计算机网络-实验4:网络层ICMP协议

姓名：翟一航 学号：23020011046

一、实验目的

In this lab, we'll explore several aspects of the ICMP protocol:

- ICMP messages generating by the Ping program;
- ICMP messages generated by the Traceroute program;
- the format and contents of an ICMP message.

二、实验环境

环境	说明
操作系统	MacOS
抓包工具	Wireshark

三、实验内容

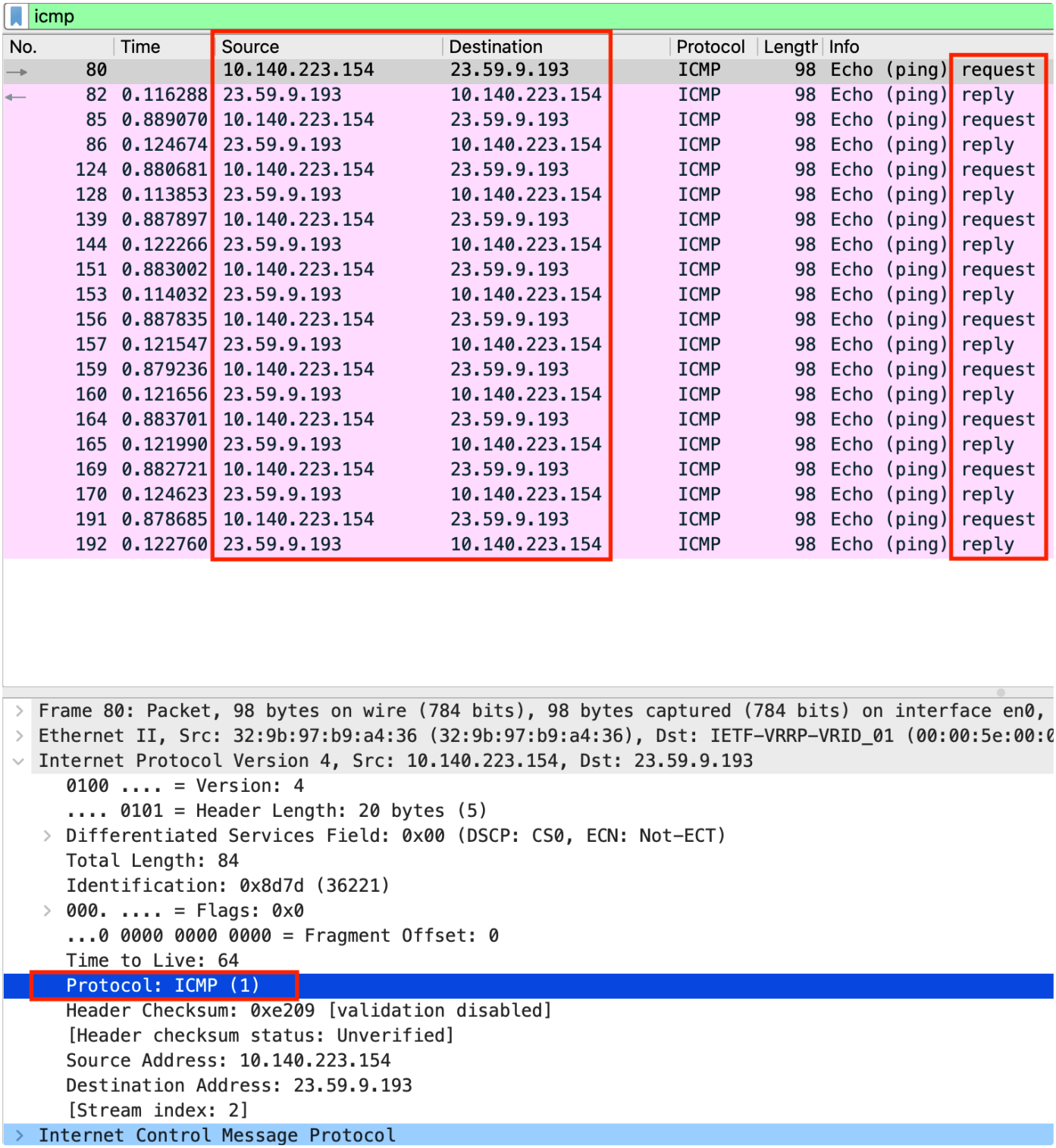
1. ICMP and Ping

图1:使用 ping 命令想 MIT 官网发送请求

```
~
❯ ping -c 10 www.mit.edu
PING www.mit.edu (23.59.9.193): 56 data bytes
64 bytes from 23.59.9.193: icmp_seq=0 ttl=43 time=116.603 ms
64 bytes from 23.59.9.193: icmp_seq=1 ttl=43 time=125.030 ms
64 bytes from 23.59.9.193: icmp_seq=2 ttl=43 time=114.277 ms
64 bytes from 23.59.9.193: icmp_seq=3 ttl=43 time=122.694 ms
64 bytes from 23.59.9.193: icmp_seq=4 ttl=43 time=114.338 ms
64 bytes from 23.59.9.193: icmp_seq=5 ttl=43 time=121.913 ms
64 bytes from 23.59.9.193: icmp_seq=6 ttl=43 time=122.122 ms
64 bytes from 23.59.9.193: icmp_seq=7 ttl=43 time=122.454 ms
64 bytes from 23.59.9.193: icmp_seq=8 ttl=43 time=125.049 ms
64 bytes from 23.59.9.193: icmp_seq=9 ttl=43 time=123.136 ms

--- www.mit.edu ping statistics ---
10 packets transmitted, 10 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 114.277/120.762/125.049/3.907 ms
```

图2:在 Wireshark 中使用 ICMP 过滤器查看发送与收到的 ICMP 数据包



可以看到20个 ICMP 数据包（10个 request，10个 reply），可以看到源端主机IP与目标主机IP，根据IP形式可以发现，源端主机IP是一个私有IP。除此之外，也能够发现 IP 数据报的协议号为1，说明IP数据报的有效载荷是一个ICMP数据包。

图3:查看request数据包的 ICMP 协议信息

```
> Frame 80: Packet, 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interface en0,
> Ethernet II, Src: 32:9b:97:b9:a4:36 (32:9b:97:b9:a4:36), Dst: IETF-VRRP-VRID_01 (00:00:5e:00:0
> Internet Protocol Version 4, Src: 10.140.223.154, Dst: 23.59.9.193
v Internet Control Message Protocol
    Type: Echo (ping) request (8)
    Code: 0
    Checksum: 0xd501 [correct]
    [Checksum Status: Good]
    Identifier (BE): 46158 (0xb44e)
    Identifier (LE): 20148 (0x4eb4)
    Sequence Number (BE): 0 (0x0000)
    Sequence Number (LE): 0 (0x0000)
    [Response frame: 82]
  > ICMP Data: 692c0444000d162f08090a0b0c0d0e0f101112131415161718191a1b1c1d1e1f2021222324252627
```

发现 ICMP 数据包的类型为8，代码为0，说明这是一个echo request数据包。

> Q1：你的主机的IP地址是什么？目标主机的IP地址是什么？

答：

主机（源端）IP地址： `10.140.223.154`

目标主机IP地址： `23.59.9.193`

> Q2：为什么ICMP数据包没有源端口和目标端口编号？

答：

因为 ICMP 是网络层协议，而端口号是传输层协议的概念，网络层的主要职责是通过IP地址进行主机到主机的通信，传输层的主要职责是在单个主机上的不同应用程序之间进行区分和通信，端口号就是用来标识这些不同应用程序或进程的。因而 ICMP 数据包没有也不需要端口号，IP头部的协议号就可以告诉目标主机的操作系统将数据包交给哪个网络层协议处理。

> Q3：检查主机发送的一个ping请求数据包。ICMP的类型和代码编号是什么？这个ICMP数据包还有哪些其他字段？校验和、序列号和标识符字段各有多少字节？

答：

如图3所示，ICMP 的 Type 为 8，Code 为 0

其他字段：

- Checksum：0xd501
    - 2字节
- Identifier：46158
    - 2字节
- Sequence Number：0
    - 2字节
- ICMP Data

BE与LE分别表示大端序与小端序

Q4: 检查相应的ping回复数据包。ICMP的类型和代码编号是什么？这个ICMP数据包还有哪些其他字段？校验和、序列号和标识符字段各有多少字节？

答:

图4:查看reply数据包的 ICMP 协议信息

```
> Frame 82: Packet, 98 bytes on wire (784 bits), 98 bytes captured (784 bits) on interfac
> Ethernet II, Src: IETF-VRRP-VRID_01 (00:00:5e:00:01:01), Dst: 32:9b:97:b9:a4:36 (32:9b:
> Internet Protocol Version 4, Src: 23.59.9.193, Dst: 10.140.223.154
v Internet Control Message Protocol
    Type: Echo (ping) reply (0)
    Code: 0
    Checksum: 0xdd01 [correct]
    [Checksum Status: Good]
    Identifier (BE): 46158 (0xb44e)
    Identifier (LE): 20148 (0x4eb4)
    Sequence Number (BE): 0 (0x0000)
    Sequence Number (LE): 0 (0x0000)
    [Request frame: 80]
    [Response time: 116.288 ms]
  > ICMP Data: 692c0444000d162f08090a0b0c0d0e0f101112131415161718191a1b1c1d1e1f202122232
```

- Type: 0
- Code: 0

包含的其余字段与这些字段的长度与Q3相同，不再赘述。

2. ICMP和traceroute

图5:使用traceroute来确定数据包从源到目标所经过的路径（此处只展示前30跳）

```
❯ traceroute -I www.inria.fr
traceroute to www.inria.fr (128.93.162.83), 64 hops max, 48 byte packets
 1  10.140.220.1 (10.140.220.1)  11.274 ms  5.259 ms  4.570 ms
 2  10.70.3.1 (10.70.3.1)  6.380 ms  10.830 ms  34.236 ms
 3  * * *
 4  211.64.145.93 (211.64.145.93)  16.466 ms  11.381 ms  10.029 ms
 5  211.64.145.61 (211.64.145.61)  9.709 ms  7.815 ms  7.602 ms
 6  100.64.163.1 (100.64.163.1)  7.396 ms  9.233 ms  6.974 ms
 7  101.4.113.50 (101.4.113.50)  39.856 ms  15.572 ms *
 8  100.64.62.1 (100.64.62.1)  22.411 ms * *
 9  * * 101.4.116.118 (101.4.116.118)  33.199 ms
10  * * *
11  * * *
12  101.4.115.250 (101.4.115.250)  36.535 ms  26.442 ms  21.325 ms
13  * * *
14  210.25.189.237 (210.25.189.237)  24.628 ms  21.653 ms  23.580 ms
15  210.25.187.50 (210.25.187.50)  21.958 ms  25.187 ms  23.619 ms
16  202.179.241.9 (202.179.241.9)  21.597 ms  22.772 ms  18.774 ms
17  202.179.241.2 (202.179.241.2)  55.112 ms  55.270 ms  57.630 ms
18  202.179.241.102 (202.179.241.102)  175.199 ms  177.071 ms  177.822 ms
19  202.179.249.34 (202.179.249.34)  503.659 ms  413.017 ms  404.071 ms
20  lag-2-0.rt0.lon2.uk.geant.net (62.40.98.65)  412.434 ms  408.031 ms  406.481 ms
21  lag-8-0.rt0.par.fr.geant.net (62.40.98.107)  415.174 ms  404.018 ms  413.160 ms
22  renater-lb1-gw.mx1.par.fr.geant.net (62.40.124.70)  409.466 ms  404.853 ms  408.546 ms
23  hu0-4-0-0-ren-nr-orsay-rtr-091.noc.renater.fr (193.51.180.131)  411.042 ms  410.100 ms  410.896 ms
24  inria-rocquencourt-vl1631-te1-4-inria-rtr-021.noc.renater.fr (193.51.184.177)  408.004 ms  351.923 ms  349.806 ms
25  unit240-reth1-vfw-ext-dc1.inria.fr (192.93.122.19)  348.062 ms  381.434 ms  409.839 ms
26  prod-inriafr-cms.inria.fr (128.93.162.83)  349.337 ms  672.954 ms  410.826 ms
```

图6:在 Wireshark 中使用 ICMP 过滤器查看traceroute过程中发送的 ICMP 数据包

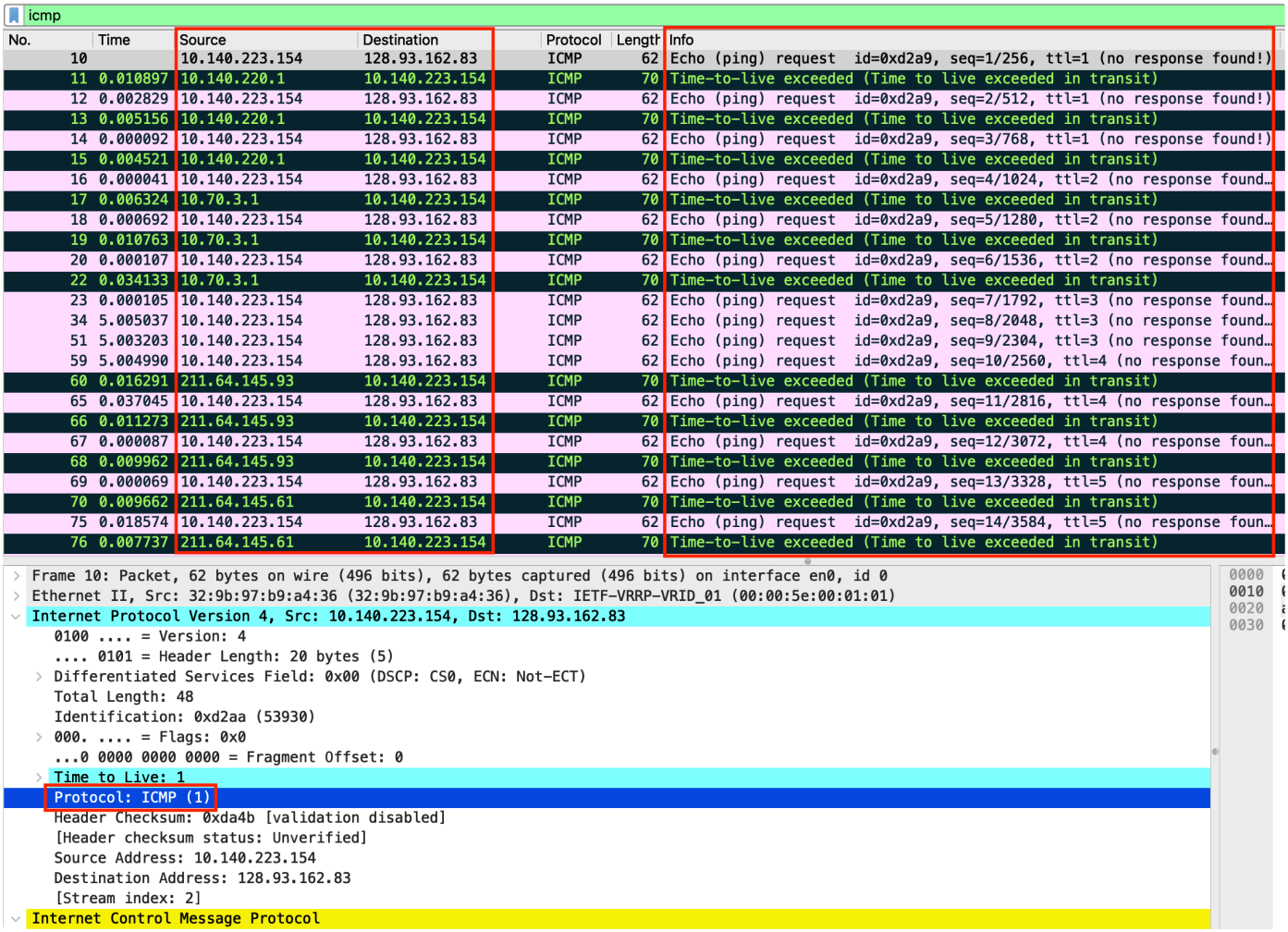


图7:查看 ICMP 字段（TTL正常超过）

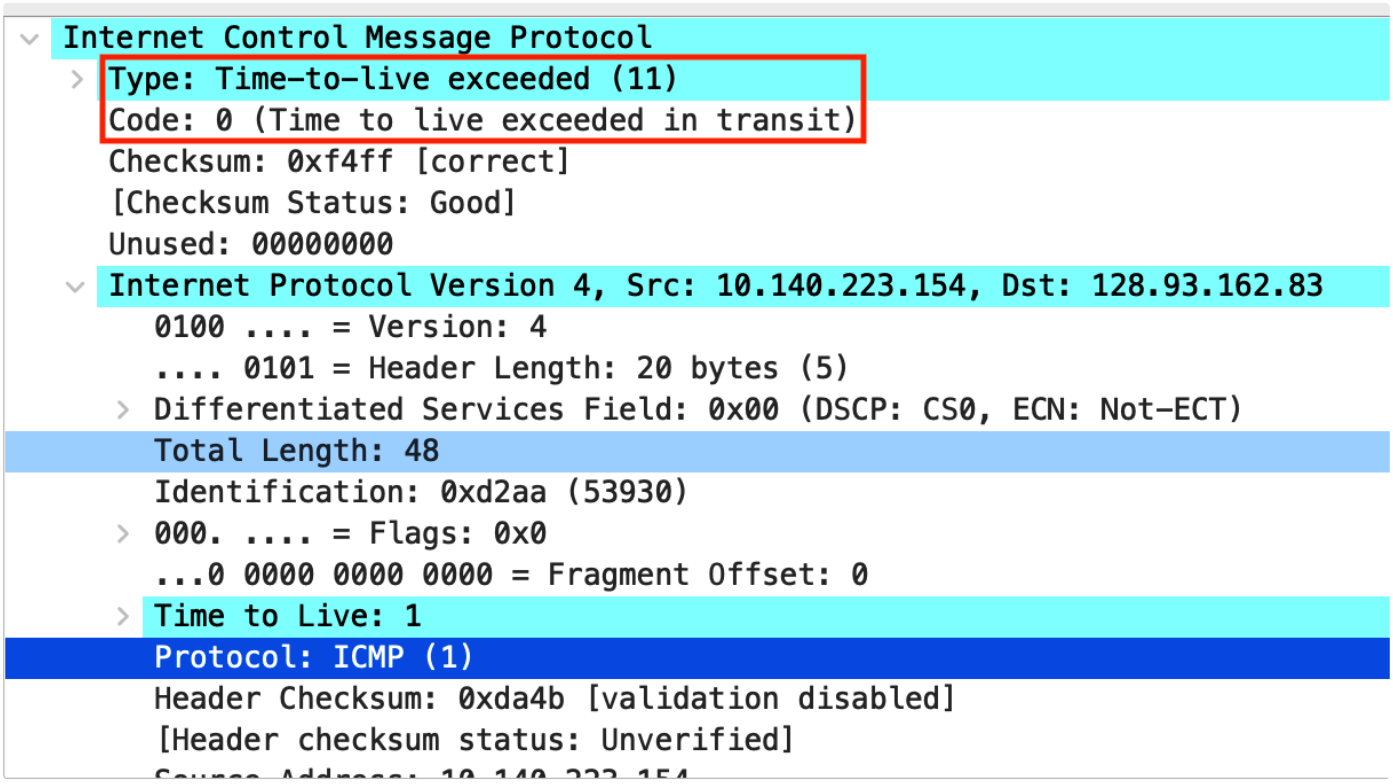


图8:查看 ICMP 字段（request）

```
> Frame 10: Packet, 62 bytes on wire (496 bits), 62 bytes captured (496 bits) on
> Ethernet II, Src: 32:9b:97:b9:a4:36 (32:9b:97:b9:a4:36), Dst: IETF-VRRP-VRID_01
> Internet Protocol Version 4, Src: 10.140.223.154, Dst: 128.93.162.83
v Internet Control Message Protocol
    Type: Echo (ping) request (8)
    Code: 0
    Checksum: 0x2555 [correct]
    [Checksum Status: Good]
    Identifier (BE): 53929 (0xd2a9)
    Identifier (LE): 43474 (0xa9d2)
    Sequence Number (BE): 1 (0x0001)
    Sequence Number (LE): 256 (0x0100)
  > [No response seen]
  > Data (20 bytes)
```

图9:查看 ICMP 字段（reply）

```
> Frame 791: Packet, 62 bytes on wire (496 bits), 62 bytes captured (496 bits) on
> Ethernet II, Src: IETF-VRRP-VRID_01 (00:00:5e:00:01:01), Dst: 32:9b:97:b9:a4:36
> Internet Protocol Version 4, Src: 128.93.162.83, Dst: 10.140.223.154
v Internet Control Message Protocol
    Type: Echo (ping) reply (0)
    Code: 0
    Checksum: 0x2d0a [correct]
    [Checksum Status: Good]
    Identifier (BE): 53929 (0xd2a9)
    Identifier (LE): 43474 (0xa9d2)
    Sequence Number (BE): 76 (0x004c)
    Sequence Number (LE): 19456 (0x4c00)
    [Request frame: 782]
    [Response time: 349.179 ms]
  > Data (20 bytes)
```

Q5: 你的主机IP地址是什么？目标目的主机的IP地址是什么？

答:

可以观察第一个 request 数据包（图6）：

主机（源）IP地址: `10.140.223.154`

目标主机IP地址: `128.93.162.83`

Q6: 如果ICMP发送UDP数据包（如Unix/Linux那样），探测数据包的IP协议号仍然是01吗？如果不是，会是什么？

答:

图10:

```
Protocol: UDP (17)
```

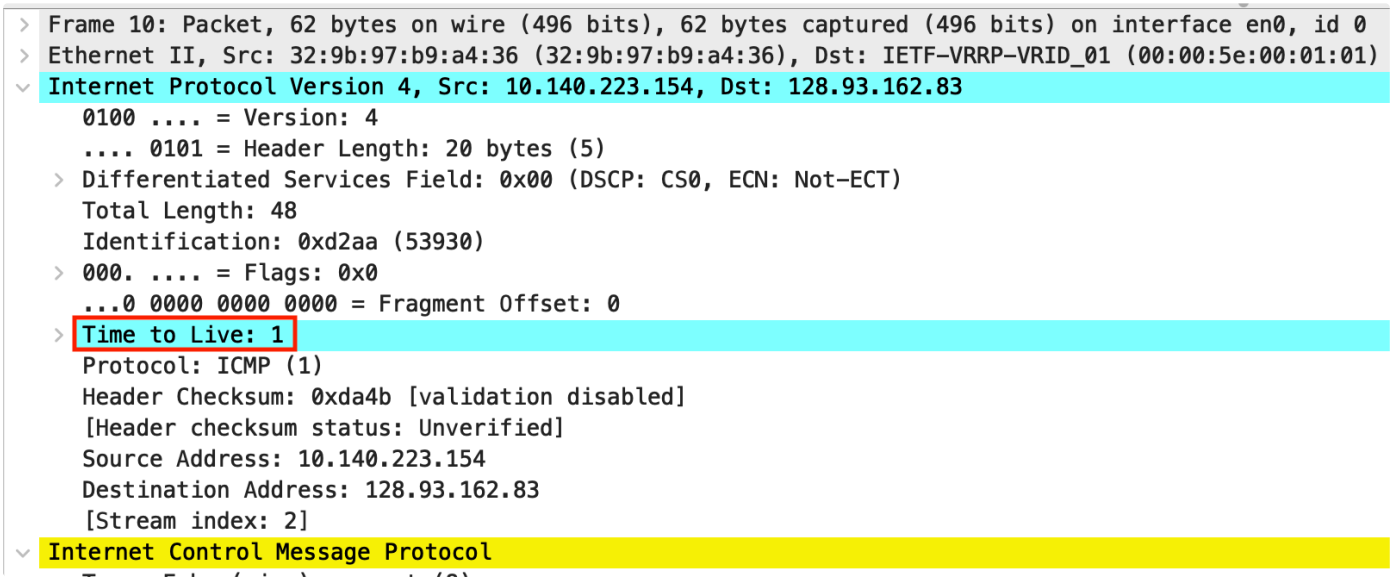
不是。 如果 traceroute 发送的是UDP探测数据包，那么IP头部的协议号将不再是 01，而会变成 17。

因为IP协议号用于标识其数据部分是哪种上层协议的数据包，01 代表 ICMP 报文，17 代表 UDP 报文。

> Q7: 检查你的截图中的ICMP回显数据包。这与本实验前半部分的ICMP ping查询数据包有什么不同？如果有，如何不同？

答：

图11:tracetoute ICMP echo request 数据包（第一个request）



对比可以发现二者主要的不同在于 TTL 的设置，普通 ping 的 TTL 一般设置为64，而traceroute的echo request数据包的TTL一般是递增的，且故意让中间路由器返回超时错误，从而发现路径。

> Q8: 检查你的截图中的ICMP错误数据包。它比ICMP回显数据包包含更多字段。这些字段包含什么内容？

答：

图12:错误数据包中的 ICMP 字段

```
Internet Control Message Protocol
  Type: Time-to-live exceeded (11)
  Code: 0 (Time to live exceeded in transit)
  Checksum: 0xf4ff [correct]
  [Checksum Status: Good]
  Unused: 00000000
  Internet Protocol Version 4, Src: 10.140.223.154, Dst: 128.93.162.83
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
    Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 48
    Identification: 0xd2aa (53930)
    000. .... = Flags: 0x0
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 1
    Protocol: ICMP (1)
    Header Checksum: 0xda4b [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 10.140.223.154
    Destination Address: 128.93.162.83
    [Stream index: 2]
  Internet Control Message Protocol
    Type: Echo (ping) request (8)
    Code: 0
    Checksum: 0x2555 [unverified] [in ICMP error packet]
    [Checksum Status: Unverified]
    Identifier (BE): 53929 (0xd2a9)
    Identifier (LE): 43474 (0xa9d2)
    Sequence Number (BE): 1 (0x0001)
    Sequence Number (LE): 256 (0x0100)
```

如图所示，多了**完整的 IPv4 头**（Version、Header Length、TTL=1、Src/Dst IP 等）、**原始 ICMP Echo Request 头的前 8 字节**（Type=8, Code=0, Checksum, Identifier, Sequence Number）

> Q9：检查源主机收到的最后三个ICMP数据包。这些数据包与ICMP错误数据包有什么不同？为什么不同？

答：

最后的三个 ICMP 数据包的 ICMP 字段如图9所示，均为reply。

不同：

- ICMP Echo Reply：Type 0，Code 0，是对 Echo 请求的正常响应，没有嵌入原始数据包IP头。
- ICMP Time Exceeded：Type 11，Code 0，是错误消息，数据部分包含原始请求包的 IP 头+8 字节。

不同的原因：

后三个 request 的 TTL 足以支持它们到达目标主机而不超时，目标主机收到 TTL 未超时的 Echo 请求后，会正常处理并返回 Echo Reply，而不是错误消息。

> Q10: 在tracert测量中，是否有链路的延迟显著长于其他链路？参考图4中的截图，是否有链路的延迟显著长于其他链路？根据路由器名称，你能猜测这两个位于该链路两端的路由器的位置吗？

答：

根据图5，延迟变化：

第17跳到第18跳的延迟从约 **55 ms** 跃升至 **175-177 ms**，紧接着第18跳到第19跳的延迟进一步飙升至 **400-500 ms**，从第19跳开始，延迟稳定在较高的水平，直到目的地。

根据IP地址查询可以得知：

- 第18跳路由器(`202.179.241.102`)：属于中国教育和科研计算机网。根据其在中国国内网络节点之后，在国际链路节点之前的位置来看，它很可能位于中国境内的国际出口网关或边界路由器。
- 第19跳路由器(`202.179.249.34`)：这个IP地址同样属于中国教育和科研计算机网。结合第20跳是位于英国伦敦的GEANT网络节点 (`lag-2-0.rt0.lon2.uk.geant.net`)，可以推断出第19跳是CERNET连接至欧洲GEANT网络的国际出口点。

因此，第18跳至第19跳实际上代表了数据包从中国境内网络穿越到欧洲国际学术网络，巨大的延迟也是因巨大的地理跨度产生的。

第18跳的路由器可能在北京或上海等主要国际出口节点，第19跳可能是对端在欧洲的接入点。

四、总结建议

实验中遇到的问题与解决方法：

> 实验中，开一开始使用 `traceroute` 追踪到 `www.inria.fr` 的路由器时，出现了大量超时未知节点

```
398.628 ms  409.155 ms
24  inria-rocquencourt-vl1631-te1-4-inria-rtr-021.noc.renater.fr (193.51.184.177
)  410.209 ms  413.327 ms  410.915 ms
25  * * *
26  * * *
27  * * *
28  * * *
29  * * *
30  * * *
31  * * *
32  * * *
33  * * *
34  * * *
35  * * *
36  * * *
37  * * *
38  * * *
39  * * *
40  * * *
41  * * *
42  * * *
43  * * *
44  * * *
45  * * *
46  * * *
```

经查询后发现，因为我使用的是MacOS操作系统，`traceroute` 命令默认使用UDP数据包进行探测，但是抵达目标网络需要数据包穿越多个网络，这些网络很可能出于安全考虑，配置了严格的防火墙规则，这些防火墙规则可能将UDP数据包视为非必要或潜在的恶意扫描流量，进而将UDP包丢弃。即使UDP包穿过了所有中间节点，目标服务器 `prod-inriafr-cms.inria.fr` 本身也可能被设置为不响应UDP端口探测。

但是将命令修改为 `traceroute -I www.inria.fr` ，指示traceroute使用ICMP Echo Request，ICMP Echo通常被视为更“无害”的网络诊断工具，可以成功探测。

心得收获：

通过本次实验，我对ICMP协议的工作原理和应用场景有了更深入的理解。在实验过程中，我不仅掌握了Ping和Traceroute这两个常用网络诊断工具的具体实现机制，还亲身体验了不同网络环境下ICMP协议的实际表现，了解了ICMP 数据包的构成。特别是在使用Traceroute命令时遇到的大量超时问题，让我深刻认识到网络防火墙策略对诊断工具的影响，以及灵活调整参数的重要性。通过对比分析ICMP Echo Request和UDP探测包的差异，我更加明确了网络协议选择在实际应用中的关键作用。