网络抓包与协议分析

方桂安*，古博老师†

中山大学 智能科学与技术 20354027

【摘　要】 通过本实验我能更加熟练地使用网络抓包软件，捕捉和分析网络数据包，掌握以太网、ARP、IP、ICMP 和 TCP 等重要协议传输单元的结构，深入理解相关网络命令和重要协议算法的工作原理，从而掌握网络故障检测、网络性能改进和网络安全分析的能力。

【关键词】 Wireshark，ARP 协议，IP 协议，ICMP 协议

1　引 言

Wireshark（导线鲨鱼，前称 Ethereal，空灵）是一个免费开源的网络数据包分析软件。网络数据包分析软件的功能是截取网络数据包，并尽可能显示出最为详细的网络数据包资料。通过该软件我们可以实现：

1. 网络管理员使用 Wireshark 来检测网络问题
2. 网络安全工程师使用 Wireshark 来检查信息安全相关问题
3. 开发者使用 Wireshark 来为新的通信协议调试
4. 普通用户使用 Wireshark 来学习网络协议的相关知识

因此，本次实验中我将借此来实现网络抓包与协议分析。

2　实验：Wireshark 软件使用与 ARP 协议分析

学习 Wireshark 的基本操作，抓取和分析有线局域网的数据包；掌握以太网 MAC 帧的基本结构，掌握 ARP 协议的特点及工作过程。

2.1　实验过程与分析

2.1.1　观察 MAC 地址并分析以太网的帧结构

经过对软件界面与功能的熟悉之后，我开始抓取以太网帧，并对其进行分析。如图1所示，主

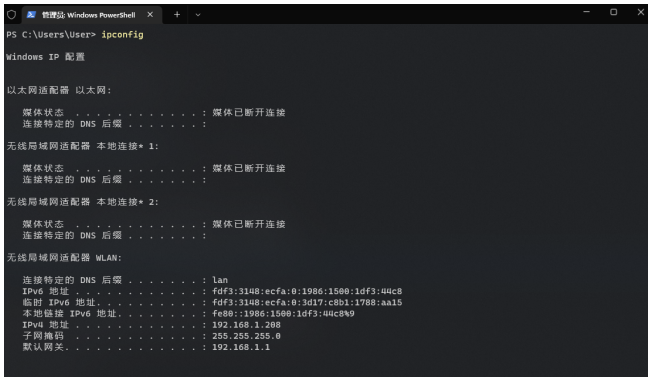

图 1　主机 ip 配置

机的 ip 配置如下：

- IPv4 地址：192.168.1.208
- 子网掩码：255.255.255.0
- 默认网关：192.168.1.1

由此我分别用笔记本在命令行窗口 ping 网关（192.168.1.1）和同一网段的我的平板（192.168.1.152）ping 笔记本，效果如图。

从图中可以看出：

- 笔记本 mac 地址为 08:d2:3e:21:fe:99
- 平板 mac 地址为 e0:d4:64:d9:a7:2e

实验时间: 2022 年 4 月 26 日
报告时间: 2022 年 5 月 1 日
† **指导教师**
***学号**：20354027
*E-mail：fanggan@mail2.sysu.edu.cn

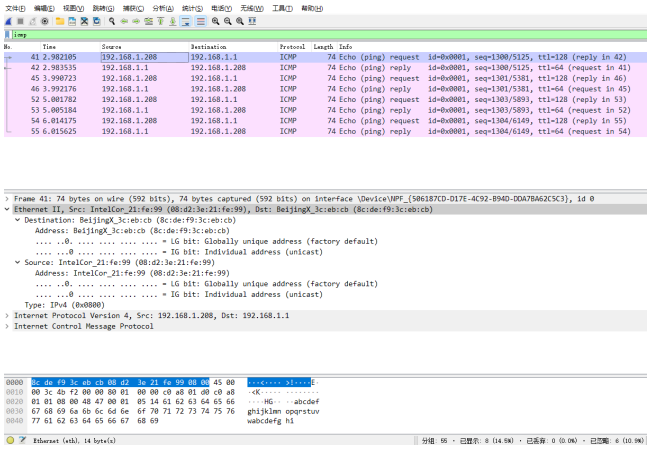

图 2 笔记本 ping 网关

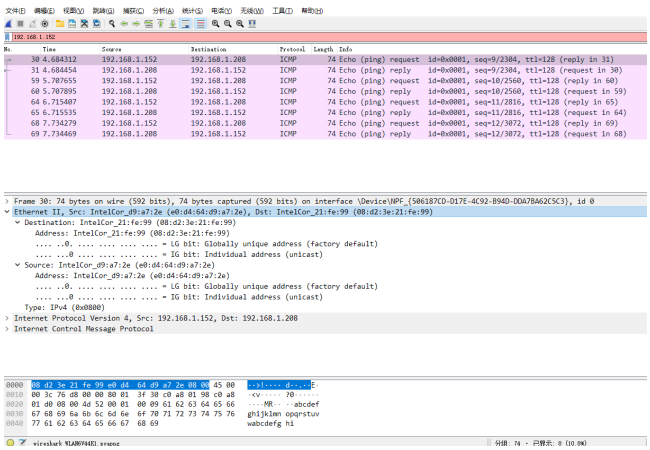

图 3 平板 ping 笔记本

- 网关 mac 地址为 8c:de:f9:3c:eb:cb

最后我再 ping 了一次 sysu.edu.cn，效果如图。

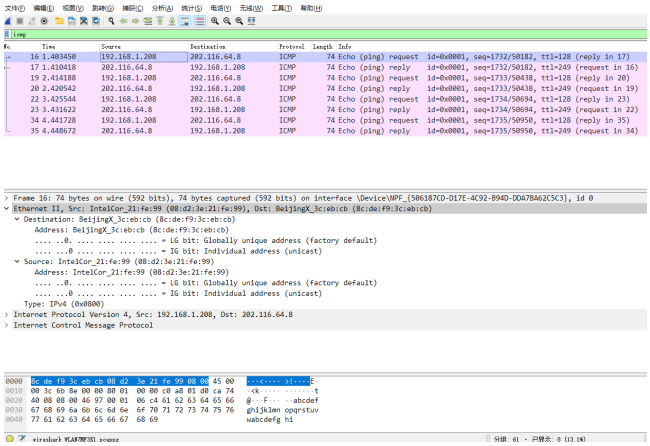

图 4 ping sysu.edu.cn

综上可知，访问本子网的计算机时，目的 MAC 是该主机的，但访问非本子网的计算机时，目的 MAC 是网关的。因为在子网之内的计算机就不需要经过网关，可以直接查询 MAC 地址并建立连接；而访问子网外的需要先将数据发送到网关处，然后 ARP 或者 DNS 解析得到 MAC 地址后到达目的子网发送到目的地址，所以无论是发送还是接受都要经过本计算机的网关。

图 5 mac 地址及其组成

MAC 地址（英语：Media Access Control Address），直译为媒体访问控制地址，也称为局域网地址（LAN Address），以太网地址（Ethernet Address）或物理地址（Physical Address），它是一个用来确认网络设备位置的地址。在 OSI 模型中，第三层网络层负责 IP 地址，第二层数据链接层则负责 MAC 地址。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。

MAC 地址共 48 位（6 个字节），以十六进制表示。I/G（Individual/Group）位，如果 I/G=0，则是某台设备的 MAC 地址，即单播地址；如果 I/G=1，则是多播地址（组播 + 广播 = 多播）。G/L（Global/Local，也称为 U/L 位，其中 U 表示 Universal）位，如果 G/L=0，则是全局管理地址，由 IEEE 分配；如果 G/L=1，则是本地管理地址，是网络管理员为了加强自己对网络管理而指定的地址。前 3-24 位由 IEEE 决定如何分配给每一家制造商，且不重复，后 24 位由实际生产该网络设备的厂商自行指定且不重复。

这里以我的网关 mac 为例分析，其 I/G 位为 0，G/L 位为 0，OUI 信息通过工具查询可知，我的路由器来自于北京的小米公司（型号为小米 AC2100，信息正确）。

表 1 以太网帧格式

前导字符	目的 MAC 地址	源 MAC 地址
8 字节	6 字节	6 字节
类型	IP 数据报	帧校验
2 字节	46-1500 字节	4 字节

图 6 OUI 信息

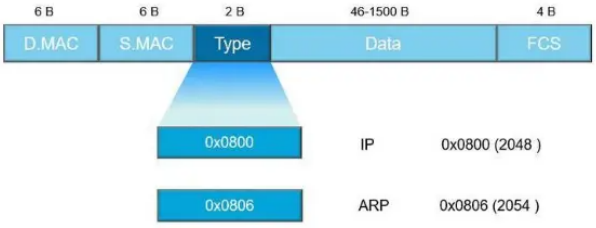

图 7 以太网帧的头尾信息及封装

接下来对 Ethernet II 进行分析，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。

除了前面提到的 mac 地址外，一个 0x0800 的以太类型说明这个帧包含的是 IPv4 数据报。同样的，一个 0x0806 的以太类型说明这个帧是一个 ARP 帧，0x8100 说明这是一个 IEEE 802.1Q 帧，而 0x86DD 说明这是一个 IPv6 帧。

由表1可知，以太网数据帧的长度在 64-1518 字节之间。数据部分在 46-1500 字节之间。数据包在以太网物理介质上传播之前必须封装头部和尾部信息。封装后的数据包称为数据帧，数据帧的封装的信息决定了数据如何传输。

2.1.2 ARP 协议分析及工作过程

最后是对 arp 协议的分析，我先后 ping 了同子网的手机（192.168.1.248）和我的域名 enderfga.cn（公网：39.108.128.240，内网：172.22.6.168），效果如图。

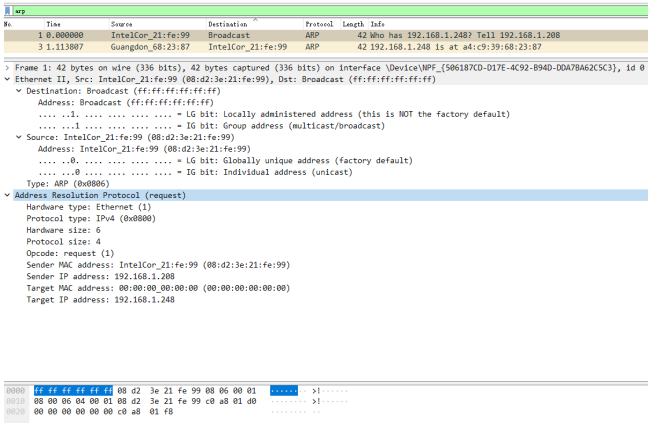

图 8 arp 协议

以笔记本主机 A（192.168.1.208）向手机主机 B（192.168.1.248）发送数据为例。从中可以知道工作过程为：

1. 当发送数据时，主机 A 会在自己的 ARP 缓存表中寻找是否有目标 IP 地址。如果找到就知道目标 MAC 地址为（a4:c9:39:68:23:87），直接把目标 MAC 地址写入帧里面发送就可。

2. 由于缓存已经被我清空，故主机 A 就会在网络上发送一个广播（ARP request），目标 MAC 地址是“ff.ff.ff.ff.ff.ff”，这表示向同一网段内的所有主机发出这样的询问：“192.168.1.248 的 MAC 地址是什么?”

3. 网络上其他主机并不响应 ARP 询问，只有主机 B 接收到这个帧时，才向主机 A 做出这样的回应（ARP response）：“192.168.38.11 的 MAC 地址是 a4:c9:39:68:23:87”，此回应以单播方式。这样，主机 A 就知道主机 B 的 MAC 地址，它就可以向主机 B 发送信息。同时它还更新自己的 ARP 高速缓存（ARP cache），下次再向主机 B 发送信息时，直接从 ARP 缓存表里查找就可。

如果是非同一子网下的主机间通讯，工作流程应为：

1. 主机 A 有数据发往主机 B，数据封装 IP 之后发现没有主机 B 的 mac 地址；然后查询 ARP，ARP 回应：“我在 192.168.1.0/24 网段，目标地址在 172.22.6.168/18, 不属于同一网段，需要使用默认网关”；ARP 发现默认网关是

192.168.1.1，但是没有网关 mac 地址，需要先进行查询；

2. 主机将数据包先放到缓存中，然后发送 ARP 查询报文：封装自己的 mac 地址为源 mac，目标 mac 地址写全 F 的广播地址，请求网关 192.168.1.1 的 mac 地址。然后以广播方式发送出去；

3. 路由器收到广播数据包，首先将原 192.168.1.1 添加到自己的 mac 地址表中，对应 mac 地址为 08:d2:3e:21:fe:99。路由发现是请求自己的 mac 地址，然后路由回复一个 ARP 应答：封装自己的 IP 地址为源 IP 自己的 mac 地址为源 mac，主机 A 的 IP 为目的 IP 主机 A 的 mac 为目的 mac，发送一个单播应答 “我是 192.168.1.1. 我的 mac 地址为 8c:de:f9:3c:eb:cb”；

4. 主机收到应答后，将网关 mac 地址对应 172.22.6.168（跨网关通信，其他网段 IP 地址的 mac 地址均为网关 mac），然后将缓存中的数据包，封装网关 mac 地址进行发送；

5. 路由收到数据包，检查目的 IP 地址，发现不是给自己的，决定要进行路由，然后查询路由表，需要发往 172.22.0.0 网段中的 172.22.6.168 地址。路由准备从相应接口上发出去，然后查询 mac 地址表，发现没有主机 B 的映射。路由器发送 arp 请求查询主机 B 的 mac 地址（原理同 2、3 步，主机 B 收到请求后首先会添加网关的 mac 地址，然后单播回复 arp 请求）；

6. 路由器收到主机 B 的 mac 地址后，将其添加到路由 mac 地址表中，然后将缓存中的数据 2 层帧头去掉，封装自己的 mac 地址为源 mac，主机 B 的 mac 地址为目的 mac（源和目的 IP 地址不变），加上二层帧头及校验，发送给主机 B；

7. 主机 B 收到数据之后，进行处理，发送过程结束；

8. 如果主机 B 收到数据后进行回复，主机 B 会进行地址判断，不在同一网段，然后决定将数据发送给网关，主机 B 查询 mac 地址表获得网关 mac 地址，将数据封装后发送（arp 地址解析的过程不再需要了，mac 地址表条目有一定的有效时间），网关收到数据后直接查询 mac 表，将二层帧 mac 地址更改为 A 的 mac 发送出去。如此，主机 A 收到主机 B 的回复；

综上所述，如果访问的是本子网的 IP，那么 ARP 解析将直接得到该 IP 对应的 MAC；如果访问的非本子网的 IP，那么 ARP 解析将得到网关的 MAC。（在子网之内的计算机不需要经过网关；而访问子网外的需要先将数据发送到网关处）

ARP 协议（英语：Address Resolution Protocol，直译为地址解析协议）是一个通过解析网络层地址来找寻数据链路层地址的网络传输协议，它在 IPv4 中极其重要。ARP 最初在 1982 年的 RFC 826（征求意见稿）中提出并纳入互联网标准 STD 37。从图8中可以知道 arp 数据包含以下内容：

- 硬件类型（HTYPE）：如以太网（0x0001）、分组无线网。
- 协议类型（PTYPE）：如网际协议 (IP)（0x0800）、IPv6（0x86DD）。
- 硬件地址长度（HLEN）：每种硬件地址的字节长度，一般为 6（以太网）。
- 协议地址长度（PLEN）：每种协议地址的字节长度，一般为 4（IPv4）。
- 操作码：1 为 ARP 请求，2 为 ARP 应答，3 为 RARP 请求，4 为 RARP 应答。
- 源硬件地址（Sender Hardware Address，简称 SHA）：n 个字节，n 由硬件地址长度得到，一般为发送方 MAC 地址。
- 源协议地址（Sender Protocol Address，简称 SPA）：m 个字节，m 由协议地址长度得到，一般为发送方 IP 地址。
- 目标硬件地址（Target Hardware Address，简称 THA）：n 个字节，n 由硬件地址长度得到，一般为目标 MAC 地址。
- 目标协议地址（Target Protocol Address，简称 TPA）：m 个字节，m 由协议地址长度得到，一般为目标 IP 地址。

2.2 实验思考与感悟

2.2.1 “同一网段”

在我根据实验指导书去 ping 同一网段的主机时，起初我选择了室友的主机，认为同为校园网的设备肯定是“同一网段”，但是结果请求超时，没有收到任何回应。

于是我查阅了相关资料，发现在中文的网络知识入门中，“网段”更经常地被误用来指代“子网”，也就是网络层中由网关或路由器等设备隔开的不同部分。例如 IP 为 192.168.0.1 - 192.168.0.254 的设备就位于掩码 255.255.255.0 的同一子网中，这句话经常被说成“位于 192.168.0.x ‘网段’中”，如果不涉及网络层之下的结构，这么说不会引起混淆，但是在深入探讨互联网底层结构的时候，应该避免使用“网段”来指代“子网”。而我想找的设备应该是同一子网下的。

想了解更多关于子网的信息，需要先认识什么是子网掩码。它是一种用来指明一个 IP 地址的哪些位标识的是主机所在的网络地址以及哪些位标识的是主机地址的位掩码。通常情况下，子网掩码的表示方法和地址本身的表示方法是一样的。在 IPv4 中，就是点分十进制四组表示法（四个取值从 0 到 255 的数字由点隔开，比如 255.128.0.0）。

我校的校园网子网掩码是 255.255.192.0，默认网关是 172.29.0.1，用 CIDR 表示即为/18。属于 B 类 IP 地址，共有 16382 个可用 IP 地址。其中 256/(256-192)=4，共 4 个子网：

- 172.29.0.0 - 172.29.63.255
- 172.29.64.0 - 172.29.127.255
- 172.29.128.0 - 172.29.191.255
- 172.29.192.0 - 172.29.255.255

2.2.2 ”广播，单播，组播”

广播（英语：broadcast）是指将信息数据包发往指定网络范围内的所有设备。其发送范围称为“广播域”。

单播（英语：unicast）是指数据包在计算机网络的传输中，目的地址为单一目标的一种传输方式。它是现今网络应用最为广泛，通常所使用的网络协议或服务大多采用单播传输，例如一切基于 TCP 的协议。

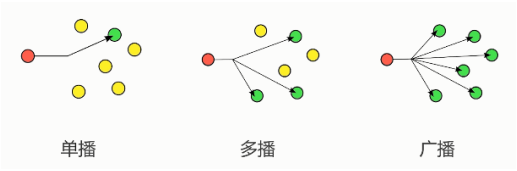

图 9 单播，多播，广播

多播（英语：multicast，又称群播，中国大陆也译作组播），是计算机网络中的一种群组通信，它把信息同时传递给一组目的计算机。多播可以是一对多或多对多布置。不应将其与物理层的点到多点通信混淆。

2.2.3 思考题

1. 使用了显示过滤器后 Wireshark 的抓包工作量不会减少，过滤只是查找只显示的信息，不会减少任何抓包工作量。但捕抓过滤会减少，对确定的捕抓类型抓包。

2. Gratuitous ARP（无回报的 ARP）是指主机发送 ARP 查询（广播）自己的 IP 地址，当 ARP 功能被开启或者是端口初始配置完成，主机向网络发送无回报的 ARP 来查询自己的 IP 地址确认地址唯一可用。作用：

   - 确定网络中是否有其他主机使用了 IP 地址，如果有应答则产生错误消息。
   - 无回报的 ARP 可以做更新 ARP 缓存用，网络中的其他主机收到该广播则在缓存中更新条目，收到该广播的主机无论是否存在与 IP 地址相关的条目都会强制更新，如果存在旧条目则会将 MAC 更新为广播包中的 MAC。

3 实验：IP 与 ICMP 分析

IP 和 ICMP 协议是 TCP/IP 协议簇中的网络层协议，在网络寻址定位、数据分组转发和路由选择等任务中发挥了重要作用。本实验要求熟练使用 Wireshark 软件，观察 IP 数据报的基本结构，分析数据报的分片；掌握基于 ICMP 协议的 ping 和 traceroute 命令及其工作原理。

3.1 实验过程与分析

互联网控制消息协议（英语：Internet Control Message Protocol，缩写：ICMP）是互联网协议族的核心协议之一。它用于网际协议（IP）中发送控

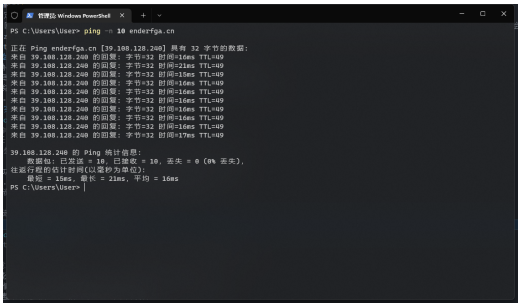

图 10 ping 域名

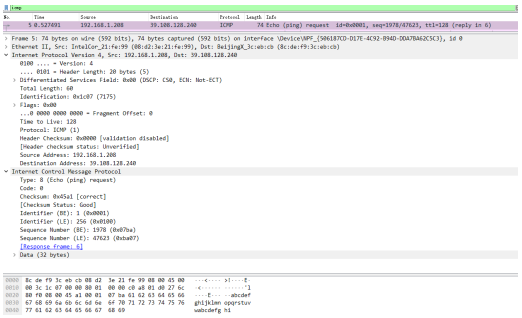

图 11 IP 数据包与 ICMP 报文

制消息，提供可能发生在通信环境中的各种问题反馈。通过这些信息，使管理者可以对所发生的问题作出诊断，然后采取适当的措施解决。ICMP 依靠 IP 来完成它的任务，它是 IP 的主要部分。它与传输协议（如 TCP 和 UDP）显著不同：它一般不用于在两点间传输数据。它通常不由网络程序直接使用，除了 ping 和 traceroute 这两个特别的例子。IPv4 中的 ICMP 被称作 ICMPv4，IPv6 中的 ICMP 则被称作 ICMPv6。

3.1.1 ping 命令

ping 是 ICMP 最著名的一个应用，通过 ping 可以测试网络的可达性，即网络上的报文能否成功到达目的地。使用 ping 命令时，源设备向目的设备发送 Echo request 消息，目的地址是目的设备的 IP 地址。目的设备收到 Echo request 消息后，向源设备回应一个 Echo reply 消息，可知目的设备是可达的。

这一次我依旧选择了 ping 我的域名，结果如图 10,11 所示。

接下来我们来解释 ip 数据报的首部，具体细节如图 12,13 所示。

版本为 4，即 IPv4；首部长度为 20 字节，即 5 个 32 位字节；服务类型为全 0，即没有特殊要求的一般服务；总长度为 60 个字节；标识为 0x1c25；

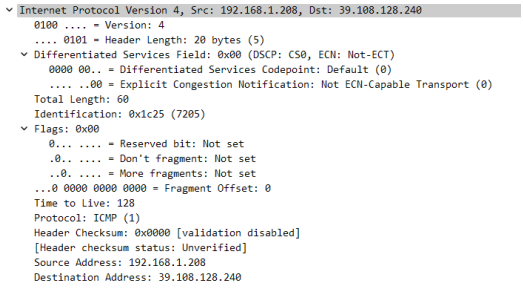

图 12 IP 数据报

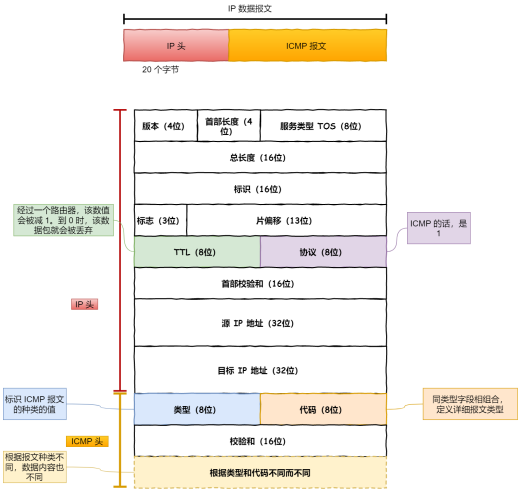

图 13 IP 数据报文结构示意图

标记为 0x00；片偏移为 0；生存时间为 128；协议为 ICMP；头部校验和为 0x0000；源地址为我的笔记本 ip 地址；目的地址为域名的 ip 地址。

对比图 14,15，ICMP Echo 请求帧和回应帧主要区别在于 type 以及 checksum。回应帧中还多了一项 Response time，即回应时间，单位为毫秒。

最后是对 ip 分片的研究。ping 命令默认为 32 字节的数据，数据较少，不会出现分片的情况。查阅资料可知，以太网帧的 MTU(最大传输单元) 是 1500 字节，ip 头部信息为 20 字节，故我认为指定 length 参数大于 1480 开始会出现分片的情况。分别执行 ping -l 1000/2000/3000 enderfga.cn，可以看到结果如图18所示。

由图可得，1000 字节时没有分片，2000 字节时有分片，多了一个数据包；3000 字节时有分片，多了两个数据包。

初步判断我的猜想正确，以 2000 的数据包为例进行计算，先捕捉到的 ipv4 数据包长度为 1514（14B 以太网帧头 + 20B IP 包头 + 1480B 的数据），id 是 0xee83，offset 为 0，为第一个分片；而接下来的 icmp 包长度为 562（8B ICMP 包头 + 14B 以

```
Internet Control Message Protocol
    Type: 8 (Echo (ping) request)
    Code: 0
    Checksum: 0x4579 [correct]
    [Checksum Status: Good]
    Identifier (BE): 1 (0x0001)
    Identifier (LE): 256 (0x0100)
    Sequence Number (BE): 2018 (0x07e2)
    Sequence Number (LE): 57863 (0xe207)
    [Response frame: 14]
```

图 14 request

```
Internet Control Message Protocol
    Type: 0 (Echo (ping) reply)
    Code: 0
    Checksum: 0x4d79 [correct]
    [Checksum Status: Good]
    Identifier (BE): 1 (0x0001)
    Identifier (LE): 256 (0x0100)
    Sequence Number (BE): 2018 (0x07e2)
    Sequence Number (LE): 57863 (0xe207)
    [Request frame: 13]
    [Response time: 16.379 ms]
```

图 15 reply

类型值	ICMP消息类型	错误消息	查询消息
0	Echo Reply 回送响应消息		✓
3	Destination Unreachable 目的不可达	✓	
5	Redirect 重定向	✓	
8	Echo Request 回送请求消息		✓
11	Time Exceeded 超时	✓	
12	Parameter Problem 参数问题	✓	
13	Timestamp Request 时间戳请求		✓
14	Timestamp Reply 时间戳响应		✓

图 16 type 值

目的不可达类型	
代码值	ICMP目的不可达消息
0	Network Unreachable 目的网络不可达
1	Host Unreachable 目的主机不可达
2	Protocol Unreachable 目的协议不可达
3	Port Unreachable 目的端口不可达
4	Fragmentation Needed and Don't Fragment was Set 需分片但DF置位
5	Source Route Failed 源路由错误
6	Destination Network Unknown 目的网络未知
7	Destination Host Unknown 目的主机未知
8	Source Host Isolated 源主机隔离
9	Communication with Destination Network is Administratively Prohibited 禁止与目的网络的通信
10	Communication with Destination Host is Administratively Prohibited 禁止与目的主机的通信
11	Destination Network Unreachable for Type of Service 服务类型的目的网络不可达
12	Destination Host Unreachable for Type of Service 服务类型的目的主机不可达

图 17 code 值

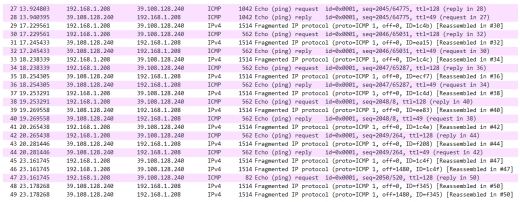

图 18 分片情况

太网帧头 + 20B IP 包头 + 520B 的数据），id 同样是 0xee83，验证分片猜想，offset 为 1480，刚好可以接上。因此，2000=1514-14-20+562-14-20-8，猜想成立。

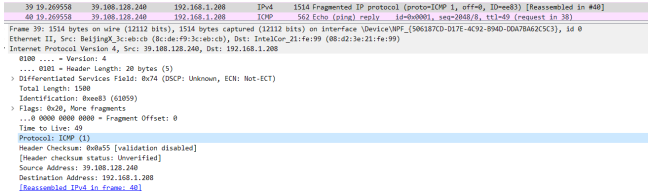

图 19 ipv4 包

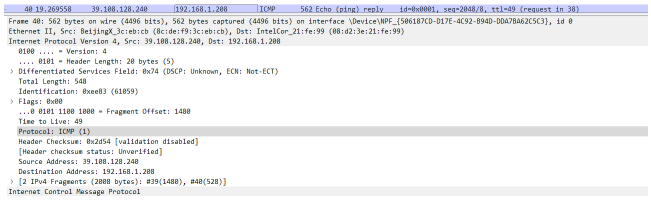

图 20 icmp 包

3.1.2 traceroute 命令

ping 工具只能测试目的设备的连通性，但是看不到数据包的传输路径。所以在网络不通的情况下，无法知道网络问题发生在哪个位置。tracert 工具可以查看数据包的整条传输路径，包括途中经过的中间设备。

在 Windows 中命令是 tracert，在 Unix、MacOS 中命令是 traceroute 。tracert enderfga.cn 可以得到如图21,22中的结果。

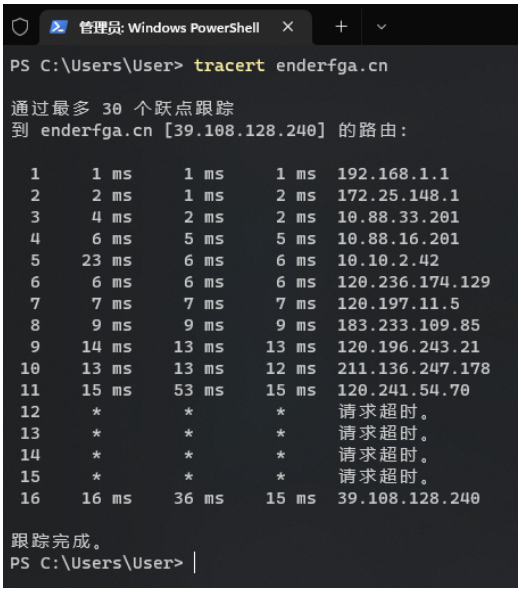

图 21 tracert 结果

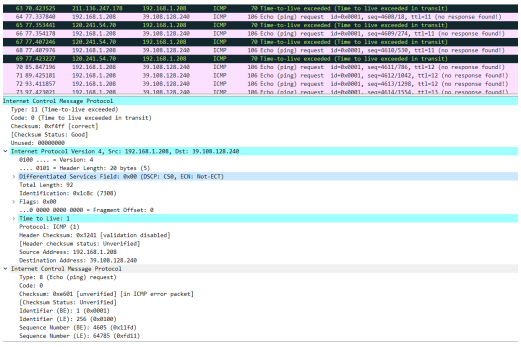

图 22 差错报文

使用 tracert 命令时，源设备的 tracert 逐跳发送数据包，并等待每一个响应报文。发送第一个数据包时，TTL 值设为 1 。第一个路由器收到数据包后 TTL 值减 1 ，随即丢弃数据包，并返回一个 Time Exceeded 消息。源设备的 tracert 收到响应报文后，取出源 IP 地址，即路径上的第一个路由器地址。然后 tracert 发送一个 TTL 值为 2 的数据包。第一个路由器将 TTL 值减 1 ，并转发数据包。第二个路由器再将 TTL 值减 1 ，丢弃数据包并返回一个 Time Exceeded 消息。tracert 收到响应报文后，取出源 IP 地址，即路径上的第二个路由器地址。类似步骤，tracert 逐跳获得每一个路由器的地址，并探测到目的设备的可达性。

差错报文包括：目标不可到达（网络、协议、主机、端口不可到达；禁止分割、目标网络不认识、目标主机不认识等等）、超时、参数问题、重定向（网络重定向、主机重定向等）等等。tracert

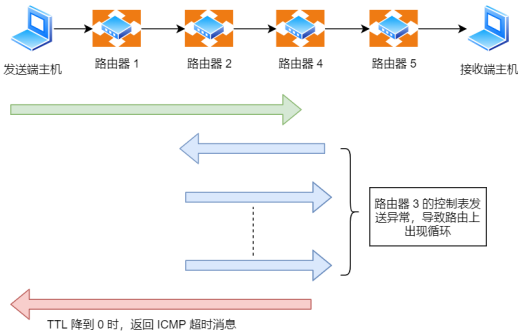

图 23 数据交互示意图

在发送数据包时，会填入一个不可能的端口号值作为目标端口号（大于 3000 ）。当目的主机，收到数据包后，会返回 ICMP 差错报文消息，且这个差错报文消息的类型是「端口不可达」。所以，当差错报文类型是端口不可达时，说明发送方发出的数据包到达了目的主机。交互过程见图23。

分析图 22 的差错报文可知，tracert 的 data 不像 ping 命令中的随机字符，内容全为 0；由于其是错误分组，因此有一个 Unused 字段，其值为 0；Internet Protocol Version 4，里面包含了 Differented Services Field、Total Length、Identification、Flags、Fragment、Time to live、Header、Source、Destination 等信息。另外路由跟踪的 ICMP 响应数据包（非超时错误数据包）的 ICMP 的 TYPE =0 代表 ICMP 响应，且每次的序列号都不同。

3.2 实验思考与感悟

3.2.1 “本机”

Ping 127.0.0.1 时，不能捕获 ICMP 报文。数据包根本没有到达网口，所以捕获不到 ICMP 报文，在 ping 本机的时候，虽然是用这种办法来判断网卡是否正常工作，但实际上并没有发送到网卡，也就是说，ping 本机也收不到 ICMP 报文，两者都是经过环路来进行处理。

3.2.2 ” 最后三个数据包”

最后三个 ICMP（响应）数据包是目标主机发送给我的 ICMP 回应数据包，因为路由查询是使用逐渐递增 TTL 的 ICMP 查询数据包，最后的 ICMP 查询数据包的 TTL 已经大于到达目的主机中间路由跃点数，因此不会被目标主机丢弃，发送 ICMP 超时的数据包，所以只会收到 ICMP 响应数据包。

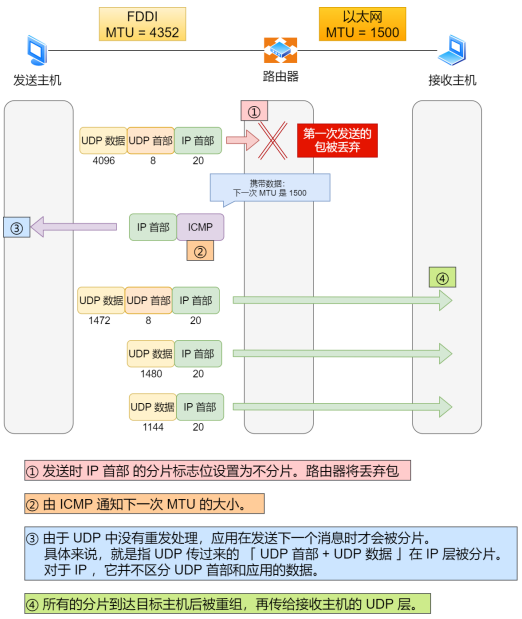

图 24 MTU 大小确定

3.2.3 思考题

1. 在实际操作中，Traceroute 命令返回的某些条目以“*”号表示。能是防火墙封掉了 ICMP 的返回信息，所以我们得不到什么相关的数据包返回数据。或者中间任何一个 router 上如果封了 ICMP Echo Request，traceroute 就不能工作；中间的 router 看不到，但能看到 packet 到达了最后的 destination；如果封了 ICMP Echo Reply，中间的全能看到，最后的 destination 看不到。

2. 发送方要怎样决定 IP 数据报分组大小，才能避免因为不同网络 MTU 不一致而引起分片呢？可以故意设置不分片，从而确定路径的 MTU。首先在发送端主机发送 IP 数据报时，将 IP 包首部的分片禁止标志位设置为 1。根据这个标志位，途中的路由器不会对大数据包进行分片，而是将包丢弃。

   随后，通过一个 ICMP 的不可达消息将数据链路上 MTU 的值一起给发送主机，不可达消息的类型为「需要进行分片但设置了不分片位」。

   发送主机端每次收到 ICMP 差错报文时就减少包的大小，以此来定位一个合适的 MTU 值，以便能到达目标主机。

4 结 论

本次实验借助 wireshark 工具，我深入了解了 IP 协议，arp 协议与 icmp 协议。这些协议都作用于网络层。

网络层（Network Layer）是 OSI 模型中的第三层（TCP/IP 模型中的网际层），提供路由和寻址的功能，使两终端系统能够互连且决定最佳路径，并具有一定的拥塞控制和流量控制的能力。相当于发送邮件时需要地址一般重要。由于 TCP/IP 协议体系中的网络层功能由 IP 协议规定和实现，故又称 IP 层。

当然网络层还有一些其他的协议，如 RARP、OSPF、IPX、RIP、IGRP 等，wireshark 的功能也远远不止在本次实验中体现的这些。

因此在未来的学习生活中，我将继续研究计算机网络的“神奇”之处，以自顶向下的方法来探索网络领域。

参考文献

[1] WRIGHT G, RICHARDSTEVENS W, 赖特, 等. TCP/IP 详解 [J]. 机械工业出版社, 2000.

[2] 库罗斯. 计算机网络自顶向下方法 (原书第 6 版)[M]. [出版地不详]: 计算机网络自顶向下方法 (原书第 6 版), 2009.

[3] 竹下隆史, 村山公保, 荒井透, 等. 图解 TCP/IP:TCP/IP[J]. 人民邮电出版社, 2013.