

暨南大学本科实验报告专用纸

课程名称 计算机网络实验 成绩评定 _____
实验项目名称 IEEE802协议和以太网 指导老师 某某某
实验项目编号 1 实验项目类型 综合类 实验地点 N117
学生姓名 某某 学号 XXXXXXX
学院 网络空间安全学院 系 _____ 专业 网络空间安全
实验时间 2020 年 10 月 7 日 晚 上 ~ 2020 年 10 月 7 日 晚 上

(一) 实验目的

- 掌握以太网的报文格式
- 掌握 MAC 地址的作用
- 掌握 MAC 广播地址的作用
- 掌握 LLC 帧报文格式
- 掌握协议编辑器和协议分析器的使用方法
- 掌握协议栈发送和接收以太网数据帧的过程

(二) 实验环境

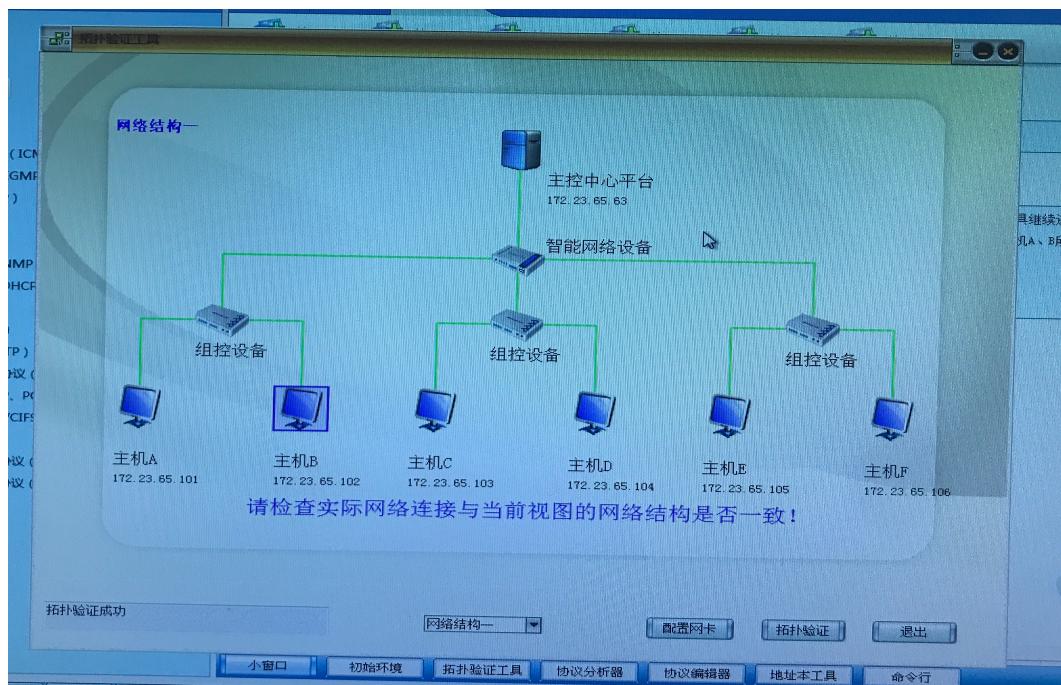


图 1：实验环境（该实验采用网络结构一，本机为主机 B）

(三) 实验原理

1. 关于 IEEE802 协议

IEEE 802 系列标准是 IEEE 802 LAN/MAN 标准委员会制定的局域网、城域网技术标准。其中最广泛使用的有以太网、令牌环、无线局域网等。这一系列标准中的每一个子标准都由委员会中的一个专门工作组负责。

2. 关于 MAC 地址

百度百科这么说：MAC 地址（英语：Media Access Control Address），直译为媒体存取控制位址，也称为局域网地址（LAN Address），MAC 位址，以太网地址或物理地址，它是一个用来确认网络设备位置的位址。在 OSI 模型中，第三层网络层负责 IP 地址，第二层数据链路层则负责 MAC 位址。MAC 地址用于在网络中唯一标示一个网卡，一台设备若有一或多个网卡，则每个网卡都需要并会有一个唯一的 MAC 地址。

快件到达【广州番禺化龙网点】 2020-09-28 09:12:16
快件离开【番禺南村集散点】已发往【广州番 禺化龙网点】 2020-09-28 03:17:49
快件到达【番禺南村集散点】 2020-09-28 03:09:58
快件离开【广州花都转运中心】已发往【番禺 南村集散点】 2020-09-27 23:28:52
快件到达【广州花都转运中心】 2020-09-27 23:24:31
快件到达【广州花都转运中心】 2020-09-27 23:24:16
快件离开【上海浦西转运中心】已发往【广州 花都转运中心】 2020-09-26 23:31:43
快件到达【上海浦西转运中心】 2020-09-26 23:29:43
快件离开【上海闵行申闵网点】已发往【上海 浦西转运中心】

延长收货

确认收货

我的理解：IP 地址类似于快递的发送地点和接收地点，而 MAC 地址就类似于各个快递网点，比如从上海闵行区某个网点到暨大南校区的菜鸟驿站，IP 只有标识这两个网点的地址，中间需要经过很多网点的转发，而联系到网络传输，网点的转发就正好代表链路两端的传输。但不巧的是，网络传输中并不一定知道下一个链路（“网点”）在哪里，但是根据路由表它知道下一站的 IP(ARP 帧中的 IP 是路由表下一跳的 IP，并非总目的地 IP)，所以它就先发一个 ARP 帧咨询一下（广播），收到讯息的主机将 IP 与自己的对比，若相同便回复告知自身的 MAC 地址，数据包就能知道下一跳的具体位置。

3. 关于 ping 命令

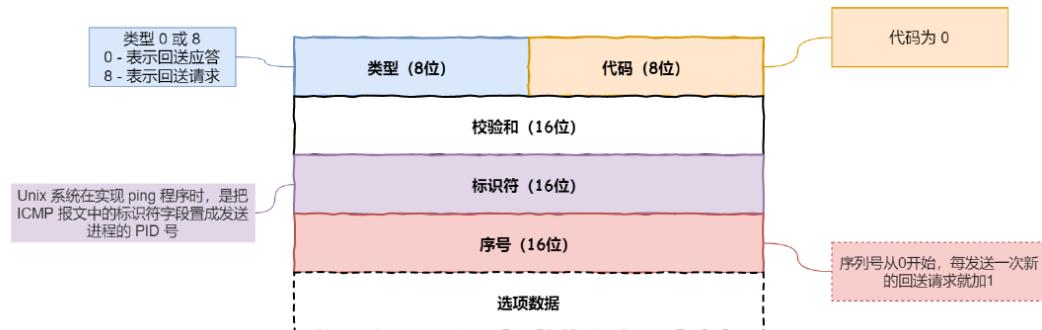
ping 是基于 ICMP 协议（不可靠协议）工作的，所以要明白 ping 的工作原理，首先应先熟悉 ICMP 协议。ICMP 全称是 Internet Control Message Protocol，也就是互联网控制报文协议。

ICMP 主要的功能包括：确认 IP 包是否成功送达目标地址、报告发送过程中 IP 包被废弃的原因和改善网络设置等。在 IP 通信中如果某个 IP 包因为某种原因未能达到目标地址，那么这个具体的原因将由 ICMP 负责通知。

ICMP 报文是封装在 IP 包里面，它工作在网络层，是 IP 协议的助手。 ICMP 包头的类型字段，大致可以分为两大类：

- 一类是用于诊断的查询消息，也就是「查询报文类型」
- 另一类是通知出错原因的错误消息，也就是「差错报文类型」

查询报文包括回送消息，回送消息用于进行通信的主机或路由器之间，判断所发送的数据包是否已经成功到达对端的一种消息，ping 命令就是利用这个消息实现的。回送信息可以向对端主机发送回送请求的消息（ICMP Echo Request Message，类型 8），也可以接收对端主机发回来的回送应答消息（ICMP Echo Reply Message，类型 0）。



在选项数据中，ping 还会存放发送请求的时间值，来计算往返时间，说明路程的长短。同个子网下的主机A和主机B，主机A执行 ping 主机B后，源主机首先会构建一个 ICMP 回送请求消息数据包。ICMP 数据包内包含多个字段，最重要的是两个：

- 第一个是**类型**，对于回送请求消息而言该字段为 8；

- 另外一个是序号，主要用于区分连续 ping 的时候发出的多个数据包。

每发出一个请求数据包，序号会自动加 1。为了能够计算往返时间RTT，它会在报文的数据部分插入发送时间。主机B收到这个数据帧后，先检查它的目的MAC地址，并和本机的MAC地址对比，如符合，则接收，否则就丢弃。接收后检查该数据帧，将IP数据包从帧中提取出来，交给本机的IP层。同样，IP层检查后，将有用的信息提取后交给ICMP协议。主机B会构建一个ICMP回送响应消息数据包，回送响应数据包的类型字段为0，序号为接收到的请求数据包中的序号，然后再发送出去给主机A。

在规定的时候内，源主机如果没有接到 ICMP 的应答包，则说明目标主机不可达；如果接到了 ICMP 回送响应消息，则说明目标主机可达。此时，源主机会检查，用当前时刻减去该数据包最初从源主机上发出的时刻，就是 ICMP 数据包的时间延迟。

当然这只是最简单的，同一个局域网里面的情况。如果跨网段的话，还会涉及网关的转发、路由器的转发等等。

4. 关于以太网帧

在以太网链路上的数据包称作以太帧。以太帧起始部分由前导码和帧开始符组成，后面紧跟着一个以太网报头，以 MAC 地址说明目的地址和源地址。帧的中部是该帧负载的包含其他协议报头的数据包（例如 IP 协议）。以太帧由一个 32 位冗余校验码结尾，它用于检验数据传输是否出现损坏。

当然，说到以太网帧，就不得不提 ARP 协议和 RARP 协议，以下是我认为概括以太网帧格式比较全面的一张示意图。



目的地址和源地址不用多说，需要注意的是目的地址和源地址都是 **MAC 地址**。我觉得值得关注的点有两个，类型和数据大小范围。类型 (0800 Vs 0806 Vs 0835) 是区分普通请求、ARP 请求或者 RARP 请求的关键。如果是 ARP 请求，其帧头部的目标地址会填充为 0xFFFFFFF-FFFFFFF。接下来是数据大小，最大 1500B 是规定的，这可能是综合考虑链路传输的效率等因素确定的，但这个 46B 就比较有趣了，由 CSMA/CD 算法可知以太网帧长为 64B(简单可认为是及时探知碰撞，当帧长过小时难以得知发生碰撞而一直发送数据占用信道资源，具体计算涉及到物理层传播，可参考相关博客)。所以 RAP 数据帧和 PRAP 数据帧都用 PAD(英文意义为“软垫”) 填充。

(四) 实验步骤

练习1 领略真实的 MAC 帧

各主机打开工具区的“拓扑验证工具”，选择相应的网络结构，配置网卡后，进行拓扑验证，如果通过拓扑验证，关闭工具继续进行实验，如果没有通过，请检查网络连接。本练习将主机 A 和 B 作为一组，主机 C 和 D 作为一组，主机 E 和 F 作为一组。现仅以主机 A、B 所在组为例，其它组的操作参考主机 A、B 所在组的操作。

1. 主机 B 启动协议分析器，新建捕获窗口进行数据捕获并设置过滤条件（提取 ICMP 协议）。
2. 主机 A ping 主机 B，查看主机 B 协议分析器捕获的数据包，分析 MAC 帧格式。
3. 将主机 B 的过滤器恢复为默认状态。

练习2 理解 MAC 地址的作用

本练习将主机 A 和主机 B 作为一组，主机 C 和主机 D 作为一组，主机 E 和主机 F 作为一组。现仅以主机 A、B 为例，其他组的操作参考主机 A、B 的操作。

1. 主机 B 启动协议分析器，打开捕获窗口进行数据捕获并设置过滤条件（源 MAC 地址为主机 A 的 MAV 地址）。
2. 主机 A ping 主机 B。
3. 主机 B 停止捕获数据，在捕获的数据中查找主机 A 所发送的 ICIP 数据帧，并分析该帧内容。

思考问题：

- 1) MAC 地址应用于 TCP/IP 协议模型的哪一层？

TCP/IP 协议共有四层，分别为网络接口层、网际层、传输层、应用层，MAC 地址在网络接口层（或网际接口层、数据链路层等，都有人叫）。

- 2) 如何区分以太网的两种标准帧格式？

如果以太网帧的第 13、14 字节包含的数值小于 1500，那么这个域可解释为分组的长度并适用于 802.3 标准，否则，该域被解释为一个类型域并适用原始以太网标准。

练习3 编辑并发送 MAC 广播帧

练习将主机 A、B、C、D、E、F 作为一组进行实验。

1. 主机 E 启动协议编辑器。
2. 主机 E 编辑一个 MAC 帧：

目的 MAC 地址：FFFF-FFFF-FFFF

源 MAC 地址：主机 E 的 MAC 地址

3. 主机 A、B、C、D、F 启动协议分析器，打开捕获窗口进行数据捕获并设置过滤条件（源 MAC 地址为主机 E 的 MAC 地址）。

4. 主机 E 发送已编辑好的数据帧。
5. 主机 A、B、C、D、F 停止捕获数据，察看捕获到的数据中是否含有主机 E 发送的数据帧。

思考问题：

- 1) 主机 A、B、C、D、F 是否可以收到主机 E 的广播帧？

都可以。

- 2) 说明 MAC 广播帧的范围？

MAC 广播帧的范围是广播域内的所有主机，或者简单解释为局域网内所有主机。

(五) 实验结果与分析

练习1 领略真实的 MAC 帧



图 2：主机 A 的地址为：172.23.65.101

分析：正如实验原理所述，ICMP 数据报会插入发送时间进而估算往返时间。这里的 TTL 指数据包的最大跳数。

```

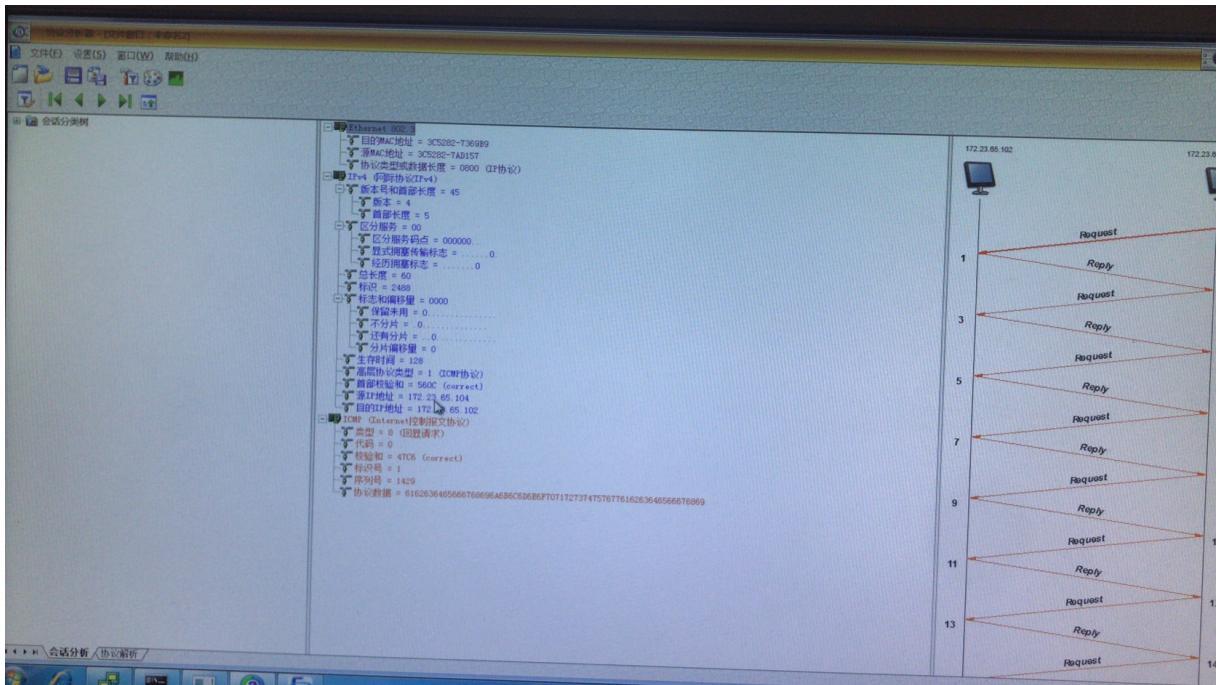
ShellNo.1
File Actions Edit View Help
loop txqueuelen 1000 (Local Loopback)
RX packets 72 bytes 4218 (4.1 KiB)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 72 bytes 4218 (4.1 KiB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions

root@kali:~/Desktop/code/simple_server# ping 192.168.56.1
PING 192.168.56.1 (192.168.56.1) 56(84) bytes of data.
64 bytes from 192.168.56.1: icmp_seq=1 ttl=128 time=0.709 ms
64 bytes from 192.168.56.1: icmp_seq=2 ttl=128 time=0.540 ms
64 bytes from 192.168.56.1: icmp_seq=3 ttl=128 time=0.630 ms
64 bytes from 192.168.56.1: icmp_seq=4 ttl=128 time=0.577 ms
64 bytes from 192.168.56.1: icmp_seq=5 ttl=128 time=1.29 ms
64 bytes from 192.168.56.1: icmp_seq=6 ttl=128 time=1.34 ms
^C
--- 192.168.56.1 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 5105ms
rtt min/avg/max/mdev = 0.540/0.848/1.340/0.335 ms
root@kali:~/Desktop/code/simple_server# █

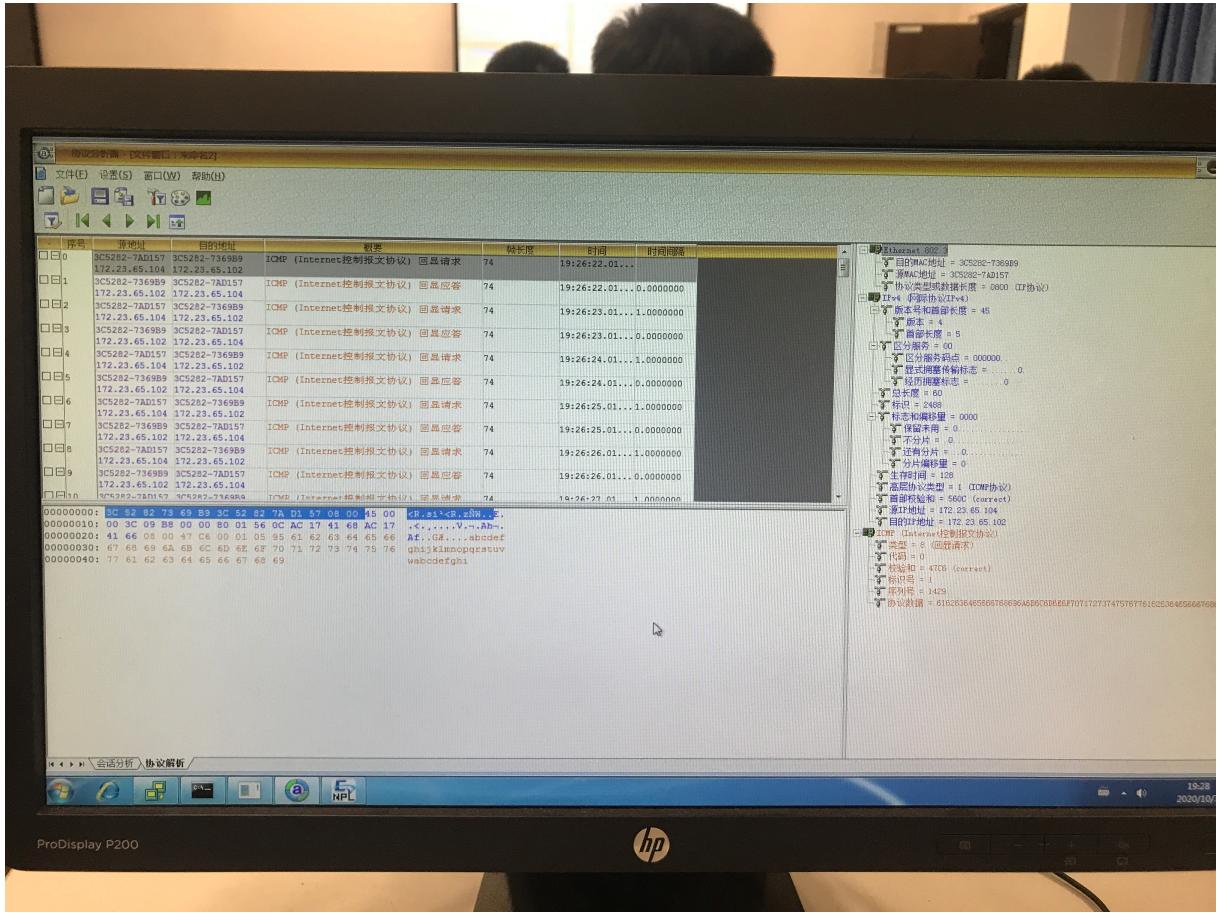
```

图 3: Linux 能显示报文更详细的信息

练习2 理解 MAC 地址的作用

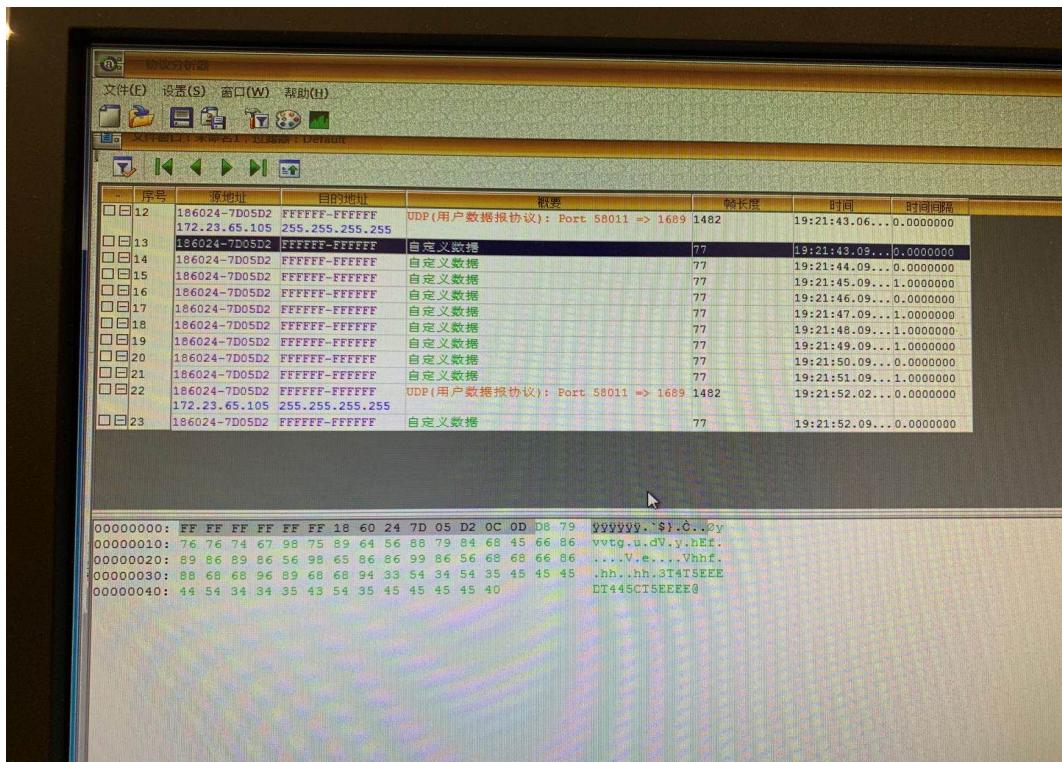


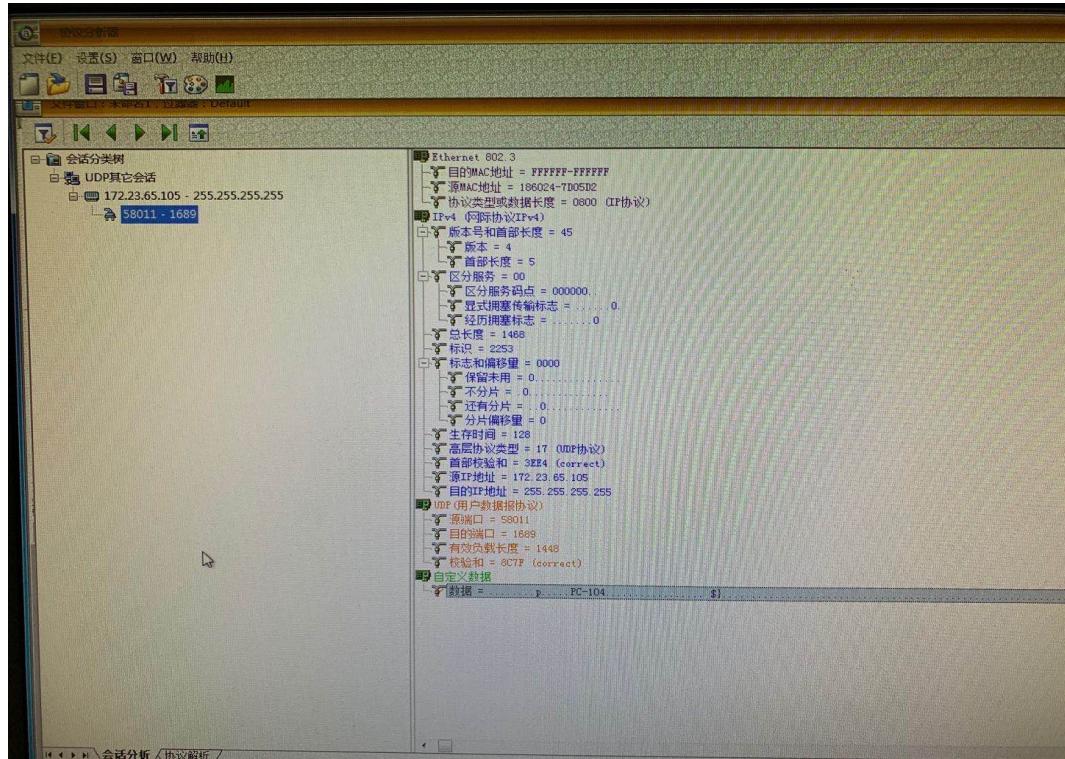
分析: 这张图便把数据包基本所有字段都显示出来了。包括以太网帧的类型，源地址和目标地址，同时也包括 IP 数据报头部的版本号 (4)、首部长度 (5)、TTL (128)、总长度 (60)、偏移量 (0)、首部校验和 (560C) 等等。除此之外，ICMP 的头部信息也都显示出来了，对于回送请求其类型为 8(参照实验原理)，还有检验和 (47C6)、标识号 (1)、序列号 (1429)、协议数据等字段。



分析: 这张图我关注一个数值, 帧长度。74 怎么来的呢, 我觉得应该是由以太网头部14个字节 (源目 MAC 各6个字节, Type 2个字节)+IP 包头20个字节+ICMP40个字节 (8个固定字节+32个 data 字节)。

练习3 理解 MAC 地址的作用





分析：观察第二张图可知我们收到了主机 E 的数据，而我们是主机 B。除此之外，主机 A、C、D 和 F 也都收到了主机 E 的数据，从而验证了以太网的广播帧。

(六) 附录：实现简单的 ping 程序

关于 ping 的一些测试网络说明

- ping 127.0.0.1。127.0.0.1 是本地循环地址，如果本地址无法 ping 通，则表明本地机 TCP/IP 协议不能正常工作。
- ping 本机的 IP 地址。通则表明网络适配器（网卡）工作正常，不通则是网络适配器出现故障。
- ping 同网段计算机的 IP。ping 一台同网段计算机的 IP，不通则表明网络线路出现故障；若网络中还包含有路由器，则应先 Ping 路由器在本网段端口的 IP，不通则此段线路有问题，通则再 ping 路由器在目标计算机所在网段的端口 IP，不通则是路由出现故障，通则再 ping 目的机 IP 地址。
- ping 网址。若要检测一个带 DNS 服务的网络，在上一步 ping 通了目标计算机的 IP 地址后，仍无法连接到该机，则可 ping 该机的网络名。

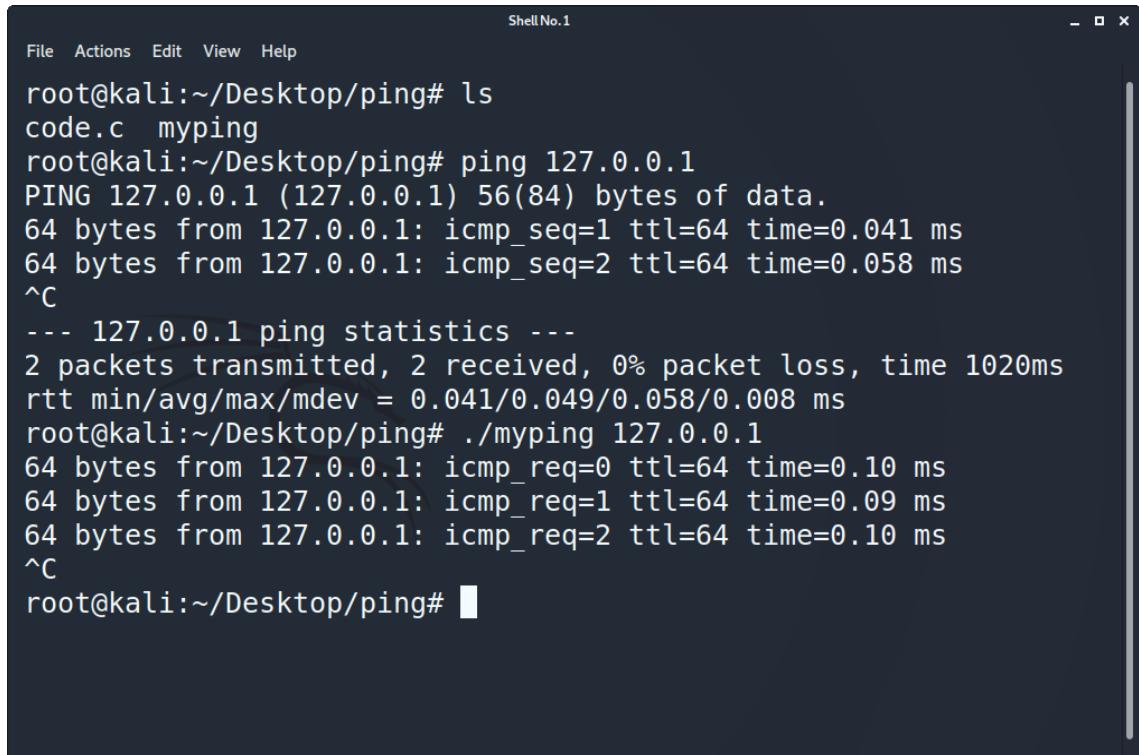
暨南大学本科实验报告专用纸(附页)

参考《UNIX 网络编程》第三版 28.5 P₅₈₄

简述

1. ICMP 协议是 IP 层的一个协议，但是由于差错报告在发送给报文源发方时可能也要经过若干子网，因此牵涉到路由选择等问题，所以 ICMP 报文需通过 IP 协议来发送。ICMP 数据报的数据发送前需要两级封装：首先添加 ICMP 报头形成 ICMP 报文，再添加 IP 报头形成 IP 数据报。
2. ICMP 协议为不可靠协议，所以不能仅仅发一个 ICMP 数据报然后由是否有应答来判断，而是定期发数据报（类似轮询），然后通过接收的帧判断是否与所发的相匹配。
3. 依据书中的思路，可以使用信号 SIGALRM 来定时，在 SIGALRM 的信号处理函数中发送 ICMP 回射请求包，主函数负责接收接收 ICMP 回显应答包。
4. 关于校验和。校验和是对每 16 位码进行反码求和（高位溢出位会加到低位）。可以先构建一个 32 的数字，对每 16 位求和，再将高 16 位右移后加到低 16 位上，最后将所得的和转为反码。
5. 为了简化程序，只实现 IPv4 部分。
6. 综上所述，该程序大概需要如下函数：检验码的构建、构建 ICMP 数据包、发送最终 IP 数据包、定时、接收捕获的 IP 数据包等。

1 // 见同目录源代码



```
File Actions Edit View Help
root@kali:~/Desktop/ping# ls
code.c myping
root@kali:~/Desktop/ping# ping 127.0.0.1
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.041 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.058 ms
^C
--- 127.0.0.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1020ms
rtt min/avg/max/mdev = 0.041/0.049/0.058/0.008 ms
root@kali:~/Desktop/ping# ./myping 127.0.0.1
64 bytes from 127.0.0.1: icmp_req=0 ttl=64 time=0.10 ms
64 bytes from 127.0.0.1: icmp_req=1 ttl=64 time=0.09 ms
64 bytes from 127.0.0.1: icmp_req=2 ttl=64 time=0.10 ms
^C
root@kali:~/Desktop/ping#
```

```
Shell No.1
File Actions Edit View Help
root@kali:~/Desktop/ping# ls
code.c myping
root@kali:~/Desktop/ping# ./myping 192.168.220.128
64 bytes from 192.168.220.128: icmp_req=0 ttl=64 time=0.12 ms
64 bytes from 192.168.220.128: icmp_req=1 ttl=64 time=0.11 ms
64 bytes from 192.168.220.128: icmp_req=2 ttl=64 time=0.09 ms
^C
root@kali:~/Desktop/ping# ping 192.168.220.128
PING 192.168.220.128 (192.168.220.128) 56(84) bytes of data.
64 bytes from 192.168.220.128: icmp_seq=1 ttl=64 time=0.076 ms
64 bytes from 192.168.220.128: icmp_seq=2 ttl=64 time=0.063 ms
64 bytes from 192.168.220.128: icmp_seq=3 ttl=64 time=0.061 ms
^C
--- 192.168.220.128 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2039ms
rtt min/avg/max/mdev = 0.061/0.066/0.076/0.006 ms
root@kali:~/Desktop/ping#
```

```
Shell No.1
File Actions Edit View Help
root@kali:~/Desktop/ping# ls
code.c myping
root@kali:~/Desktop/ping# ping www.baidu.com
PING www.a.shifen.com (14.215.177.39) 56(84) bytes of data.
64 bytes from 14.215.177.39 (14.215.177.39): icmp_seq=1 ttl=128
time=13.9 ms
64 bytes from 14.215.177.39 (14.215.177.39): icmp_seq=2 ttl=128
time=13.5 ms
^C
--- www.a.shifen.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1003ms
rtt min/avg/max/mdev = 13.504/13.708/13.913/0.204 ms
root@kali:~/Desktop/ping#
root@kali:~/Desktop/ping# ./myping 14.215.177.39
64 bytes from 14.215.177.39: icmp_req=0 ttl=128 time=28.28 ms
64 bytes from 14.215.177.39: icmp_req=1 ttl=128 time=15.46 ms
64 bytes from 14.215.177.39: icmp_req=2 ttl=128 time=15.97 ms
^C
root@kali:~/Desktop/ping#
```