计算机网络

实验二

学　　院	信息学院
专　　业	计算机科学与技术
年　　级	2021 级
学　　号	22920212204066
姓　　名	邓语苏
日　　期	2023 年 10 月 19 日
地　　点	厦门大学翔安校区

实验二

目录

1 实验目的

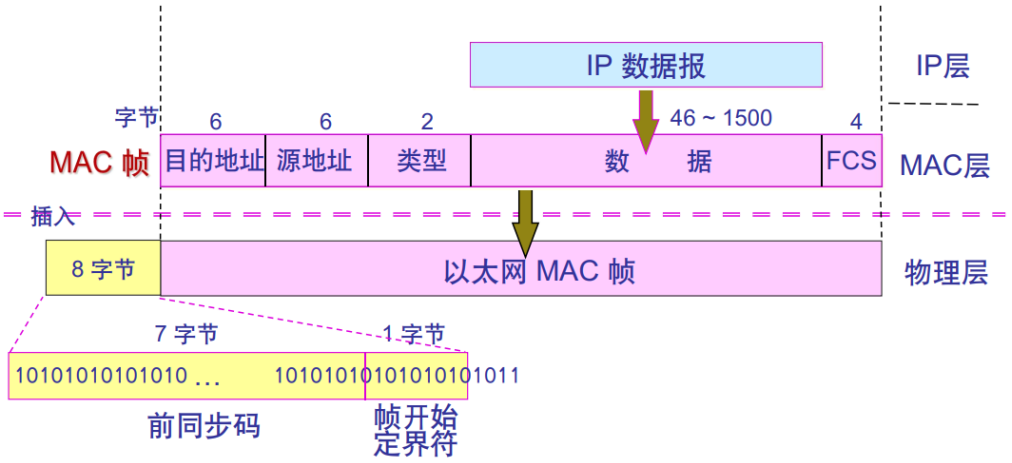
- 学习捕获和分析网络数据包
- 掌握以太网 MAC 帧、802.11 数据帧和 IPV4 数据包的构成，了解各字段的含义
- 掌握 ICMP 协议，ping 和 tracert 指令的工作原理
- 掌握 ARP 协议的请求/响应机理

2 实验内容

1. 捕获和分析有线以太网数据包

   - 分析 MAC 帧

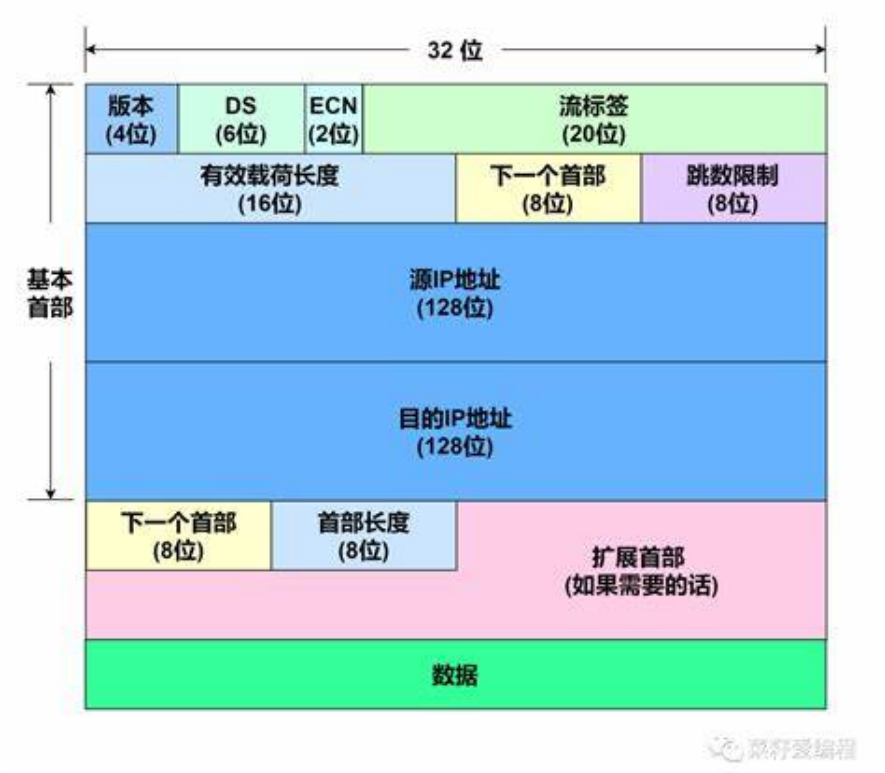
图 1: MAC 帧



   - 分析 IP 数据报首部

图 2: IPv4 数据报报文格式

图 3: IPv6 数据报报文格式

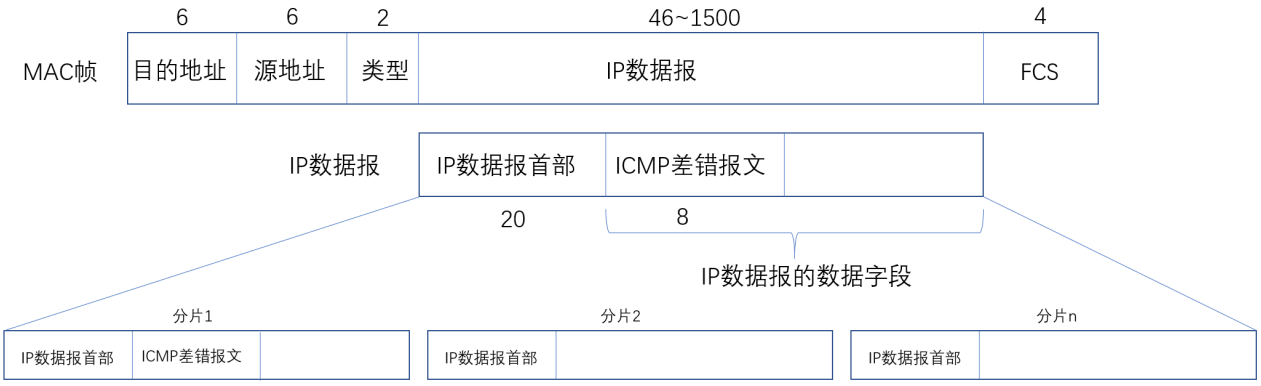


- 观察 IP 分片

  给 www.xmu.edu.cn 发送以下 ping 命令，结合 CMD 和 Wireshark 分析命令结果

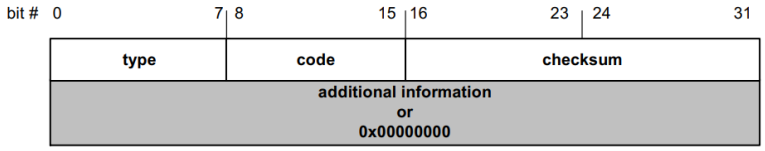
  - ping -4 www.xmu.edu.cn
  - ping www.xmu.edu.cn -l 1472 -f -n 1
  - ping www.xmu.edu.cn -l 1473 -f -n 1
  - ping www.xmu.edu.cn -l 1473 -n 1

图 4: 分片示意图



- ICMP 协议分析（以 ping 指令为例）

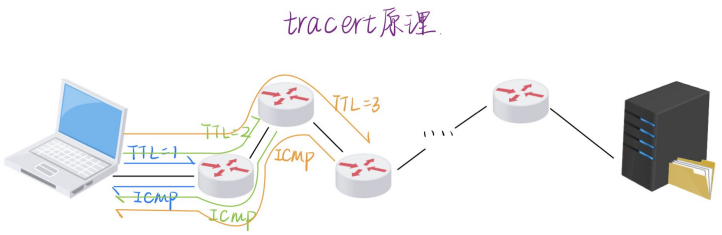
图 5: ICMP 报文的格式



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		√

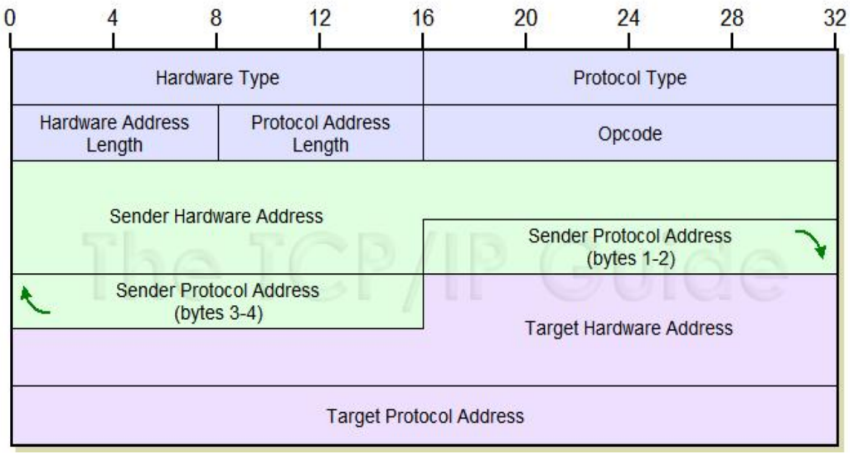
- tracert 工作原理分析

图 6: tracert 原理示意图



- ARP 协议分析

图 7: ARP 报文的格式



2. 捕获和分析 802.11 数据

构建无线环境，捕获无线数据包、分析 802.11 数据

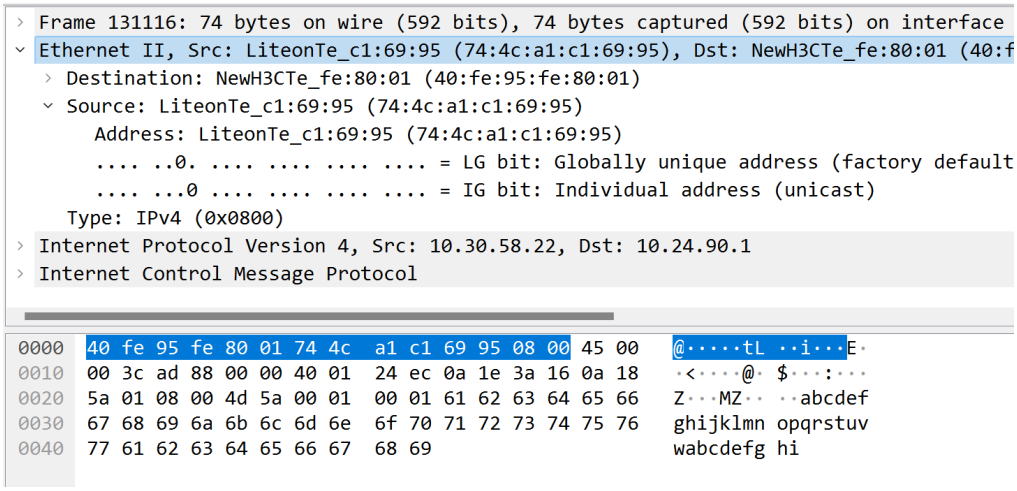
3. 探索 Wireshark 更多功能和其它抓包工具（选做）

3 任务一：捕获和分析有线以太网数据包

3.1 观察 MAC 帧格式

使用 Wireshark 抓取数据包，查看其以太网帧头部

图 8: 以太网帧头部信息



MAC 帧前 12 字节分别为目的地址、源地址；图中所示的数据包目的地址的 MAC 地址为 40:fe:95:fe:80:01，源地址的 MAC 地址为 74:4c:a1:c1:69:95。紧接着的两个字节标志类型；图中的类型标志的值为 0x8000，表示下一层为 IP 协议。

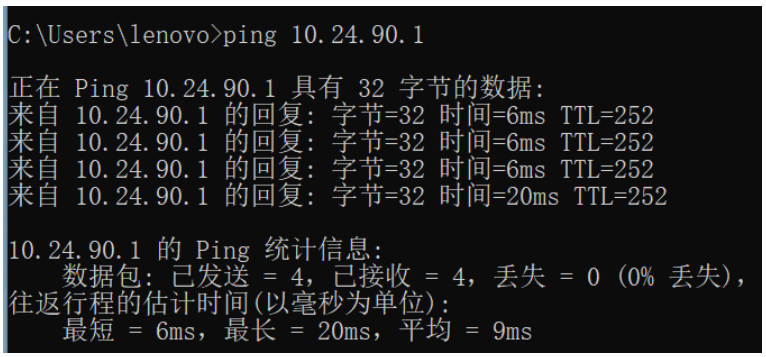
MAC 地址，是一个 48 位的二进制数。它的前 24 位是厂商的 OUI。其中第一个字节的后两位分别表示了是否是全局唯一的地址和是否是多播地址。例如目的地址的前两位是 00，表示是全局唯一的地址，进行单播；源地址也是如此。

3.2 观察 IP 数据报的首部结构

3.2.1 IPv4 数据报结构

打开 CMD 窗口 ping 一个格式为 IPv4 的 IP 地址

图 9: ping IPv4 地址



使用 Wireshark 筛选出 IP 地址为 10.24.90.1 的数据包

图 10

ip.addr==10.24.90.1

Source	Destination	Protocol	Length
10.30.58.22	10.24.90.1	ICMP	74
10.24.90.1	10.30.58.22	ICMP	74
10.30.58.22	10.24.90.1	ICMP	74
10.24.90.1	10.30.58.22	ICMP	74
10.30.58.22	10.24.90.1	ICMP	74
10.24.90.1	10.30.58.22	ICMP	74
10.30.58.22	10.24.90.1	ICMP	74
10.24.90.1	10.30.58.22	ICMP	74

点击任意一个数据包，观察其 IP 数据报首部结构

图 11: IPv4 数据报首部结构

```
> Ethernet II, Src: NewH3CTe_fe:80:01 (40:fe:95:fe:80:01), Dst: LiteonTe_c1:69
v Internet Protocol Version 4, Src: 10.24.90.1, Dst: 10.30.58.22
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x60 (DSCP: CS3, ECN: Not-ECT)
    Total Length: 60
    Identification: 0xf736 (63286)
  > 000. .... = Flags: 0x0
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 252
    Protocol: ICMP (1)
    Header Checksum: 0x1edd [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 10.24.90.1
    Destination Address: 10.30.58.22
> Internet Control Message Protocol
```

```
0000  74 4c a1 c1 69 95 40 fe  95 fe 80 01 08 00 45 60   tL··i·@· ·····E`
0010  00 3c f7 36 00 00 fc 01  1e dd 0a 18 5a 01 0a 1e   ·<·6···· ····Z···
0020  3a 16 00 00 55 57 00 01  00 04 61 62 63 64 65 66   :···UW·· ··abcdef
0030  67 68 69 6a 6b 6c 6d 6e  6f 70 71 72 73 74 75 76   ghijklmn opqrstuv
0040  77 61 62 63 64 65 66 67  68 69                    wabcdefg hi
```

第一个字节长度为 8 位，前 4 位为版本号，后四位为首部长度。在该数据报中，版本号为 4，表示为 IPv4 类型的 IP 数据报；首部长度为 20 字节；接着为服务类型 TOS（也称区分服务 DS），占 1 字节；该 IP 数据包的总长度为 60 字节；标识占两字节为 0xf736；标志位 flag 占 3 位，为 0x0；偏位移占 13 位，为 0；生存时间 TTL 为 252,；协议字段占 8 位，表明此数据包携带的数据是适用何种协议，本数据报适用协议为 ICMP；首部检验和为 0x1edd；源 IP 地址为 10.24.90.1；目的 IP 地址为 10.30.58.22

3.2.2 IPv6 数据报结构

打开 CMD 窗口 ping 一个格式为 IPv6 的 IP 地址

```
C:\Users\lenovo>ping -6 2001:dc7:dd01:0:218:241:97:42

正在 Ping 2001:dc7:dd01:0:218:241:97:42 具有 32 字节的数据:
来自 2001:dc7:dd01:0:218:241:97:42 的回复: 时间=50ms
来自 2001:dc7:dd01:0:218:241:97:42 的回复: 时间=53ms
来自 2001:dc7:dd01:0:218:241:97:42 的回复: 时间=48ms
来自 2001:dc7:dd01:0:218:241:97:42 的回复: 时间=51ms

2001:dc7:dd01:0:218:241:97:42 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 = 48ms，最长 = 53ms，平均 = 50ms
```

图 12: ping IPv6 地址

使用 Wireshark 抓取对应数据包，观察其 IPv6 数据报首部结构如下

```
> Ethernet II, Src: LiteonTe_c1:69:95 (74:4c:a1:c1:69:95), Dst: NewH3CTe_fe:80:01 (40:fe:95:fe:80:01)
v Internet Protocol Version 6, Src: 2409:8734:1a70:7e2:c7c:6cac:3729:f2bb, Dst: 2001:dc7:dd01:0:218:241
    0110 .... = Version: 6
  > .... 0000 0000 .... .... .... .... .... = Traffic Class: 0x00 (DSCP: CS0, ECN: Not-ECT)
    .... 0000 0000 0000 0000 0000 = Flow Label: 0x00000
    Payload Length: 40
    Next Header: ICMPv6 (58)
    Hop Limit: 64
    Source Address: 2409:8734:1a70:7e2:c7c:6cac:3729:f2bb
    Destination Address: 2001:dc7:dd01:0:218:241:97:42
> Internet Control Message Protocol v6
```

```
0000  40 fe 95 fe 80 01 74 4c  a1 c1 69 95 86 dd 60 00   @·····tL ··i···`·
0010  00 00 00 28 3a 40 24 09  87 34 1a 70 07 e2 0c 7c   ···(:@$· ·4·p···|
0020  6c ac 37 29 f2 bb 20 01  0d c7 dd 01 00 00 02 18   l·7)·· · ········
0030  02 41 00 97 00 42 80 00  54 5e 00 01 00 01 61 62   ·A···B·· T^····ab
0040  63 64 65 66 67 68 69 6a  6b 6c 6d 6e 6f 70 71 72   cdefghij klmnopqr
0050  73 74 75 76 77 61 62 63  64 65 66 67 68 69        stuvwabc defghi
```

图 13: IPv6 数据报首部结构

前 4 为位版本，值为 6，为 IPv6 数据报；接着 8 位为服务类型 (包括 DS、ECN)；后面位 20 位流标签；有效载荷长度为 16；下一个首部为 ICMPv6; 跳数限制为 8 位；源 IP 地址为 2409:8734:1a70:7e2:c7c:6cac:3729:f2bb、目的 IP 地址为 2001:dc7:dd01:0:218:241:97:42。

3.3 观察 IP 分片

使用-4 选项指定使用 IPv4 地址 ping 网站 www.xmu.edu.cn；-f 选项表示在数据包发送前不分片；-n 1 表示只发送一个 ping 请求

```
C:\Users\lenovo>ping -4 www.xmu.edu.cn

正在 Ping cmsn1.xmu.edu.cn [219.229.81.200] 具有 32 字节的数据:
来自 219.229.81.200 的回复: 字节=32 时间=3ms TTL=59
来自 219.229.81.200 的回复: 字节=32 时间=6ms TTL=59
来自 219.229.81.200 的回复: 字节=32 时间=5ms TTL=59
来自 219.229.81.200 的回复: 字节=32 时间=4ms TTL=59

219.229.81.200 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 = 3ms，最长 = 6ms，平均 = 4ms

C:\Users\lenovo>ping www.xmu.edu.cn -l 1472 -f -n 1

正在 Ping cmsn1.xmu.edu.cn [219.229.81.200] 具有 1472 字节的数据:
来自 219.229.81.200 的回复: 字节=1472 时间=7ms TTL=59

219.229.81.200 的 Ping 统计信息:
    数据包: 已发送 = 1，已接收 = 1，丢失 = 0 (0% 丢失)，
往返行程的估计时间(以毫秒为单位):
    最短 = 7ms，最长 = 7ms，平均 = 7ms
```

```
C:\Users\lenovo>ping www.xmu.edu.cn -l 1473 -f -n 1

正在 Ping cmsn1.xmu.edu.cn [219.229.81.200] 具有 1473 字节的数据:
需要拆分数据包但是设置 DF。

219.229.81.200 的 Ping 统计信息:
    数据包: 已发送 = 1，已接收 = 0，丢失 = 1 (100% 丢失)，

C:\Users\lenovo>ping www.xmu.edu.cn -l 1473 -n 1

正在 Ping cmsn1.xmu.edu.cn [2001:da8:e800:251c::200] 具有 1473 字节的数据:
来自 2001:da8:e800:251c::200 的回复: 时间=3ms

2001:da8:e800:251c::200 的 Ping 统计信息:
    数据包: 已发送 = 1，已接收 = 1，丢失 = 0 (0% 丢失)，
往返行程的估计时间(以毫秒为单位):
    最短 = 3ms，最长 = 3ms，平均 = 3ms
```

分析 CMD 窗口

除非使用-n 1 指定仅请求一次，否则 ping 指令默认请求网站 4 次；1473 字节超过最大传输单元 MTU，需要进行分片处理，若使用-f 强制数据包不分片会造成丢包。同时可以看到 ping1472 不分片的数据，比 ping1473 分片的数据更慢。

使用 Wireshark 抓包分析

单独分析 ping -4 www.xmu.edu.cn、ping -4 www.xmu.edu.cn -l 1473 -n 1 两个特殊的指令。

1. ping -4 www.xmu.edu.cn

Source	Destination	Protocol	Length	Info
10.30.58.22	219.229.81.200	ICMP	74	Echo (ping) request
219.229.81.200	10.30.58.22	ICMP	74	Echo (ping) reply
10.30.58.22	219.229.81.200	ICMP	74	Echo (ping) request
219.229.81.200	10.30.58.22	ICMP	74	Echo (ping) reply
10.30.58.22	219.229.81.200	ICMP	74	Echo (ping) request
219.229.81.200	10.30.58.22	ICMP	74	Echo (ping) reply
10.30.58.22	219.229.81.200	ICMP	74	Echo (ping) request
219.229.81.200	10.30.58.22	ICMP	74	Echo (ping) reply

图 14: ping -4 www.xmu.edu.cn

   ping 命令默认请求 4 次，收到回复 4 次，共抓取 8 个数据包。

2. ping -4 www.xmu.edu.cn -l 1473 -n 1

   原数据字段为 1473 字节，超出了 IP 数据报的最大长度 (1500)。原因如下

$$\text{IP 数据报长度} = \text{IP 数据报首部长度} + \text{ICMP 差错报文前 8 字节} + \text{传输数据长度}$$

即此时的 IP 数据报长度为 20+8+1473=1501 字节，超出范围，需要分片。数据字段总长为 8+1473=1481 字节

	Time	Source	Destination	Protocol	Length	Info
6877	43.069650	10.30.58.22	219.229.81.200	IPv4	1514	Fragmented IP protocol (proto=ICMP 1, of
6878	43.069650	10.30.58.22	219.229.81.200	ICMP	35	Echo (ping) request id=0x0001, seq=1/25
6879	43.090736	219.229.81.200	10.30.58.22	IPv4	1514	Fragmented IP protocol (proto=ICMP 1, of
6880	43.121814	219.229.81.200	10.30.58.22	ICMP	60	Echo (ping) reply id=0x0001, seq=1/25
29662	193.012717	10.30.58.22	219.229.81.200	IPv4	1514	Fragmented IP protocol (proto=ICMP 1, of
29663	193.012717	10.30.58.22	219.229.81.200	IPv4	1514	Fragmented IP protocol (proto=ICMP 1, of
29664	193.012717	10.30.58.22	219.229.81.200	ICMP	82	Echo (ping) request id=0x0001, seq=2/51
29665	193.091762	219.229.81.200	10.30.58.22	IPv4	1514	Fragmented IP protocol (proto=ICMP 1, of
29666	193.092644	219.229.81.200	10.30.58.22	IPv4	1514	Fragmented IP protocol (proto=ICMP 1, of
29667	193.092983	219.229.81.200	10.30.58.22	ICMP	82	Echo (ping) reply id=0x0001, seq=2/51

```
> 000. .... = Flags: 0x0
  ...0 0000 1011 1001 = Fragment Offset: 1480
  Time to Live: 64
  Protocol: ICMP (1)
  Header Checksum: 0x0d1e [validation disabled]
  [Header checksum status: Unverified]
  Source Address: 10.30.58.22
  Destination Address: 219.229.81.200
v [2 IPv4 Fragments (1481 bytes): #6877(1480), #6878(1)]
    [Frame: 6877, payload: 0-1479 (1480 bytes)]
    [Frame: 6878, payload: 1480-1480 (1 byte)]
    [Fragment count: 2]
    [Reassembled IPv4 length: 1481]
    [Reassembled IPv4 data: 0800df4500010001616263646566676869 6a6b6c6d6e
```

```
0000  40 fe 95 fe 80 01 74 4c  a1 c1 69 9
0010  00 15 fb 2f 00 b9 40 01  0d 1e 0a 1
0020  51 c8 61
```

图 15: ping -4 www.xmu.edu.cn -l 1473 -n 1

观察 fragments 的信息可以看到分为两个分片，编号分别为 6877、6878。其中编号 6877 的 IP 数据报带有 1480 字节的原数据字段，编号为 6878 的 IP 数据报带有 1 字节的原数据字段。1+1480=1481 字节，与原 IP 数据报的数据字段长一致。

编号为 6877 的帧长为 1514=14(MAC 帧首部)+20(IP 数据报首部)+8(ICMP 差错报文前 8 字节)+1472

编号为 6878 的帧长为 35=14(MAC 帧首部)+20(IP 数据报首部)+1

3.4 ICMP 协议分析

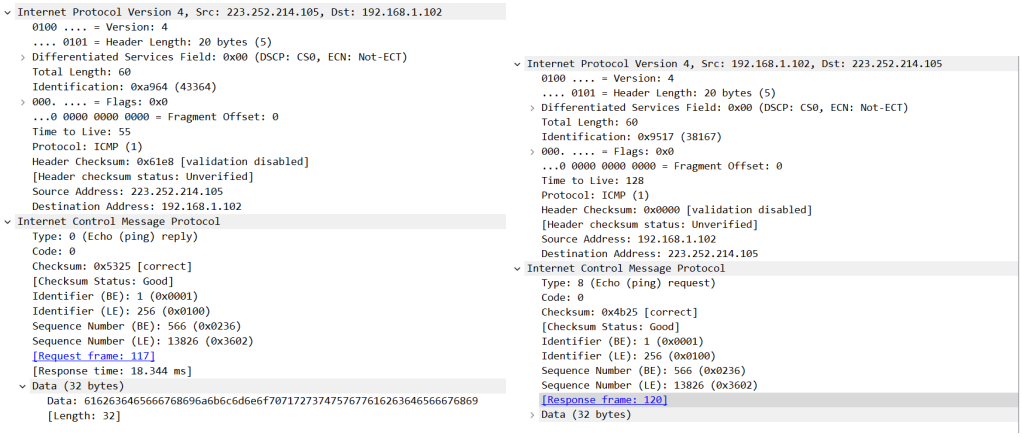
在 Windows 下，一次 Ping 命令会进行 4 次 ICMP 请求，因此有 8 个数据报，其中四次请求四次回应。如图 16所示。

图 16: Ping 产生的数据报

Destination	Protocol	Length	Info
223.252.214.105	ICMP	74	Echo (ping) request
192.168.1.102	ICMP	74	Echo (ping) reply
223.252.214.105	ICMP	74	Echo (ping) request
192.168.1.102	ICMP	74	Echo (ping) reply
223.252.214.105	ICMP	74	Echo (ping) request
192.168.1.102	ICMP	74	Echo (ping) reply
223.252.214.105	ICMP	74	Echo (ping) request
192.168.1.102	ICMP	74	Echo (ping) reply

数据包内容如图 21，ICMP 请求和回应的 Code 分别为 8-0 和 0-0。IP 部分首先两者的源地址和目的地址相反，其次 TTL 也不一样，可能是由于操作系统不同。

图 17: ICMP 请求



(a) 请求 (b) 回复

3.5 tracert 工作原理分析

tracert 程序的设计利用 ICMP 及 TTL(生存时间/跳) 来列出所有经过的节点。

首先，tracert 送出一个 TTL 是 1 的数据报到目的地，当路径上的第一个路由器收到时，它将 TTL 减 1。此时，TTL 变为 0，所以该路由器会将数据报丢掉，并送回一个 ICMP 包，这包括发 IP 包的源地址，IP 包的所有内容及路由器的 IP 地址，tracert 收到这个消息后便知道这个路由器存在于这个路径上，接着 tracert 再送出另一个 TTL 是 2 的数据报，以此类推，traceroute 每次将送出的数据报的 TTL 加 1 来发现下一个路由器，这个重复的动作一直持续到某个数据报抵达目的地。这时，该主机并不会送回 ICMP 消息，因为它已是目的地了。

图 18: tracert 原理示意图

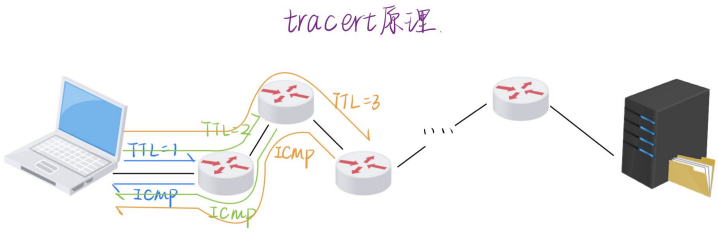
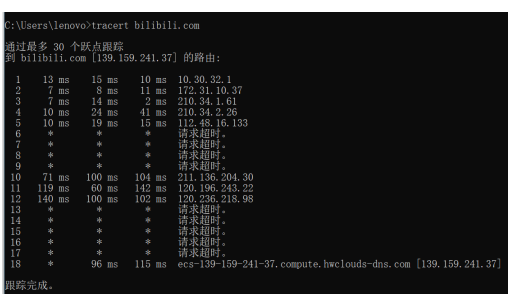
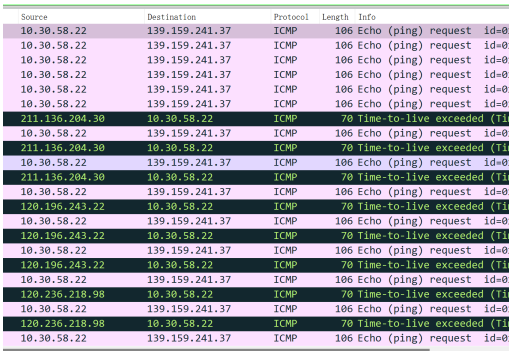


图 19: tracert 分析





(a) tracert 输出 (b) 产生的数据报

3.6 ARP 协议分析

实验要求 ping 同一局域网和局域网外的计算机，比较产生的 ARP 有何不同。

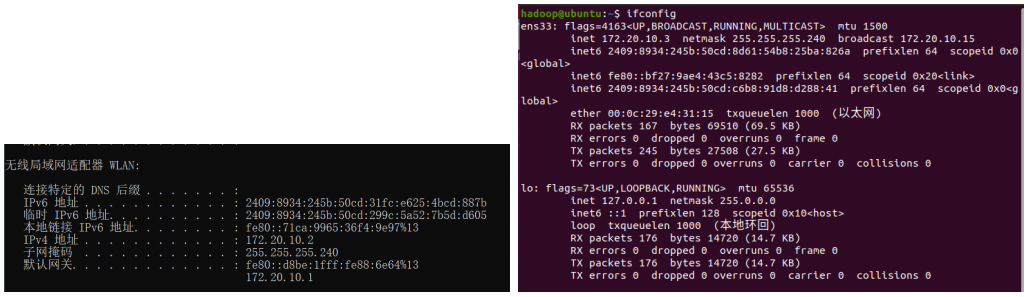
3.6.1 ping 局域网内主机

在学校或公共 Wi-Fi 热点中，可能存在多个 Wi-Fi 网络，每个网络都可以被视为一个独立的局域网。在这种情况下，连接到不同 Wi-Fi 网络的设备可能不在同一个局域网中。因此去邻座同学的 ip 也可能不在同一局域网下。

解决方案为 ping 虚拟机的 ip。将虚拟机的网络适配器设为桥接模式，如此可以将虚拟机视为同一局域网下的另一独立主机。

分别查看本机 IP 和虚拟机 IP

图 20: IP 地址



(a) 本机 (b) 虚拟机

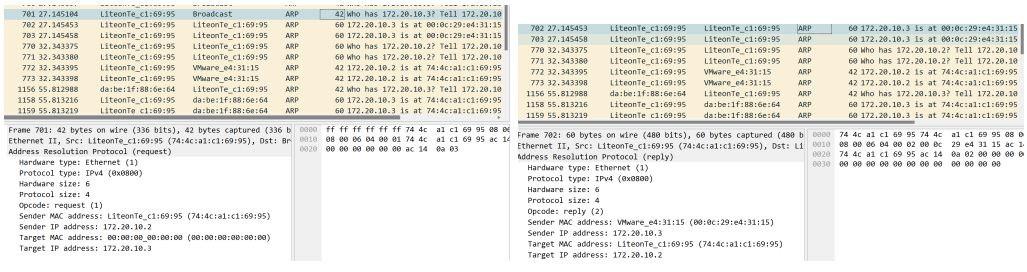
看得出本机的 IP 地址为 172.20.10.2，网关为 172.20.10.1; 虚拟机的 IP 地址为 172.20.10.3

使用管理员权限打开 CMD 窗口，输入以下代码，ping 虚拟机

```
arp -d
ping 172.20.10.3
```

使用 wireshark 抓包分析

图 21: ARP 协议



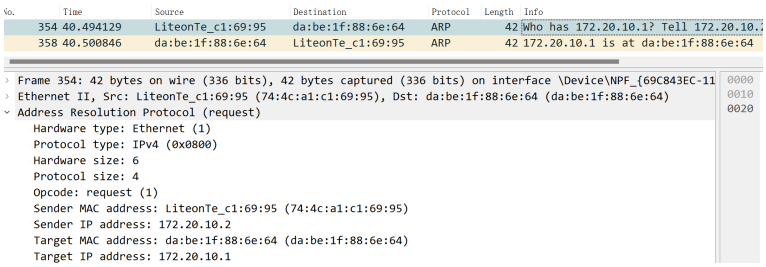
(a) 请求 (b) 回复

因为根据 ARP 的工作方式，所有帧都必须传送到本地网段中的节点。如果目的 IPv4 主机在本地网络上，帧将使用此设备的 MAC 地址作为目的 MAC 地址。

请求中 Target MAC address 为全 0，代表广播地址。回复会将请求的 IP 地址的 MAC 地址发回后填入 ARP 表中。

3.6.2 ping 局域网外主机

图 22: ARP 协议



如果目的 IPv4 主机不在本地网络上，则源节点需要将帧传送到作为网关的路由器接口，或用于到达该目的地的下一跳。源节点将使用网关的 MAC 地址作为帧（其中含有发往其它网络上主机的 IPv4 数据包）的目的地址。因此会产生这种现象。

4 任务二：捕获和分析 802.11 数据

在按照实验要求进行实验时，需要注意：在 Linux 中启动 Wireshark 需要 root 权限。否则无法访问虚拟网卡。

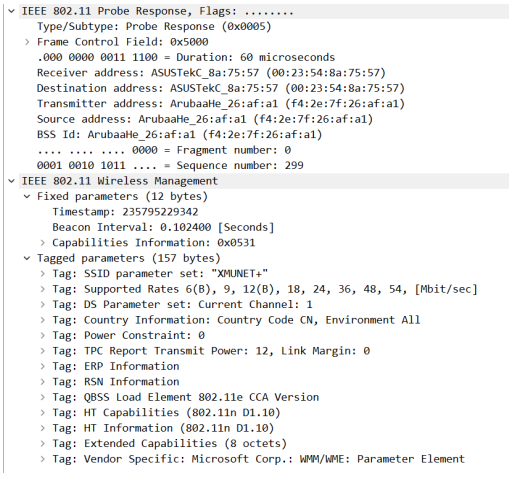
4.1 管理帧

在抓到的包中可以发现 Probe Request 和 Beacon 两种管理帧。如图 23。Beacon 帧主要来声明网络的存在。定期传送的信标可让移动式工作站该网络的存在，从而调整加入该网络所必需的参数。设备通过 Probe Request 帧来扫描所在区域内的 802.11 网络。若 Probe Request 帧探查的网络与之兼容，该网络就会回复 Probe Response 帧给予响应。

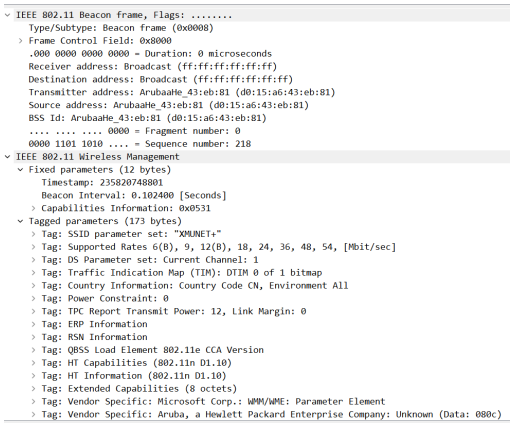
其中 Beacon 帧广播了 SSID、信道、BSSID 等信息，如图 23b。

可以看出 SSID 是 XMUNET+，信道是 1，BSSID 是 d0:15:a6:43:eb:81。

图 23: 管理帧



(a) Probe Response

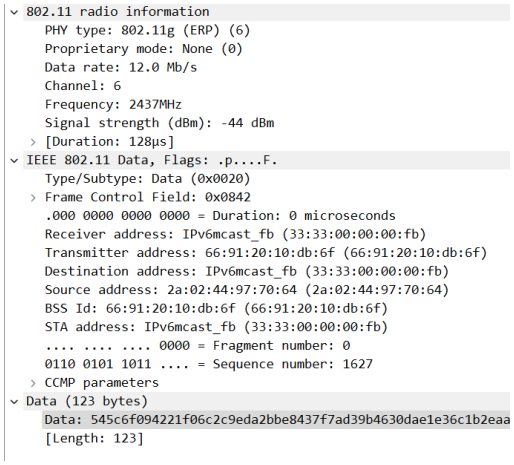


(b) Beacon

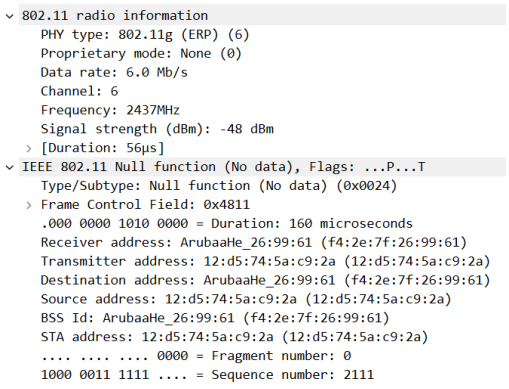
4.2 数据帧

在抓到的包中可以发现 Data、NULL Data、QoS Data 和 NULL QoS Data4 种数据帧。如图 24。数据帧会将上层协议的数据置于帧主体加以传递。

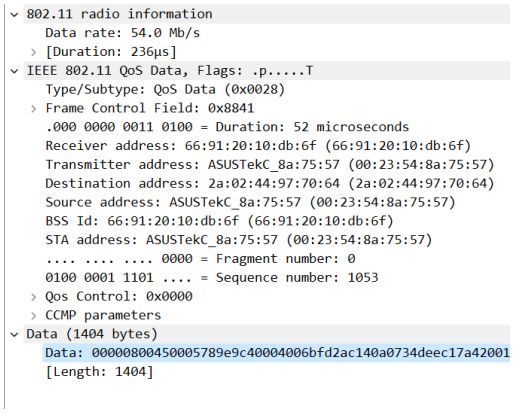
图 24: 数据帧



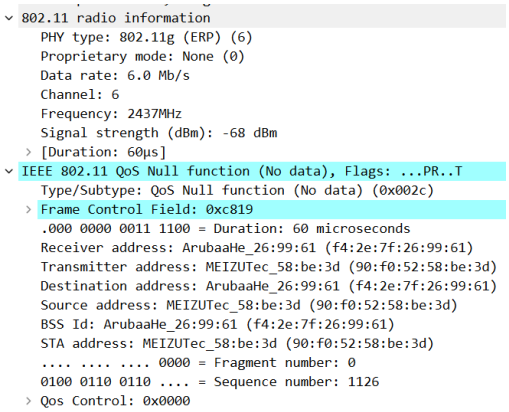
(a) Data



(b) Null Function



(c) QOS Data



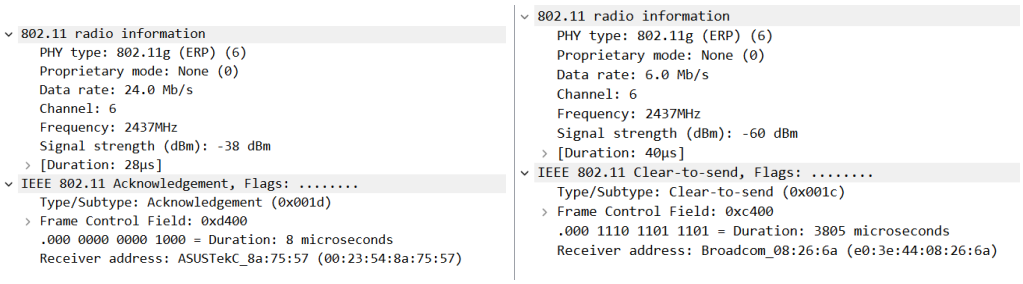
(d) QOS Null Function

4.3 控制帧

在抓到的包中可以发现 Acknowledgment 和 Clear to Send 两种控制帧。如图 25。每个发送的单播报文，接收者在成功接收到发送报文后，都要发送一个应答 ACK 进行确认。它的 Duration 是 8 微秒，能够反映出 ACK 信号在整个帧交换过程中位居何处。

目的客户端收到 RTS 后，发送一个 CTS 报文，这样在客户端覆盖范围内所有的设备都会在指定的时间内不发送数据。这里指定的时间是 3805 微秒。

图 25: 控制帧



(a) Acknowledgment (b) Clear to Send

5 任务三：探索 Wireshark 更多功能和其他抓包工具

5.1 数据流追踪

向百度图片搜索发送图片，可以使用数据流追踪来追踪数据流的传输过程。如图 26。

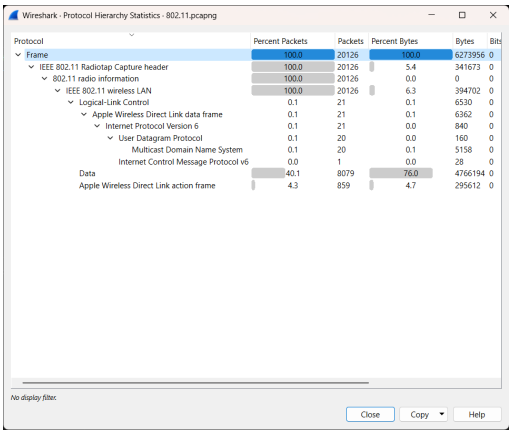
图 26: 数据流追踪

5.2 协议分层统计

以 802.11 为例，可以通过协议分层统计统计使用到了哪些协议，如图 27。

可以发现所有的包都包含在 802.11 中。

图 27: 协议分层统计



6 相关代码文档和文件记录

图 28: 抓包文件记录