网络层数据分组的捕获和分析

1. 实验内容和实验目的

本次实验内容：

1）捕获在连接 Internet 过程中产生的网络层分组：DHCP 分组，ARP 分组，IP 数据分组，ICMP 分组。

2）分析各种分组的格式，说明各种分组在建立网络连接过程中的作用。

3）分析 IP 数据分组分片的结构。

通过本次实验了解计算机上网的工作过程，学习各种网络层分组的格式及其作用，理解长度大于 1500 字节 IP 数据组分片传输的结构。

2. 实验过程

1、 ICMP

ICMP（Internet Control Message Protocol）是 Internet 控制报文协议，是一种面向无连接的协议，用于传输出错报告控制信息，对于网络安全具有极其重要的意义。它是 TCP/IP 协议簇的一个子协议，用于在 IP 主机、路由器之间传递控制消息。控制消息是指网络通不通、主机是否可达、路由是否可用等网络本身的消息。这些控制消息虽然并不传输用户数据，但是对于用户数据的传递起着重要的作用。

ICMP 报文分为差错报告报文和询问报文：

ICMP 差错报告报文共有 5 种

- 终点不可达
- 源站抑制
- 时间超过
- 参数问题
- 改变路由（重定向）

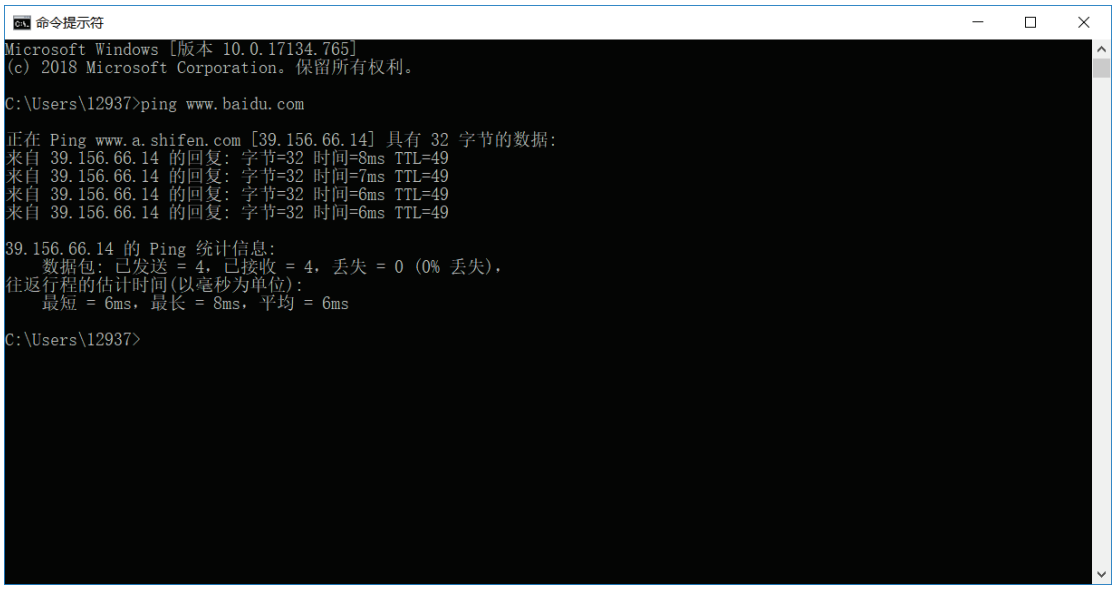
ICMP 询问报文有四种

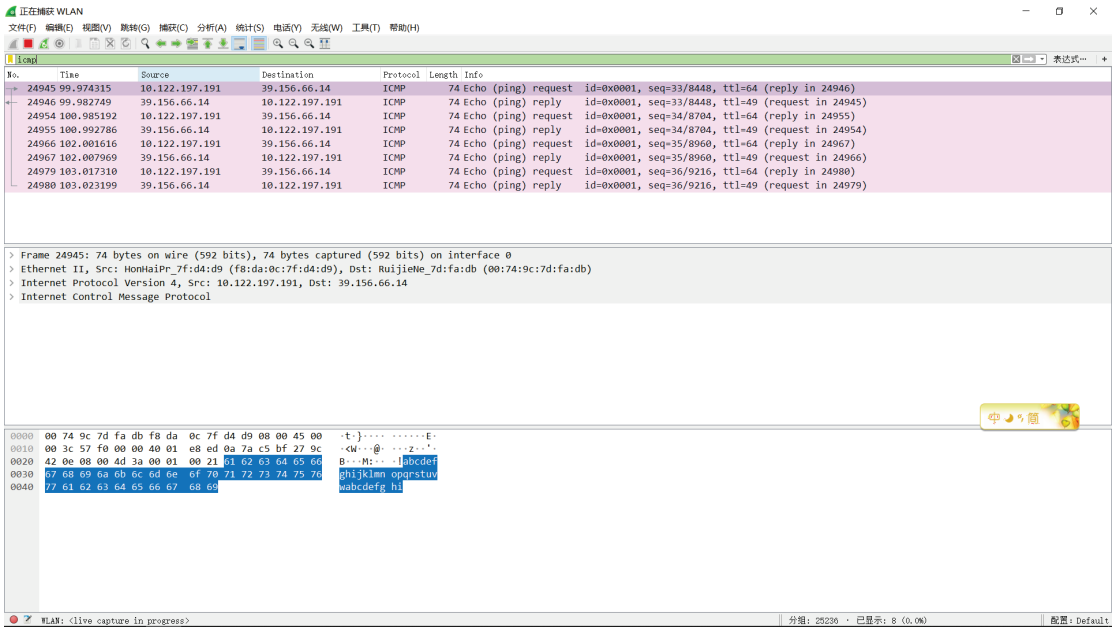
- 回送请求和回答报文
- 时间戳请求和回答报文
- 掩码地址请求和回答报文
- 路由器询问和通告报文

ICMP 可以应用在 ping 上，检查目的主机是否联通；可以用来测试到达目的主机的路径和跳数等。

操作步骤：

1、在【wireshark】过滤器中设置只捕获 ICMP 报文，在【cmd】中输入“ping www.baidu.com”。在 wireshark 中捕获到了 ICMP 报文，如下图所示：

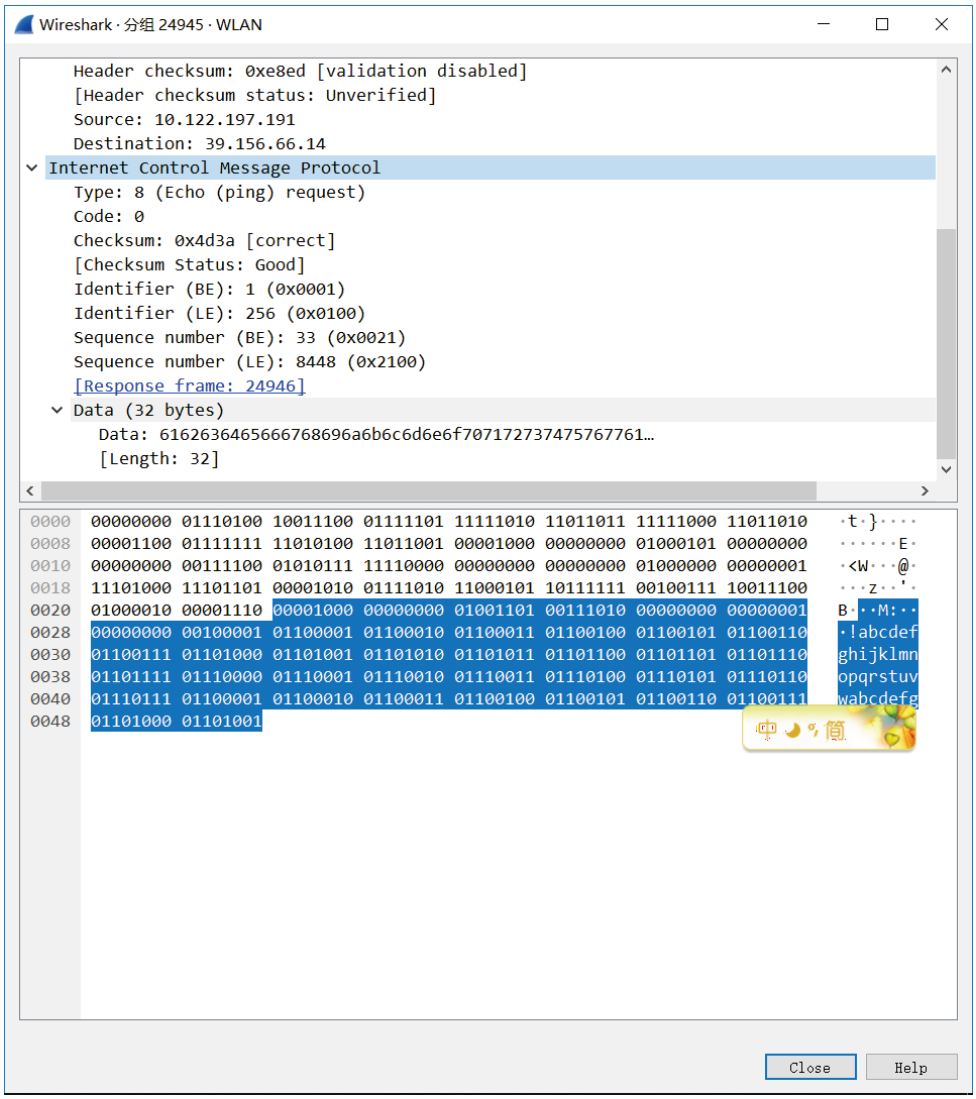




分析:

一次 ping 命令会发送 4 个包，观察发现他们的 ID 是相同的，序号是递增的。现在我们取前两个包进行分析。

请求包:



【00001000】为 ICMP 报文类型：8（ping 请求）

【00000000】为代码：0

【01001101 00111010】为校验和：0x4d3a（正确）

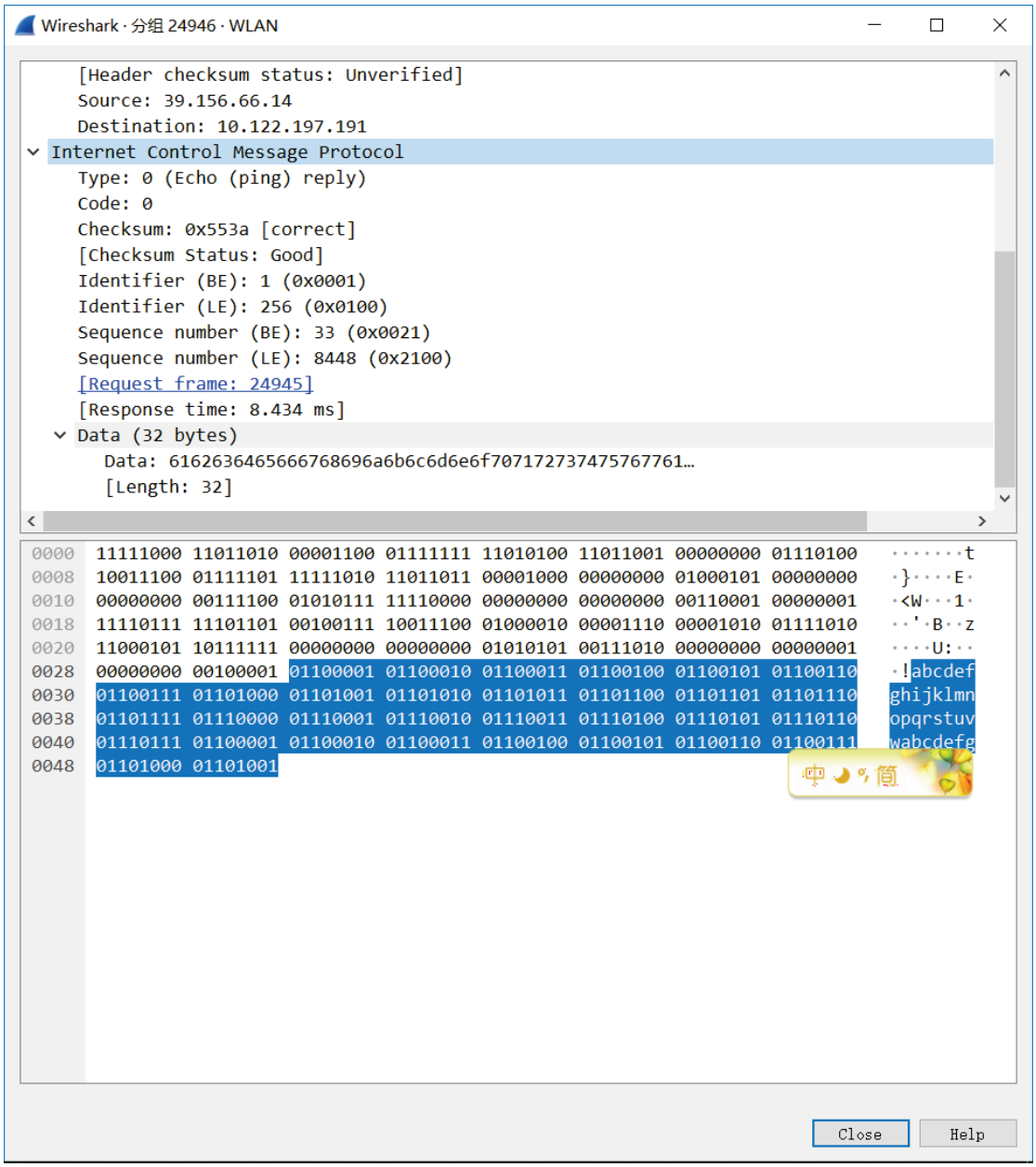
【00000000 00000001】为标志位 1/256。

【00000000 00100001】为序列号 33/8448。

windows 系统为小端法（little-endian byte order，即 LE），linux 系统为大端法（big-endian byte order，即 BE）。

后面 32 字节为数据部分。

应答包：



【00000000】为 ICMP 报文类型：0（ping 应答）

【00000000】为代码：0

【01010101 00111010】为校验和：0x553a（正确）

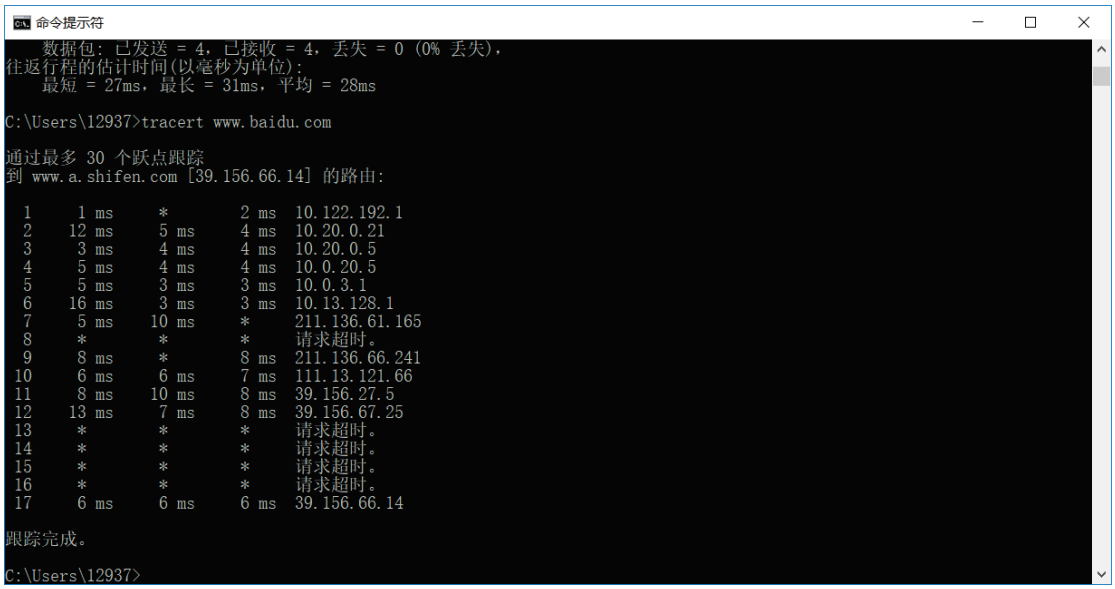
【00000000 00000001】为标志位 1/256。

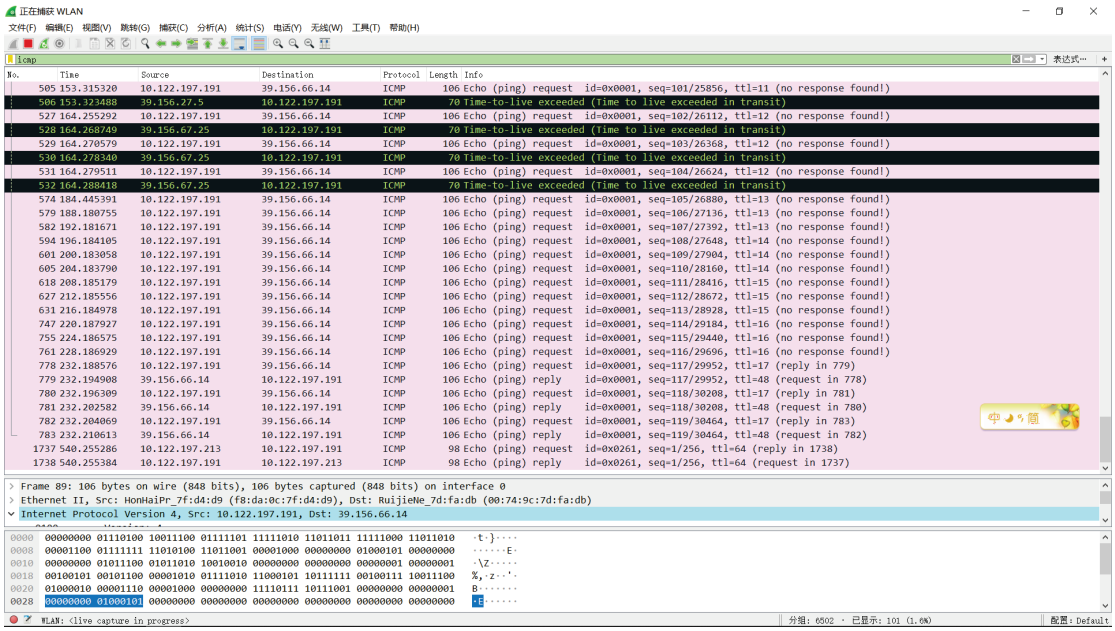
【00000000 00100001】为序列号 33/8448。

windows 系统为小端法（little-endian byte order，即 LE），linux 系统为大端法（big-endian byte order，即 BE）。

后面 32 字节为数据部分。

tracert 命令：





包的分析同上。

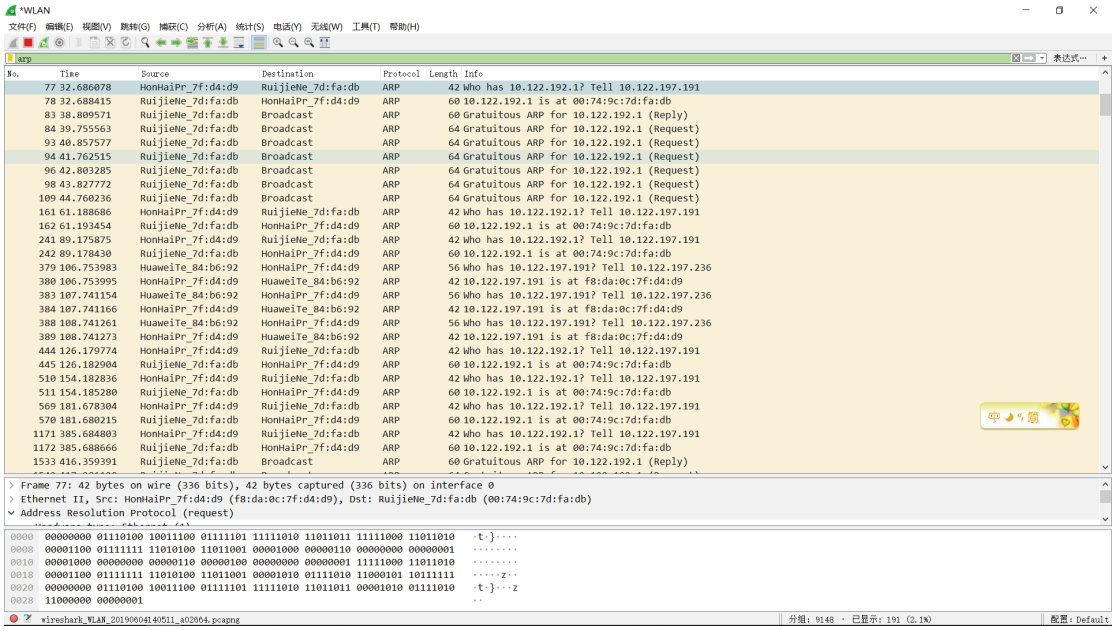
2、 ARP

ARP（Address Resolution Protocol），是地址解析协议，是根据 IP 地址获取物理地址的一个 TCP/IP 协议。主机发送信息时将包含目标 IP 地址的 ARP 请求广播到网络上的所有主机，并接收返回消息，以此确定目标的物理地址；收到返回消息后将该 IP 地址和物理地址存入 本机 ARP 缓存中并保留一定时间，下次请求时直接查询 ARP 缓存以节约资源。

ARP 的原理就是对于不知道物理地址的 IP 地址，主机发送一个广播包，内容就是询问 IP 地址对应的主机。若该 IP 对应的主机存在，那对应主机就回复一个包，包内包含自己的 MAC 地址。如果主机问了三遍还是没人回复，就认为这个 IP 地址没人使用。

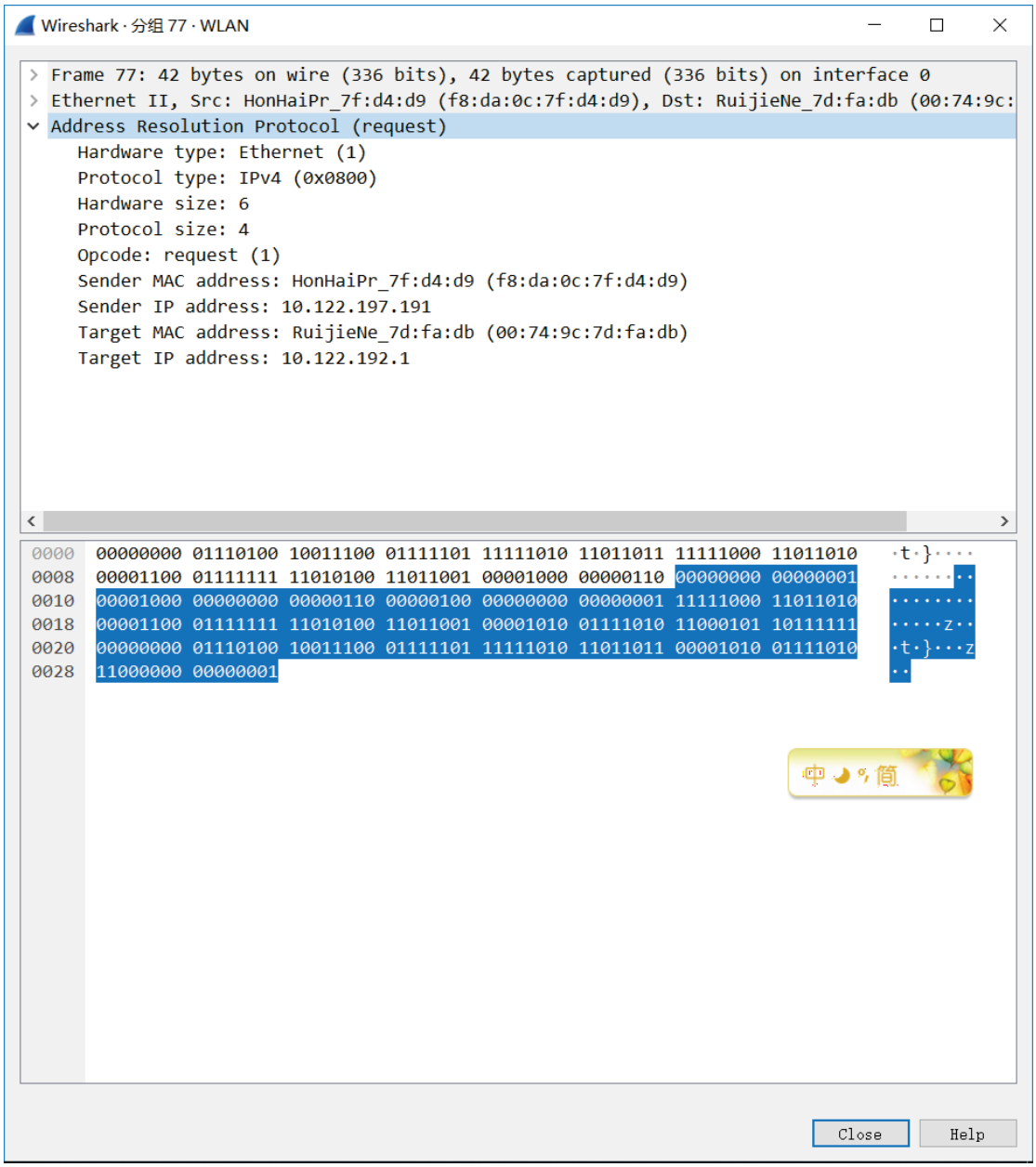
操作步骤：

1、在【wireshark】过滤器中设置只捕获 ARP 报文，在【wireshark】中收到了 ARP 包，取前两个进行分析。



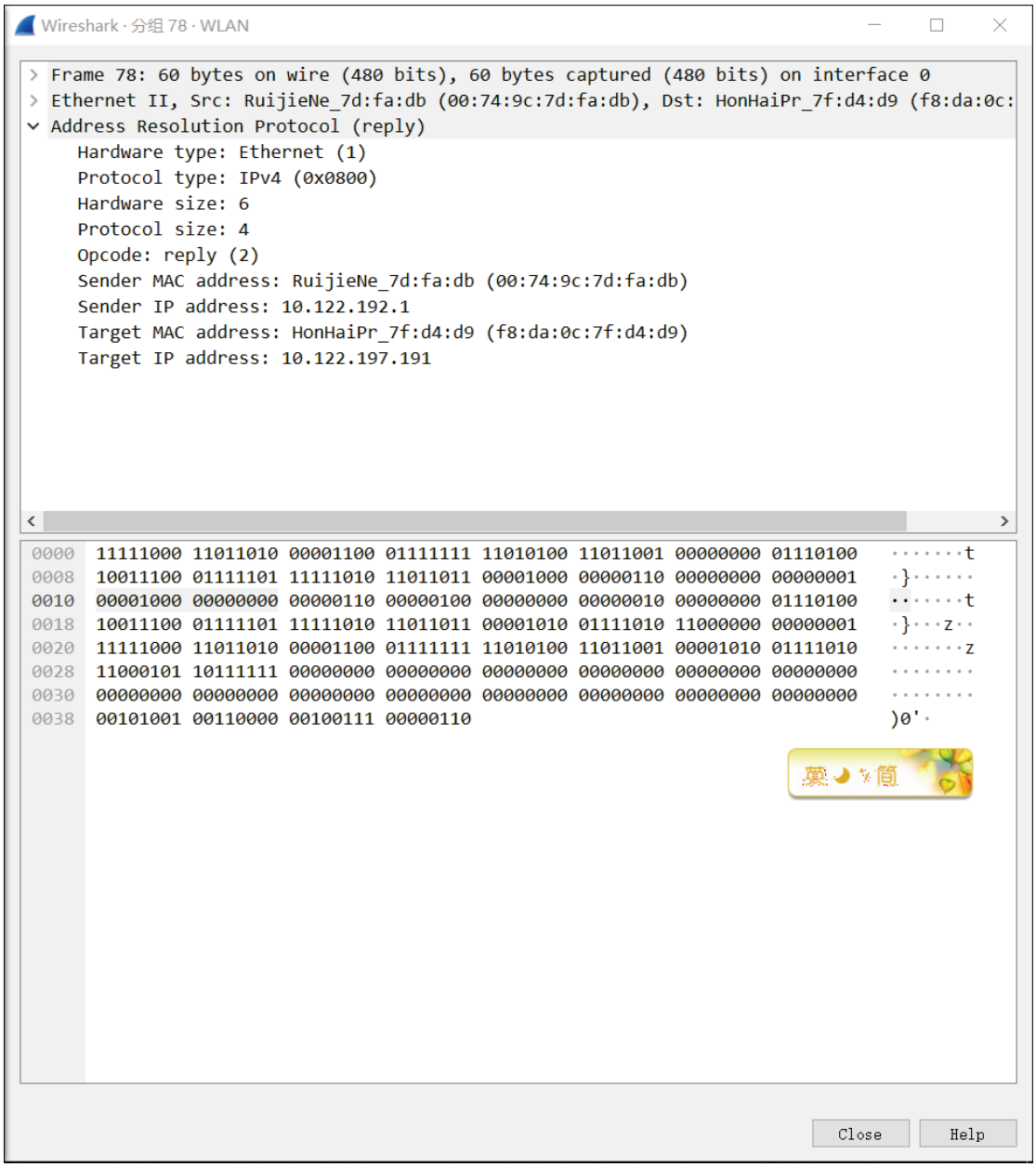
分析：

请求包：



【00000000 00000001】硬件地址：硬件地址表示硬件接口类型，1 表示以太网
【00001000 01111111】协议种类：指明协议种类，IPV4 是 0x0800
【00000110】硬件地址长度：指明硬件地址长度，单位为比特
【00000100】协议长度：指明协议长度
【00000000 00000001】操作码：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
【f8:da:0c:7f:d4:d9】发送方 MAC 地址
【10.122.197.191】发送方 IP 地址
【00:74:9c:7d:fa:db】目的地 MAC 地址
【10.122.192.1】目的地 IP 地址

应答包：



【00000000 00000001】硬件地址：硬件地址表示硬件接口类型，1 表示以太网
【00001000 01111111】协议种类：指明协议种类，IPV4 是 0x0800
【00000110】硬件地址长度：指明硬件地址长度，单位为比特
【00000100】协议长度：指明协议长度
【00000000 00000010】操作码：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
【00:74:9c:7d:fa:db】发送方 MAC 地址
【10.122.192.1】发送方 IP 地址
【f8:da:0c:7f:d4:d9】目的地 MAC 地址
【10.122.197.191】目的地 IP 地址
本机希望知道IP为【10.122.192.1】主机的MAC地址，如果本机的ARP缓存表中没有目标IP地址，那么本机将会发送一个广播本机MAC地址是【f8:da:0c:7f:d4:d9】，这表示向同一网段内的所有主机发出这样的询问：我是【10.122.197.191】，我的硬件地址是【f8:da:0c:7f:d4:d9】，请问IP地址为

【10.122.192.1】的MAC地址是什么？网络上其他主机并不响应，只有目标主机接收到时，才向本机做出这样的回应：【10.122.192.1】的MAC地址是【00:74:9c:7d:fa:db】。本机知道了目标主机的MAC地址，同时更新了ARP缓存表，下次本机再向目标主机或者目标主机向本机发送信息时，直接从各自的ARP缓存表里查找就可以了。

3、 DHCP

动态主机设置协议（Dynamic Host Configuration Protocol，DHCP）是一个局域网的网络协议，使用UDP协议工作，主要有两个用途：用于内部网或网络服务供应商自动分配IP地址；给用户用于内部网管理员作为对所有计算机作中央管理的手段。

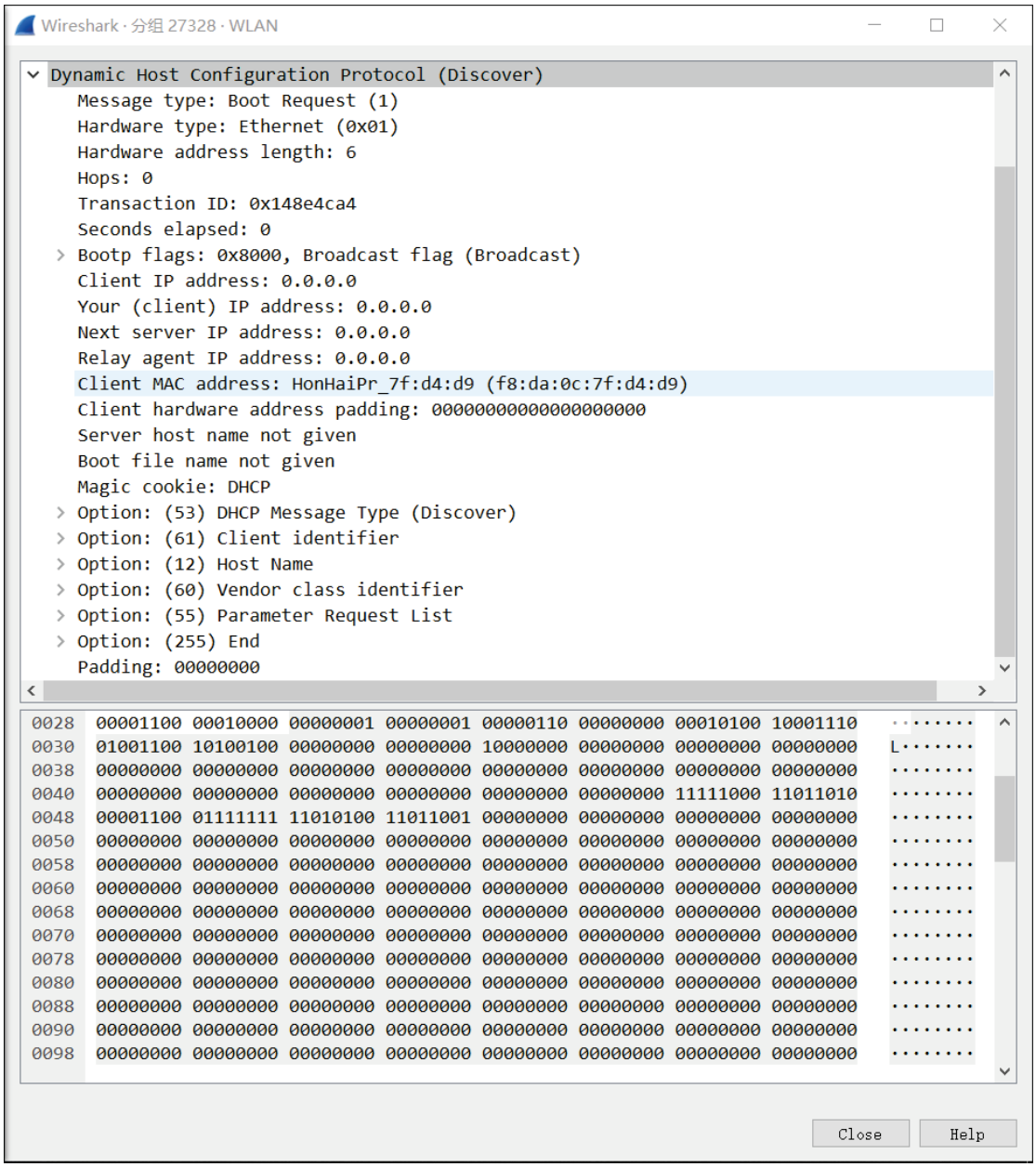
步骤：

1、在【wireshark】过滤器中设置只捕获 DHCP 包。
2、在【cmd】中运行“ipconfig /release”和“ipconfig /renew”命令。
3、在【wireshark】中收到的 DHCP 包如下图所示。

26934	3225.819685	10.122.197.191	10.3.9.31	DHCP	342	DHCP Release - Transaction ID 0xc3eb547
27328	3268.540457	0.0.0.0	255.255.255.255	DHCP	342	DHCP Discover - Transaction ID 0x148e4ca4
27359	3269.549013	10.122.192.1	255.255.255.255	DHCP	342	DHCP Offer - Transaction ID 0x148e4ca4
27360	3269.549414	0.0.0.0	255.255.255.255	DHCP	370	DHCP Request - Transaction ID 0x148e4ca4
27361	3269.570444	10.122.192.1	255.255.255.255	DHCP	342	DHCP ACK - Transaction ID 0x148e4ca4

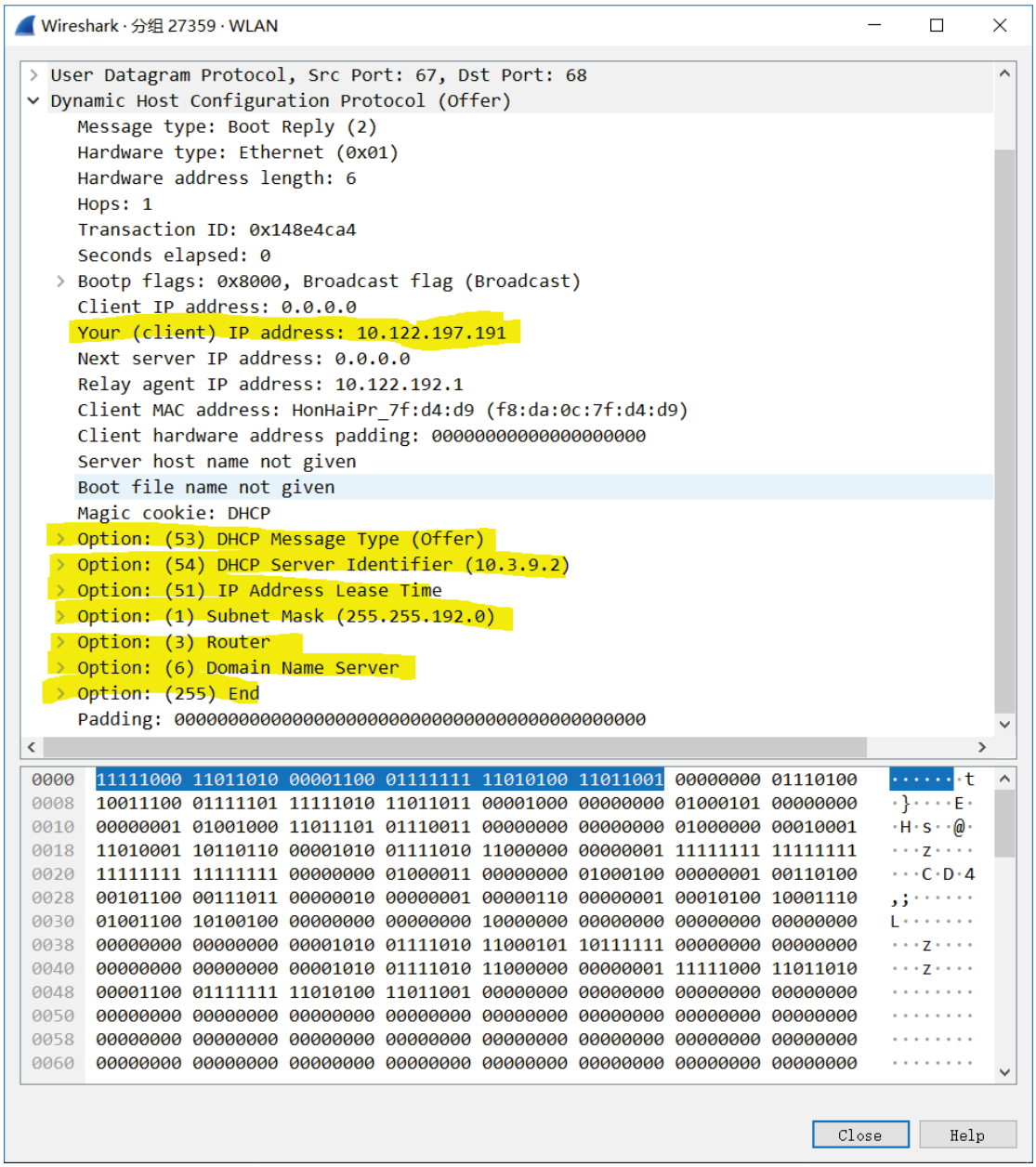
分析：

一次会收到4个数据包
数据包1：DHCP Discover
数据包2：DHCP Offer
数据包3：DHCP Request
数据包4：DHCP ACK
下面分别进行分析：
1、



Client端使用IP地址0.0.0.0发送了一个广播包，目的IP为255.255.255.255。Client想通过这个数据包发现可以给它提供服务的DHCP服务器。

2、

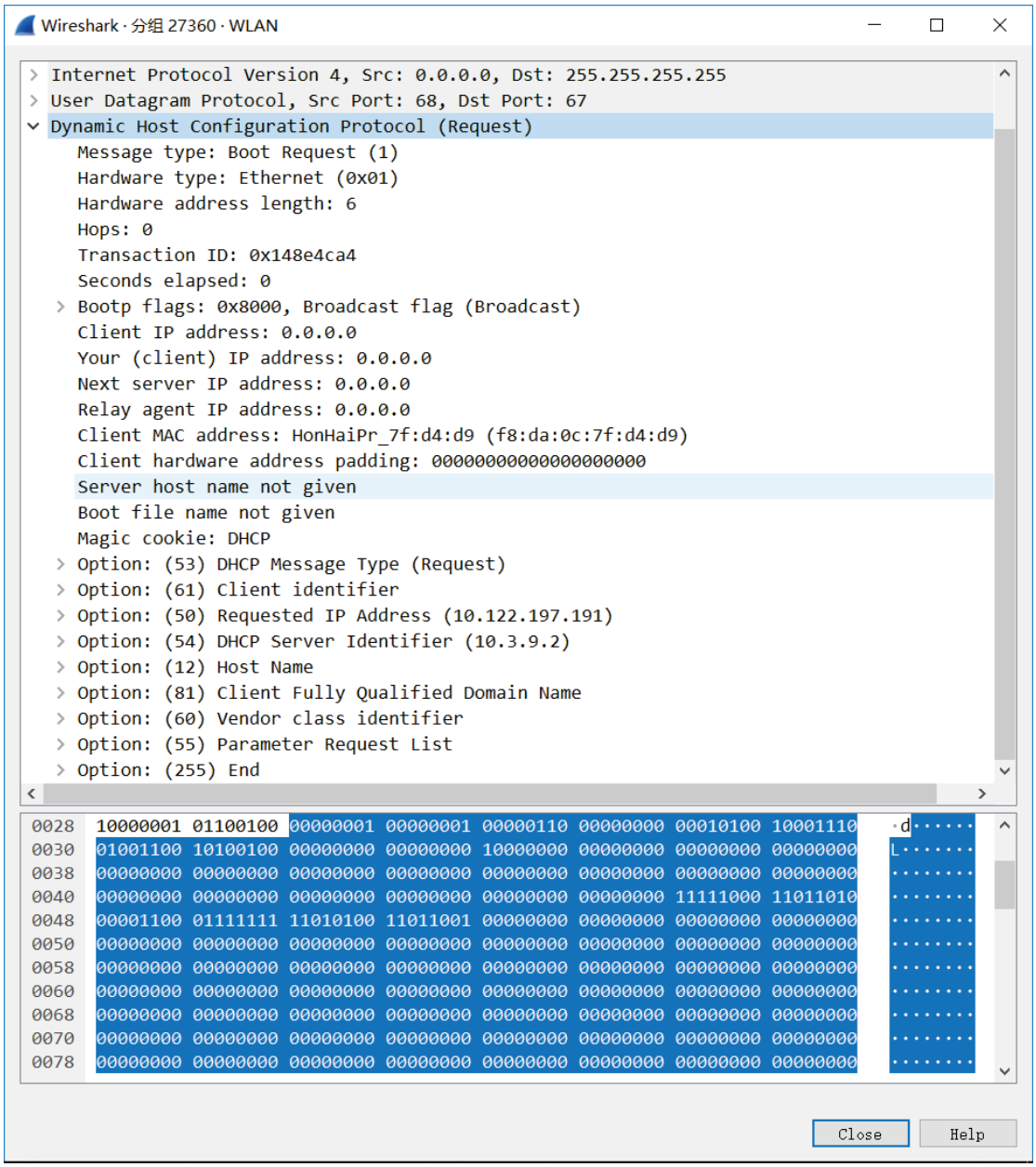


①当DHCP服务器（10.3.9.2）收到一条DHCP Discover数据包时，用一个DHCP Offerr包给予客户端响应。因为同时可能有多个Client在使用0.0.0.0这个IP向DHCP服务器发出IP分配请求，于是DHCP服务器采用广播的方式，告诉正在请求的Client，这是一台可以使用的DHCP服务器。

②DHCP服务器提供了一个可用的IP，Your（client）IP Address字段可以看到DHCP服务器提供的可用IP是10.122.197.191。

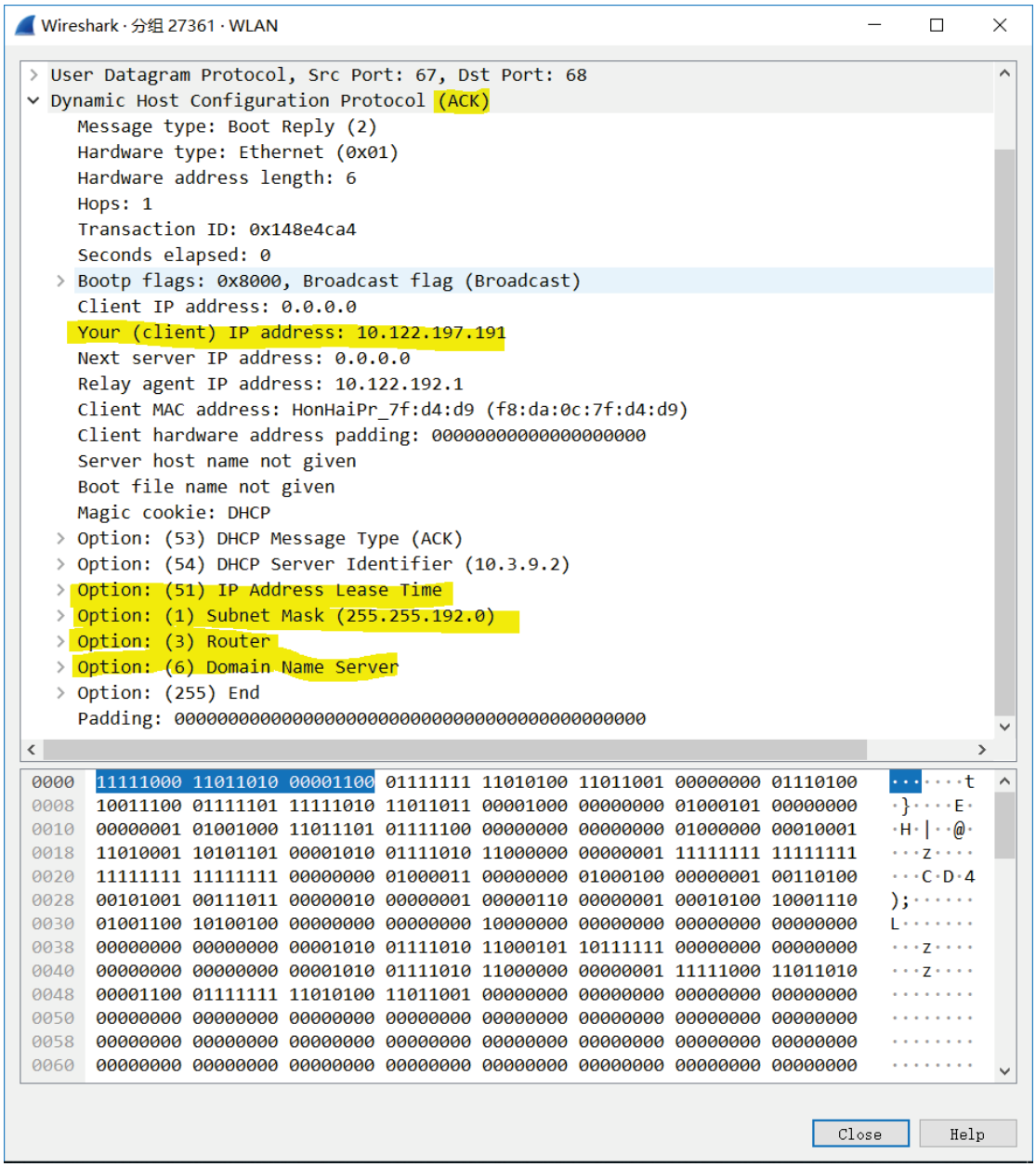
③在Option字段，服务器还发送了IP地址租用期、子网掩码、路由器、域名等信息。

3、



当Client收到了Offer包以后（如果有多个可用的DHCP服务器，那么可能会收到多个DHCP Offer包），确认有可以和它交互的DHCP服务器存在，于是Client发送Request包，请求分配IP。此时的源IP和目的IP依然是0.0.0.0和255.255.255.255。

4、



DHCP服务器用ACK包对DHCP请求进行响应，以确认 IP 租约的正式生效，至此结束了一个完整的 DHCP 工作过程。

【Your(client) IP address】：分配给Client的可用IP（10.122.197.191）

【IP Address Lease Time】：IP租用期

【Subnet Mask】：Client端分配到的IP的子网掩码

【Router】：路由器

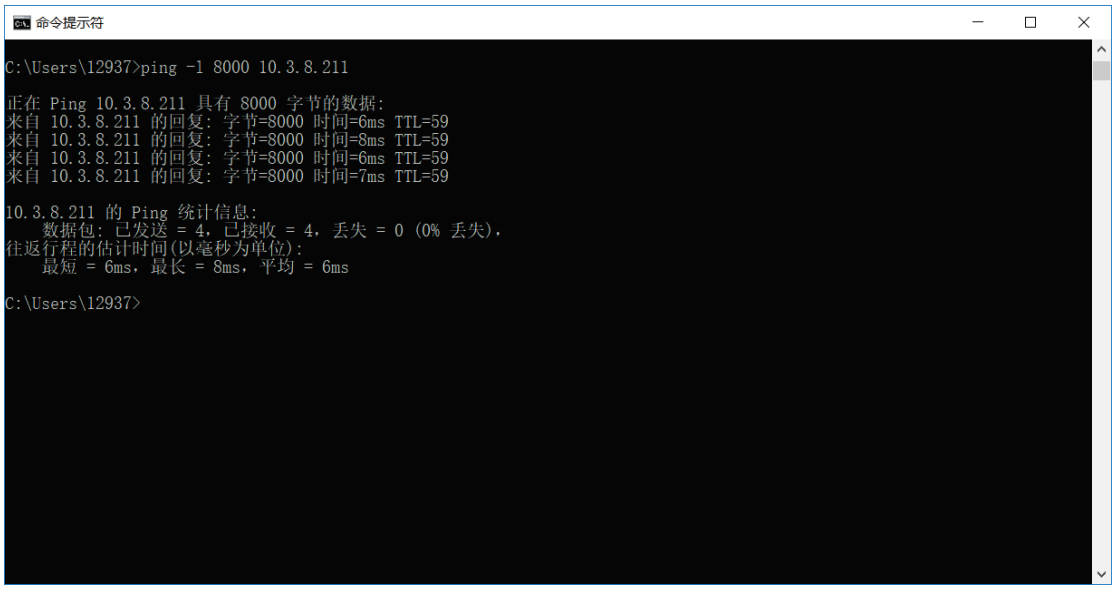
【Domain Name Server】：域名服务器

4、 IP 数据包

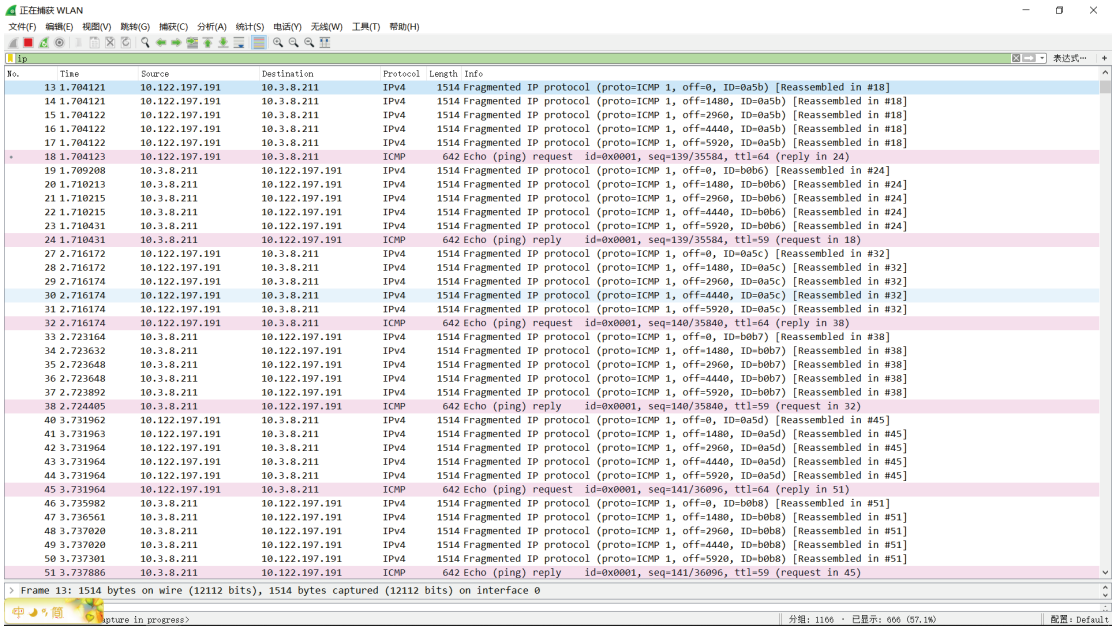
步骤：

1、在【wireshark】过滤器中设置只捕获 IP 包。

2、在【cmd】中运行“ping -l 8000 10.3.8.211”命令，发送一个 8000 字节的包通过 6 片进行分组传输。



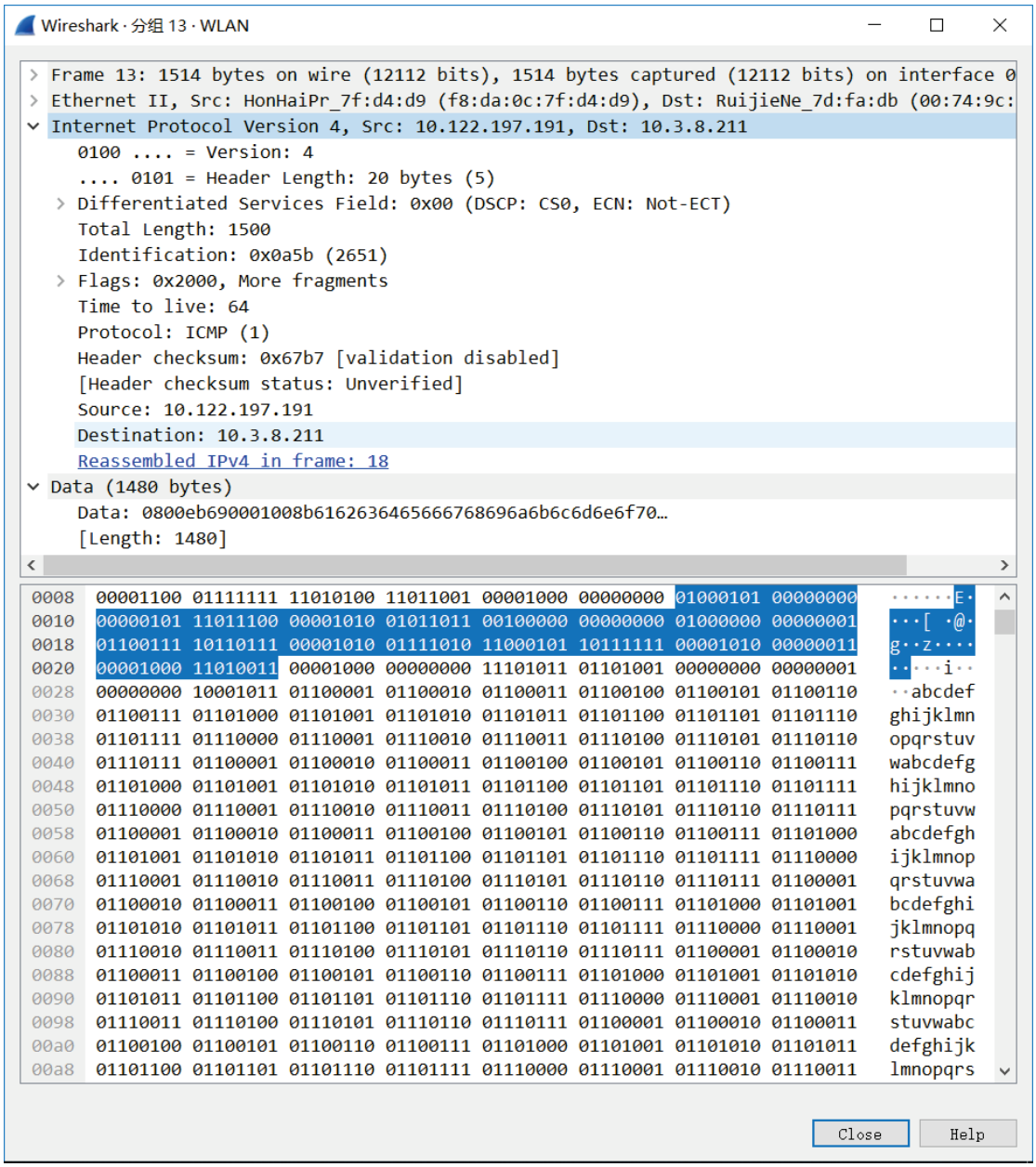
3、在【wireshark】中收到的 IP 包如下图所示，可以发现 ping 命令发送了 4 次 request，回复了 4 次，每次刚好是通过 6 片传输（ICMP 中也有一片）。



分析:

1、

取第一个分片进行分析



【0100】为协议类型：4.

【0101】为首部长度：5*4=20Bytes

【00000000】为服务类型。

【00000101 11011100】为总长度：1500，头部为20，数据为1480

【01010111 11110000】为标识：0x0a5b(2651)

【00100000 00000000】为标志位DF=0，MF=1，说明后面还有。Fragment offset=0

```
Flags: 0x2000, More fragments
    0... .... .... .... = Reserved bit: Not set
    .0.. .... .... .... = Don't fragment: Not set
    ..1. .... .... .... = More fragments: Set
    ...0 0000 0000 0000 = Fragment offset: 0
```

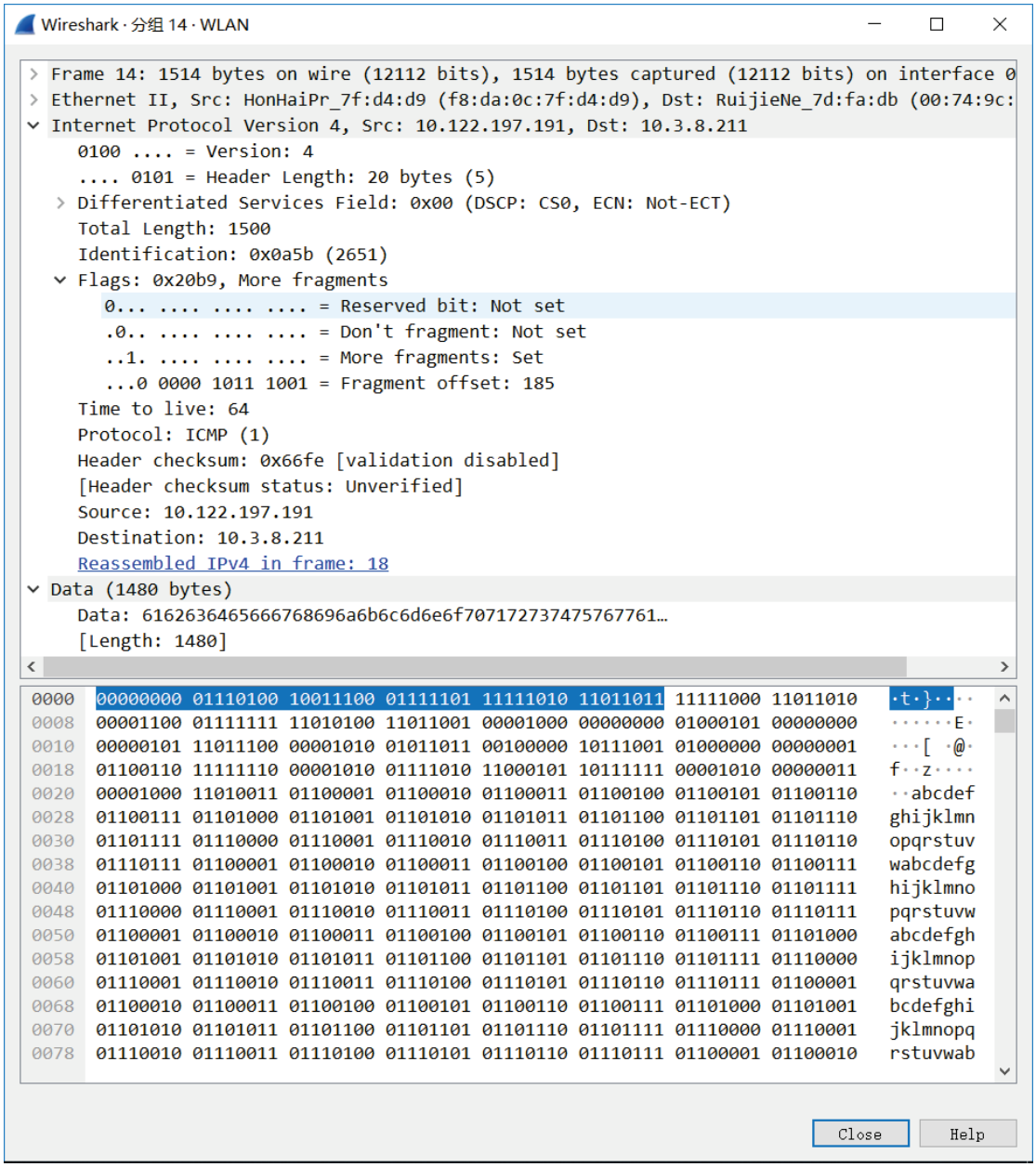
【01000000】 为生存时间：64

【00000001】为协议：ICMP（1）
【11101000 11101101】为首部校验和：0x67b7
【00001010 01111010 11000101 10111111】为源地址：10.122.197.191
【00001010 00000011 00001000 11010011】为目标地址：10.3.8.211
以下1480字节为数据部分。

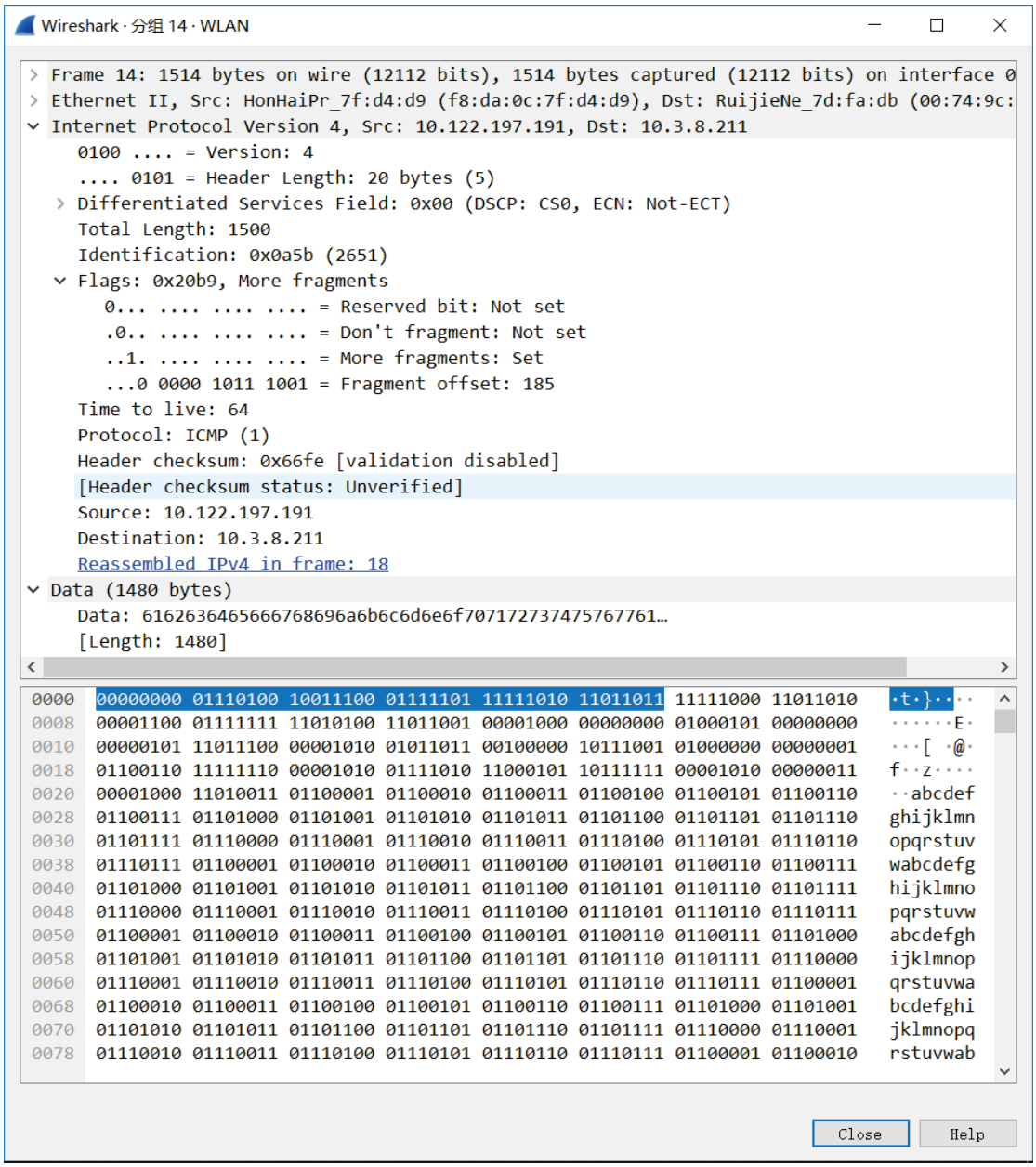
2、

后面4片与第一片相同，只是Fragment offset不再是0，而是185。该字段是以8个“八比特组”为单位的，所以1480（前面所有片的数据长度之和）在该字段的值是除以8之后的185。同理第n片的Fragment offset应为1480*(n-1)/8，经检验后发现是正确的。

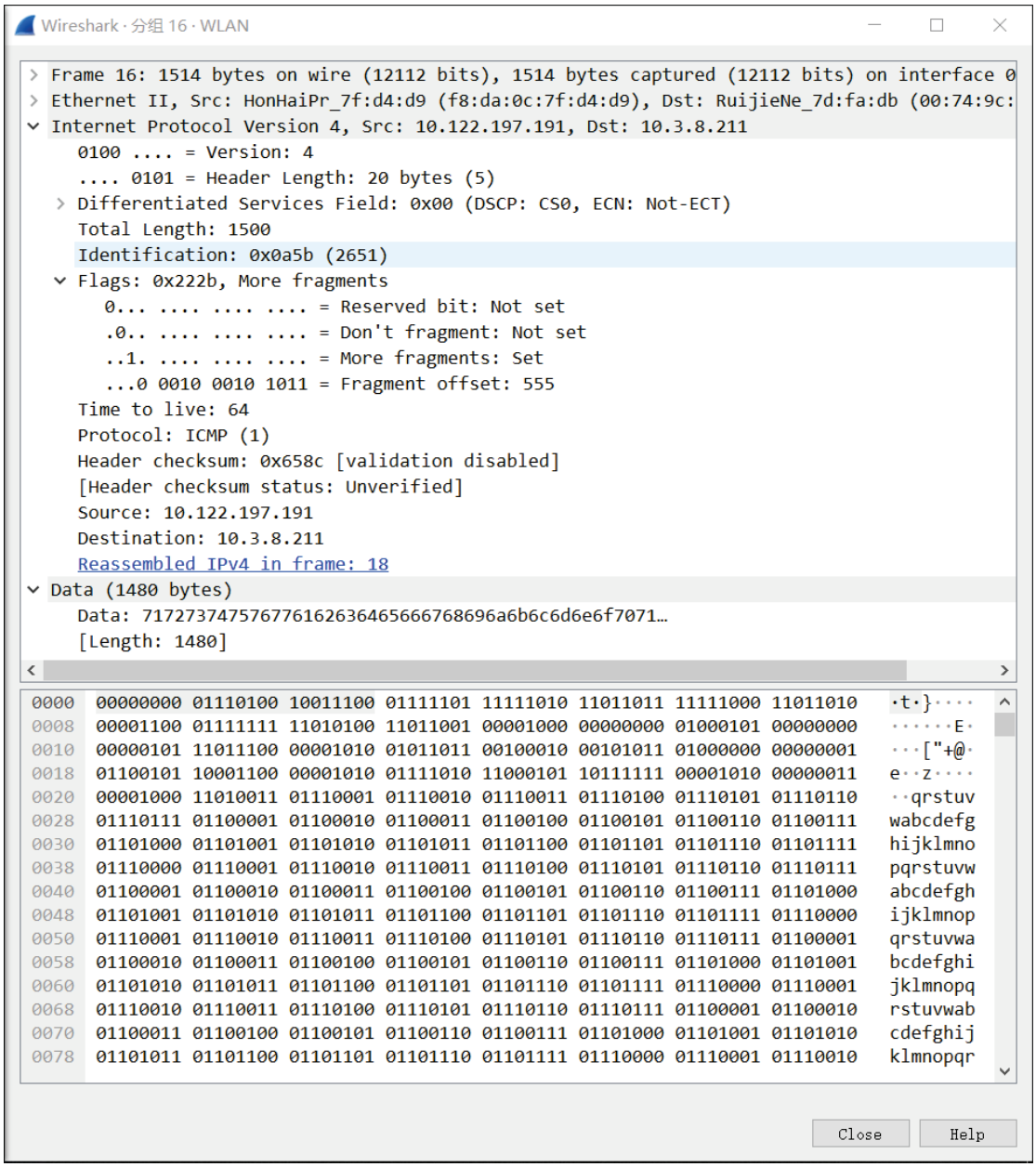
第二片：



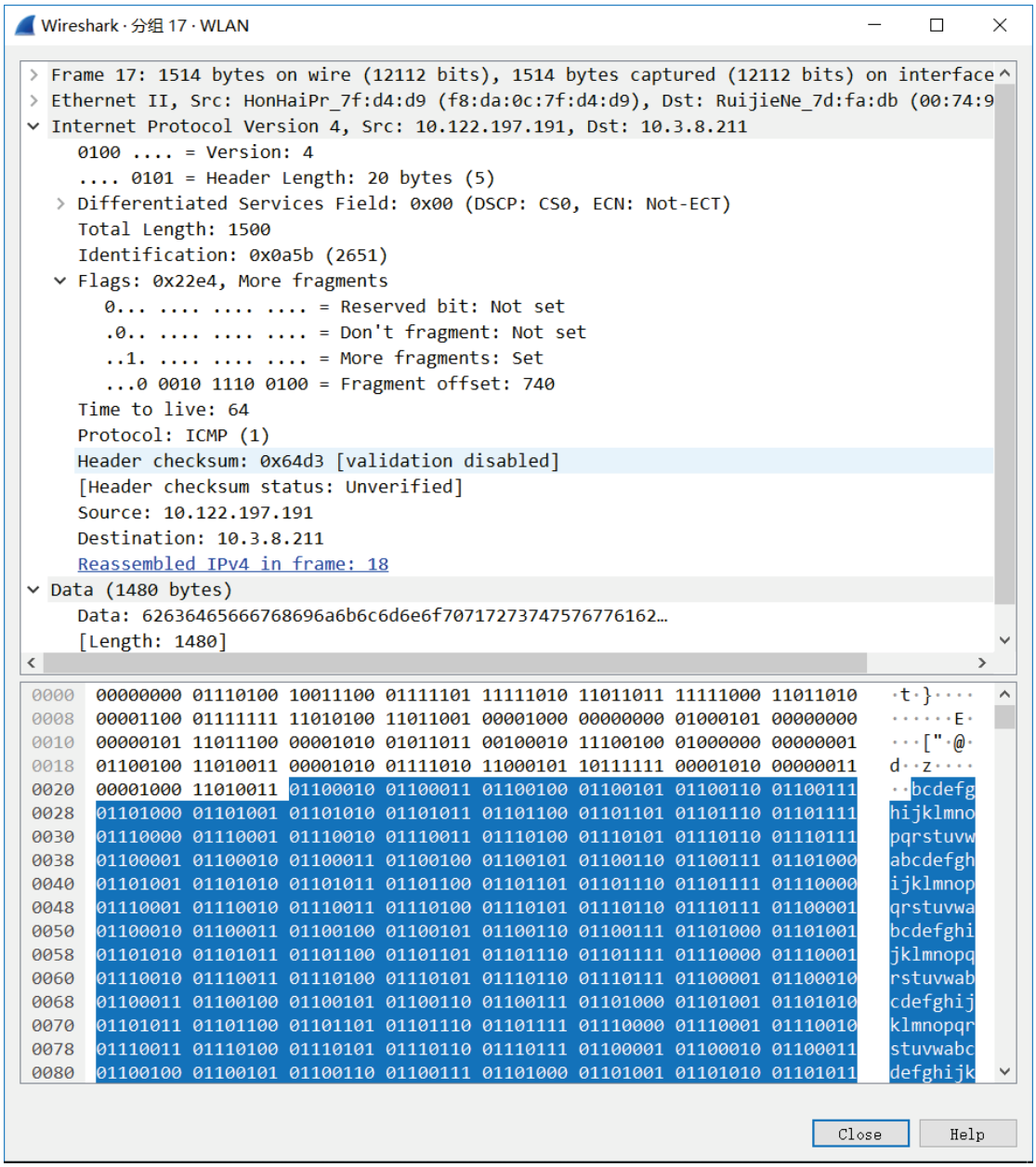
第三片：


Wireshark · 分组 14 · WLAN
Frame 14: 1514 bytes on wire (12112 bits), 1514 bytes captured (12112 bits) on interface 0
Ethernet II, Src: HonHaiPr_7f:d4:d9 (f8:da:0c:7f:d4:d9), Dst: RuijieNe_7d:fa:db (00:74:9c:
Internet Protocol Version 4, Src: 10.122.197.191, Dst: 10.3.8.211
0100 .... = Version: 4
.... 0101 = Header Length: 20 bytes (5)
Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
Total Length: 1500
Identification: 0x0a5b (2651)
Flags: 0x20b9, More fragments
0... .... = Reserved bit: Not set
.0.. .... = Don't fragment: Not set
..1. .... = More fragments: Set
...0 0000 1011 1001 = Fragment offset: 185
Time to live: 64
Protocol: ICMP (1)
Header checksum: 0x66fe [validation disabled]
[Header checksum status: Unverified]
Source: 10.122.197.191
Destination: 10.3.8.211
Reassembled IPv4 in frame: 18
Data (1480 bytes)
Data: 6162636465666768696a6b6c6d6e6f707172737475767761…
[Length: 1480]
0000 00000000 01110100 10011100 01111101 11111010 11011011 11111000 11011010 ·t·}····
0008 00001100 01111111 11010100 11011001 00001000 00000000 01000101 00000000 ·····E·
0010 00000101 11011100 00001010 01011011 00100000 10111001 01000000 00000001 ···[·@·
0018 01100110 11111110 00001010 01111010 11000101 10111111 00001010 00000011 f··z····
0020 00001000 11010011 01100001 01100010 01100011 01100100 01100101 01100110 ··abcdef
0028 01100111 01101000 01101001 01101010 01101011 01101100 01101101 01101110 ghijklmn
0030 01101111 01110000 01110001 01110010 01110011 01110100 01110101 01110110 opqrstuv
0038 01110111 01100001 01100010 01100011 01100100 01100101 01100110 01100111 wabcdefg
0040 01101000 01101001 01101010 01101011 01101100 01101101 01101110 01101111 hijklmno
0048 01110000 01110001 01110010 01110011 01110100 01110101 01110110 01110111 pqrstuvw
0050 01100001 01100010 01100011 01100100 01100101 01100110 01100111 01101000 abcdefgh
0058 01101001 01101010 01101011 01101100 01101101 01101110 01101111 01110000 ijklmnop
0060 01110001 01110010 01110011 01110100 01110101 01110110 01110111 01100001 qrstuvwa
0068 01100010 01100011 01100100 01100101 01100110 01100111 01101000 01101001 bcdefghi
0070 01101010 01101011 01101100 01101101 01101110 01101111 01110000 01110001 jklmnopq
0078 01110010 01110011 01110100 01110101 01110110 01110111 01100001 01100010 rstuvwab
Close
Help

第四片：

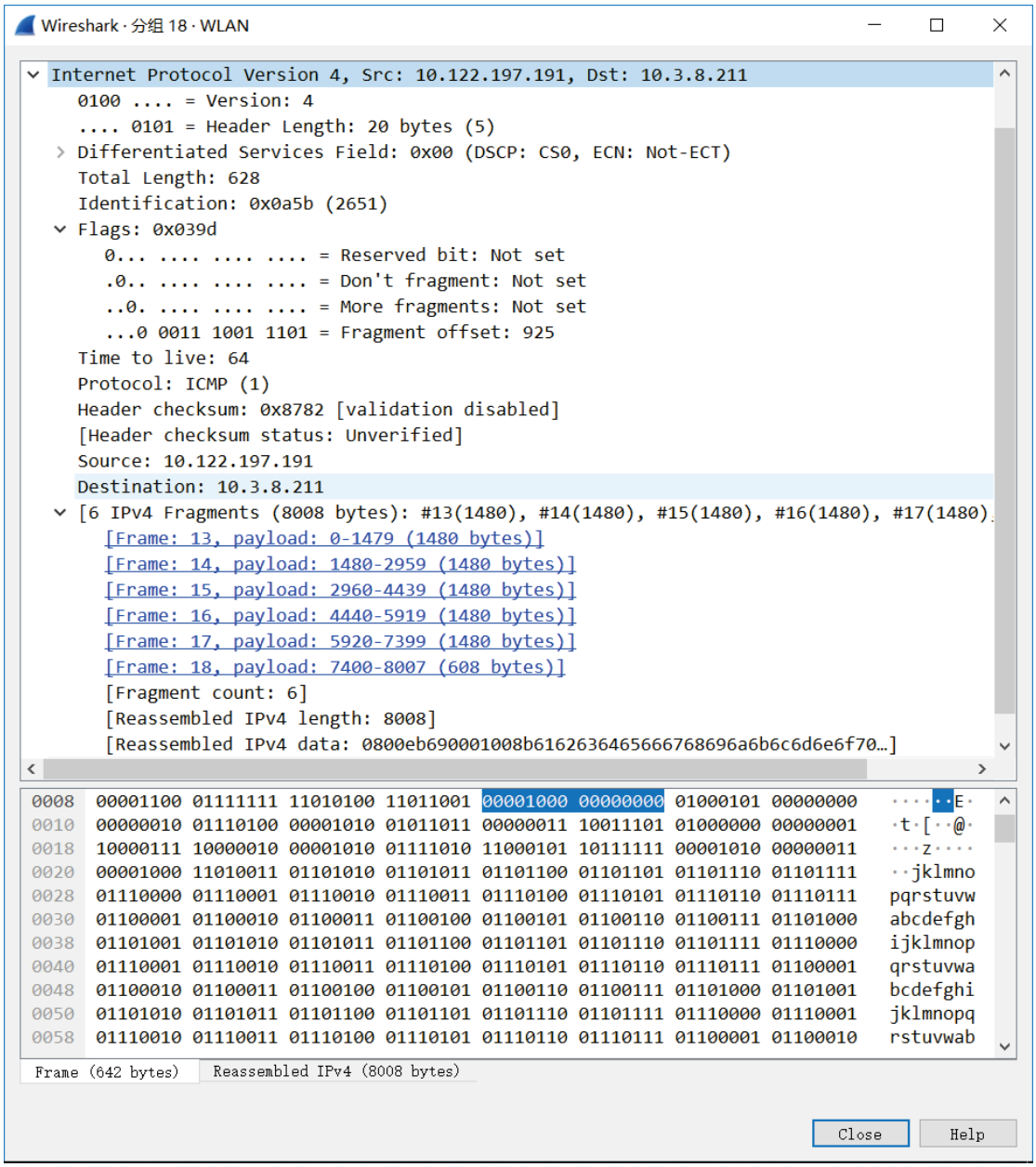

Wireshark · 分组 16 · WLAN
Frame 16: 1514 bytes on wire (12112 bits), 1514 bytes captured (12112 bits) on interface 0
Ethernet II, Src: HonHaiPr_7f:d4:d9 (f8:da:0c:7f:d4:d9), Dst: RuijieNe_7d:fa:db (00:74:9c:
Internet Protocol Version 4, Src: 10.122.197.191, Dst: 10.3.8.211
0100 .... = Version: 4
.... 0101 = Header Length: 20 bytes (5)
Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
Total Length: 1500
Identification: 0x0a5b (2651)
Flags: 0x222b, More fragments
0... .... = Reserved bit: Not set
.0.. .... = Don't fragment: Not set
..1. .... = More fragments: Set
...0 0010 0010 1011 = Fragment offset: 555
Time to live: 64
Protocol: ICMP (1)
Header checksum: 0x658c [validation disabled]
[Header checksum status: Unverified]
Source: 10.122.197.191
Destination: 10.3.8.211
Reassembled IPv4 in frame: 18
Data (1480 bytes)
Data: 71727374757677616263646566676869 6a6b6c6d6e6f7071…
[Length: 1480]
0000 00000000 01110100 10011100 01111101 11111010 11011011 11111000 11011010 ·t·}····
0008 00001100 01111111 11010100 11011001 00001000 00000000 01000101 00000000 ·····E·
0010 00000101 11011100 00001010 01011011 00100010 00101011 01000000 00000001 ···["+@·
0018 01100101 10001100 00001010 01111010 11000101 10111111 00001010 00000011 e··z····
0020 00001000 11010011 01110001 01110010 01110011 01110100 01110101 01110110 ··qrstuv
0028 01110111 01100001 01100010 01100011 01100100 01100101 01100110 01100111 wabcdefg
0030 01101000 01101001 01101010 01101011 01101100 01101101 01101110 01101111 hijklmno
0038 01110000 01110001 01110010 01110011 01110100 01110101 01110110 01110111 pqrstuvw
0040 01100001 01100010 01100011 01100100 01100101 01100110 01100111 01101000 abcdefgh
0048 01101001 01101010 01101011 01101100 01101101 01101110 01101111 01110000 ijklmnop
0050 01110001 01110010 01110011 01110100 01110101 01110110 01110111 01100001 qrstuvwa
0058 01100010 01100011 01100100 01100101 01100110 01100111 01101000 01101001 bcdefghi
0060 01101010 01101011 01101100 01101101 01101110 01101111 01110000 01110001 jklmnopq
0068 01110010 01110011 01110100 01110101 01110110 01110111 01100001 01100010 rstuvwab
0070 01100011 01100100 01100101 01100110 01100111 01101000 01101001 01101010 cdefghij
0078 01101011 01101100 01101101 01101110 01101111 01110000 01110001 01110010 klmnopqr
Close
Help

第五片：



3、

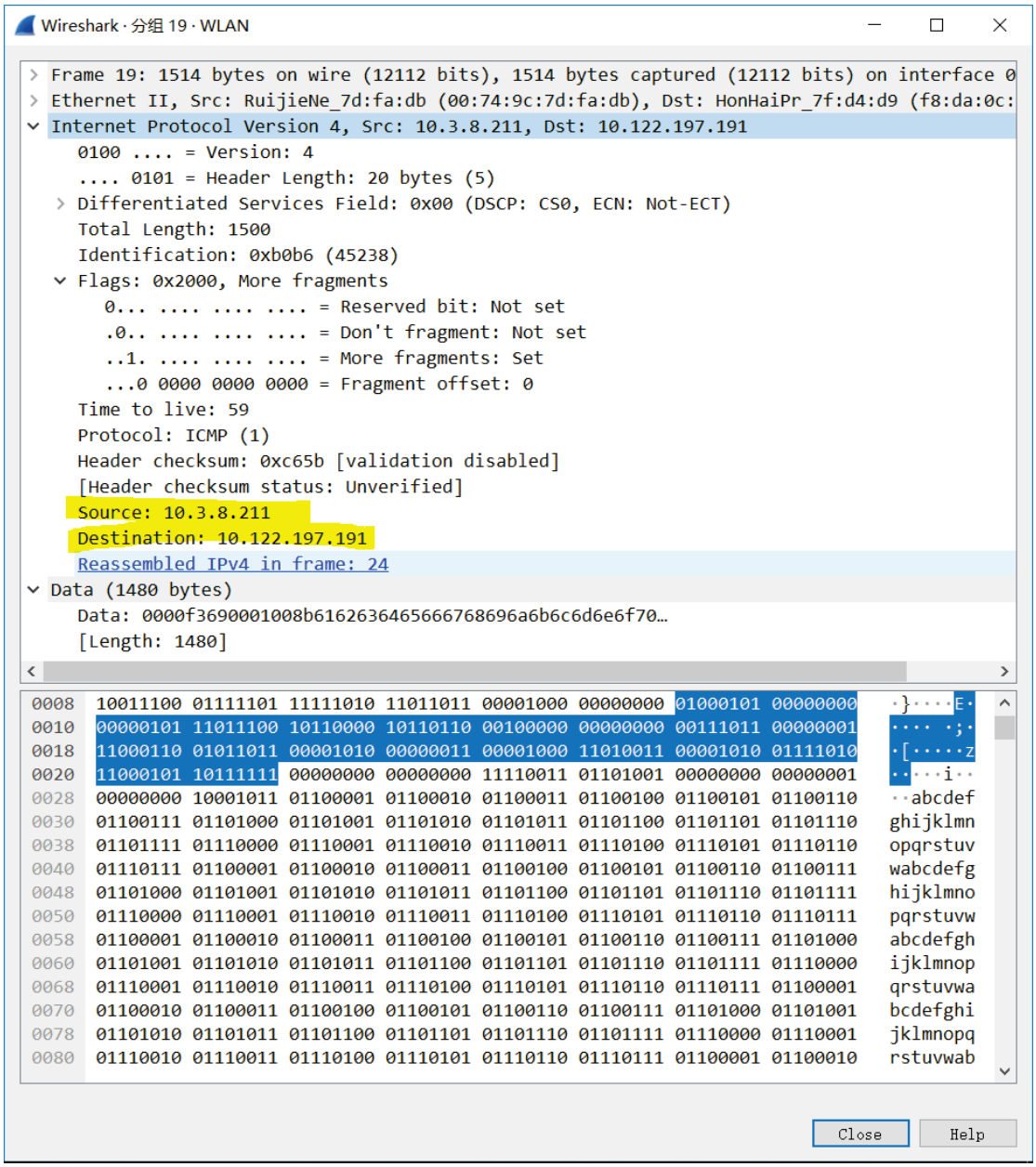
再取最后一片（在ICMP包中）进行分析：



与前五段不同的是在标志位字段DF=0。MF=0，说明该片是最后一片。Fragment offset=925，925*8=7400，刚好等于前面5片数据长度之和5*1480=7400。在后面的蓝字中也写明了分组发送情况，刚好是6组，共8008字节（其中有8字节是ICMP包，剩余8000字节是数据，刚好符合“ping -l 8000”发送8000字节的数据的命令）。

4、

应答部分的包与请求部分的包类似，主要区别是源地址和目的地址对换一下。



3. 实验心得

整个实验用了我大约一下午的时间，包括查询各种资料、进行实验和撰写实验报告。

在此次实验中，我遇到了很多的问题，首先是wireshark软件的使用，开始并不知道如何过滤协议，通过上网查询，对这个软件的使用也越来越熟练。其次是cmd命令，通过上网查询也解决了。还有就是对包的分析，开始比较生疏，但通过与同学讨论分析和上网查询，逐渐掌握了对数据包的分析方法。

我通过此次实验，对于各个协议有了更深的了解。通过对每种分组（ICMP分组，ARP分组，DHCP分组，IP数据分组）的分析，也对几个分组的内部结构有了一定的认识，把课堂上学到的知识应用到了实践之中，让我受益匪浅。