哈爾濱工業大學
Harbin Institute of Technology

计算机网络
课程实验报告

实验名称	利用 Wireshark 进行协议分析					
姓名	田一间		院系	计算机学院		
班级	1636101		学号	1160300617		
任课教师	李全龙		指导教师	李全龙		
实验地点	格物楼 213		实验时间	2018 年 11 月 17 日		
实验课表现	出勤、表现得分(10)		实验报告得分(40)		实验总分	
	操作结果得分(50)					
教师评语						

计算机科学与技术学院 SINCE 1956...
School of Computer Science and Technology

实验目的：

熟悉并掌握 Wireshark 的基本操作，了解网络协议实体间进行交互以及报文交换的情况。

实验内容：

1) 学习 Wireshark 的使用
2) 利用 Wireshark 分析 HTTP 协议
3) 利用 Wireshark 分析 TCP 协议
4) 利用 Wireshark 分析 IP 协议
5) 利用 Wireshark 分析 Ethernet 数据帧

选做内容：

a) 利用 Wireshark 分析 DNS 协议
b) 利用 Wireshark 分析 UDP 协议
c) 利用 Wireshark 分析 ARP 协议

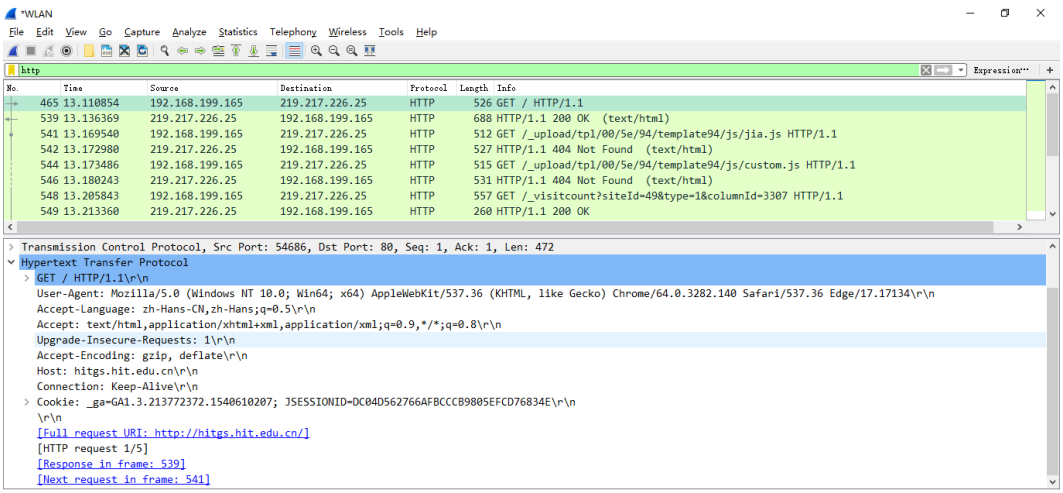
实验过程：

以文字描述、实验结果截图等形式阐述实验过程，必要时可附相应的代码截图或以附件形式提交。

（一） HTTP分析

1） HTTP GET/response 交互

输入 http://hitgs.hit.edu.cn/ ， Wireshark 抓包情况如下：

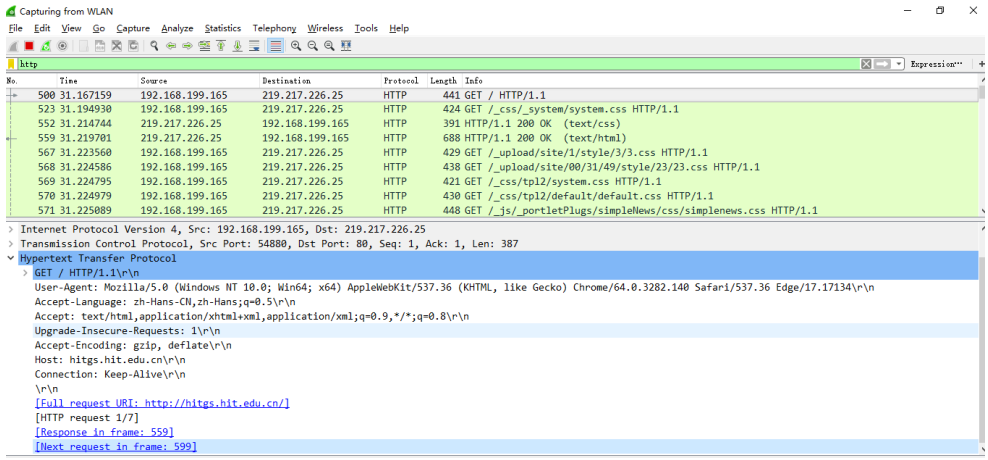


问题解答：

- 浏览器运行 HTTP1.1，服务器运行 HTTP1.1
- Accept-Language: zh-Hans-CN,zh-Hans;q=0.5
- 计算机 IP：192.168.199.165，服务器 IP：219.217.226.25
- 服务器返回状态码：HTTP/1.1 200 OK

2） HTTP 条件GET/response 交互

清除浏览器缓存，输入 http://hitgs.hit.edu.cn/



问题解答：

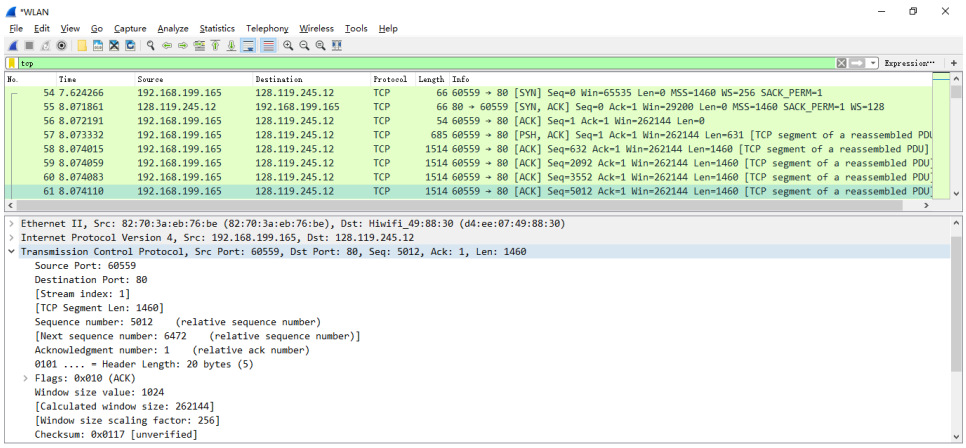
- 第一个 HTTP GET 请求无 IF-MODIFIED-SINCE 字段
- 服务器明确返回了文件内容，从 Line-based text data: text/html (753 lines) 内容可以看出
- 较晚的“HTTP GET”请求有 IF-MODIFIED-SINCE，后面跟的信息是时间，如：If-Modified-Since: Tue, 17 Apr 2018 07:06:42 GMT\r\n
- 服务器对较晚的 HTTP GET 请求的响应中 HTTP 转态码为 HTTP/1.1 304 Not Modified\r\n，没有明确返回文件内容。

  解释：浏览器第一次访问后对该内容进行缓存，再次访问则先发送 IF-MODIFIED-SINCE，询问服务器该内容是否是最新，服务器根据修改时间确定是最新的内容，返回 304 Not Modified。

（二） TCP分析

1） 俘获大量的由本地主机到远程服务器的 TCP 分组

2） 浏览追踪信息



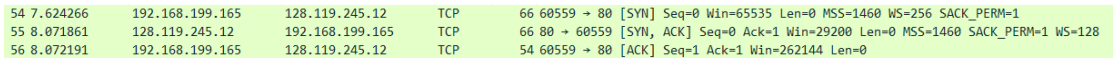
问题解答：

- 客户端主机的 IP 地址是 192.168.199.165，TCP 端口号是 60559
- Gaia.cs.umass.edu 服务器的 IP 地址是 128.119.245.12。对这一连接，它用来发送和接收 TCP 报文的端口号是 80。

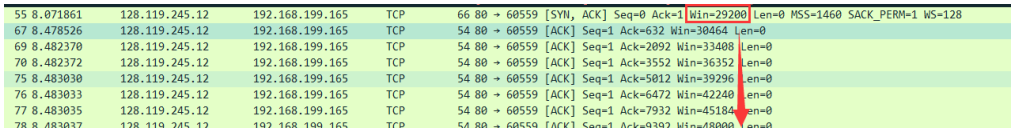
3） TCP基础

问题解答：

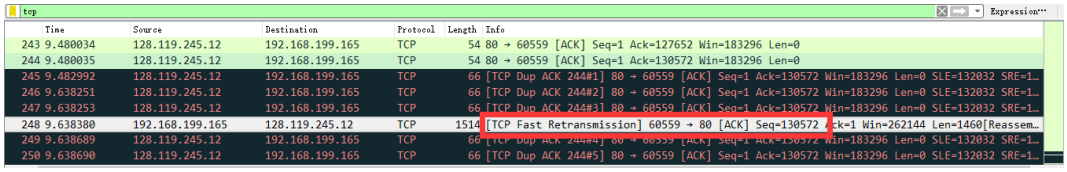
- 客户服务器之间用于初始化 TCP 连接的 TCP SYN 报文段的序号（sequence number）是 0，在该报文段中，是用 Flags: 0x002 (SYN)来标示该报文段是 SYN 报文段的。
- 服务器向客户端发送的 SYNACK 报文段序号是，该报文段中，Acknowledgement 字段的值是 1。Gaia.cs.umass.edu 服务器决定此值方法：客户服务器之间初始化 TCP 连接的报文段序号 0+1=1。在该报文段中，是用 Flags: 0x012 (SYN, ACK)来标示该报文段是 SYNACK 报文段的。
- 从捕获的数据包中分析出 tcp 三次握手过程：
  客户端发送 SYN 报文，服务器回 SYNACK 报文，客户端回复 ACK 报文，握手完成。



- 包含 HTTP POST 命令的 TCP 报文段的序号是 1。
- 如果将包含 HTTP POST 命令的 TCP 报文段看作是 TCP 连接上的第一个报文段，那么该 TCP 连接上的第六个报文段的序号是 6472。
  是何时发送的： Nov 11, 2018 11:30:09.106464000 中国标准时间。该报文段所对应的 ACK 是何时接收的： Nov 11, 2018 11:30:09.511658000 中国标准时间。
- 前六个 TCP 报文段的长度各是多少：包含头部：651(631+20)，1480(1460+20)，1480，1480,1480，1480。
- 在整个跟踪过程中，接收端公示的最小的可用缓存空间是多少？限制发送端的传输以后，接收端的缓存是否仍然不够用？
  答：最小的为 29200，接收端窗口一直在增大，缓存够用，并没抑制发送方速率。

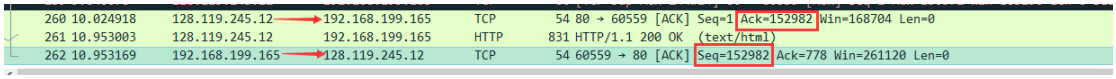


- 在跟踪文件中是否有重传的报文段？进行判断的依据是什么？
  报文段 Seq=130572 发生了快速重传，该报文段序列号重复出现，且标识为快速重传。截图如下：



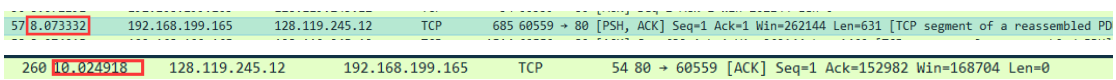
- TCP 连接的 throughput (bytes transferred per unit time)是多少？请写出你的计算过程。

传输数据为：152982 - 1 = 152981 bytes



所用时间为：

10.024918 - 8.073332 = 1.951586s



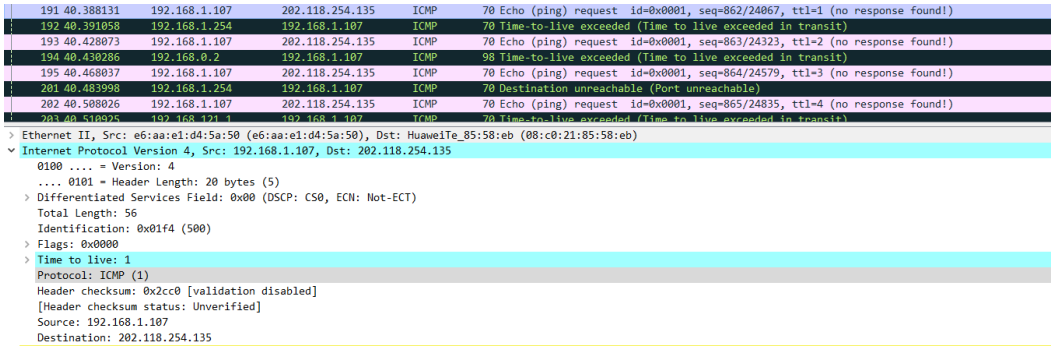
吞吐量：152981/1.951586 = 78.388Kbps

（三） IP分析

A. 通过执行 traceroute 执行捕获数据包

B. 对捕获的数据包进行分析

1） 选择第一个你的主机发出的ICMP Echo Request消息，在packet details窗口展开数据包的Internet Protocol部分。

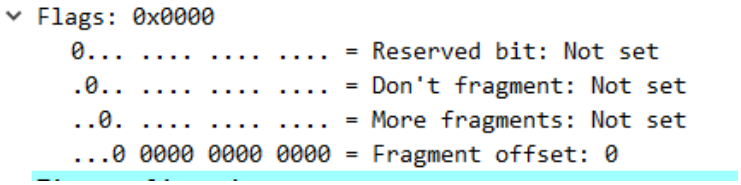


问题解答：

- 主机 IP 地址：192.168.1.107
- 在 IP 数据包头中，上层协议（upper layer）字段的值是：1
- IP 头有多少字节？该 IP 数据包的净载为多少字节？并解释你是怎样确定该 IP 数据包的净载大小的

  IP 头：20 字节，净载荷：36 字节。净载荷=总长度 56-IP 头 20=36 字节。
- 该 IP 数据包分片了吗？解释你是如何确定该 P 数据包是否进行了分片

  没有分片，Flags 字段全为 0



2）单击Source列按钮，这样将对捕获的数据包按源IP地址排序。选择第一个你的主机发出的ICMP Echo Request消息，在packet details窗口展开数据包的Internet Protocol部分。

问题解答：

- 你主机发出的一系列 ICMP 消息中 IP 数据报中哪些字段总是发生改变？

  Identification、Time to live、Header checksum
- 哪些字段必须保持常量？哪些字段必须改变？为什么？

  常量：Version、Source、Destination

  改变：Identification、Time to live、Header checksum

  原因：id 是 IP 报文的唯一 id，traceRoute 每次 TTL 加一，checksum 会重新计算
- 描述你看到的 IP 数据包 Identification 字段值的形式

  每一个 IP 数据包的标志位都不一样，且每次都加 1.

3） 找到由最近的路由器（第一跳）返回给你主机的ICMP Time-to-live exceeded消息。

问题解答：

- Identification 字段和 TTL 字段的值是什么？

  Identification: 0xe74b (59211)、Time to live: 255

- 最近的路由器（第一跳）返回给你主机的 ICMP Time-to-live exceeded 消息中这些值是否保持不变？为什么？

  Identification 字段变化，因为其是数据报的唯一标识

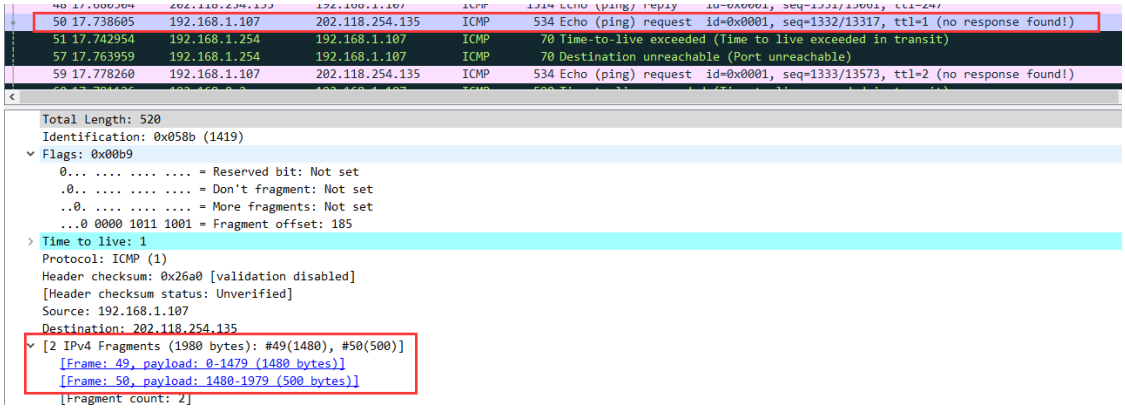
  TTL 值不变，因为路由器是固定的，其捕获的 ttl 值也是固定的。

4） 单击Time列按钮，这样将对捕获的数据包按时间排序。找到在将包大小改为2000字节后你的主机发送的第一个ICMP Echo Request消息。

问题解答：

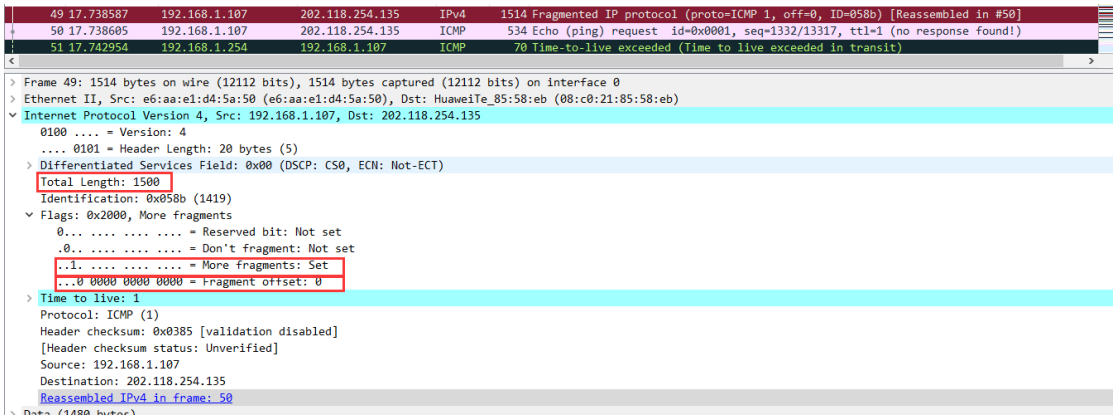
- 该消息是否被分解成不止一个IP数据报？

  答：是的，被分成了两个数据包



- 观察第一个IP分片，IP头部的哪些信息表明数据包被进行了分片？IP头部的哪些信息表明数据包是第一个而不是最后一个分片？该分片的长度是多少

  答：MF为1表明被进行了分片，片偏移Fragment offset: 0表明数据包是第一个分片，该分片长度为1500bytes。

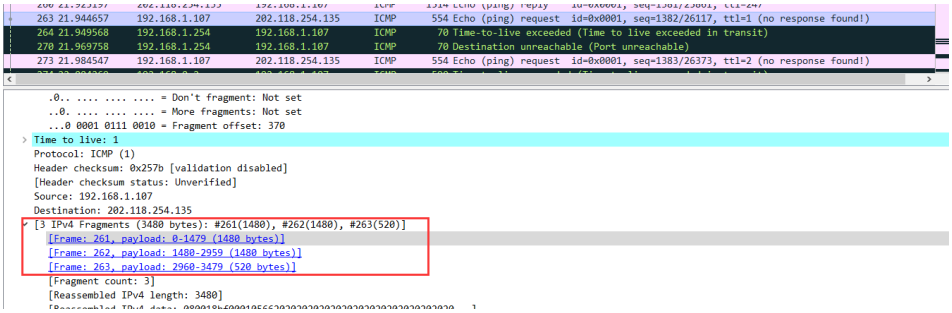


C. 找到在将包大小改为3500字节后你的主机发送的第一个ICMP Echo Request消息。

问题解答：

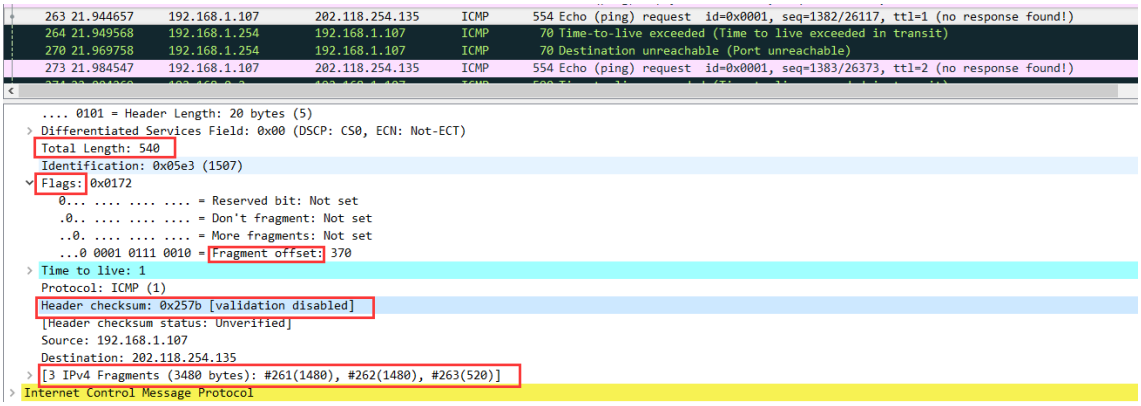
- 原始数据包被分成了多少片？

  三片



- 这些分片中IP数据报头部哪些字段发生了变化？

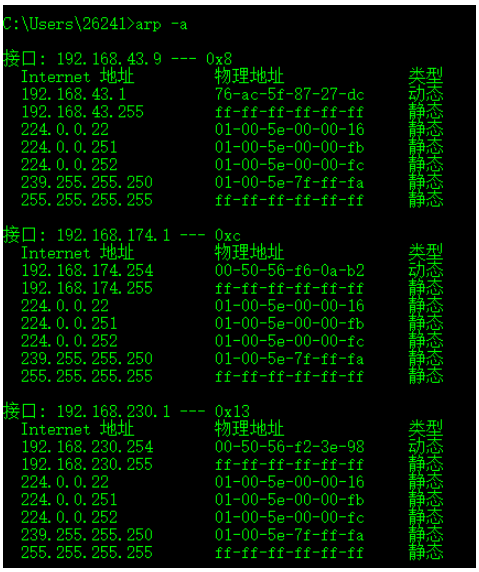
  Total Length、Flags、 Fragment offset、Header checksum



（四） 抓取ARP数据包

（1）利用 MS-DOS 命令：arp 或 c:\windows\system32\arp 查看主机上 ARP 缓存的内容。说明 ARP 缓存中每一列的含义是什么?

答：ARP缓存如下，三列分别为IP地址、MAC地址以及类型。



（2）清除主机上 ARP 缓存的内容,抓取 ping 命令时的数据包。分析数据包,回答下面的问题：

问题解答：

- ARP 数据包的格式是怎样的？由几部分构成，各个部分所占的字节数是多少？

图 6-11 ARP 请求和应答的分组格式

```
Address Resolution Protocol (request)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (1)
    Sender MAC address: 76:ac:5f:87:27:dc (76:ac:5f:87:27:dc)
    Sender IP address: 192.168.43.1
    Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)
    Target IP address: 192.168.43.9
```

- 如何判断一个 ARP 数据是请求包还是应答包？

  答：通过 OP 字段，当其值为 0x0001 时是请求，为 0x0002 时是应答。

```
Address Resolution Protocol (request)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (1)
    Sender MAC address: 76:ac:5f:87:27:dc (76:ac:5f:87:27:dc)
    Sender IP address: 192.168.43.1
    Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)
    Target IP address: 192.168.43.9
```

```
0000  72 ce 99 49 e7 ff 76 ac  5f 87 27 dc 08 06 00 01   r··I··v· _·'·····
0010  08 00 06 04 00 01 76 ac  5f 87 27 dc c0 a8 2b 01   ······v· _·'···+·
0020  00 00 00 00 00 00 c0 a8  2b 09                     ········ +·
```

```
Address Resolution Protocol (reply)
    Hardware type: Ethernet (1)
    Protocol type: IPv4 (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: reply (2)
    Sender MAC address: 72:ce:99:49:e7:ff (72:ce:99:49:e7:ff)
    Sender IP address: 192.168.43.9
    Target MAC address: 76:ac:5f:87:27:dc (76:ac:5f:87:27:dc)
    Target IP address: 192.168.43.1
```

```
0000  76 ac 5f 87 27 dc 72 ce  99 49 e7 ff 08 06 00 01   v·_·'·r· ·I·····
0010  08 00 06 04 00 02 72 ce  99 49 e7 ff c0 a8 2b 09   ······r· ·I····+·
0020  76 ac 5f 87 27 dc c0 a8  2b 01                     v·_·'··· +·
```

- 为什么 ARP 查询要在广播帧中传送，而 ARP 响应要在一个有着明确目的局域网地址的帧中传送？

  答：由于刚开始查询方并不知道目的主机的 MAC 地址，因此需要广播查询。

  而ARP响应时，目的主机获取到查询主机的MAC地址以及IP信息，可以在有明确目的的局域网地址的帧中传送。

（五） 抓取UDP数据包

问题解答：

- 消息是基于 UDP 的还是 TCP 的？

  答：UDP

- 你的主机 ip 地址是什么？目的主机 ip 地址是什么？

  答：主机 ip：192.168.43.9　目的主机 ip：123.151.78.14

8	13.239285	123.151.78.14	192.168.43.9	OICQ	121 OICQ Protocol
9	24.672932	192.168.43.9	123.151.78.14	UDP	209 4002 → 8000 Len=167
10	24.814747	123.151.78.14	192.168.43.9	UDP	73 8000 → 4002 Len=31
14	28.965404	192.168.43.9	123.151.78.14	UDP	881 4002 → 8000 Len=839

- 你的主机发送 QQ 消息的端口号和 QQ 服务器的端口号分别是多少？

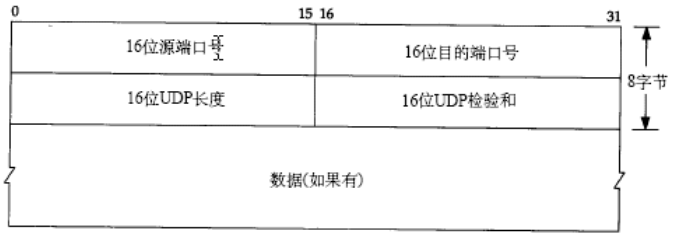
  答：发送端口：4002　服务器端口号：8000

```
    9 24.672932     192.168.43.9          123.151.78.14         UDP      209 4002 → 8000 Len=167

Frame 9: 209 bytes on wire (1672 bits), 209 bytes captured (1672 bits) on interface 0
Ethernet II, Src: 72:ce:99:49:e7:ff (72:ce:99:49:e7:ff), Dst: 76:ac:5f:87:27:dc (76:ac:5f:87:27:dc)
Internet Protocol Version 4, Src: 192.168.43.9, Dst: 123.151.78.14
User Datagram Protocol, Src Port: 4002, Dst Port: 8000
Data (167 bytes)
```

- 数据报的格式是什么样的？都包含哪些字段，分别占多少字节？

  答：格式：




```
User Datagram Protocol, Src Port: 4002, Dst Port: 8000
    Source Port: 4002
    Destination Port: 8000
    Length: 175          175 = 167+8
    Checksum: 0x5b8e [unverified]
    [Checksum Status: Unverified]
    [Stream index: 0]
Data (167 bytes)
```

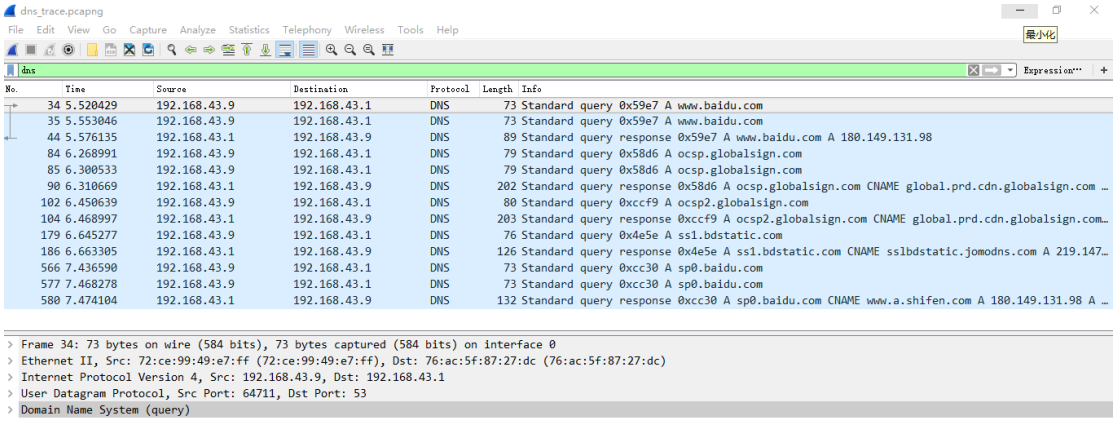
- 为什么你发送一个 ICQ 数据包后，服务器又返回给你的主机一个 ICQ 数据包？这 UDP 的不可靠数据传输有什么联系？对比前面的 TCP 协议分析，你能看出 UDP 是无连接的吗？

  答：因为 UDP 是不可靠数据传输，因此当你发送数据包之后，服务器返回一个数据包，以确认对方接收到了消息，在一定程度上实现可靠的数据传输。

  可以看出 UDP 是无连接的。TCP 在传输数据之前，需要先经历三次握手阶段，而 UDP 无此过程，

（六） 利用 WireShark 进行 DNS 协议分析

打开浏览器键入:www.baidu.com。打开 Wireshark，启动抓包。在控制台回车执行完毕后停止抓包。Wireshark 捕获的 DNS 报文如图所示：



心得体会：

结合实验过程和结果给出实验的体会和收获。

本次实验利用了Wireshark抓包工具分析HTTP、TCP、IP、以太网帧、ARP、DNS等数据包情况，了解网络协议实体间进行交互以及报文交换的情况。

以前的学习都停留在理论上，而通过这次实验，自己对于很多协议的处理过程都有了更为直观的认识，比如HTTP的GET交互，TCP的三次握手，IP数据报的分片，ARP协议获取MAC地址等。

整个实验细节十分多，尤其是TCP和IP部分的分析，自己也花费了很大的功夫。不知不觉间，已经是最后一个实验了，计网的四次实验真的让自己受益匪浅，在这里感谢老师无私的教导，也感谢助教一直的辛劳，谢谢你们。