计算机网络第六次实验实验报告

舒文炫

2021 年 11 月 13 日

目录

Chapter 1

实验内容

本次实验内容是 IP 协议，主要关注 IP 数据报，这次将会使用 traceroute 程序发送和接收的一系列数据报，供观察，详细的学习 IP 数据报文结构，以及分片操作。

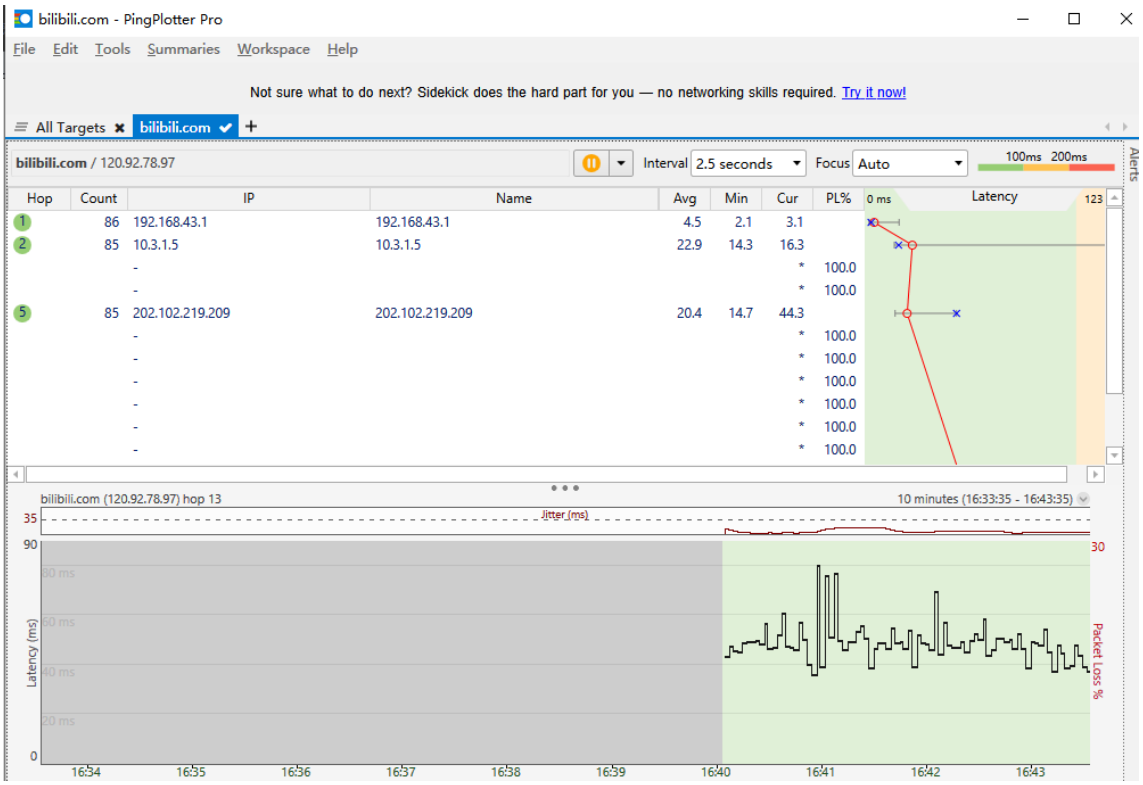
Chapter 2

实验过程

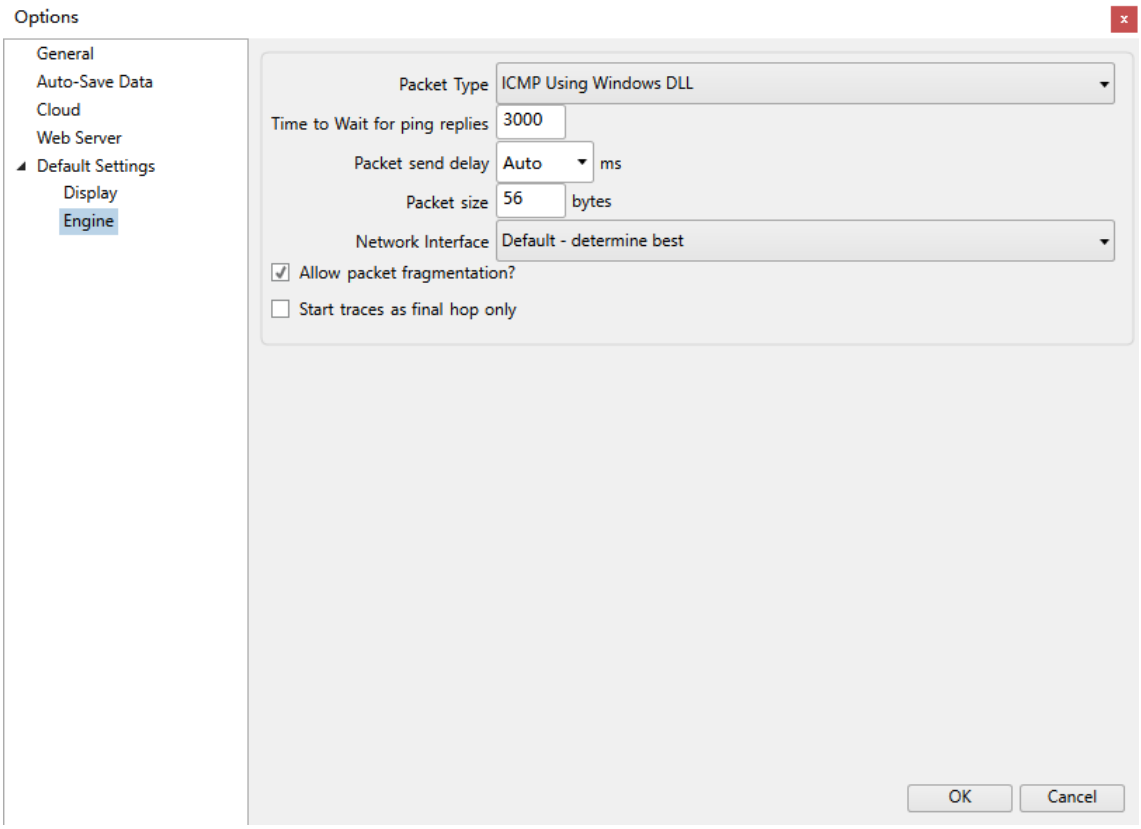
2.1 1. Capturing packets from an execution of traceroute

2.1.1 过程截图

首先安装 pingplotter 软件，下面是该程序的截图，这是第五版的软件和实验指导书上的软件有一些区别，下面是我往 bilibili.com 发送 ping 消息，软件运行的截图



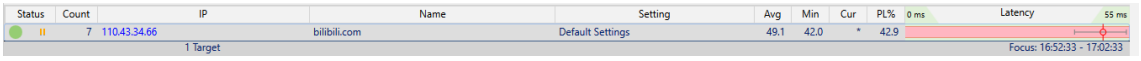
这个版本调整发送字节大小的选项稍有不同，在 edit-options-engine 里面，下面是截图



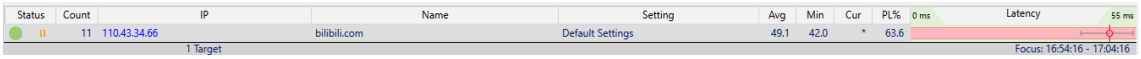
然后这里并没有设置 time to trace 的地方，所以后面我只能手动暂停

首先打开 wireshark 进行捕获，然后使用 pingplotter 对 bilibili.com 进行 ping 操作，这里我等到 count 为 3 时暂停，并且包的大小设置的是 56KB。不过这个地方我忘记截图下来了，看第一张截图凑合凑合 qwq

然后将包的大小设置为 2000KB，同样等一段时间，因为这里没用设置 time to trace 的地方，手动停止可能不能保证每隔 3 个停一下，不过这不重要，下面是 count 为 7 时我暂停了



然后包大小设置为 3500KB，和上面相同，在 count 为 11 时我停止了

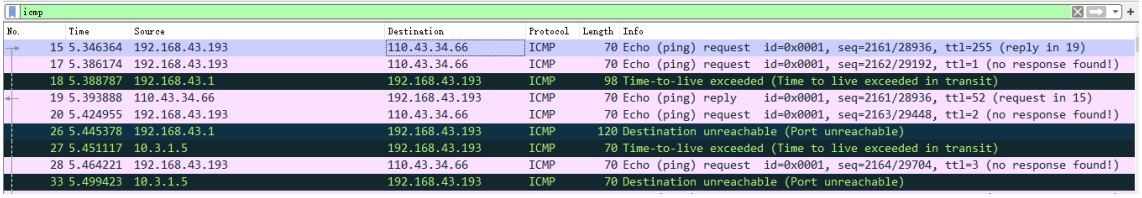


上面做完之后，停止 wireshark 捕获, 到这里这部分结束，同时准备工作完成，下面只需要观察捕获的包就可以了

2.2 2. A look at the captured trace

2.2.1 需要的截图

下面是捕获到的包的列表开始的一部分，这里有很多其他的协议混杂，方便起见后面使用过滤器找出 icmp 协议的包。



这第 15 个包就是我电脑发送的第一个 ICMP 响应请求报文，其详细信息截图如下

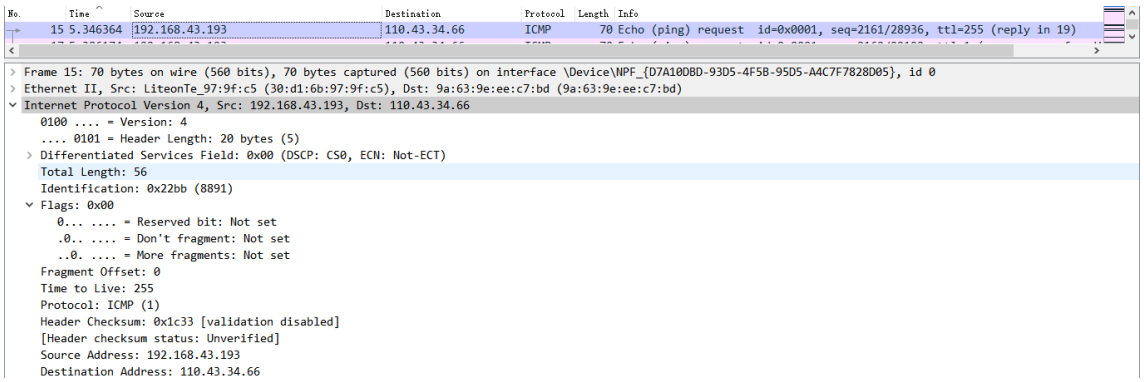
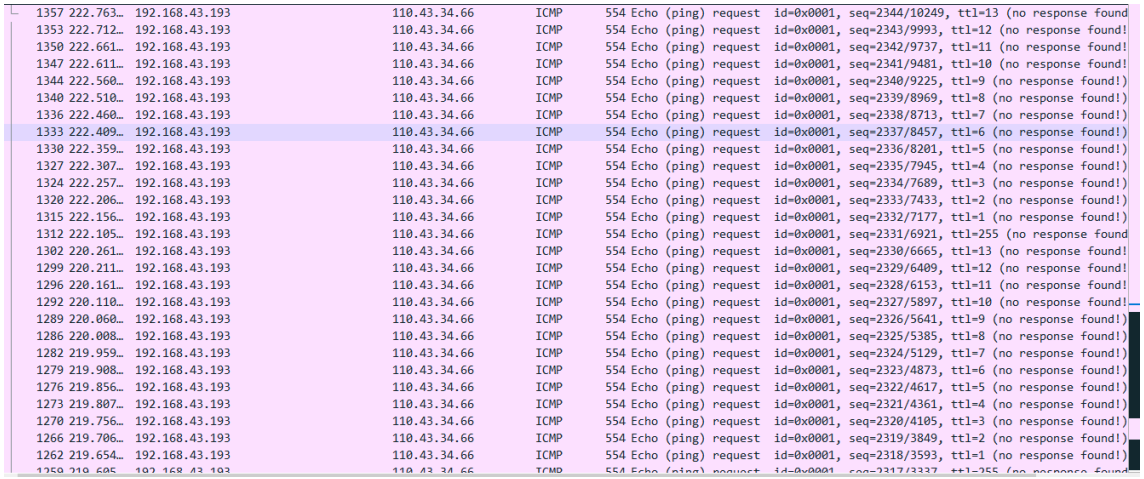
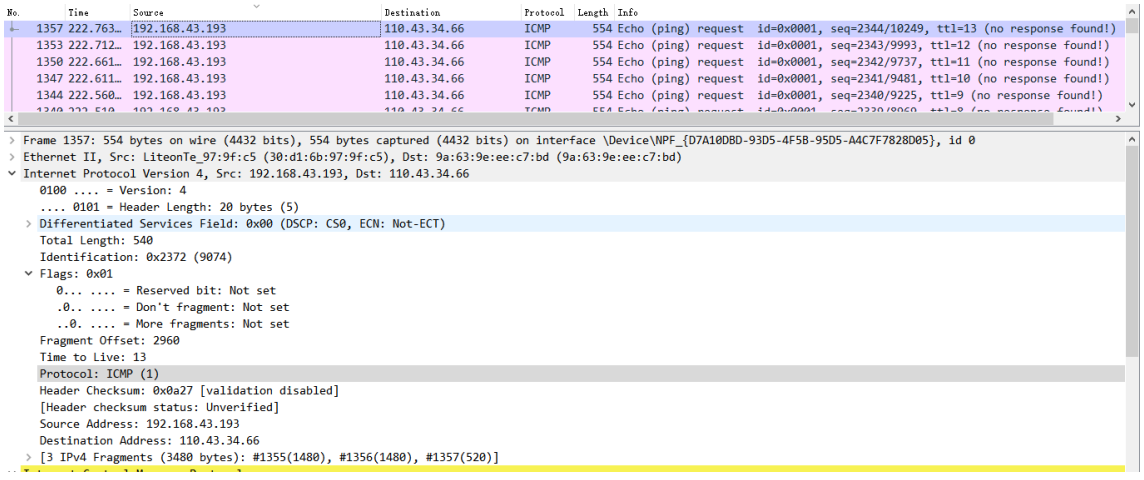


图 1

然后将这些包按照源地址排序，得到结果如下



要看具体内容的话，变成下图这样，然后就可以一个个的翻看比较了



然后至于 ICMP TTL-exceeded replies，可以看到比较靠前的如下，第 27 个包应该是从第一跳的路由器发送回来的

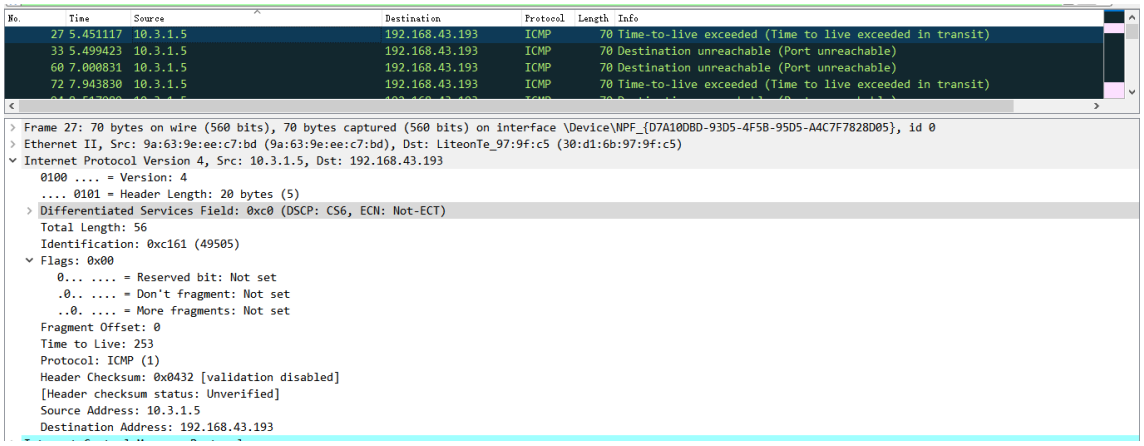


图 2

2.2.2 小节思考题

1. Select the first ICMP Echo Request message sent by your computer, and expand the Internet Protocol part of the packet in the packet details window. What is the IP address of your computer?

看我上面的图 1，其源地址就是我电脑的 IP 地址，即为 192.168.43.193

2. Within the IP packet header, what is the value in the upper layer protocol field?

在我的截图 1 里面有一行，Protocol: ICMP (1)，表示上层协议是 ICMP 后面跟着的 1，即为上层协议的值

3. How many bytes are in the IP header? How many bytes are in the payload of the IP datagram? Explain how you determined the number of payload bytes.

在我的截图 1 里面有一行.... 0101 = Header Length: 20 bytes (5)，表示 IP 头有 20 字节，然后在后面有一行 Total Length: 56，即总长为 56 字节，那么其有效载荷为 36 字节，即将总长减去头长度得到有效载荷大小

4. Has this IP datagram been fragmented? Explain how you determined whether or not the datagram has been fragmented. 没有，从截图 1 里面看到 flag 行为 0，可以看到里面 more fragments 是 not set，偏移量也为 0，这表明并没有被分段，同时如果分段了，可以在后面看到分了几段，这里并没有显示。

5. Which fields in the IP datagram always change from one datagram to the next within this series of ICMP messages sent by your computer?

通过一个个的查看这一系列数据报文，可以发现 Identification,Time to live,Header checksum 一直在改变

6. Which fields stay constant? Which of the fields must stay constant? Which fields must change? Why?

可以注意到这个 protocol 一直都是 ICMP 不变，然后 header length 一直为 20，version 一直为 4，Differentiated Services Field 一直为 0x00 (DSCP: CS0, ECN: Not-ECT)，源地址 192.168.43.193 和目的地址 110.43.34.66 都不变，然后 total length，flag 会在一段时间内不变，因为在这一段时间内，pingplotter

发送的包大小始终为一个定值，而是否会分片是由这个大小决定，所以 flag 也会维持一段时间不变。

至于必须保持不变的应该是版本是 ipv4，首部长度，服务和协议这样的，因为这些都是规定好的规范

必须改变的应该是 Identification，我们需要不同的标识来区分数据报，time to live 也必须变化，这个是由这个软件决定的，会发送不同 ttl 的包，因为这个结果由首部的各项决定，而首部存在必定会变化的项，所以检验和随之变化。

7. Describe the pattern you see in the values in the Identification field of the IP datagram

可以发现每次随着 wireshark 包序号增加，这个 identification 的值都会加 1.

8. What is the value in the Identification field and the TTL field?

观察上面标为图 2 的截图，可以看到这里 identification 是 49505，TTL 是 253.

9. Do these values remain unchanged for all of the ICMP TTL-exceeded replies sent to your computer by the nearest (first hop) router? Why?

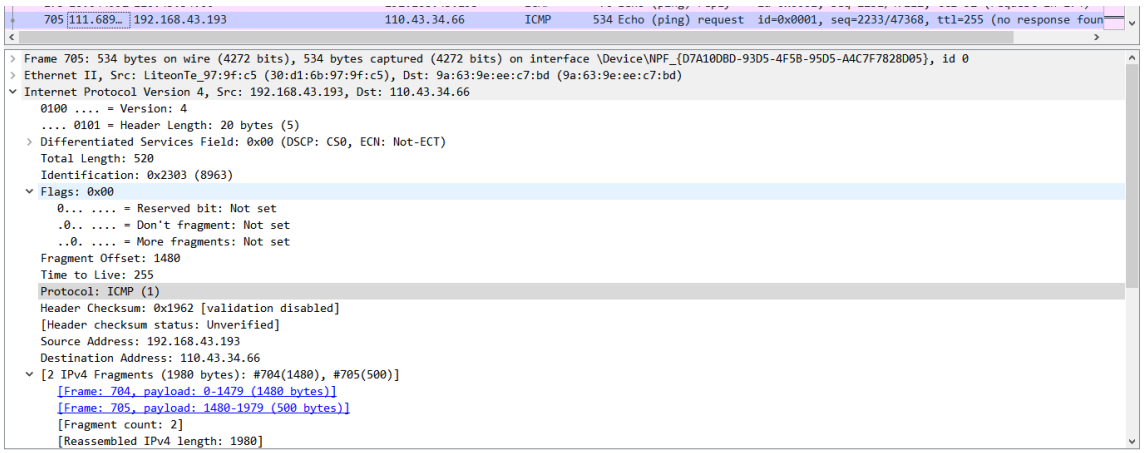
并没有，identification 是会变化的，因为 identification 是不同数据报的标识，而 ttl 貌似没有变化，毕竟这是第一跳的路由器，减少的都会一样。

2.3 Fragmentation

2.3.1 需要的截图

这一小节观察 IP 数据报分片，这里体现了前面修改包大小的作用，如果包太小就不会分片了

这里将这些包按照时间排序，然后找到改成 2000KB 后我的电脑第一个 ICMP 响应请求报文，如下截图



如果只看 ICMP 的话，貌似只会看到最后一个分片，也即将最后一个分片识别为了 ICMP 协议，要看全部的分片需要查看未经过滤的包列表，如下

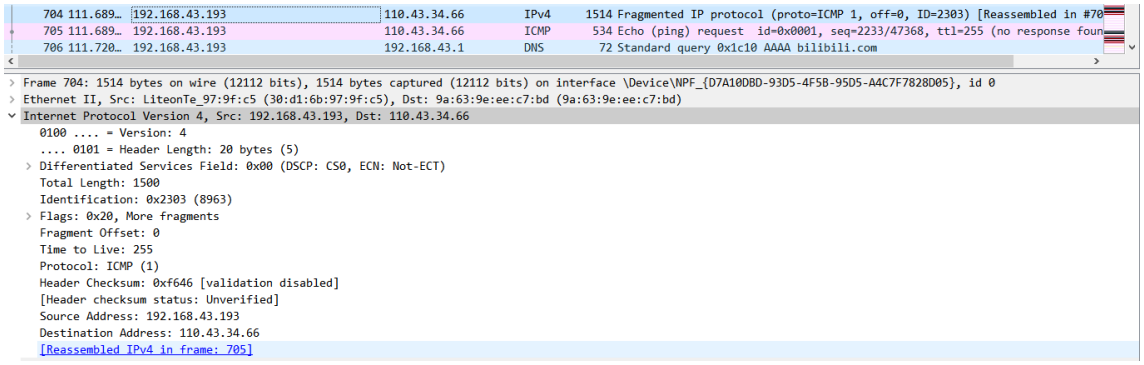
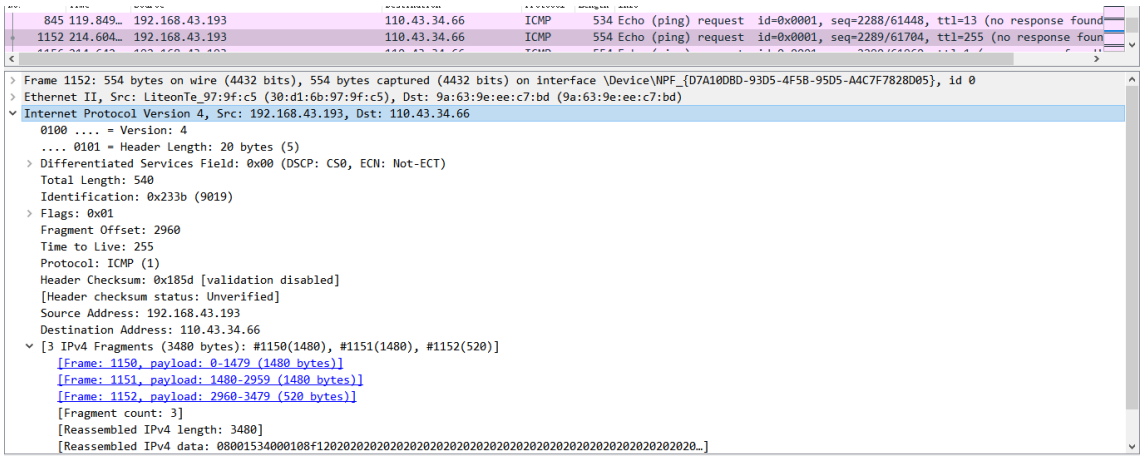
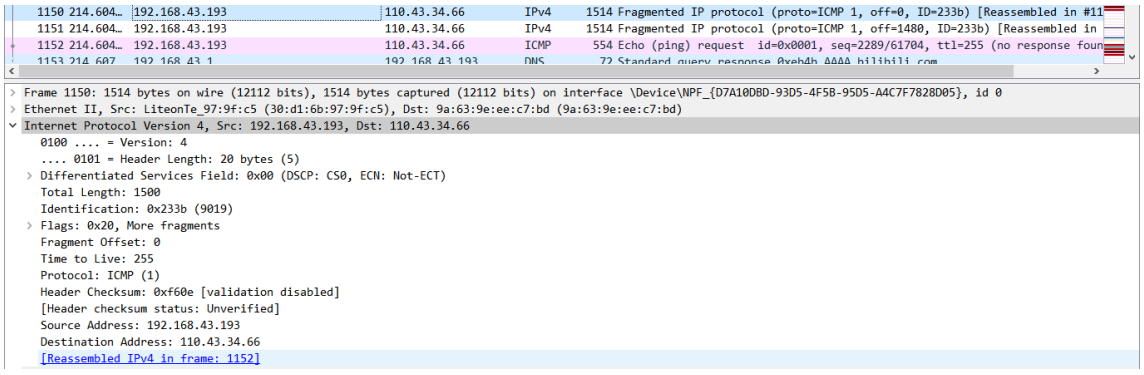
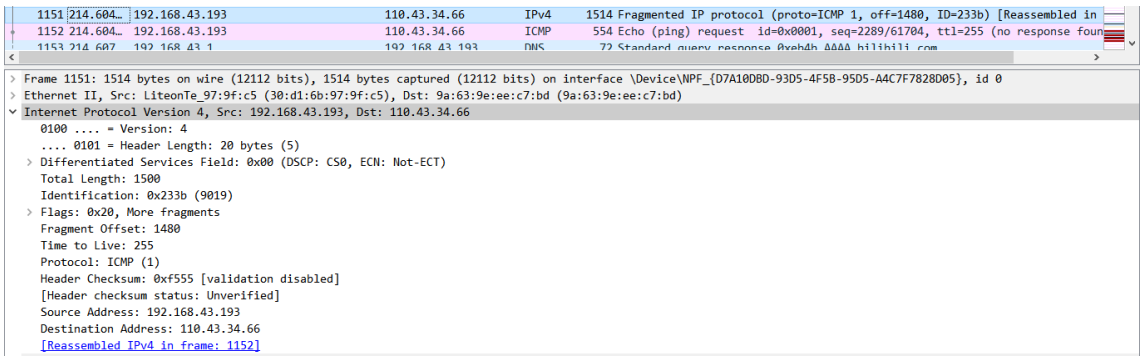


图 3

同理接下来对包大小为 3500KB，有三个分片，如截图所示







2.3.2 小节思考题

10. Find the first ICMP Echo Request message that was sent by your computer after you changed the Packet Size in pingplotter to be 2000. Has that message been fragmented across more than one IP datagram? [Note: if you find your packet has not been fragmented, you should download the zip file

http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip and extract the ip- ethereal-trace-1packet trace. If your computer has an Ethernet interface, a packet size of 2000 should cause fragmentation.3]

由上面的截图 3 看到，第一个 ICMP 响应请求报文，包序号为 705，前面序号 704 就是前面一个片，从而可以看到确实分片了，对 2000KB 的数据分成了两片

11. Print out the first fragment of the fragmented IP datagram. What information in the IP header indicates that the datagram been fragmented? What information in the IP header indicates whether this is the first fragment versus a latter fragment? How long is this IP datagram?

将这个第一片的数据报打印出来如下

```
No.     Time           Source                Destination           Protocol Length Info
    704 111.689992     192.168.43.193        110.43.34.66          IPv4     1514   Fragmented IP protocol
(proto=ICMP 1, off=0, ID=2303) [Reassembled in #705]
Frame 704: 1514 bytes on wire (12112 bits), 1514 bytes captured (12112 bits) on interface
\Device\NPF_{D7A10DBD-93D5-4F5B-95D5-A4C7F7828D05}, id 0
Ethernet II, Src: LiteonTe_97:9f:c5 (30:d1:6b:97:9f:c5), Dst: 9a:63:9e:ee:c7:bd (9a:63:9e:ee:c7:bd)
Internet Protocol Version 4, Src: 192.168.43.193, Dst: 110.43.34.66
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
    Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 1500
    Identification: 0x2303 (8963)
    Flags: 0x20, More fragments
        0... .... = Reserved bit: Not set
        .0.. .... = Don't fragment: Not set
        ..1. .... = More fragments: Set
    Fragment Offset: 0
    Time to Live: 255
    Protocol: ICMP (1)
    Header Checksum: 0xf646 [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 192.168.43.193
    Destination Address: 110.43.34.66
    [Reassembled IPv4 in frame: 705]
```

可以在里面看到有 Flags: 0x20, More fragments，说明后面还有其他的片，也即这个被分片了，然后由 Fragment Offset: 0 知其偏移量为 0，也即这是第一片这个数据报长度可由 Total Length: 1500 得出是 1500KB

12. Print out the second fragment of the fragmented IP datagram. What information in the IP header indicates that this is not the first datagram fragment? Are the more fragments? How can you tell?

第二片数据报打印出来如下

```
No.     Time           Source                Destination           Protocol Length Info
    705 111.689992     192.168.43.193        110.43.34.66          ICMP     534    Echo (ping) request
id=0x0001, seq=2233/47368, ttl=255 (no response found!)
Frame 705: 534 bytes on wire (4272 bits), 534 bytes captured (4272 bits) on interface
\Device\NPF_{D7A10DBD-93D5-4F5B-95D5-A4C7F7828D05}, id 0
Ethernet II, Src: LiteonTe_97:9f:c5 (30:d1:6b:97:9f:c5), Dst: 9a:63:9e:ee:c7:bd (9a:63:9e:ee:c7:bd)
Internet Protocol Version 4, Src: 192.168.43.193, Dst: 110.43.34.66
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
    Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 520
    Identification: 0x2303 (8963)
    Flags: 0x00
        0... .... = Reserved bit: Not set
        .0.. .... = Don't fragment: Not set
        ..0. .... = More fragments: Not set
    Fragment Offset: 1480
    Time to Live: 255
    Protocol: ICMP (1)
    Header Checksum: 0x1962 [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 192.168.43.193
    Destination Address: 110.43.34.66
    [2 IPv4 Fragments (1980 bytes): #704(1480), #705(500)]
        [Frame: 704, payload: 0-1479 (1480 bytes)]
        [Frame: 705, payload: 1480-1979 (500 bytes)]
        [Fragment count: 2]
        [Reassembled IPv4 length: 1980]
        [Reassembled IPv4 data: 0800338a000108b9202020202020202020202020202020202020202020202020202020…]
Internet Control Message Protocol
```

可以看到 Fragment Offset: 1480，表明这里偏移量为 1480KB，从而知道这不是第一片，然后由 Flags:

0x00，More fragments: Not set 知道后面没有其他的片了这是最后一片。

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

由上面两题知道 Flags，Fragment Offset，total length 都有变化，那么首部检查和也会相应变化，所以这些域都变了，其他的没有变化。

14. How many fragments were created from the original datagram?

根据我上面 3500KB 对应的截图，这里分成了 3 片。

15. What fields change in the IP header among the fragments?

与 13 题同理，Flags，Fragment Offset，total length 都有变化，首部检查和也会相应变化。

Chapter 3

总结与思考

本次实验学习了 IP 协议，了解其数据报的格式，也看到了分片的操作，对 IP 协议有了更具体的感受，收获很多。