计算机学院_计算机网络_课程实验报告

实验题目： IP		学号：202200130048
日期：4.23	班级： 6	姓名： 陈静雯
Email：1205037094@qq.com		

实验方法介绍：

从 traceroute 的执行中捕获数据包，分析查看具体数据报

实验过程描述：

1.从 traceroute 的执行中捕获数据包

2.基本 IPv4

3.碎片

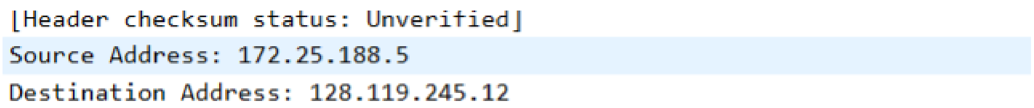
4. IPv6

In the first part, we' ll analyze packets in a trace of IPv4 datagrams sent and received by the traceroute program (the traceroute program itself is explored in more detail in the Wireshark ICMP lab). We' ll study IP fragmentation in Part 2 of this lab, and take a quick look at IPv6 in Part 3 of this lab.

结论分析：

1. What is the IP address of your computer?

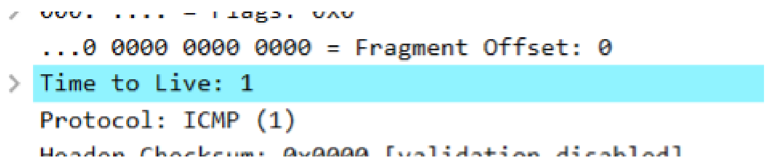
172.25.188.5



```
[Header checksum status: Unverified]
Source Address: 172.25.188.5
Destination Address: 128.119.245.12
```

2. What is the value in the time-to-live (TTL) field in this IPv4 datagram' s header?

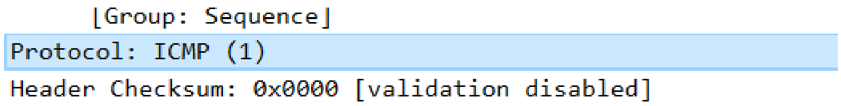
Time to live : 1



```
000. .... = Flags: 0x0
...0 0000 0000 0000 = Fragment Offset: 0
Time to Live: 1
Protocol: ICMP (1)
Header Checksum: 0x0000 [validation disabled]
```

3. What is the value in the upper layer protocol field in this IPv4 datagram' s header?

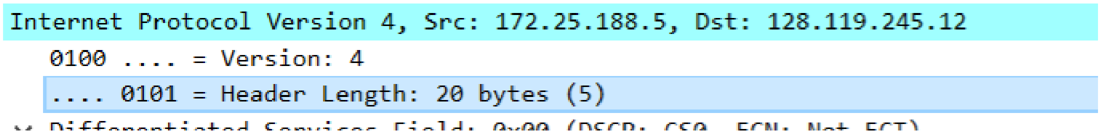
ICMP，protocal 表示 IPv4 数据报载荷中的高层协议



```
[Group: Sequence]
Protocol: ICMP (1)
Header Checksum: 0x0000 [validation disabled]
```

4. How many bytes are in the IP header?

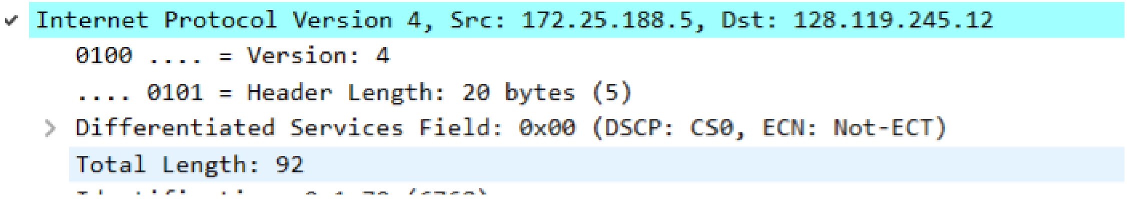
20 bytes



```
Internet Protocol Version 4, Src: 172.25.188.5, Dst: 128.119.245.12
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  ∨ Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
```

5. How many bytes are in the payload of the IP datagram?

Payload=total-header=72 bytes



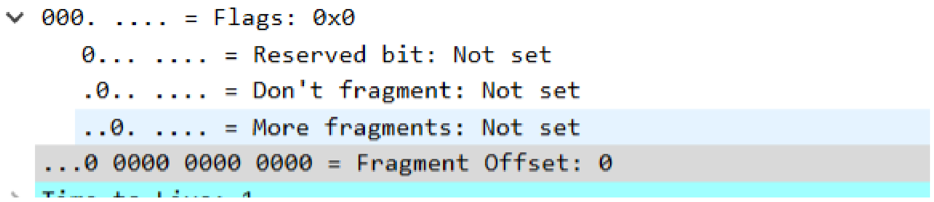
```
✓ Internet Protocol Version 4, Src: 172.25.188.5, Dst: 128.119.245.12
      0100 .... = Version: 4
      .... 0101 = Header Length: 20 bytes (5)
    > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
      Total Length: 92
```

6. Has this IP datagram been fragmented?

没有被分片

MF（More Fragments）位： MF 位用于指示是否还有更多分片。如果 MF 位为 1，说明当前数据包不是最后一个分片；若 MF 位为 0，则表示这是最后一个分片或数据报未被分片。

分片偏移量（Fragment Offset）： 对于分片的数据报，此字段表示当前分片相对于原始数据报起始处的偏移量，以 8 字节为单位。非零的分片偏移量是数据报已被分片的一个明显标志。



```
∨ 000. .... = Flags: 0x0
      0... .... = Reserved bit: Not set
      .0.. .... = Don't fragment: Not set
      ..0. .... = More fragments: Not set
  ...0 0000 0000 0000 = Fragment Offset: 0
  Time to Live: 1
```

7. Which fields in the IP datagram always change from one datagram to the next within this series of UDP segments sent by your computer destined to 128.119.245.12, via traceroute? Why?

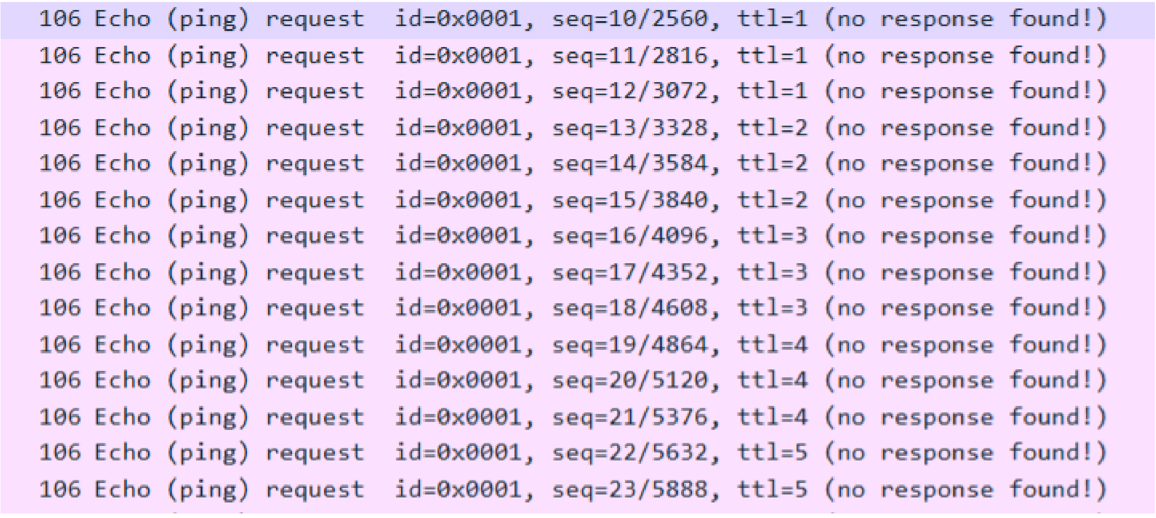
Identification、TTL、Checksum、Sequence Number

Identification：这个字段用于唯一标识一个 IP 数据报，当数据报需要分片时，所有分片都会共享同一标识符。在 traceroute 过程中，每次发送新的 UDP 段时，为了确保不同数据报之间的唯一性，系统通常会生成一个新的标识符。

TTL：TTL 字段决定了数据报在网络中能够经过的最大跳数（路由器数量）。Traceroute 正是利用了这一特性，每次发送 UDP 段时递增 TTL 值，从而迫使数据报在达到目标之前依次经过不同的中间路由器。每经过一个路由器，TTL 值就会减一，当 TTL 降至 0 时，路由器会丢弃该数据报并返回一个 ICMP 超时报文。通过这种方式，traceroute 能够追踪路径上的每一个跃点。

Checksum：校验和是对 IP 头部（包括可变长的选项部分）以及数据报载荷进行计算得出的。由于标识符和 TTL 字段的变化会影响头部内容，因此对应的校验和也会随之更新，以保证头部数据的完整性。

Sequence Number：序列号字段与标识符一起，用于确保 ICMP 请求和应答之间的关联。对于每个 ICMP 回显请求，发送端会选择一个递增的序列号，接收端在回应时会复制这个序列号。这样，即使在同一时刻发送了多个 ICMP 请求，发送端也能根据标识符和序列号的组合正确地将收到的 ICMP 回显应答与对应的请求相匹配。



```
106 Echo (ping) request  id=0x0001, seq=10/2560, ttl=1 (no response found!)
106 Echo (ping) request  id=0x0001, seq=11/2816, ttl=1 (no response found!)
106 Echo (ping) request  id=0x0001, seq=12/3072, ttl=1 (no response found!)
106 Echo (ping) request  id=0x0001, seq=13/3328, ttl=2 (no response found!)
106 Echo (ping) request  id=0x0001, seq=14/3584, ttl=2 (no response found!)
106 Echo (ping) request  id=0x0001, seq=15/3840, ttl=2 (no response found!)
106 Echo (ping) request  id=0x0001, seq=16/4096, ttl=3 (no response found!)
106 Echo (ping) request  id=0x0001, seq=17/4352, ttl=3 (no response found!)
106 Echo (ping) request  id=0x0001, seq=18/4608, ttl=3 (no response found!)
106 Echo (ping) request  id=0x0001, seq=19/4864, ttl=4 (no response found!)
106 Echo (ping) request  id=0x0001, seq=20/5120, ttl=4 (no response found!)
106 Echo (ping) request  id=0x0001, seq=21/5376, ttl=4 (no response found!)
106 Echo (ping) request  id=0x0001, seq=22/5632, ttl=5 (no response found!)
106 Echo (ping) request  id=0x0001, seq=23/5888, ttl=5 (no response found!)
```

8.Which fields in this sequence of IP datagrams (containing UDP segments) stay constant? Why?

Version、Header Length、DSCP、Total Length、Protocol

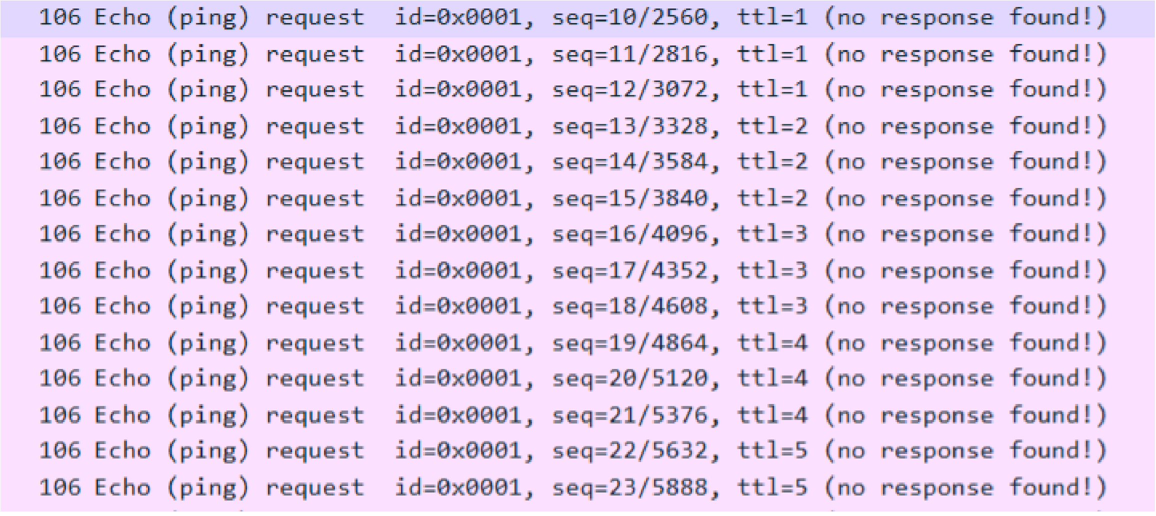
Version：在整个 IPv4 数据报的生命周期内，其使用的协议版本始终保持一致。

Header Length：对于同一类型（如无选项字段）的 IPv4 数据报，其头部长度是固定的。

DSCP：除非在 traceroute 过程中明确更改了数据报的服务质量要求，否则这些设置通常保持不变。

Total Length：虽然实际的载荷（如 UDP 段）内容可能不同，但由于 traceroute 通常使用固定大小的 UDP 段，因此总长度在一系列数据报中可能保持一致。

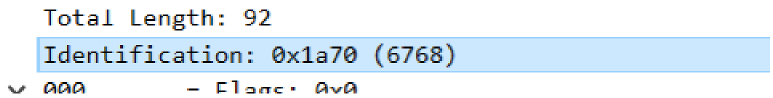
Protocol（协议）：在 traceroute 过程中会保持不变。



```
106 Echo (ping) request  id=0x0001, seq=10/2560, ttl=1 (no response found!)
106 Echo (ping) request  id=0x0001, seq=11/2816, ttl=1 (no response found!)
106 Echo (ping) request  id=0x0001, seq=12/3072, ttl=1 (no response found!)
106 Echo (ping) request  id=0x0001, seq=13/3328, ttl=2 (no response found!)
106 Echo (ping) request  id=0x0001, seq=14/3584, ttl=2 (no response found!)
106 Echo (ping) request  id=0x0001, seq=15/3840, ttl=2 (no response found!)
106 Echo (ping) request  id=0x0001, seq=16/4096, ttl=3 (no response found!)
106 Echo (ping) request  id=0x0001, seq=17/4352, ttl=3 (no response found!)
106 Echo (ping) request  id=0x0001, seq=18/4608, ttl=3 (no response found!)
106 Echo (ping) request  id=0x0001, seq=19/4864, ttl=4 (no response found!)
106 Echo (ping) request  id=0x0001, seq=20/5120, ttl=4 (no response found!)
106 Echo (ping) request  id=0x0001, seq=21/5376, ttl=4 (no response found!)
106 Echo (ping) request  id=0x0001, seq=22/5632, ttl=5 (no response found!)
106 Echo (ping) request  id=0x0001, seq=23/5888, ttl=5 (no response found!)
```

9. Describe the pattern you see in the values in the Identification field of the IP datagrams being sent by your computer.

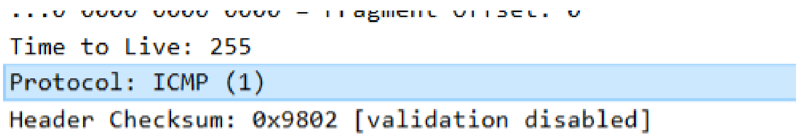
16 进制（十进制）



```
    Total Length: 92
    Identification: 0x1a70 (6768)
  ⌄ 000.        = Flags: 0x0
```

10.What is the upper layer protocol specified in the IP datagrams returned from the routers?

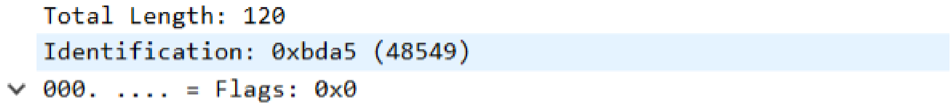
Protocol：ICMP



```
...0 0000 0000 0000 = Fragment Offset: 0
Time to Live: 255
Protocol: ICMP (1)
Header Checksum: 0x9802 [validation disabled]
```

11. Are the values in the Identification fields (across the sequence of all of ICMP packets from all of the routers) similar in behavior to your answer to question 9 above?

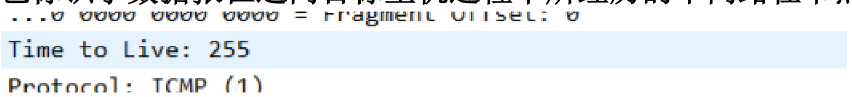
Yes。



```
  Total Length: 120
  Identification: 0xbda5 (48549)
v 000. .... = Flags: 0x0
```

12. Are the values of the TTL fields similar, across all of ICMP packets from all of the routers?

TTL 不断减小，这些不同的 TTL 值反映了数据报在到达每个路由器时剩余的生命值，同时也标识了数据报在通向目标主机过程中所经历的不同路径节点。

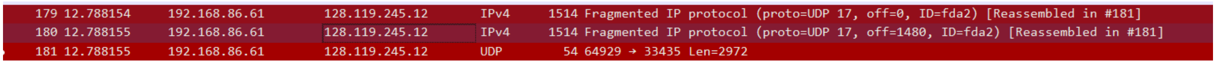


```
...0 0000 0000 0000 = Fragment Offset: 0
Time to Live: 255
Protocol: ICMP (1)
```

后面用的是题设给的数据包信息（Windows 的 trace 最大长度到不了 3000）

13. Has that segment been fragmented across more than one IP datagram?

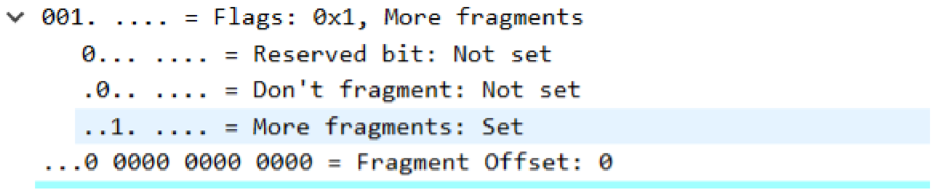
Yes



```
179 12.788154   192.168.86.61   128.119.245.12   IPv4   1514 Fragmented IP protocol (proto=UDP 17, off=0, ID=fda2) [Reassembled in #181]
180 12.788155   192.168.86.61   128.119.245.12   IPv4   1514 Fragmented IP protocol (proto=UDP 17, off=1480, ID=fda2) [Reassembled in #181]
181 12.788155   192.168.86.61   128.119.245.12   UDP      54 64929 → 33435 Len=2972
```

14. What information in the IP header indicates that this datagram been fragmented?

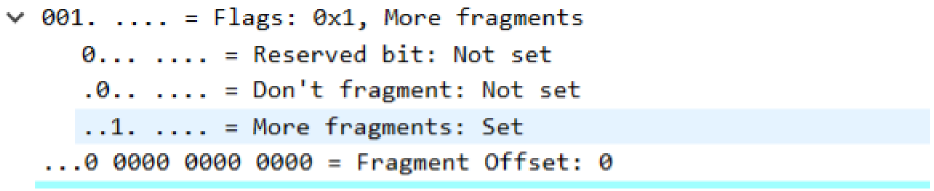
More fragments=1,表示有更多的分片



```
v 001. .... = Flags: 0x1, More fragments
     0... .... = Reserved bit: Not set
     .0.. .... = Don't fragment: Not set
     ..1. .... = More fragments: Set
  ...0 0000 0000 0000 = Fragment Offset: 0
```

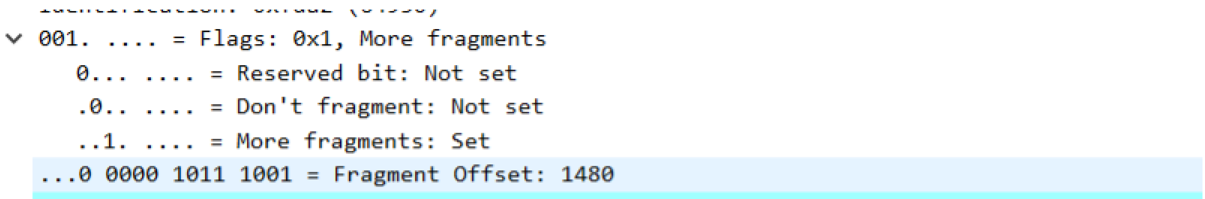
15. What information in the IP header for this packet indicates whether this is the first fragment versus a latter fragment?

Fragment offset=0，表示这是第一个分片（偏移量为 0）



```
v 001. .... = Flags: 0x1, More fragments
     0... .... = Reserved bit: Not set
     .0.. .... = Don't fragment: Not set
     ..1. .... = More fragments: Set
  ...0 0000 0000 0000 = Fragment Offset: 0
```

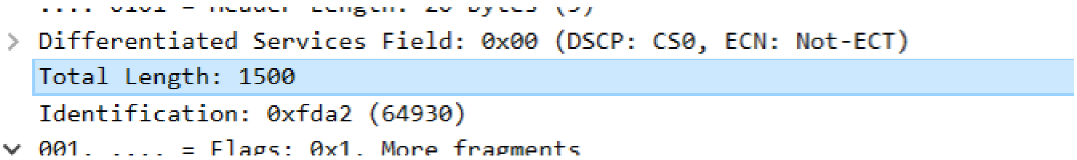
180 数据包偏移量为 1480，如下图



```
    Identification: 0xfda2 (64930)
  ∨ 001. .... = Flags: 0x1, More fragments
      0... .... = Reserved bit: Not set
      .0.. .... = Don't fragment: Not set
      ..1. .... = More fragments: Set
    ...0 0000 1011 1001 = Fragment Offset: 1480
```

16. How many bytes are there in is this IP datagram (header plus payload)?

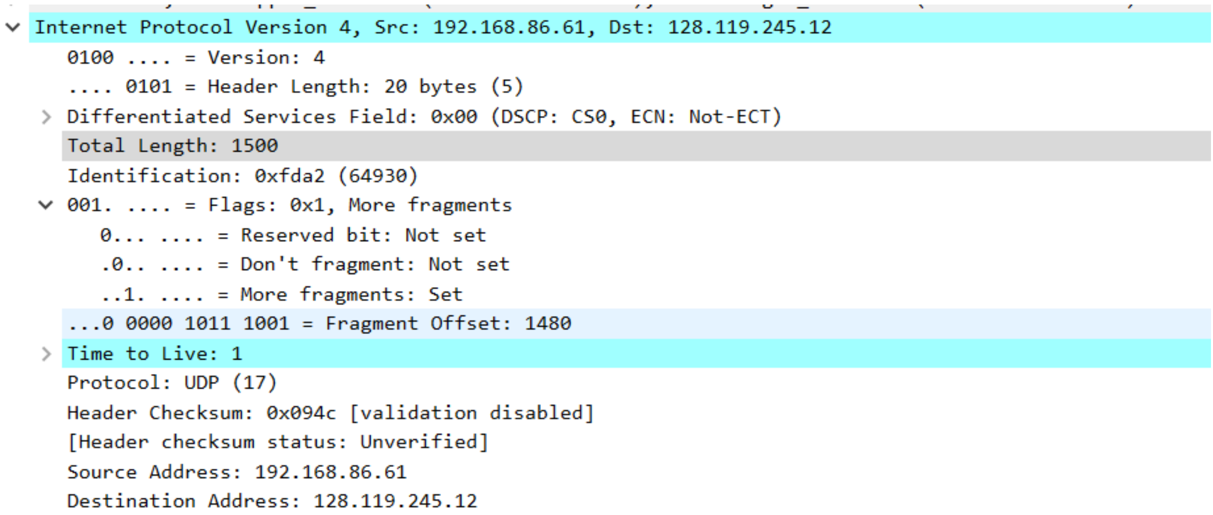
1500



```
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 1500
    Identification: 0xfda2 (64930)
  ∨ 001. .... = Flags: 0x1, More fragments
```

17. What information in the IP header indicates that this is not the first datagram fragment?

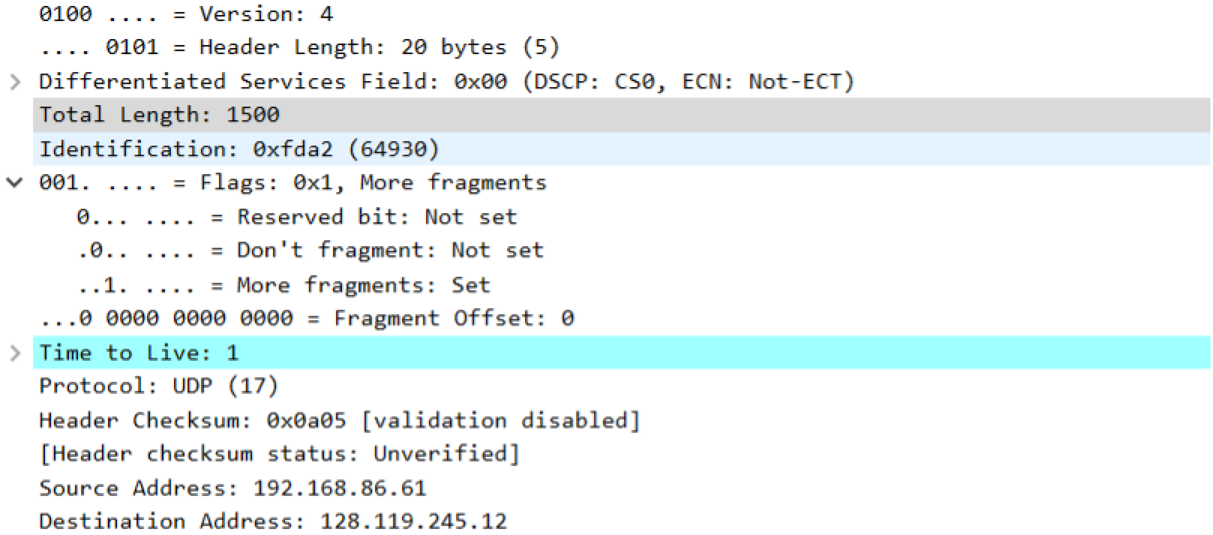
Fragment Offset 为 1480，identification 与前一个一样



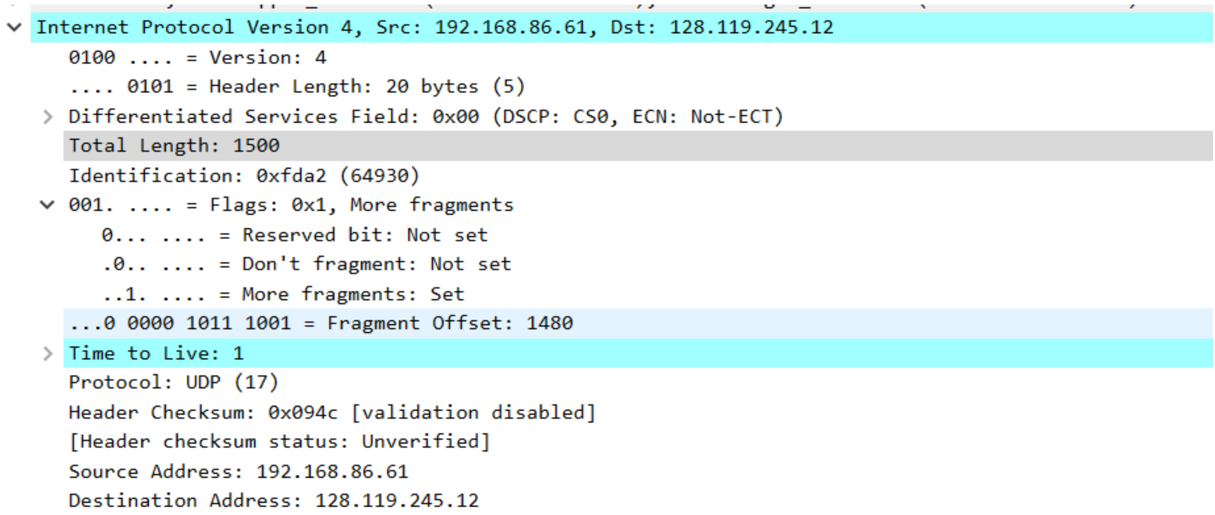
```
∨ Internet Protocol Version 4, Src: 192.168.86.61, Dst: 128.119.245.12
      0100 .... = Version: 4
      .... 0101 = Header Length: 20 bytes (5)
    > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
      Total Length: 1500
      Identification: 0xfda2 (64930)
    ∨ 001. .... = Flags: 0x1, More fragments
        0... .... = Reserved bit: Not set
        .0.. .... = Don't fragment: Not set
        ..1. .... = More fragments: Set
      ...0 0000 1011 1001 = Fragment Offset: 1480
    > Time to Live: 1
      Protocol: UDP (17)
      Header Checksum: 0x094c [validation disabled]
      [Header checksum status: Unverified]
      Source Address: 192.168.86.61
      Destination Address: 128.119.245.12
```

18. What fields change in the IP header between the first and second fragment?

Fragment offset



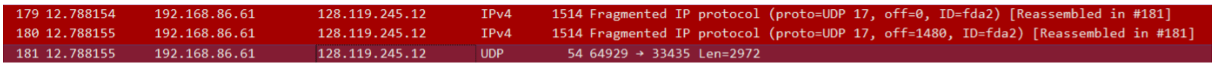
```
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 1500
    Identification: 0xfda2 (64930)
  ∨ 001. .... = Flags: 0x1, More fragments
      0... .... = Reserved bit: Not set
      .0.. .... = Don't fragment: Not set
      ..1. .... = More fragments: Set
    ...0 0000 0000 0000 = Fragment Offset: 0
  > Time to Live: 1
    Protocol: UDP (17)
    Header Checksum: 0x0a05 [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 192.168.86.61
    Destination Address: 128.119.245.12
```



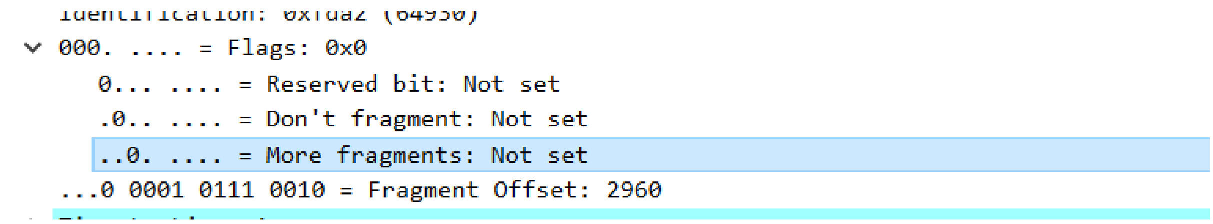
```
Internet Protocol Version 4, Src: 192.168.86.61, Dst: 128.119.245.12
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 1500
    Identification: 0xfda2 (64930)
  v 001. .... = Flags: 0x1, More fragments
        0... .... = Reserved bit: Not set
        .0.. .... = Don't fragment: Not set
        ..1. .... = More fragments: Set
    ...0 0000 1011 1001 = Fragment Offset: 1480
  > Time to Live: 1
    Protocol: UDP (17)
    Header Checksum: 0x094c [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 192.168.86.61
    Destination Address: 128.119.245.12
```

19. What information in the IP header indicates that this is the last fragment of that segment?

偏移量为 2960，more fragments 为 0，说明后面没有更多分片了，所以这个是最后一个



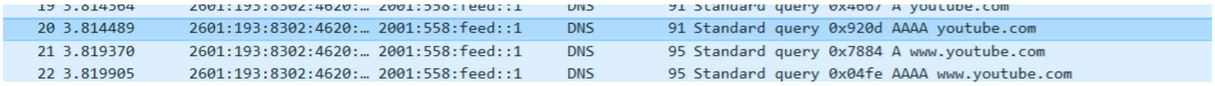
```
179 12.788154   192.168.86.61   128.119.245.12   IPv4   1514 Fragmented IP protocol (proto=UDP 17, off=0, ID=fda2) [Reassembled in #181]
180 12.788155   192.168.86.61   128.119.245.12   IPv4   1514 Fragmented IP protocol (proto=UDP 17, off=1480, ID=fda2) [Reassembled in #181]
181 12.788155   192.168.86.61   128.119.245.12   UDP      54 64929 → 33435 Len=2972
```



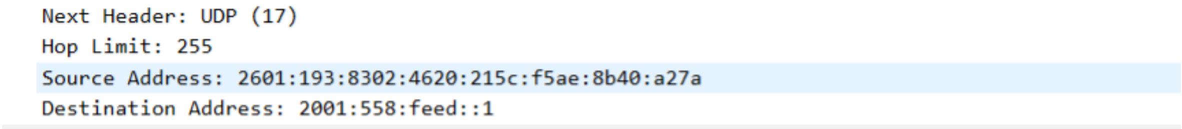
```
   Identification: 0xfda2 (64930)
 v 000. .... = Flags: 0x0
       0... .... = Reserved bit: Not set
       .0.. .... = Don't fragment: Not set
       ..0. .... = More fragments: Not set
   ...0 0001 0111 0010 = Fragment Offset: 2960
```

20. What is the IPv6 address of the computer making the DNS AAAA request?

Src add：2601:193:8302:4620:215c:f5ae:8b40:a27a



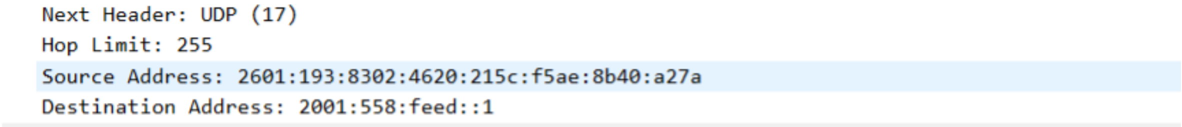
```
 19 3.814364   2601:193:8302:4620:… 2001:558:feed::1   DNS   91 Standard query 0x4667 A youtube.com
 20 3.814489   2601:193:8302:4620:… 2001:558:feed::1   DNS   91 Standard query 0x920d AAAA youtube.com
 21 3.819370   2601:193:8302:4620:… 2001:558:feed::1   DNS   95 Standard query 0x7884 A www.youtube.com
 22 3.819905   2601:193:8302:4620:… 2001:558:feed::1   DNS   95 Standard query 0x04fe AAAA www.youtube.com
```



```
Next Header: UDP (17)
Hop Limit: 255
Source Address: 2601:193:8302:4620:215c:f5ae:8b40:a27a
Destination Address: 2001:558:feed::1
```

21. What is the IPv6 destination address for this datagram?

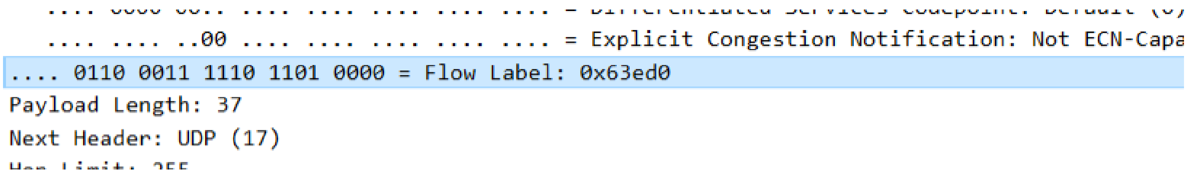
Dst：2001:558:feed::1



```
Next Header: UDP (17)
Hop Limit: 255
Source Address: 2601:193:8302:4620:215c:f5ae:8b40:a27a
Destination Address: 2001:558:feed::1
```

22. What is the value of the flow label for this datagram?

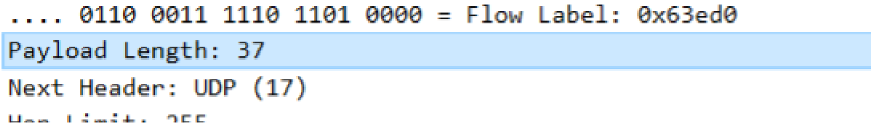
0x63ed0



```
    .... 0000 00.. .... .... .... .... .... = Differentiated Services Codepoint: Default (0)
    .... .... ..00 .... .... .... .... .... = Explicit Congestion Notification: Not ECN-Capa
.... 0110 0011 1110 1101 0000 = Flow Label: 0x63ed0
Payload Length: 37
Next Header: UDP (17)
Hop Limit: 255
```

23. How much payload data is carried in this datagram?

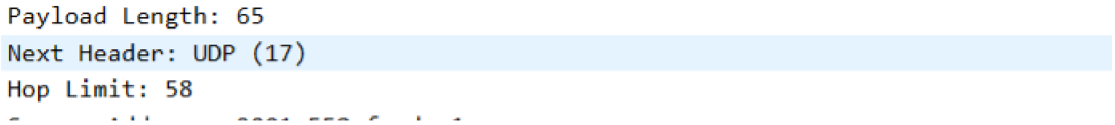
37 bytes



```
.... 0110 0011 1110 1101 0000 = Flow Label: 0x63ed0
Payload Length: 37
Next Header: UDP (17)
Hop Limit: 255
```

24. What is the upper layer protocol to which this datagram’ s payload will be delivered at the destination?

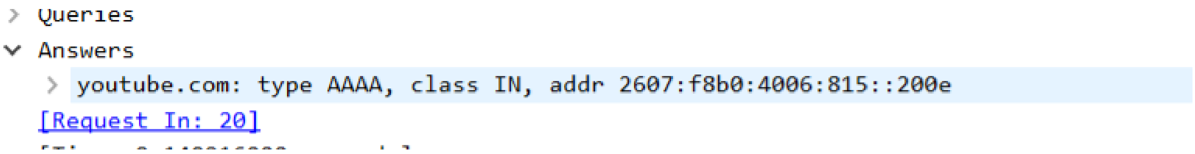
上层协议：UDP



```
Payload Length: 65
Next Header: UDP (17)
Hop Limit: 58
```

25.How many IPv6 addresses are returned in the response to this AAAA request?

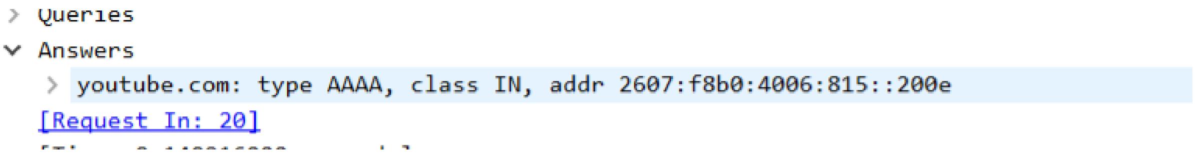
1 address



```
> Queries
v Answers
   > youtube.com: type AAAA, class IN, addr 2607:f8b0:4006:815::200e
  [Request In: 20]
```

26. What is the first of the IPv6 addresses returned by the DNS for youtube.com

youtube.com: type AAAA, class IN, addr 2607:f8b0:4006:815::200e



```
> Queries
v Answers
   > youtube.com: type AAAA, class IN, addr 2607:f8b0:4006:815::200e
  [Request In: 20]
```

结论：

1. 标识符（Identification）：对于分片的 IP 数据报，所有分片共享相同的标识符。如果看到连续几个 IP 数据包具有相同的标识符，这可能意味着它们属于同一个原始数据报的不同分片。

2.标志（Flags）字段：在 IP 头部中，有两个标志位与分片相关：

DF（Don't Fragment）位： 如果 DF 位设置为 1，表示数据报不允许被分片。如果该位为 0，则分片是可能的。在未分片的数据包中，此位通常不重要，但在排查分片问题时，确认 DF 位的状态有助于理解网络设备是否遵循了发送端的分片要求。

MF（More Fragments）位： MF 位用于指示是否还有更多分片。如果 MF 位为 1，说明当前数据包不是最后一个分片；若 MF 位为 0，则表示这是最后一个分片或数据报未被分片。

3.分片偏移量（Fragment Offset）： 对于分片的数据报，此字段表示当前分片相对于原始数据报起始处的偏移量，以 8 字节为单位。非零的分片偏移量是数据报已被分片的一个明显标志。

4. TTL 字段的特点如下：

递增发送： 发送端在执行 traceroute 时，会按照预设策略（通常为线性递增）逐次增加 IP 数据报的 TTL 值。例如，首次发送的 UDP 段可能带有 TTL 值为 1，随后的 UDP 段 TTL 值依次为 2、3、4 等，直到达到某个预设阈值或到达目标。

逐跳递减： 每当 IP 数据报经过一个路由器（即完成一次“跳”），路由器会在转发前将 TTL 值减 1。当 TTL 值减至 0 时，路由器不再转发该数据报，而是向源地址返回一个 ICMP 超时报文（类型为 11，代码为 0）。

唯一标识路径： 由于traceroute过程中发送的每个UDP段具有不同的TTL值，且每个TTL值仅允许数据报经过特定数量的路由器，因此收到的ICMP超时报文可以唯一对应到途经的某个路由器。换句话说，每个路由器返回的ICMP超时报文中携带的TTL值与触发该响应的原始数据报的TTL值相同，并且这个TTL值在所有返回的ICMP报文中是唯一的。