
实验 3.1 Wireshark 软件使用与 ARP 协议分析

一．实验目的

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。

二．实验内容

使用 Wireshark 抓取局域网的数据包并进行分析：

1. **学习 Wireshark 基本操作：**重点掌握捕获过滤器和显示过滤器。
2. **观察 MAC 地址：**了解 MAC 地址的组成，辨识 MAC 地址类型。
3. **分析以太网帧结构：**观察以太网帧的首部和尾部，了解数据封装成帧的原理。
4. **分析 ARP 协议：**抓取 ARP 请求和应答报文，分析其工作过程。

三．实验原理

3.1 Wireshark 简介

Wireshark 软件是目前全球使用最广泛的开源网络数据包分析工具（前身为 Ethereal），由 Gerald Combs 编写并于 1988 年获开源许可发布。网络数据包分析是指进入网络通信系统、捕获和解码网络上实时传输数据以及搜集统计信息的过程。通过 Wireshark 对网络数据进行分析，我们能够了解网络是如何运行的、数据包是如何被转发的、应用是如何被访问的；能够分析各层网络协议的性能、掌握通信主体的运行情况，确认带宽分配和时延大小、查看应用的快慢并改进优化，识别网络中存在的攻击或恶意行为、解决网络异常和故障。Wireshark 可以在 Windows、Linux 和 macOS 操作系统中运行，具备友好的图形界面、丰富的统计及图表分析功能。

3.2 以太网 MAC 帧格式

本实验基于使用最广泛的有线局域网（以太网 Ethernet II），以太网的帧结构如表 1.1-1 所示。其中，MAC 地址（Media Access Control Address，媒体存取控制位址）或

称物理地址（Physical Address），用于在网络中标识网卡。MAC 地址的长度为 48 位 (6 个字节)，通常表示为 12 个 16 进制数，如：00-16-EA-AE-3C-40。其中前 3 个字节的 16 进制数 00-16-EA 代表网络硬件制造商的编号、即组织唯一标志符 (OUI)，它由 IEEE 分配；而后 3 个字节的 16 进制数 AE-3C-40 代表该制造商所生产的某个网络产品 (如网卡) 的系列号。

表 1.1-1 以太网帧格式

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

3.3 ARP 协议及数据报格式

地址解析协议（Address Resolution Protocol，ARP），主要作用是将 IP 地址解析为 MAC 地址。当某主机或网络设备要发送数据给目标主机时，必须知道对方的网络层地址（即 IP 地址），而且在数据链路层封装成帧时，还必须有目标主机（或下一跳路由器）的 MAC 地址。本实验重点观察最简单的情形：同一个网段内，主机 A 要向主机 B 发送信息时，ARP 解析的过程（主机 A 和 B 不在同一网段的情况请参阅课本相关内容）。具体如下：

1. 主机 A 首先查看自己的 ARP 表。如果找到了主机 B 的 MAC 地址，则利用这个地址对 IP 数据报进行帧封装，并将数据报发送给主机 B。
2. 如果主机 A 在 ARP 表中找不到主机 B 的 MAC 地址，则以广播方式发送一个 ARP 请求报文。ARP 请求报文中的发送端 IP 地址和发送端 MAC 地址为主机 A 的 IP 地址和 MAC 地址，目标 IP 地址和目标 MAC 地址为主机 B 的 IP 地址和全 0 的 MAC 地址。由于 ARP 请求报文以广播方式发送，该网段上的所有主机都可以接收到该请求，但只有被请求的主机 B 会对该请求进行处理。
3. 主机 B 比较自己的 IP 地址和 ARP 请求报文中的目标 IP 地址，当两者相同时进行如下处理：将 ARP 请求报文中的发送端（即主机 A）的 IP 地址和 MAC 地址存入自己的 ARP 表中。然后以单播方式发送 ARP 响应报文给主机 A，其中包含了自己的 MAC 地址。

4. 主机 A 收到 ARP 响应报文后，将主机 B 的 MAC 地址加入到自己的 ARP 表中以用于后续报文的转发，同时将 IP 数据报进行封装后发送出去。

ARP 报文结构如图 1.1-1 所示，ARP 报文总长度为 28 字节，MAC 地址长度为 6 字节，IP 地址长度为 4 字节。每个字段的含义如下：

- **硬件类型**：指明了发送方想知道的硬件接口类型，以太网的值为 1。
- **协议类型**：表示要映射的协议地址类型。IP 地址的类型值为 0x0800。
- **硬件地址长度和协议地址长度**：分别指出硬件地址和协议地址的长度，以字节为单位。在以太网中，它们的值分别为 6 和 4。
- **操作码 (op)**：用来表示这个报文的类型，ARP 请求为 1，ARP 响应为 2，RARP 请求为 3，RARP 响应为 4。



图 1.1-1 ARP 报文结构示意图

3.4 实验方法及手段

使用 Wireshark 软件在有线局域网中捕捉相关网络操作的数据包，运用观察对比、计算验证、分析统计等方法，掌握以太网 MAC 帧和 IP 数据报的结构以及 ARP 协议的工作原理。

四． 实验条件

- PC 机一台，连入局域网；

- Wireshark 软件，建议 3.0 以上版本。

五．实验步骤

5.1 WireShark 基本使用

1. 通过 Wireshark 官网下载最新版软件，按默认选项安装。
2. 运行 Wireshark 软件，程序界面会显示当前的网络接口列表，双击要观察的网络接口，开始捕捉数据包，Wireshark 软件选择网络接口的界面如图 1.1-2 所示。

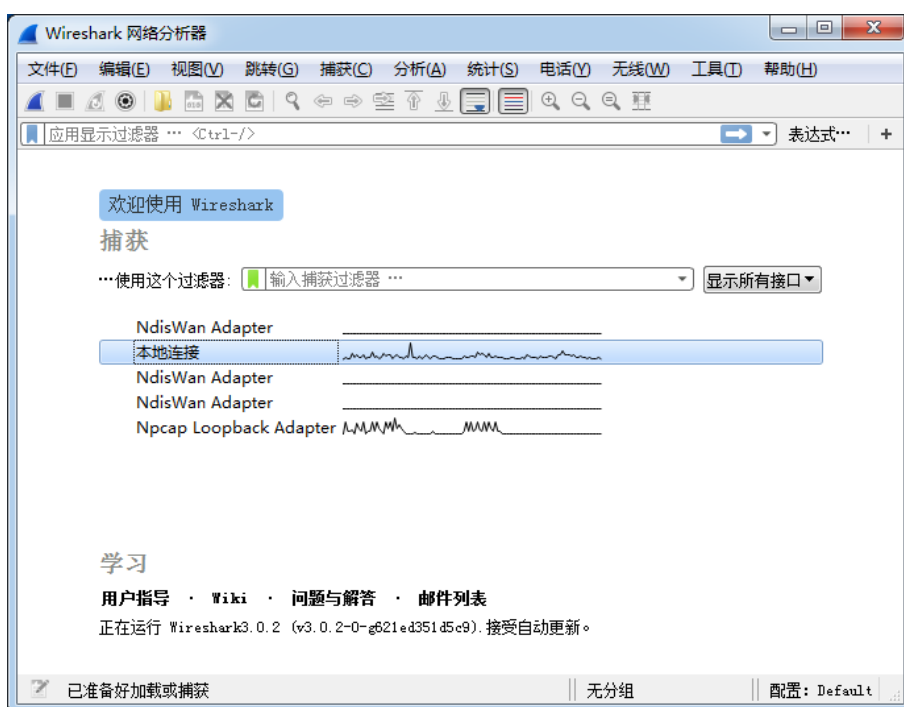


图 1.1-2 Wireshark 软件启动界面

3. 点击工具栏上的红色方块按钮停止捕捉。
4. 菜单、工具栏、状态栏和主窗口如图 1.1-3 所示，可以根据需要通过菜单“视图”以及“编辑/首选项/外观”的相关选项对基本设置进行更改。例如图 1.1-4 中的语言、字体缩放、颜色、布局等项目。
5. 使用“显示过滤器”可以方便地从捕获的数据包中筛选出要观察的数据包。显示过滤器支持若干的过滤选项：源 MAC、目的 MAC、源 IP、目的 IP、TCP/UDP 传输协议、应用层协议（HTTP, DHCP）、源端口 Port、目的端口 Port 等。在显示

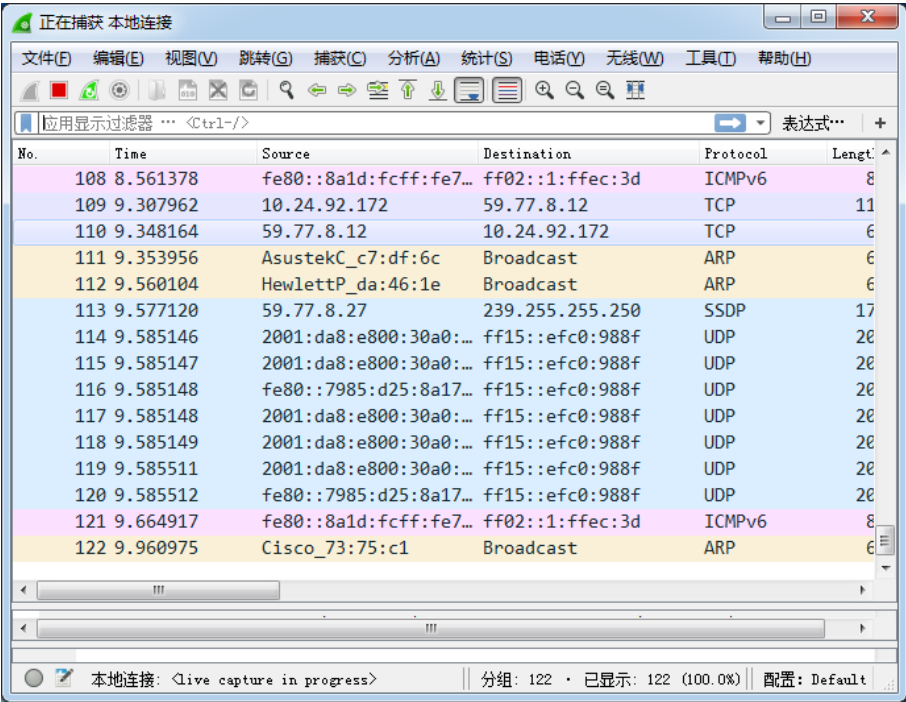


图 1.1-3 Wireshark 主窗口界面

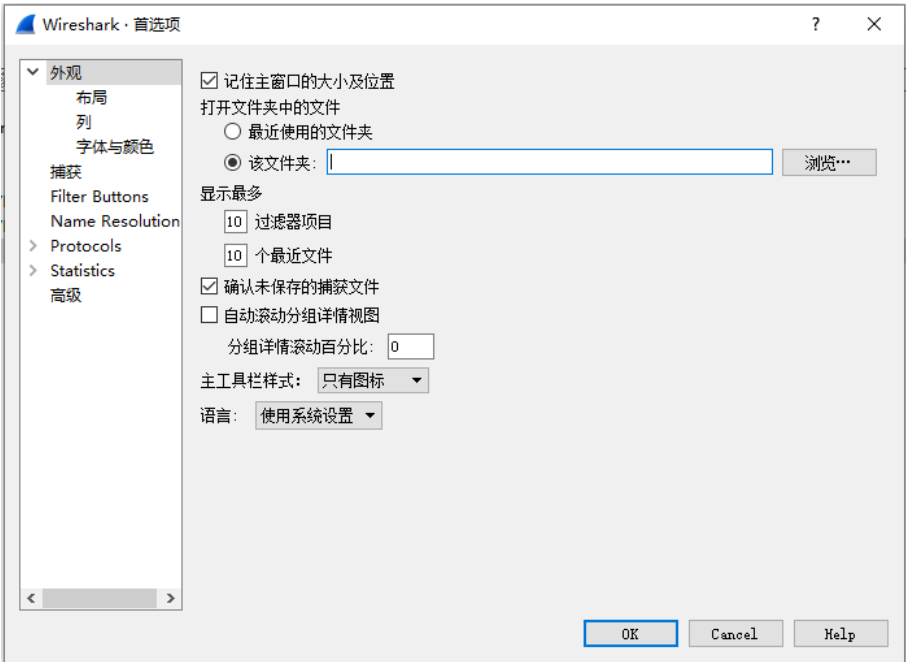


图 1.1-4 Wireshark 的设置界面

过滤器栏中输入过滤表达式（更详细的显示过滤语法可以查看 [Wireshark 的官方文档¹](#)），例如下面的命令：

- arp //显示 arp 协议报文，例如图1.1--5
- ip.src == a.b.c.d && icmp //显示源地址为 a.b.c.d 的 icmp 报文

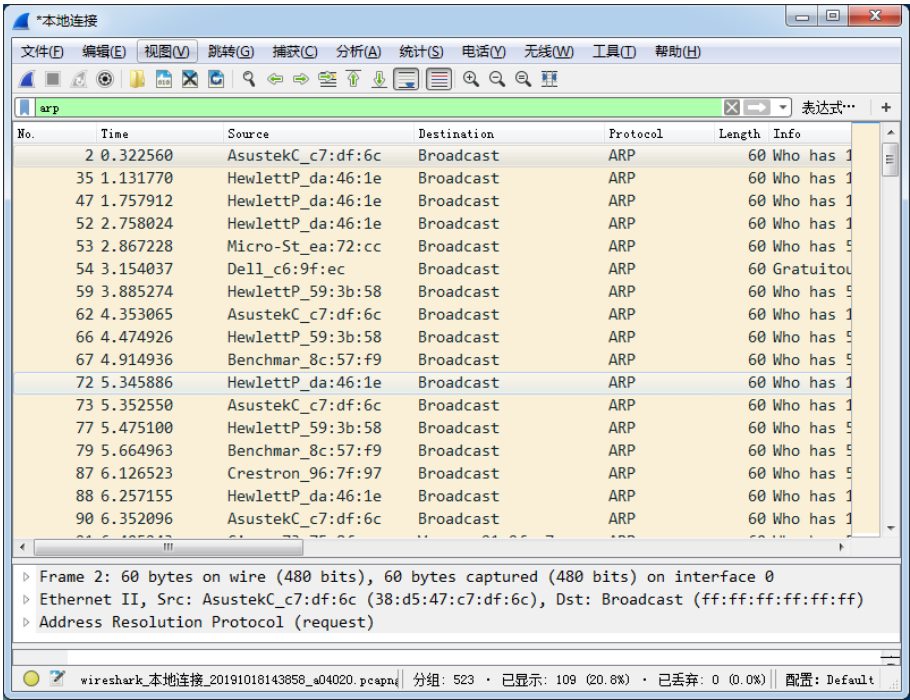


图 1.1-5 显示过滤规则的示例

6. 通过主菜单“文件” / “导出特定分组”（如图1.1-6），可以保存捕获的网络数据（也可以先选中某些包，只保存部分数据）。
7. 如果只想捕捉特定的数据包，可以使用菜单“捕获” / “捕获过滤器”选定想要的类型（如图1.1-7）。例如，选择“IPv4 only”，Wireshark 只抓取 ipv4 类型的数据包。Wireshark 过滤器官方文档提供了更加全面详细的语法和常用示例²。
8. Wireshark 还提供了丰富的统计功能供用户选用，如图1.1-8。更多文档可以查询 Wireshark 使用帮助³。

¹[Wireshark 显示过滤器语法](#)

²[Wireshark 常用过滤器语法](#)

³[Wireshark 学习手册](#)

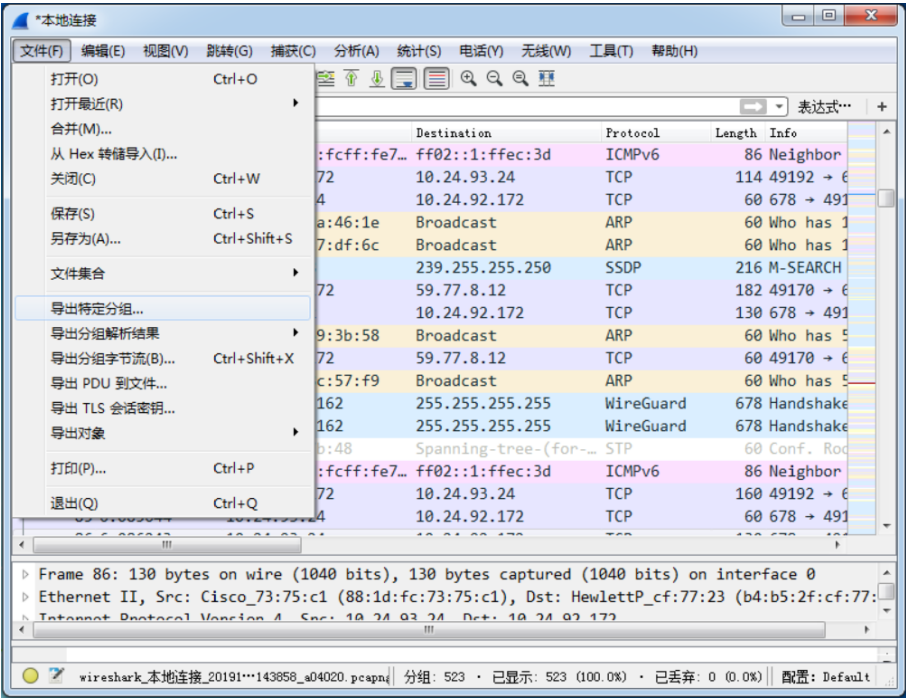


图 1.1-6 操作主菜单保存数据文件

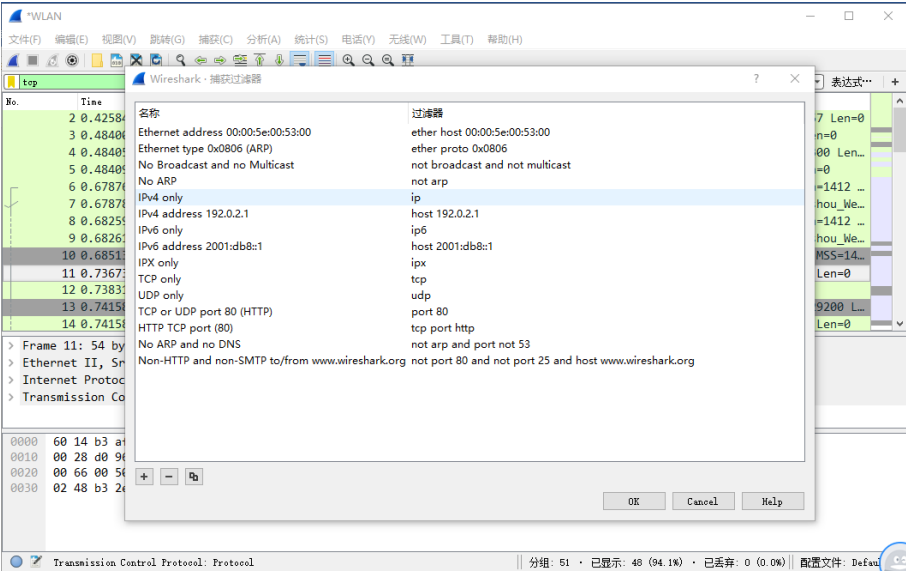


图 1.1-7 选中特定的捕获类型

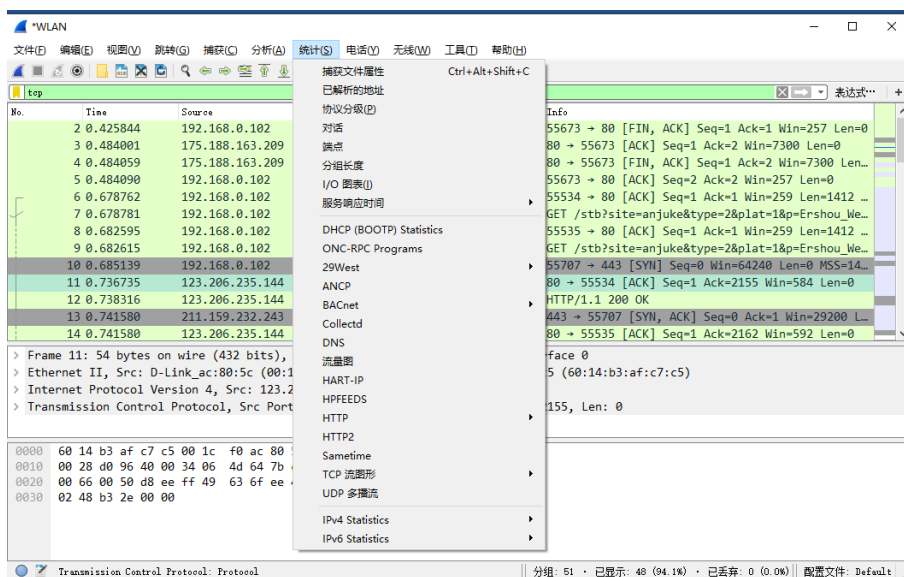


图 1.1-8 统计功能

5.2 观察 MAC 地址

启动 Wireshark 捕捉数据包，在命令行窗口分别 ping 网关和 ping 同网段的一台主机，分析本机发出的数据包。重点观察以太网帧的 Destination 和 Source 的 MAC 地址，辨识 MAC 地址类型，解读 OUI 信息、I/G 和 G/L 位。

5.3 分析以太网的帧结构

选择其中一个数据包，点击 Ethernet II 展开（图1.1-9），查看 MAC 帧的各个字段。

5.4 ARP 协议分析

1. 使用 `arp -d` 命令（其语法见图1.1-10），清空本机的 ARP 缓存，开启 Wireshark，ping 本机的同网段地址，在显示过滤器条框中输入“arp”，观察捕获的 ARP 报文的各个字段，分析请求/响应的过程。
2. 使用 `arp -d` 命令，清空本机的 ARP 缓存。开启 Wireshark，ping 与本机网段不同的 IP 地址或域名，观察捕获的 ARP 报文的各个字段，分析请求/响应的过程。

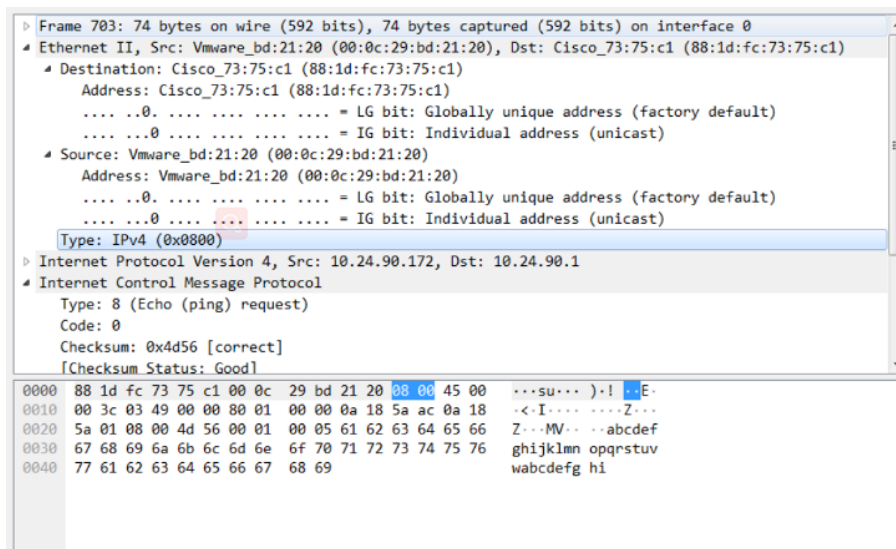


图 1.1-9 以太网帧结构展开界面

```
arp [-a [InetAddr] [-N IfaceAddr]] [-g [InetAddr] [-N IfaceAddr]]
    [-d InetAddr [IfaceAddr]] [-s InetAddr EtherAddr [IfaceAddr]]
```

- a 显示所有接口/特定接口的当前 ARP 缓存表
- g 同 -a
- d 删除所有/指定的 IP 地址项
- s 在 ARP 缓存中添加对应 InetAddr 地址的 EtherAddr 地址静态项

图 1.1-10 arp 命令语法及参数

六． 思考题

1. 使用了显示过滤器后，Wireshark 的抓包工作量会减少吗？
2. MAC 帧的长度和 IP 数据报的长度有怎样的关系？请用你的数据记录进行验证。
3. ping 同一局域网内的主机和局域网外的主机，都会产生 ARP 报文么？所产生的 ARP 报文有何不同，为什么？

实验 3.2 IP 与 ICMP 分析

一．实验目的

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。

二．实验内容

启动 Wireshark，捕捉网络命令执行过程中本机接受和发送的数据报。

1. **执行 ping 命令，观察 IP 数据报和 ICMP 询问报文的结构：**通过 Wireshark 监视器观察捕获流量中的 ICMP 询问报文和 IP 数据报的结构。注意比较 ICMP 请求帧与回应帧，及其 IP 头部数据字段的异同。
2. **改变 ping 命令的参数，观察 IP 数据报分片：**更改 ping 命令参数 MTU，使其发出长报文以触发 IP 数据报分片，再观察 IP 数据报的结构变化。
3. **执行 Traceroute 命令，观察 ICMP 差错报文的结构，并分析其工作原理：**使用 Linux 操作系统提供的 traceroute 命令（或者 Windows 系统提供的 tracert 命令），捕获和分析该命令所产生的 IP 数据报，特别注意相关的 ICMP 差错报文。结合捕获的具体数据，画出命令执行过程中数据交互的示意图，掌握 traceroute 的工作原理。

三．实验原理、方法和手段

3.1 IP 协议及数据报格式

网际互连协议（Internet Protocol, IP），是 TCP/IP 体系中的网络层协议，可实现大规模的异构网络互联互通，为主机提供无连接的、尽力而为的数据包传输服务。在网际协议第 4 版（IPv4）中，IP 数据报是一个可变长分组，包括首部和数据两部分（如图 1.2-1）。首部由 20~60 字节组成，包含与路由选择和传输有关的重要信息，其各字段意义如下：

1. **版本（4 位）：**该字段定义 IP 协议版本，所有字段都要按照此版本的协议来解释。

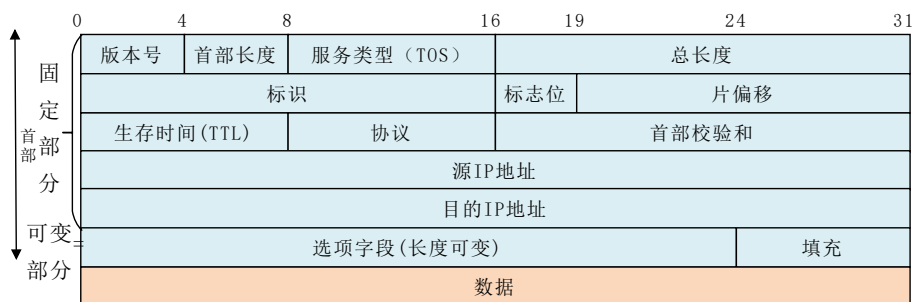


图 1.2-1 IP 数据报结构示意图

2. **首部长度 (4 位)**: 该字段定义数据报协议头长度, 表示协议首部具有 32 位字长的数量, 最小值为 5, 最大值为 15。
3. **服务 (8 位)**: 该字段定义上层协议对处理当前数据报所期望的服务质量, 并对数据报按照重要性级别进行分配。前 3 位为优先位, 后面 4 位为服务类型, 最后 1 位没有定义。这 8 位可用于分配优先级、延迟、吞吐量以及可靠性。
4. **总长度 (16 位)**: 该字段定义整个 IP 数据报的字节长度, 包括协议首部和数据, 其最大值为 65535 字节。
5. **标识 (16 位)**: 该字段包含一个整数, 用于标识当前数据报。当数据报分片时, 标识字段的值被复制到所有的分片中。
6. **标记 (3 位)**: 该字段由 3 位字段构成, 其中最低位 (MF) 控制分片: 若存在下一个分片则值为 1; 否则置 0 代表该分片是最后一个。中间位 (DF) 指出数据报是否可进行分片, 若置 1 则不允许该数据报进行分片。第三位即最高位保留不使用, 值为 0。
7. **分片偏移 (13 位)**: 该字段指出数据分片在源数据报中的相对位置, 以 8 字节为长度单位。
8. **生存时间 (8 位)**: 该字段是计数器, 转发该数据报的路由器依次减 1 直至减少为 0。
9. **协议 (8 位)**: 该字段指出在 IP 层处理后, 由哪种上层协议接收该数据报。
10. **头部校验和 (16 位)**: 该字段帮助确保 IP 协议头的正确性。计算过程是先将校验和字段置为 0, 然后将整个头部每 16 位划分为一部分, 并将各部分相加, 其计算结果取反码, 填入校验和字段中。

11. 源地址 (32 位): 源主机的 IP 地址。

12. 目的地址 (32 位): 目标主机的 IP 地址。

一个 IP 包从源主机传输到目标主机可能需要经过多个传输媒介不同的网络。每种网络对数据帧都设置了一个最大传输单元 (MTU) 的限制 (例如以太网的 MTU 是 1500 字节)。因此, 当路由器在转发 IP 包时, 如果数据包的大小超过了出口链路网络的 MTU 时, 需对该 IP 数据报进行分片, 才能在目标链路上顺利传输。每个 IP 分片将独立传输, 直到所有分片都到达目的地后, 目标主机才会把他们重组成一个完整的 IP 数据报。在 IP 数据报的分片与重组过程中, 以下三个首部字段发挥了重要作用:

1. 标记的后两位: 最低位记为 MF (More Fragment), MF = 1 代表还有后续分片, MF = 0 表示此为原始数据报的最后分片。次低位 DF (Don't Fragment), 用来控制数据报是否允许分片。DF = 1 表示该数据报不允许分片; DF = 0 允许分片。
2. 标识符: 用于目的主机将 IP 数据报的各个分片重装成原来的数据报。
3. 片偏移: 以 8 字节为单位, 目的主机在重装 IP 数据报时需要根据该字段提供偏移量进行排序。这是因为数据分片的独立传输使各分片的到达顺序难以确定。

3.2 ICMP 协议及报文格式

因特网控制报文协议 (Internet Control Message Protocol, ICMP), 用于 IP 主机、路由器之间传递控制消息。控制消息是指网络是否连通、主机是否可达、路由是否可用等网络本身的控制管理消息, 对网络正常运行起着重要的作用。

ICMP 报文的类型可以分为 ICMP 差错报文和 ICMP 询问报文两种 (其结构如图 1.2-2)。ICMP 差错报告报文主要有终点不可达、源站抑制、超时、参数问题和路由重定向 5 种。ICMP 询问报文有回送请求和应答、时间戳请求和应答、地址掩码请求和应答以及路由器询问和通告 4 种。其常见的类型与代码如表 1.2-1 所示。

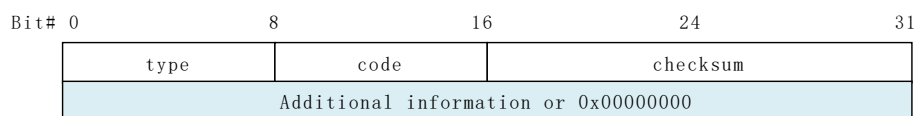


图 1.2-2 ICMP 报文结构示意图

本实验涉及以下两个常用网络命令, 都属于 ICMP 协议的典型应用。

表 1.2-1 ICMP 各类型报文的格式

类型 (TYPE)	代码 (CODE)	描述 (Description)	查询类 (Query)	差错类 (Error)
0	0	Echo Reply——回显应答 (Ping 应答)	✓	
3	1	Host Unreachable——主机不可达		✓
3	3	Port Unreachable——端口不可达		✓
3	4	Fragmentation needed but no frag. bit set ——需要进行分片但设置不分片比特		✓
8	0	Echo request——回显请求 (Ping 请求)	✓	
11	0	TTL equals 0 during transit ——传输期间生存时间为 0		✓

1. ping 命令，是测试网络最有效的工具之一。它是由主机或路由器执行 ping 命令向一个特定的目的主机发送一份 ICMP 回显请求（Echo request）报文，并等待其返回 ICMP 回显应答（Echo Reply）。ping 命令可以检测网络的连通性，简单估测数据包的往返时间（Round Trip Time），确定是否有数据包丢失或损坏，从而帮助分析网络故障。ping 命令格式和常用参数罗列如下：

```
ping [-t] [-a] [-n count] [-l length] [-f] [-i ttl] [-v tos]
    [-r count] [-s count] [-j computer-list] | [-k computer-list]
    [-w timeout] destination-list
```

- a 将地址解析为计算机名。
- n count 发送 count 指定的 ECHO 数据包数。默认值为 4。
- l length 发送包含由 length 指定的数据量的 ECHO 数据包。
默认为 32 字节；最大值是 65,527。
- f 在数据包中发送“不要分片”标志，避免数据包被路由上的网关分片。
- i ttl 将“生存时间”字段设置为 ttl 指定的值。

2. traceroute/tracert 命令，利用 TTL 字段和 ICMP 差错类型报文，查找 IP 数据包的路由转发路径（含路由器信息）。源主机执行该命令向目的主机发送生存时间（TTL）不同的 ICMP 回送请求报文，直至收到目的主机应答，并通过分析应答

报文获得转发路径和时延信息。

首先源主机发起一个 TTL=1 的 ICMP 报文。第一个路由器收到该报文后，TTL 减 1 变为 0 并丢弃此报文，返回一个 [ICMP time exceeded] 的消息。源主机通过这个消息获知 IP 数据报转发路径上的第一个路由器信息。然后，依次增加发送 ICMP 报文的 TTL 值，可以获取路径上的后续路由器的信息。当到达目的地时，目标主机返回一个 [ICMP port unreachable] 的消息，使发起者确认 IP 数据报已经正常到达。至此，tracert 命令发起者已经获得了通向目标主机路径上的所有路由信息。tracert 命令（Linux）格式和常用参数罗列如下：

```
tracert [-d] [-h maximum_hops] [-j host-list] [-w timeout] [-R]
        [-S srcaddr] [-4] [-6] target_name
```

```
-d 将地址解析成主机名
-h maximum_hops 搜索目标的最大跃点数
-j host-list 与主机列表一起的松散源路由，用于 IPv4
-w timeout 等待每个回复的超时时间（单位：毫秒）
-R 跟踪往返行程路径，用于 IPv6
-S srcaddr 要使用的源地址，用于 IPv6
-4 强制使用 IPv4
-6 强制使用 IPv6
target_name 指定目标，可以是 IP 地址或主机名
```

3.3 实验方法和手段

1. 使用 Wireshark 软件，捕获本机在 ping 和 traceroute 网络命令执行过程中接收和发出的全部数据流量。
2. 合理设置过滤条件，观察 IP 数据报和 ICMP 报文，着重分析报文首部和内容变化，从而掌握协议的工作原理。
3. 调整 ping 命令的参数，观察并分析 IP 数据报分片情况。
4. 结合所捕获的数据报，画出 traceroute 命令过程中数据交互示意图。

四．实验条件

装有 Wireshark 软件的 PC 机一台，处于局域网环境。

参考资料：

- [J.F Kurose and K.W. Ross, Wireshark Lab: ICMP v8.0](#)
- [Wireshark 官方过滤器语法指导书](#)
- [IP 协议的 RFC](#)

五．实验步骤

5.1 ping 命令

本机（示例 IP 为 192.168.1.251）启动 Wireshark 软件，选择要监听的网络接口（如 eth0、wlan0）；然后在终端发起网络命令：ping IP 地址/域名。

1. 在 Wireshark 监视器中设置过滤条件。例如图 1.2-3 设置过滤条件为 icmp，则显示出所捕获的 ICMP 数据包。

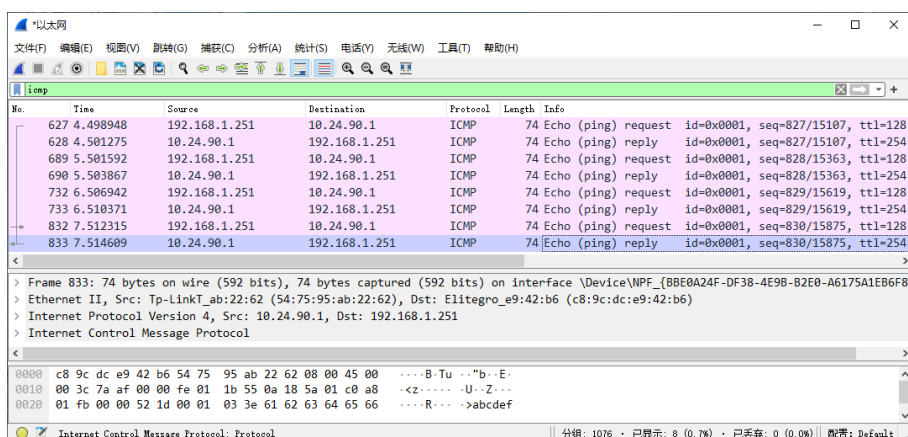


图 1.2-3 Wireshark 监视器界面

2. 点击 Internet Protocol Version 4 展开（如图 1.2-4），查看 IP 数据报，特别观察 IP 数据报的首部字段及其内容。
3. 点击 Internet Control Message Protocol 展开（如图 1.2-5），查看 ICMP 报文，并解释回显（Echo Request 和 Echo Reply）报文的首部字段。

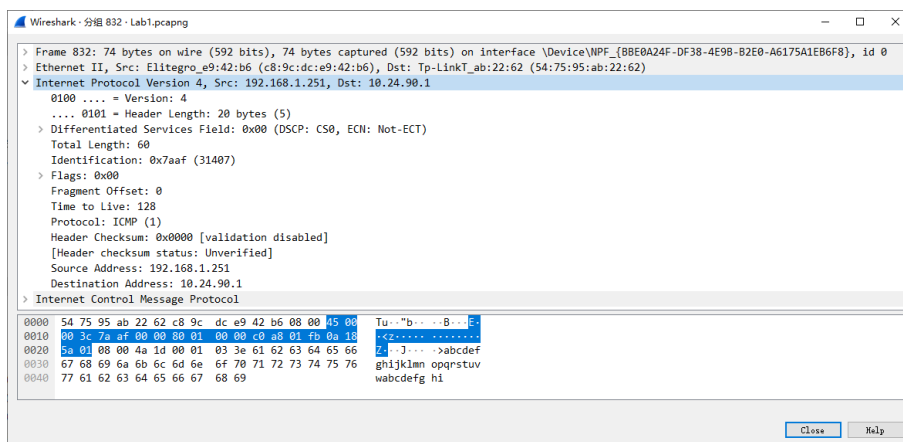


图 1.2-4 查看 IP 数据报

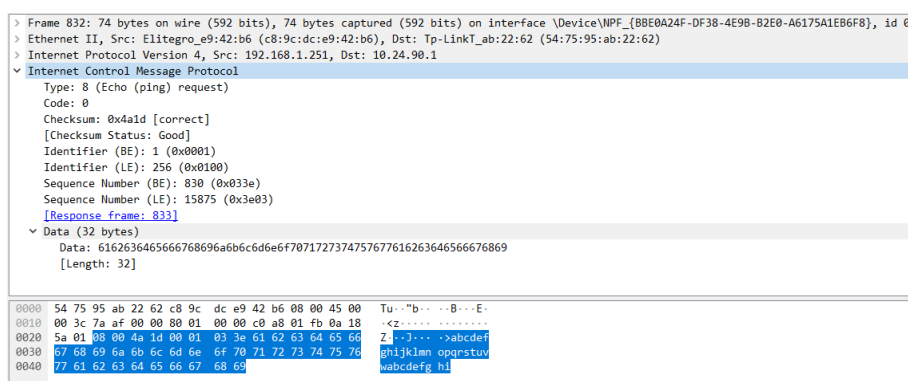


图 1.2-5 Echo request 示例

4. 清空 Wireshark 监控器，重新发起网络命令（如图 1.2-6）：ping IP 地址/域名 -l #length，并解释对比前后两次执行 ping 命令的结果。其中，-l #length 确定 echo 数据报的长度为 #length，其默认值为 32 字节，且小于 65,527 字节。
5. 可以多次改变 #length 的大小（例如 1000 字节、2000 字节和 4000 字节），观察 IP 数据报何时会分片？请解释 IP 数据报分片的原因和具体情况。提示：请先确认该网络的 MTU，可在 Wireshark 记录中查找“IPv4 fragments”项目。

5.2 traceroute 命令

本机（示例 IP 为 192.168.1.251）启动 Wireshark 软件，选择要监听的网络接口（如 eth0、wlan0）；然后在终端发起网络命令：traceroute IP 地址/域名。


```
管理员: C:\Windows\System32\cmd.exe
C:\Windows\system32>ping 10.24.90.1

正在 Ping 10.24.90.1 具有 32 字节的数据:
来自 10.24.90.1 的回复: 字节=32 时间=7ms TTL=255
来自 10.24.90.1 的回复: 字节=32 时间=2ms TTL=255
来自 10.24.90.1 的回复: 字节=32 时间=3ms TTL=255
来自 10.24.90.1 的回复: 字节=32 时间=2ms TTL=255

10.24.90.1 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 4, 丢失 = 0 (0% 丢失),
    往返行程的估计时间(以毫秒为单位):
        最短 = 2ms, 最长 = 7ms, 平均 = 3ms

C:\Windows\system32>ping 10.24.90.1 -l 4000

正在 Ping 10.24.90.1 具有 4000 字节的数据:
来自 10.24.90.1 的回复: 字节=4000 时间=8ms TTL=255
来自 10.24.90.1 的回复: 字节=4000 时间=4ms TTL=255
来自 10.24.90.1 的回复: 字节=4000 时间=5ms TTL=255
来自 10.24.90.1 的回复: 字节=4000 时间=5ms TTL=255

10.24.90.1 的 Ping 统计信息:
    数据包: 已发送 = 4, 已接收 = 4, 丢失 = 0 (0% 丢失),
    往返行程的估计时间(以毫秒为单位):
        最短 = 4ms, 最长 = 8ms, 平均 = 5ms

C:\Windows\system32>
```

图 1.2-6 ping 命令执行示例

1. 启动 Wireshark 软件，选择要监听的网络接口，设置过滤条件 icmp（如图1.2-7）。

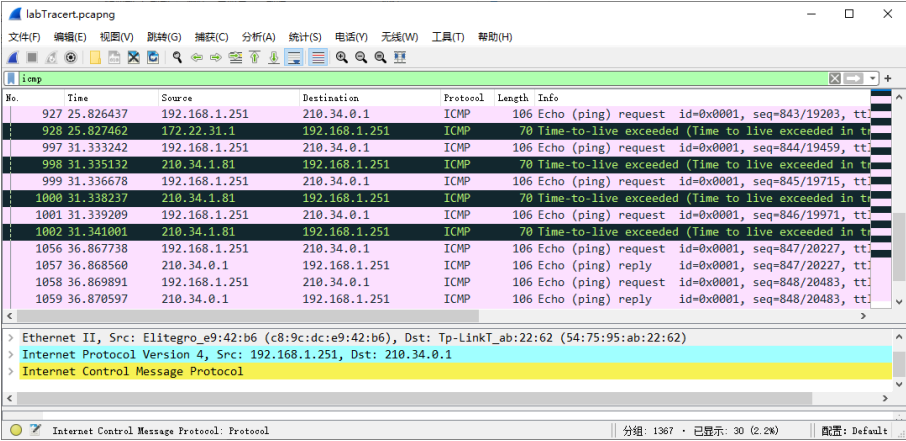


图 1.2-7 在 Wireshark 中设置过滤条件

2. 在终端中使用 traceroute 命令，目的主机是外网的一台设备（如图1.2-8，示例 IP 为 210.34.0.1）。
3. 点击 Internet Control Message Protocol 展开，查看 ICMP 差错报文，观察并解释 ICMP 报文结构和字段内容。
4. 结合 ICMP 报文记录画出数据交互示意图，并描述 traceroute 工作原理。

```
C:\WINDOWS\system32\cmd.exe
C:\>tracert 210.34.0.1
通过最多 30 个跃点跟踪
到 router.xmu.edu.cn [210.34.0.1] 的路由:
 1  <1 毫秒    <1 毫秒    <1 毫秒  192.168.1.1
 2  2 ms      3 ms      2 ms    59.77.8.1
 3  1 ms      <1 毫秒    <1 毫秒  172.22.31.1
 4  2 ms      1 ms      1 ms    210.34.1.81
 5  1 ms      <1 毫秒    <1 毫秒  router.xmu.edu.cn [210.34.0.1]
跟踪完成。
C:\>
```

图 1.2-8 在终端中执行 traceroute 命令示例

六．考核方法

本次实验需提交一份实验报告和一组 Wireshark 数据存储文件。报告内容应当包括以下两个部分，其中的分析解释都有对应的截图，且与数据存储文件记录相符。

1. 实施ping 命令，记录引发的 IP 数据报和 ICMP 报文，保存为 pcapng 文件；解释任一个 IP 数据报的首部，并对比 ICMP Echo 请求帧和回应帧；改变 ping 的长度参数，解释 IP 数据报分片情况。
2. 记录自己在本次实验中所遇到的问题，以心得感悟。如果遇到异常情况，或者无法完成任务时，也请分析错误产生的原因。