**实验 5：利用 Wireshark 进行协议分析**

|  |  |
| --- | --- |
| **学号：** | 1130310128 |
| **姓名：** | 杨尚斌 |
| **专业：** | 计算机科学与技术 |
| **指导老师：** | 聂兰顺 |

1. **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。

1. **实验环境**

* 接入 Internet 的实验主机
* Windows 10
* Wireshark

1. **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：**

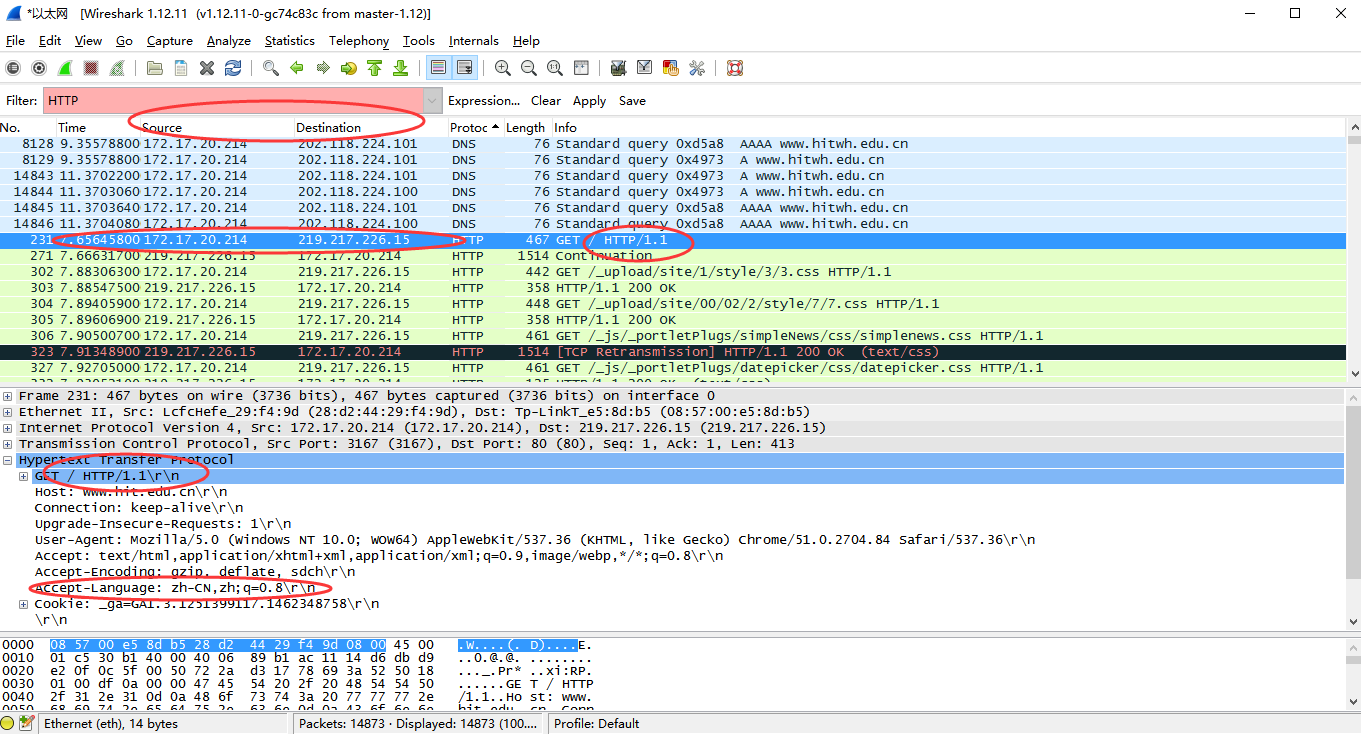
a) 利用 Wireshark 分析 DNS 协议

b) 利用 Wireshark 分析 UDP 协议

c) 利用 Wireshark 分析 ARP 协议

1. **实验过程**
2. **HTTP 分析**

**HTTP GET/response 交互**



**问题：**

* + **″你的浏览器运行的是 HTTP1.0，还是 HTTP1.1？你所访问的服务器所运行 HTTP 协议的版本号是多少？**

都是HTTP 1.1

* + **你的浏览器向服务器指出它能接收何种语言版本的对象？**

zh-cn

* + **你的计算机的 IP 地址是什么？服务器 http://www.hit.edu.cn 的 IP 地址是多少？**

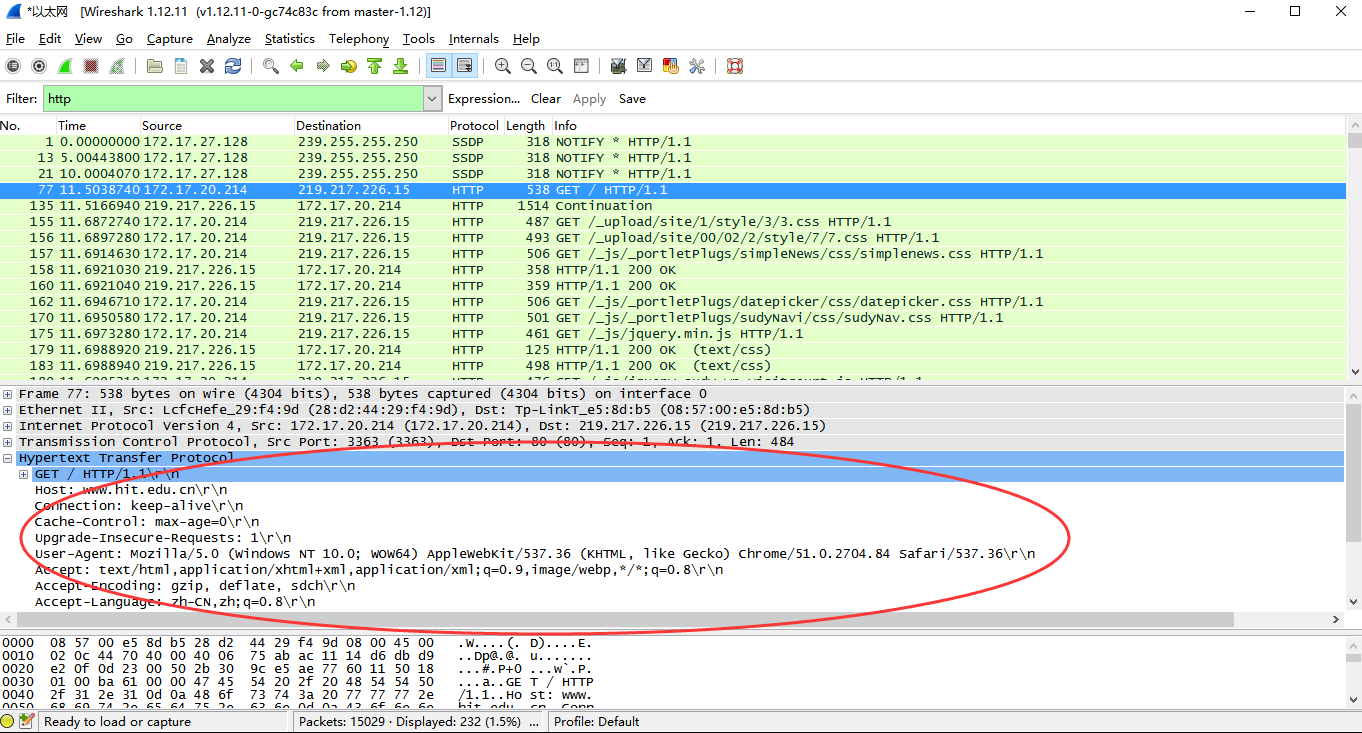
本地计算机IP：172.17.20.214

服务器 IP：219.217.226.15

* + **从服务器向你的浏览器返回的状态代码是多少？**

200

**HTTP GET/response 交互**



* + **分析你的浏览器向服务器发出的第一个 HTTP GET 请求的内容， 在该请求报文中，是否有一行是：IF-MODIFIED-SINCE？**

没有

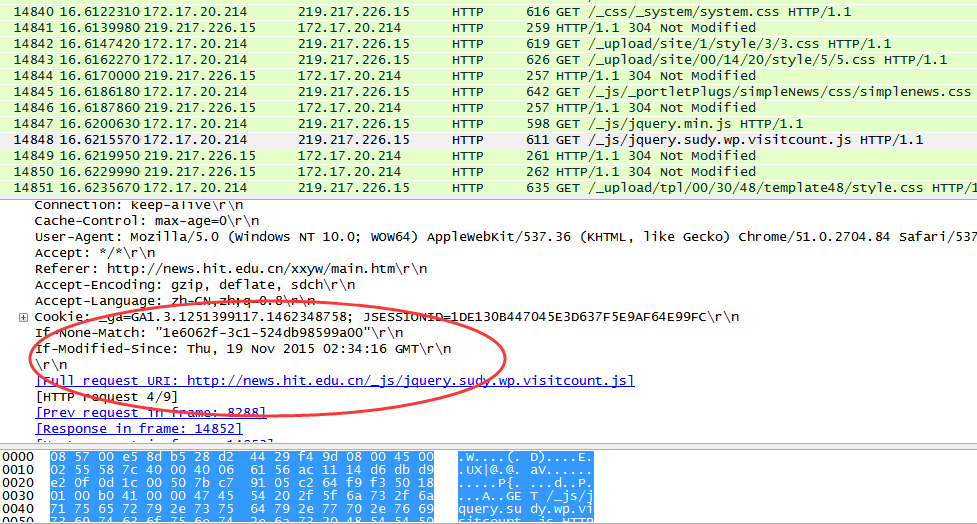
* + **分析服务器响应报文的内容，服务器是否明确返回了文件的内容？如何获知？**

明确返回了内容

HTTP Status Code 为 304时不返回文件

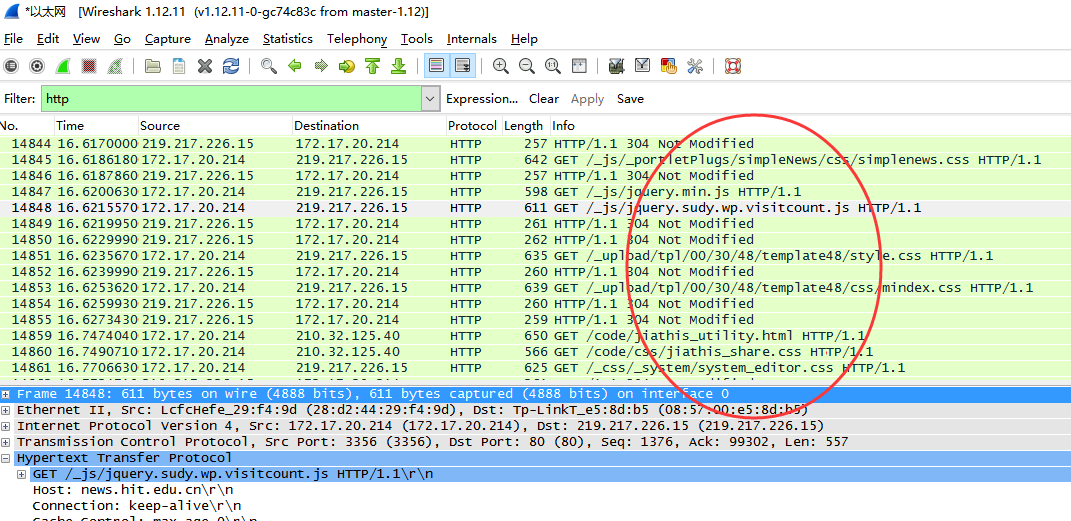
HTTP Status Code 为 200 时返回文件

* + **分析你的浏览器向服务器发出的较晚的“HTTP GET”请求，在该请 求报文中是否有一行是：IF-MODIFIED-SINCE？如果有，在该首 部行后面跟着的信息是什么？**



有，代表浏览器有缓存，后面代表时间，即服务器在这个事件之后是否有更新

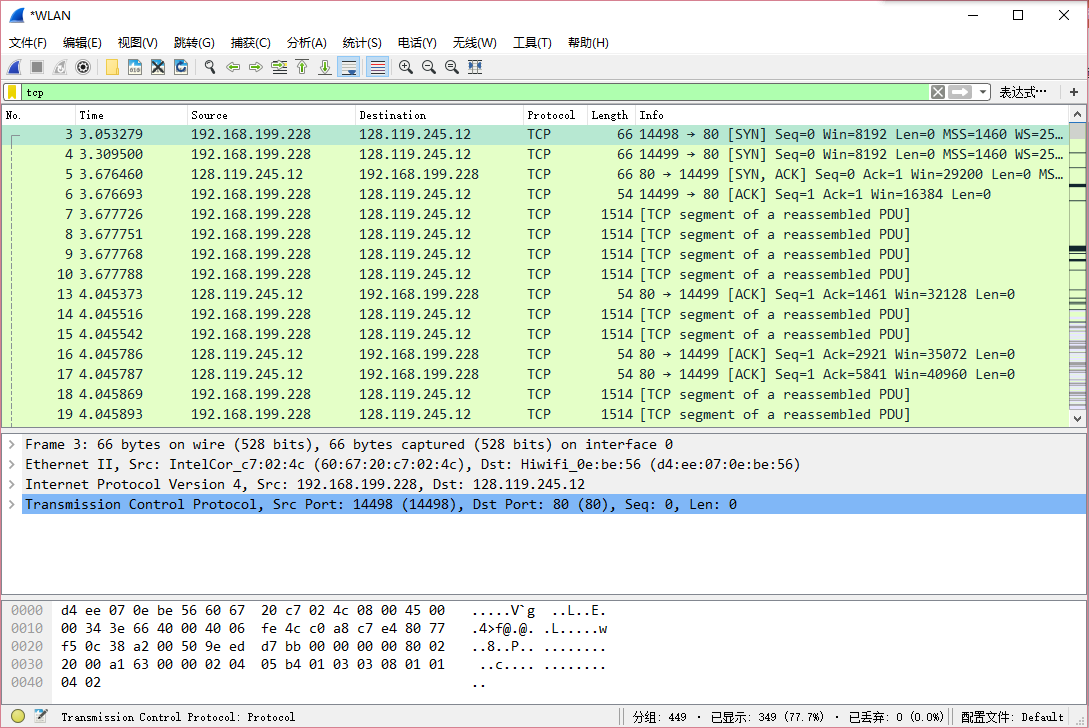
* + **服务器对较晚的 HTTP GET 请求的响应中的 HTTP 状态代码是多 少？服务器是否明确返回了文件的内容？请解释。**



304, 不会返回明确文件，使用没有过期的缓存文件

1. **TCP 分析**

**报文捕获:**



1. **向 gaia.cs.umass.edu服务器传送文件的客户端主机的 IP 地址和TCP端口号是多少？**

192.168.199.228

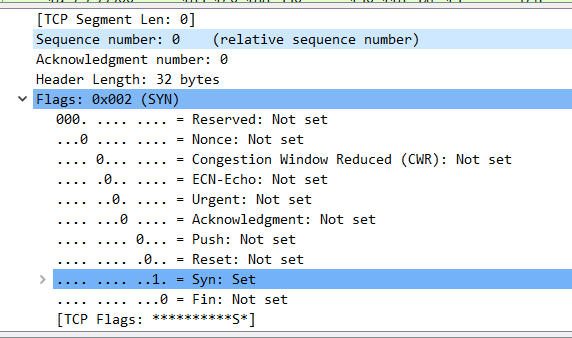
14498

**2) Gaia.cs.umass.edu 服务器的 IP 地址是多少？对这一连接，它用来发送和接收 TCP 报文的端口号是多少？**

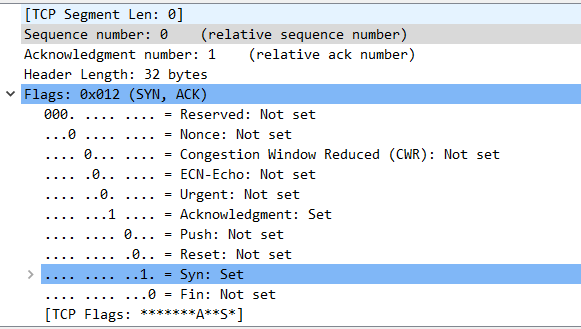
128.119.245.12

80

**3)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）是多少？在该报文段中，是用什么来标示该报文段是 SYN 报文段的？**



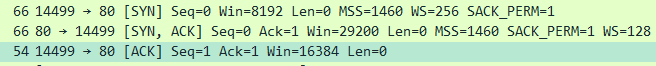
**4) 服务器向客户端发送的 SYN ACK 报文段序号是多少？该报文段 中，Acknowledgement 字段的值是多少？Gaia.cs.umass.edu 服务器 是如何决定此值的？在该报文段中，是用什么来标示该报文段是 SYNACK 报文段的？**



ack是根据上一次客户端发给服务器的seq+1得到的

1. **你能从捕获的数据包中分析出 tcp 三次握手过程吗？**



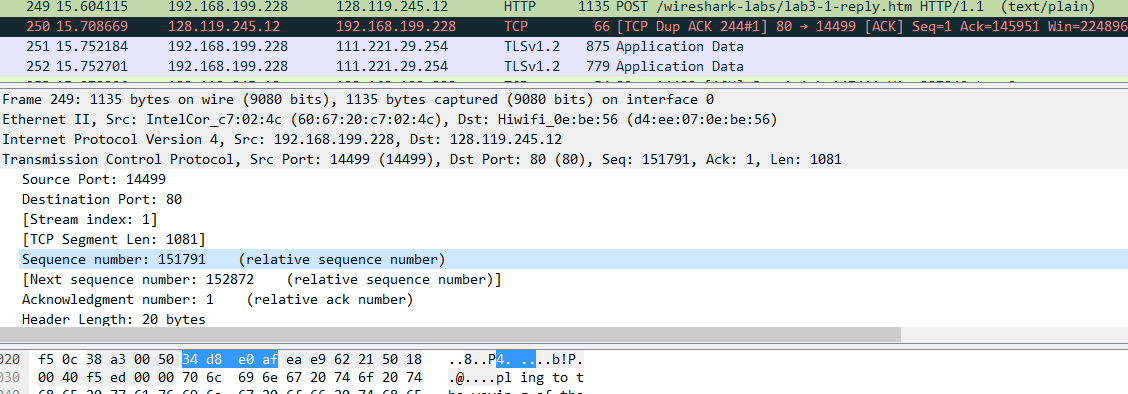


首先客户端向服务器发送seq=0的建立连接的请求

然后服务器向客户端返回seq=0,ack=0+1=1的响应

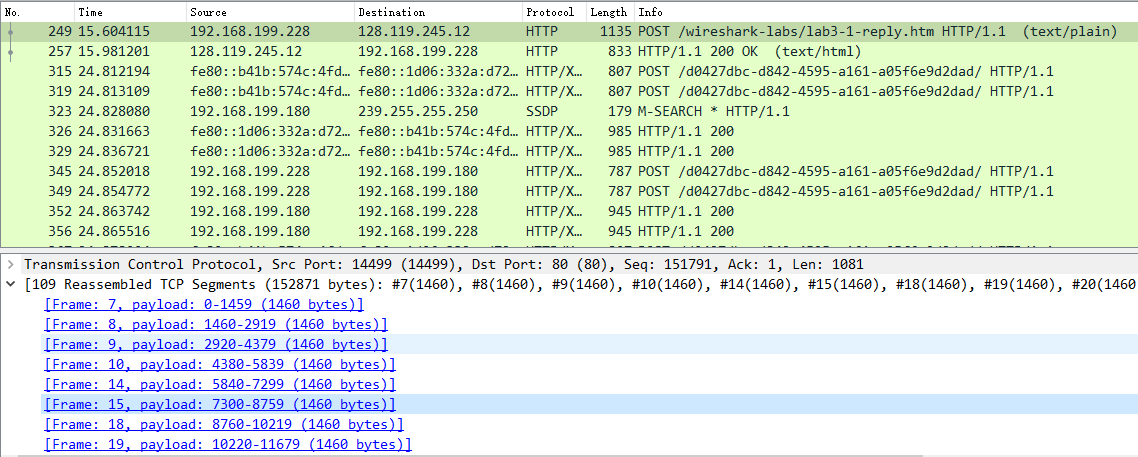
客户端收到响应，返回seq=1,ack=0+1=1的确认报文，连接建立

1. **包含 HTTP POST 命令的 TCP 报文段的序号是多少？**

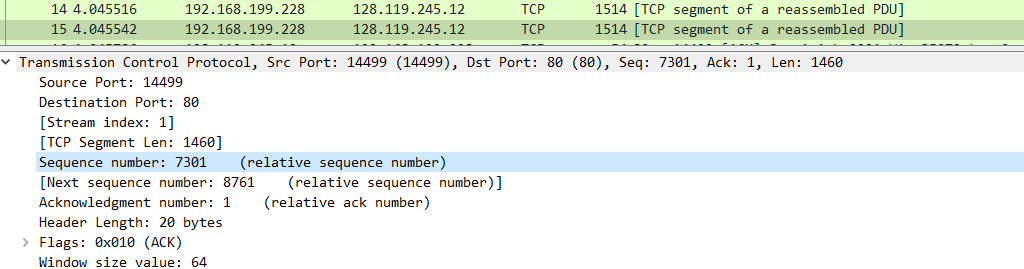


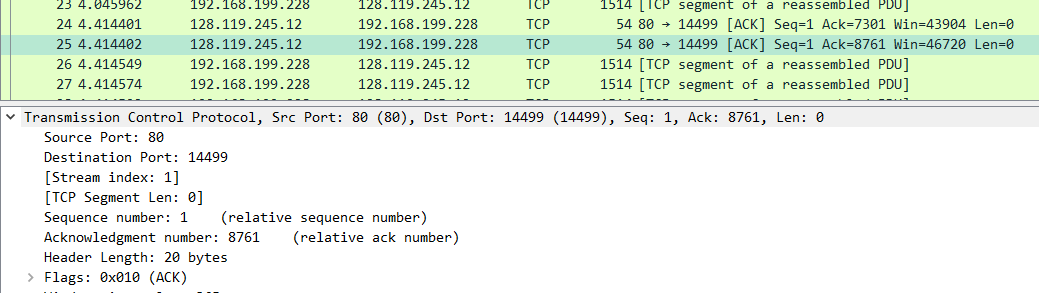
Seq=151791

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多 少？是何时发送的？该报文段所对应的 ACK 是何时接收的？**



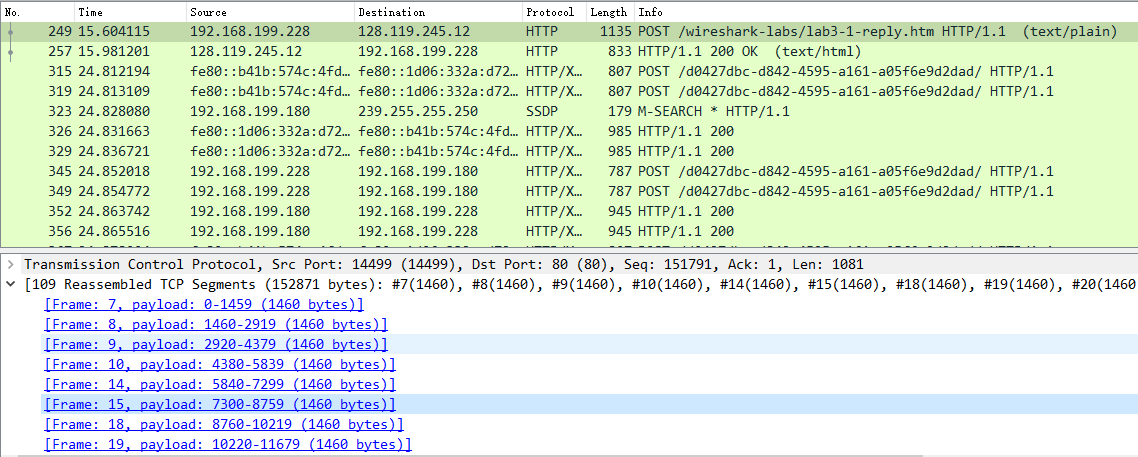
第六个报文段Seq=7301,在http post发送之前，tcp连接建立之后发送。





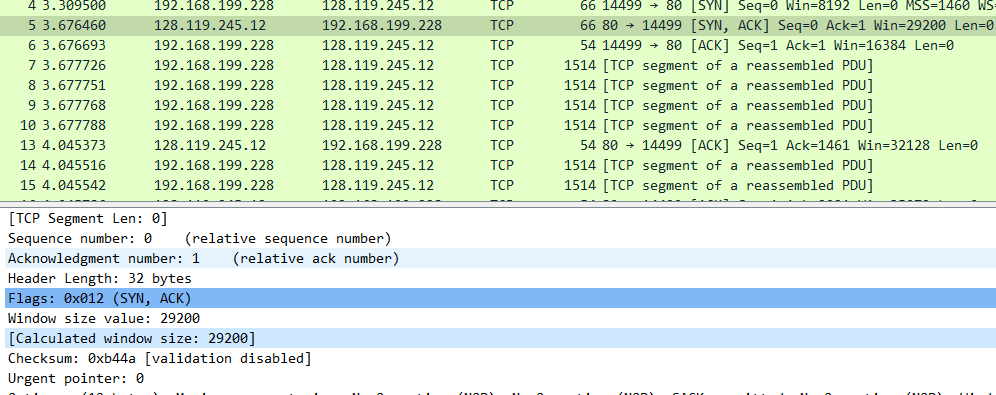
对应的ack即为服务器返回的第六个ack。

1. **前六个 TCP 报文段的长度各是多少？**



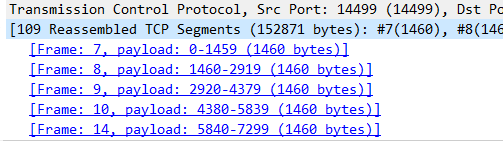
1. **在整个跟踪过程中，接收端公示的最小的可用缓存空间是多少？ 限制发送端的传输以后，接收端的缓存是否仍然不够用？**

如图，接收端公示的最小的可用缓存空间是29200，该窗口大小会一直增加，所以不会出现接收端的缓存是否仍然不够用的情况。



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

没有出现重传，因为客户端发送的报文序列号没有出现重复。



由图可知，发送数据总的长度为152871B+109\*54B=158757B

发送时间间隔约为1.673847s

因此吞吐量为158757B/1.673847s=94845.59bps

1. **IP 分析**

A.对捕获的数据包进行分析

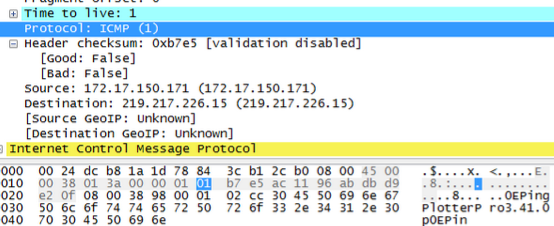
1.在你的捕获窗口中，应该能看到由你的主机发出的一系列ICMP Echo Request包和中间路由器返回的一系列ICMP TTL-exceeded消息。选 择第一个你的主机发出的ICMP Echo Request消息，在packet details窗口 展开数据包的Internet Protocol部分

**1)你主机的IP地址是什么？**



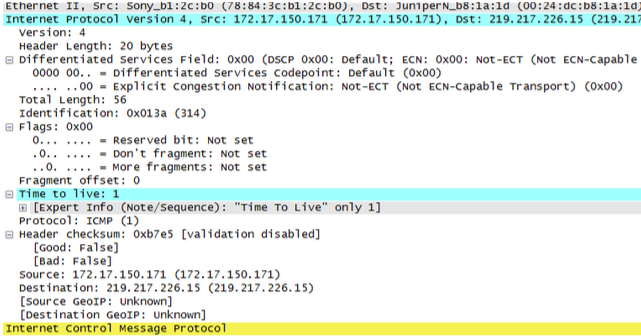
主机IP地址:172.17.150.171

1. **在IP数据包头中，上层协议（upper layer）字段的值是什么？**



上层协议字段的值是01。

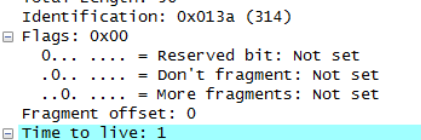
1. **IP头有多少字节？该IP数据包的净载为多少字节？并解释你是怎样确定该IP数据包的净载大小的？**



IP头有20字节。

IP包的净载为Total Length-Header Length=56B-20B=36B

**4)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，分片位移为0，More fragments为0表示后面无分段。

**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。在“listing of captured packets”窗口，你会看到许多后续的ICMP消息（或许还有你主机上运行的其他协议的数据包）**

1. **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？**

查看多个ICMP消息，发现ID、TTL、Header checksun这三个字段总在变化。

1. **哪些字段必须保持常量？哪些字段必须改变？为什么？**

ID必须改变：鉴别码，⽤于区分不同的数据包；

TTL必须改变：来⾃于traceroute的要求，⽤来测试路径上的路由信息；

Header Checksum必须改变：⾸部校验和，前⾯的字段改变，该值也跟着改变；

除以上外的字段保持常量。

1. **描述你看到的IP数据包Identification字段值的形式。**

16位，在某⼀范围内是+1递增的。

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

思考下列问题：

 **Identification字段和TTL字段的值是什么？ 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？**

不变， IP是⽆连接服务，相同的标识是为了分段后组装成同⼀段，给同⼀个主机返回的ICMP，

标识不代表序号， TTL消息是相同的，因此Identification不变;因为是第⼀跳路由器发回的数据报，故TTL是最⼤值减1，总是等于254。

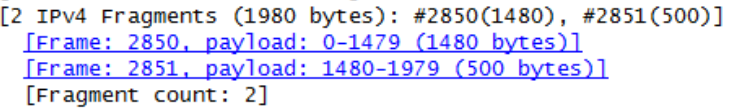
**（ 4）单击Time列按钮，这样将对捕获的数据包按时间排序。找到在将包大小改为2000字节后你的主机发送的第一个ICMP Echo Request消息。**

思考下列问题：

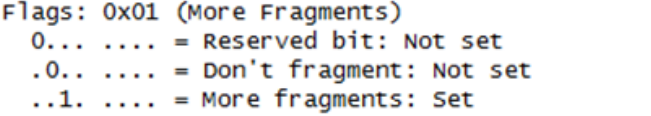
** 该消息是否被分解成不止一个IP数据报？**

** 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**



**是的，分成了两个。**



More fragments=1表⽰分⽚了且不是最后⼀⽚，该分⽚的长度是1500B

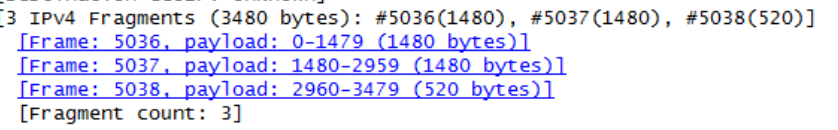
C. 找到在将包大小改为3500字节后你的主机发送的第一个ICMP

Echo Request消息。

思考下列问题：

** 原始数据包被分成了多少片？**

** 这些分片中IP数据报头部哪些字段发生了变化？**



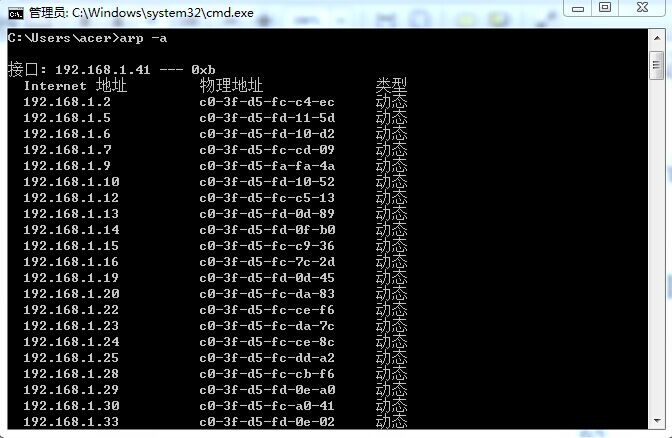
三片

前2个分⽚More fragments=1，后两个分⽚offset变为1480和2960

**（4）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?**

输入apr –a查看主机上ARP缓存的内容，结果如下图所示（截图显示部分）：



ARP缓存中的每一列分别表示IP地址所对应的物理地址和类型（动态配置或静态配置）

**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：**

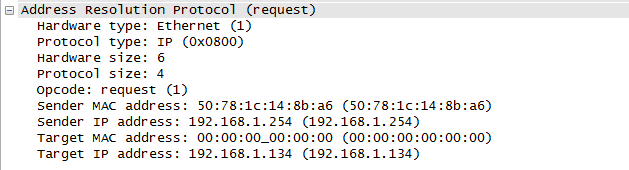
①**ARP数据包的格式是怎样的？由几部分构成，各个部分所占的字节数是多少？**

ARP数据包格式如下图：



由9部分构成，分别是硬件类型（2字节），协议类型（2字节），硬件地址长度（1字节），协议地址长度（1字节），OP（2字节)，发送端MAC地址（6字节），发送端IP地址（4字节），目的MAC地址（6字节），目的IP地址（4字节）。

截取的一个ARP数据包如下：



**②如何判断一个ARP数据是请求包还是应答包？**

通过OP字段。当OP字段值为0x0001时是请求包，当OP字段值为0x0002时是应答包。

③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？

因为进行ARP查询时并不知道目的IP地址对应的MAC地址，所以需要广播查询；而ARP响应报文知道查询主机的MAC地址（通过查询主机发出的查询报文获得），且局域网中的其他主机不需要此次查询的结果，因此ARP响应要在一个有着明确目的局域网地址的帧中传送。

**（5）抓取UDP数据包**

**①消息是基于UDP的还是TCP的？**

UDP。如图（User Datagram Protocol）：

4

**②你的主机IP地址是什么？目的主机IP地址是什么？**

我的主机IP地址：192.168.1.41

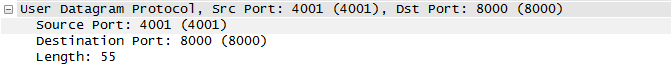
目的主机IP地址：182.254.110.92

1

③**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？**

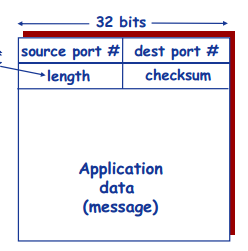
发送QQ消息端口号：4001

QQ服务器端口号：8000



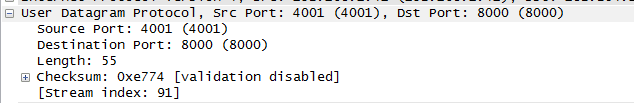
**④数据报的格式是什么样的？都包含哪些字段，分别占多少字节？**

UDP数据报格式如下图：



由5部分构成，分别是源端口号（4字节），目的端口号（4字节），长度（4字节），校验和（4字节）和应用层数据。

抓取的一个UDP数据报如下所示：



⑤**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？**

因为服务器需返回接收的结果给客户端。

因为服务器只提供了一次返回的ACK，所以不保证数据一定送达。

可以看出。UDP数据包没有序列号，因此不能像TCP协议那样先握手再发送数据，因为每次只发送一个数据报，然后等待服务器响应。

**（5）利用WireShark进行DNS协议分析**

**①打开浏览器，输入**[**www.baidu.com,DNS**](http://www.baidu.com,DNS)**查询消息如下图：**

1

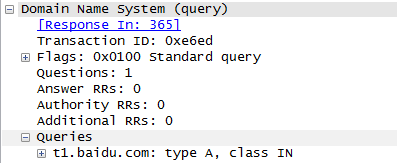
**②我的电脑IP地址：192.168.1.41，本地域名服务器IP地址：202.118.224.100.如图：**

2

**③UDP报文的源端口号65388，目的端口号53**

3

**④DNS查询报文内容如下图，表示查询主机域名为t1.baidu.com的主机的IP地址**



**⑤DNS回复信息：**

8

主机域名为t1.baidu.com的主机IP地址为：222.199.191.48

