 10 priority 4096可以变化跟交换机的角色。
    2. 测试。当主链路处于down状态时候，能够自动的切换到备份链路，保证数据的正常转发。

可以看出，在刚刚进行连接的时候，交换机其中的链路表未收敛，需要一段时间。

1. 广播报文
2. 单播报文

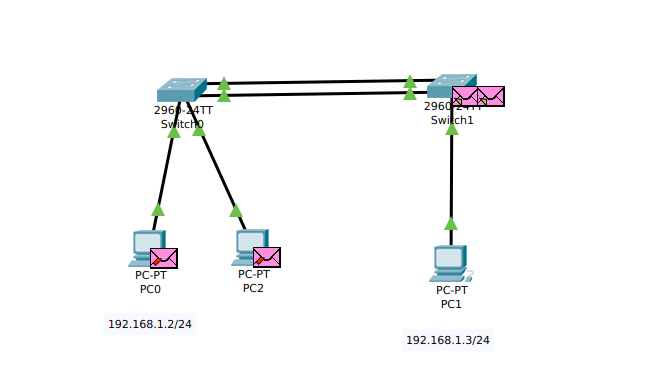


图3-4 广播报文

1. 未知单播报文

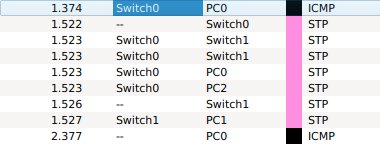


图3-5 初始交换机未知单播测试

### 3.4.3 单步调试集线器及广播报文

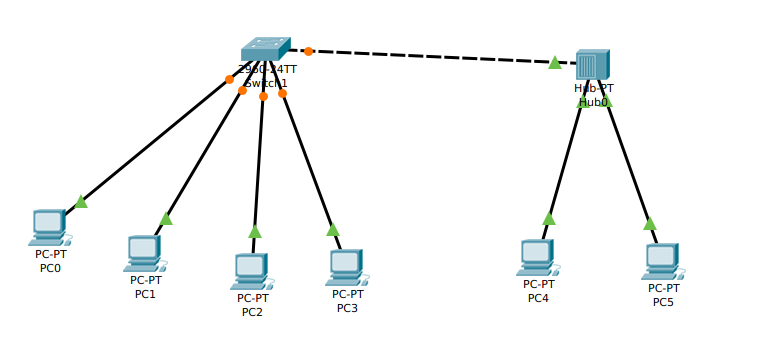


图3-6 网络的初始状态

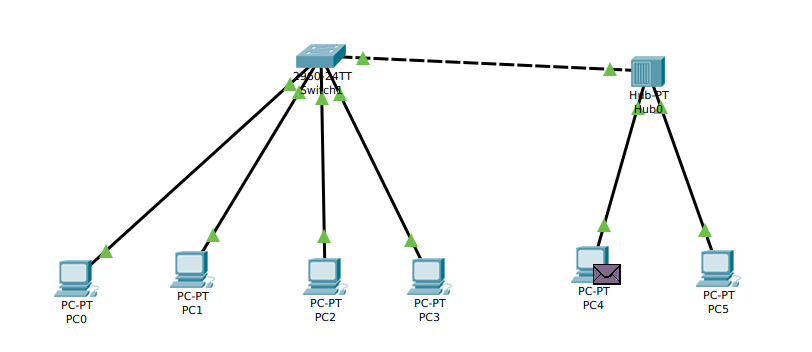


图3-7 默认执行STP后的状态

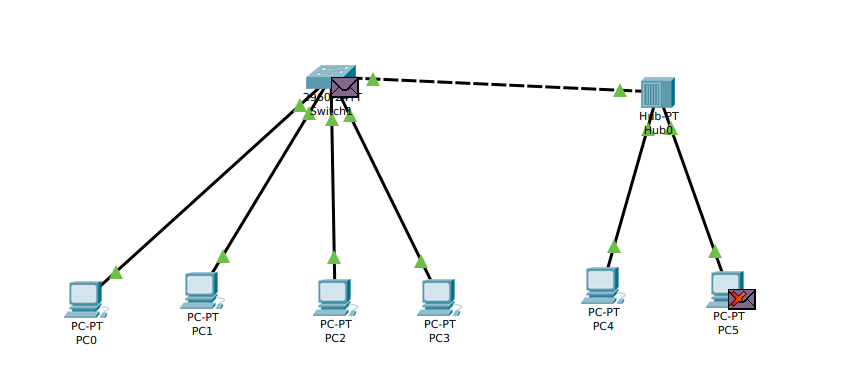


图3-8 集线器所有端口转发

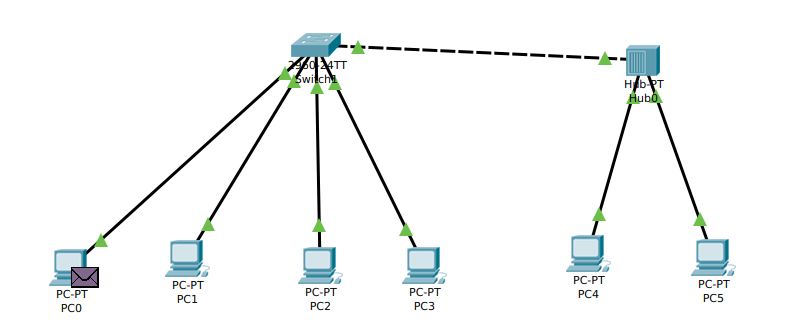


图3-9 已知PC0 MAC地址的情况下直接进行转发

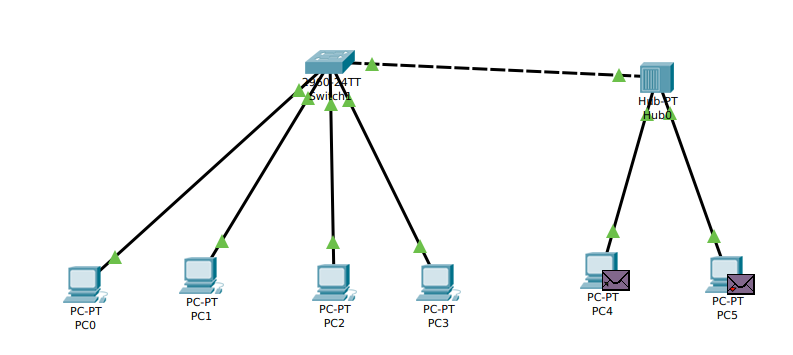


图3-10 回复报文再次进行转发

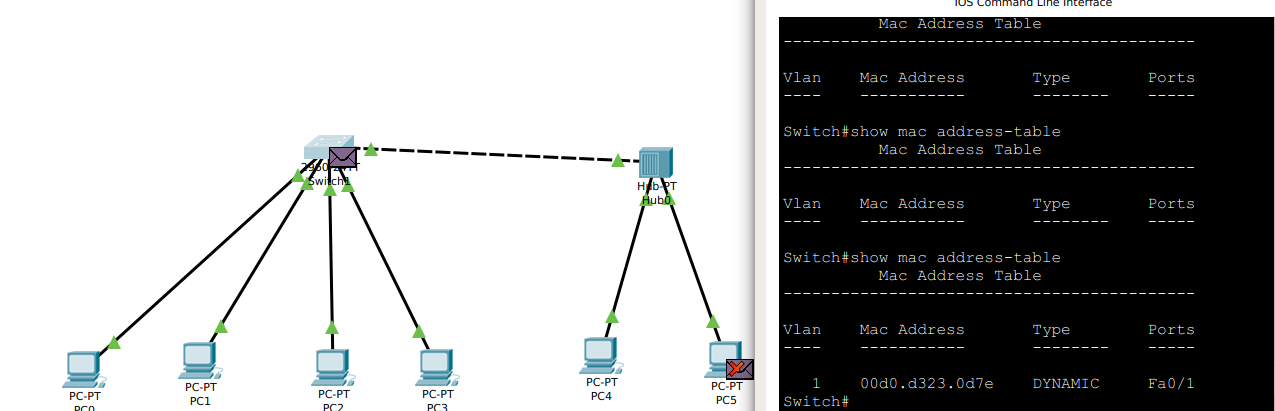


图3-11 此时添加PC5 MAC地址

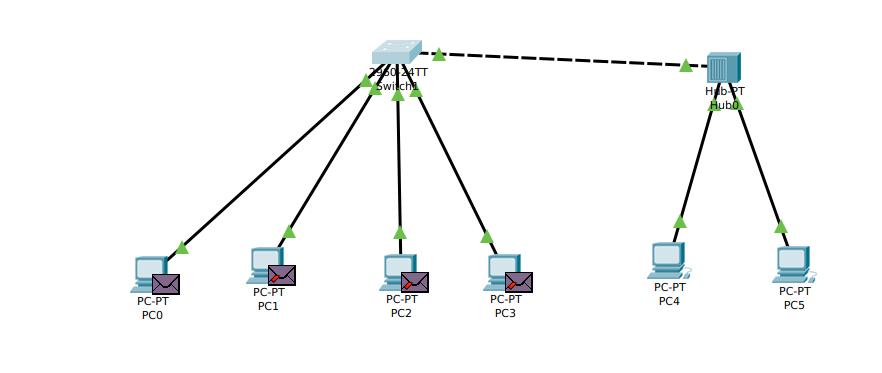


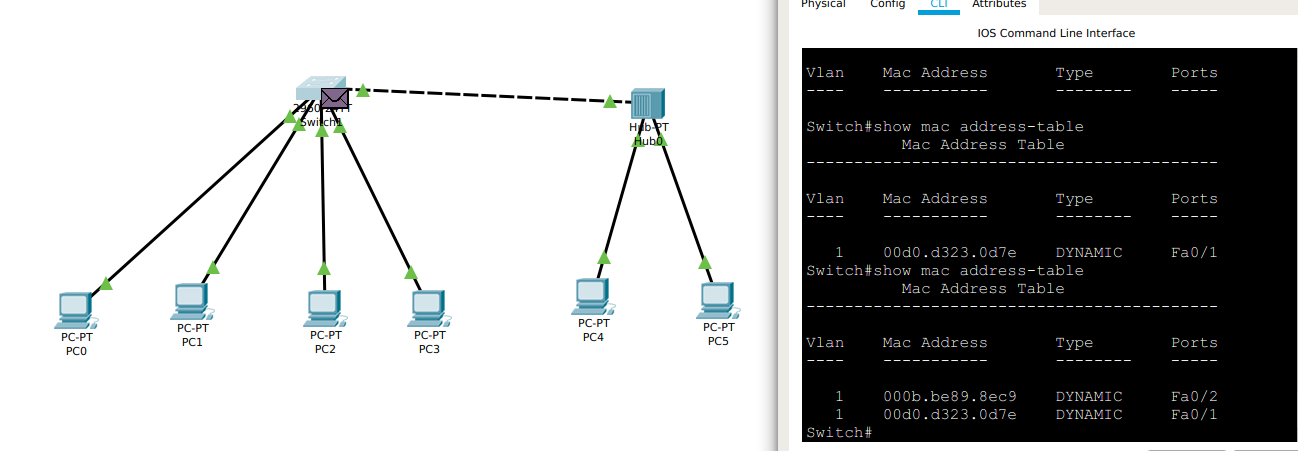
图3-12 广播报文

图3-13此时添加PC0 MAC地址

集线器工作在物理层，仅对电信号进行放大整形并向所有端口转发，不识别数据链路层的帧，不执行CSMA/CD协议。

## 3.5 实验中的问题及心得

在网络拓扑的架构中，比较标准的操作是先进行配置再进行连线，类似于电路中的先进行电路设置，再进行连线操作，因为不配置的话，容易造成广播风暴，造成交换机长时间不收敛。

普通集线器广播策略不传播至本身的原因是集线器为物理结构，不提供控制功能，故其中的线连接方式直接通过物理实现，自然不会形成回环，而在实验中也得到了验证。

而交换机采用转发表的过程，其中可能会形成环路，在未进行生成树配置的时候，可以看到存在环路的存在，默认STP开启进行处理，但是在较大的网络中国存在收敛较慢，且链路强制阻塞，效率并不高，采用RSTP进行尝试，可以看到在根节点选择不同的情况下，会有不同的链路生成树，在进行配置的时候可验证，不同的vlan共享同一个生成树，并不可按照vlan阻塞链路。

实验过程中熟悉了关于集线器，路由器，交换机不同层级的概念，加深了网络层次的理解。

## 参考文献

1. CiscoPacketTracer网络实验手册

**华中科技大学计算机学院**

**《计算机通信与网络》实验报告**

实验名称 分析IP和ARP协议

|  |  |  |  |
| --- | --- | --- | --- |
| 姓 名 | 班 级 | 学 号 | 得 分 |
| 潘翔 | IOT1601 | U201614898 |  |

教师评语：

# Lab4 分析IP和ARP协议

## 4.1 环境

操作系统： Manjaro-4.18.10-1 x86\_64 (Arch-Based Distribution)

网络平台： Wireshark 2.6.3

网络环境：

inet 222.20.100.153  netmask 255.255.254.0  broadcast 222.20.101.255   
inet6 2001:250:4000:803c:e9a0:1b72:2be3:b69e  prefixlen 64  scopeid 0x0<global>   
inet6 fe80::3781:b076:5845:ec9d  prefixlen 64  scopeid 0x20<link>   
ether a0:8c:fd:24:5d:4c  txqueuelen 1000  (Ethernet)

## 4.2 实验目的

1. 深入理解IP报文结构和工作原理。
2. 深入理解ARP协议的工作原理：

理解IP和以太网协议的关系，掌握IP地址和MAC地址的映射机制，搞清楚IP报文是如何利用底层的以太网帧进行传输的。

## 4.3 实验内容及步骤

1. 分析IP报文结构
   1. 分析俘获的分组
   2. 分析IP报文结构
2. 分析ARP协议
   1. 查看本机因特网硬件地址
   2. 使用ARP命令
   3. 分析ARP协议工作过程

## 4.4 实验结果

### 4.4.1 分析IP协议

1. 选取报文分析

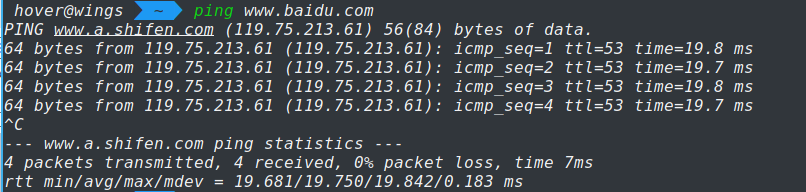


图4-1 Ping [www.baidu.com](http://www.baidu.com)

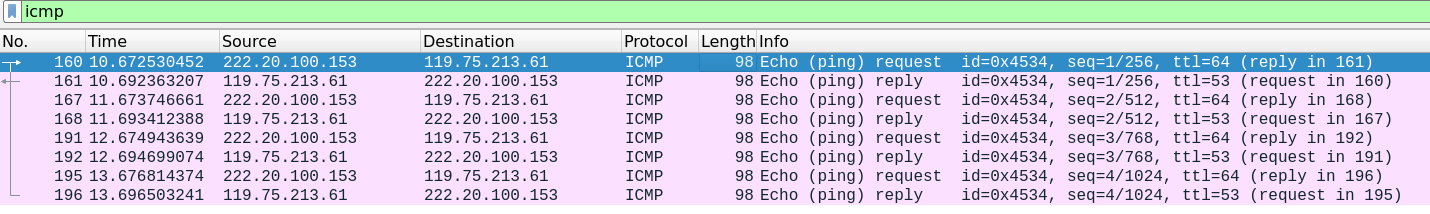


图4-2 Ping [www.baidu.com](http://www.baidu.com) wireshark抓包

1. 分析IP报文结构

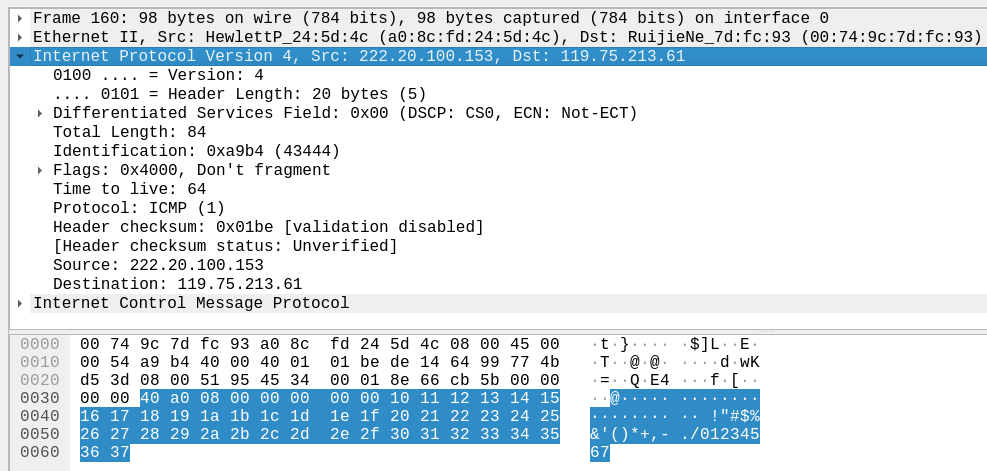


图4-3 IP部分报文选取

可以看到，报文结构为，Frame结构为整个链路层传输的PDU，封装Ethernet II作为链路层PCI,此时传输层IP报文段作为链路层的SDU，在之后

Frame报文描述：

Frame 160: 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface 0

Ethernet II报文描述：

Ethernet II, Src: HewlettP\_24:5d:4c (a0:8c:fd:24:5d:4c), Dst: RuijieNe\_7d:fc:93 (00:74:9c:7d:fc:93)

IP报文描述：

0100 .... = Version: 4

.... 0101 = Header Length: 20 bytes (5)

Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)

差分字段：

表4-1差分字段类型

| 业务类型 | DSCP PHB | DSCP值 |
| --- | --- | --- |
| Network Control | CS7(111000) | 56 |
| IP Routing | CS6(110000) | 48 |
| Interactive Voice | EF(101110) | 46 |
| Interactive Video | AF41(100010) | 34 |
| Video control | AF31(011010) | 26 |
| Transactional/Interactive(对应高优先级应用) | AF2x(010xx0) | 18、20、22 |
| Bulk Data(对应中优先级应用) | AF1x(001xx0) | 10、12、14 |
| Streaming Video | CS4(000100) | 4 |
| Telephony Signaling | CS3(000011) | 3 |
| Network Management | CS2(000010) | 2 |
| Scavenger | CS1(000001) | 1 |
| Best Effort | 0 | 0 |

此时为Best Effort，在传统的IP网络中多为Best Effort，对报文传送的可靠性、传送延迟等性能不提供任何保证。

而在特殊的情形下，需要对QoS进行保证，故需要采用标志位。

且提供了IPV4与IPV6的服务类型接口，同时在骨干网和边缘网的同时可能需要进行服务类型的转换，此时此标志位需要映射。

Total Length: 84

Identification: 0xa9b4 (43444)

标识：

当数据报由于长度超过网络的MTU而必须分片时，这个标识字段的值就被复制到所有的数据报片的标识字段中。相同的标识字段的值使分片后的各数据报片最后能正确地重装成为原来的数据报。

Flags: 0x4000, Don't fragment

1. .. .... .... .... = Reserved bit: Not set

.1.. .... .... .... = Don't fragment: Set

..0. .... .... .... = More fragments: Not set

...0 0000 0000 0000 = Fragment offset: 0

Time to live: 64

Protocol: ICMP (1)

Header checksum: 0x01be [validation disabled]

[Header checksum status: Unverified]

Source: 222.20.100.153

Destination: 119.75.213.61

之后为数据段部分，即IP协议的SDU，为整个ICMP报文段。

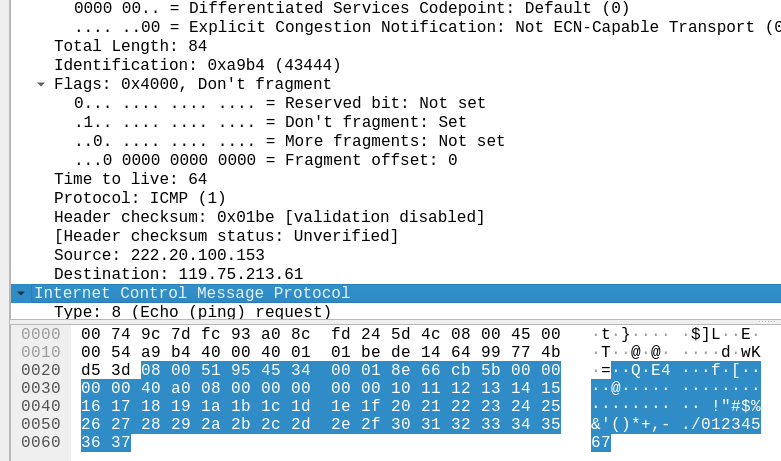


图4-4 ICMP报文段

### 4.4.1 分析ARP协议

1. 查看本机MAC地址

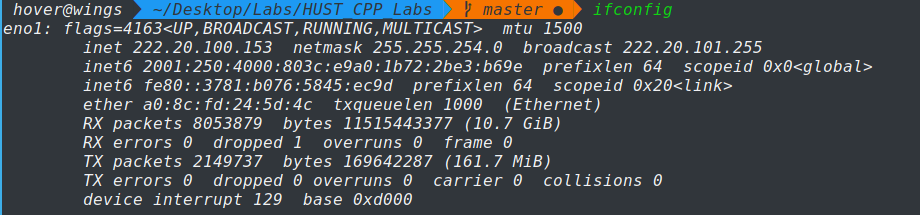


图4-5 查看本机MAC地址图

1. 查看本机ARP表

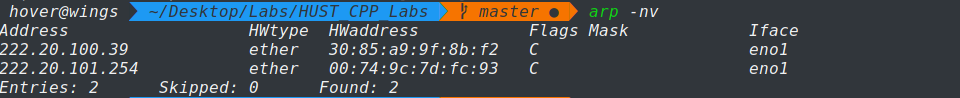


图4-6 查看本地arp表图

1. 清除ARP表

可使用ip neigh flush dev <dev>

或者arp -d 删除

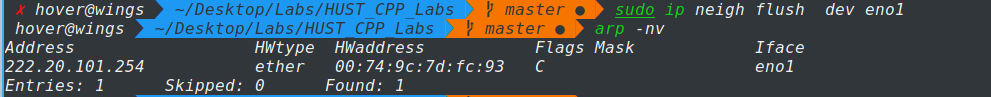


图4-7 删除本地arp表图



图4-8 删除本地arp表验证图

此时为了测试ARP表清空之后的重建过程，因为网络环境的差异，与下面实验环境稍有不同。

1. arp过程分析

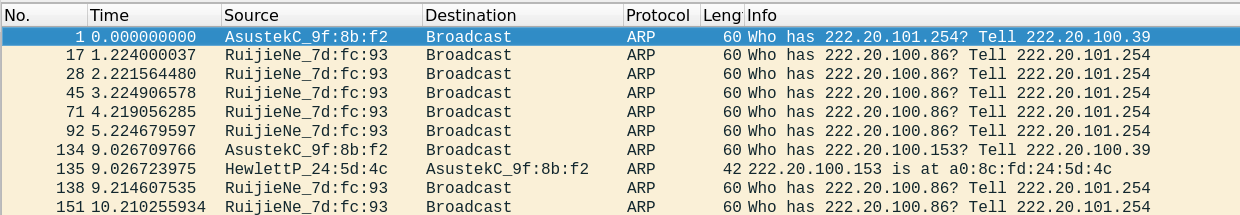


图4-9 本地arp报文抓取图

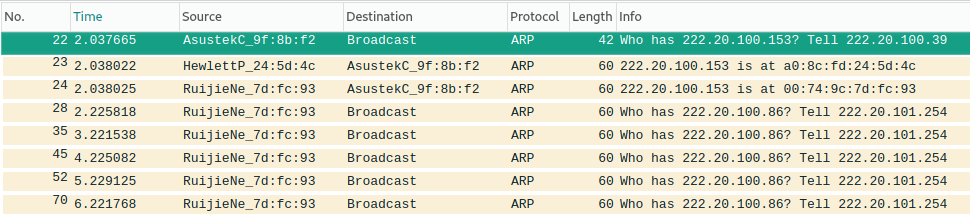


图4-10 目标主机arp报文抓取图

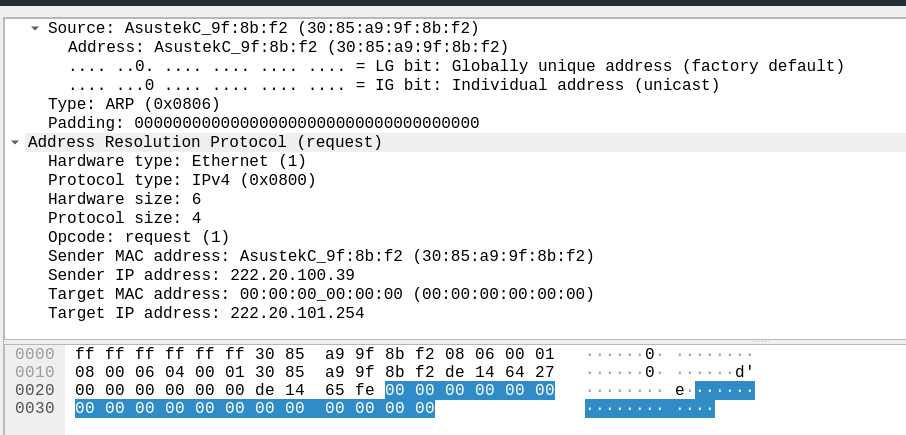


图4-11主机arp报文字段图

Hardware type: Ethernet (1)

Protocol type: IPv4 (0x0800)

Hardware size: 6

Protocol size: 4

Opcode: request (1)

Sender MAC address: RuijieNe\_7d:fc:93 (00:74:9c:7d:fc:93)

Sender IP address: 222.20.100.39

Target MAC address: 00:00:00\_00:00:00 (00:00:00:00:00:00)

Target IP address: 222.20.101.254

只知道IP地址而不知道广播地址，故MAC地址为缺省值

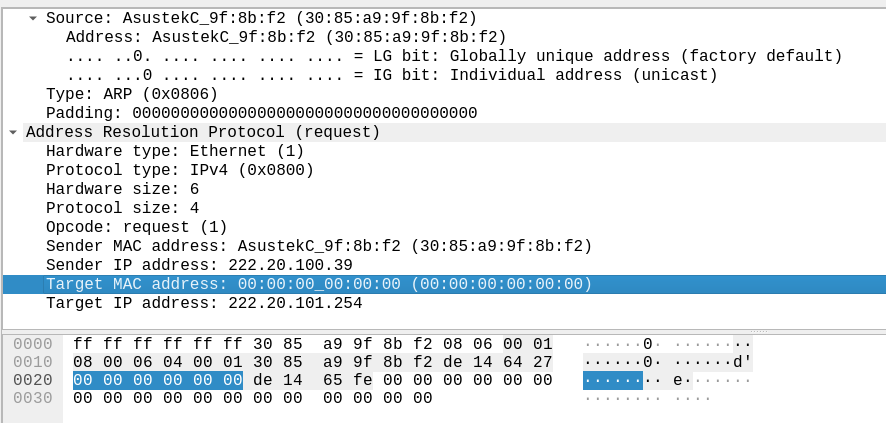


图4-12 MAC地址图

路由器接受到这样的报文，不会向网络进行转发这样的广播包的，在子网内转发包的时候，MAC地址会被修改后再转发。

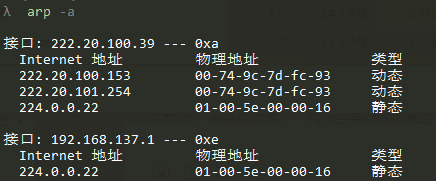


图4-13 接收端arp表图

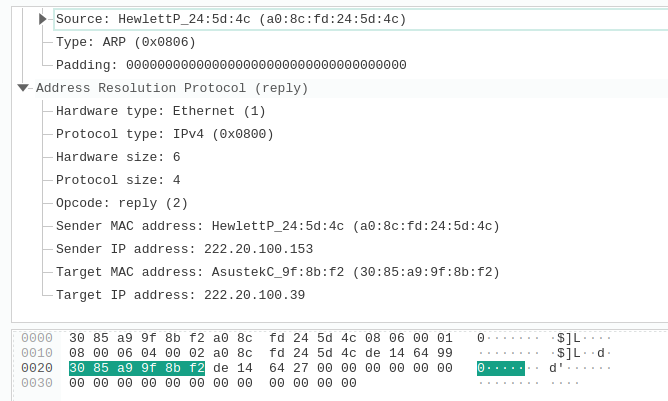


图4-14 arp回复报文图

此时已经填充目标MAC地址，不过此时为网关的MAC地址。

### 4.4.3 相关问题

1. IP地址

222.20.100.153

1. IP数据报首部，较高层协议字段中的值是什么？

icmp报文，较高层协议字段的值为1

1. IP首部字节，载荷字节

.... 0101 = Header Length: 20 bytes (5)

Total Length: 84

载荷字段有64个字节

1. 是否分段？

.1.. .... .... .... = Don't fragment: Set

1. 高层协议有用信息
   1. Hardware type
   2. Protocol type
   3. Hardware size
   4. Protocol size
   5. Opcode
   6. Sender MAC address
   7. Sender IP address
   8. Target MAC address
   9. Target IP address

## 4.5 实验中的问题及心得

在实验的过程中，曾经出现过主机A Ping 主机B超时，主机B Ping 主机A成功，此时主机A为Linux环境，B为Windows环境，考虑二者网络管理策略不同，两台主机ARP表有表项更新的情况，经过检查，当主机B开启系统自带防火墙时，作为目的节点被Ping会出现超时的情况，防火墙将ping分组拦截了下来，导致返回超时，但不影响其他数据包的通讯，故该主机仍会返回ARP应答报文，也正因如此两者能正常通讯。

在arp报文中，我们知道其MAC地址会被网关替换，实际上这是网络下层对上层透明的思想，即底层提供服务，而无需知道如何实现，对于链路层，只需要知道转发的链路即可，是不是最终的MAC地址，不重要。但此时想到该如何获取实际的目标MAC地址呢？通过查阅资料，可使用NetBIOS协议进行询问，获取远程主机的实际MAC地址。在早期windows环境下，提供nbstat命令进行同一子网的实际MAC地址查询，而如果非同一子网，则可采用封装“UDP-NetBIOS-NS”询问包进行问询。但在实际过程中，同一子网关心MAC地址，不同子网关心IP，才是正常的网络策略，所有的限制和管理也是基于此原则进行。

## 参考文献

1. 实验文档
2. IP 网络 QoS

<https://blog.csdn.net/jianchaolv/article/details/7926537>