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	田雪洋		院系	计算机科学与技术学院		
班级	1937102		学号	1190202110		
任课教师	李全龙		指导教师	李全龙		
实验地点	格物 207		实验时间	2021.11.20		
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：

本次实验的主要目的。

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：

概述本次实验的主要内容，包含的实验项等。

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：

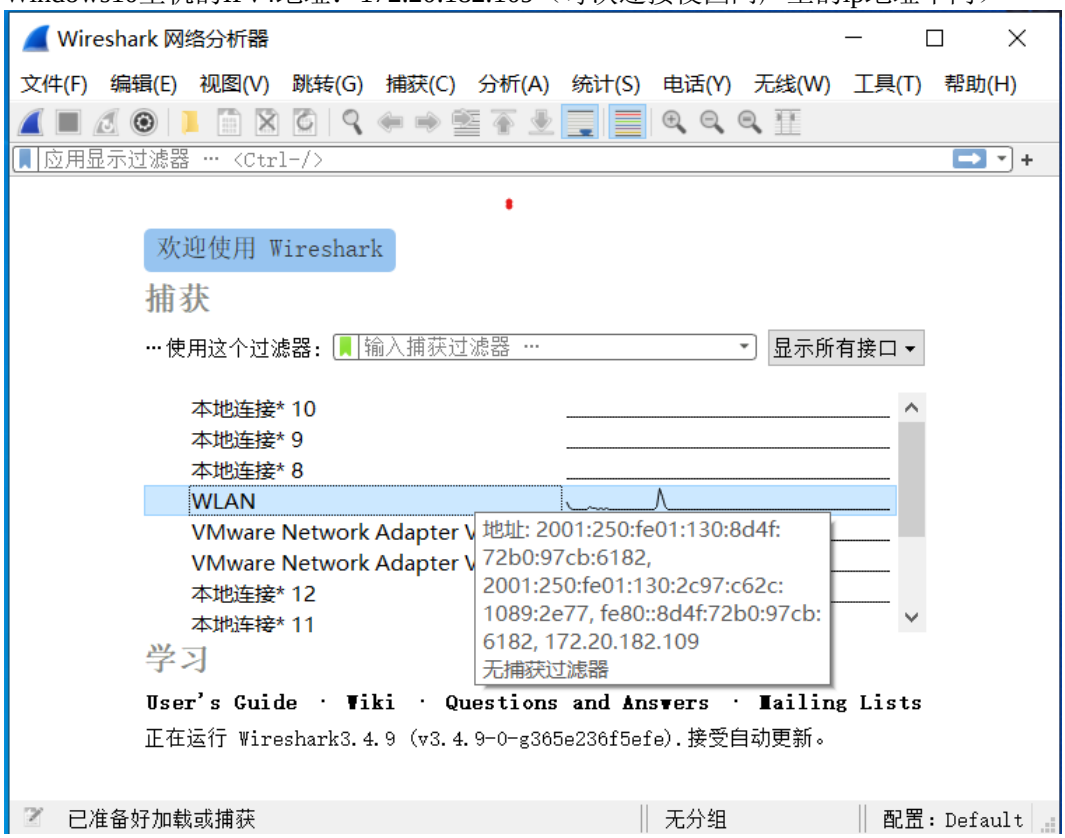
a) 利用 Wireshark 分析 DNS 协议
b) 利用 Wireshark 分析 UDP 协议
c) 利用 Wireshark 分析 ARP 协议

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。

在实验开始前，首先需要知道我的Windows 10主机的IPv4地址：

Windows10主机的IPv4地址：172.20.182.109（每次连接校园网产生的ip地址不同）



（1） Wireshark的使用

本次实验选取windows10主机的无线网卡来进行分组捕获，然后在运行分组俘获的同时，在浏览器地址栏中输入某网页的 URL：http://www.hit.edu.cn。进行访问。当完整的页面下载完成后，单击 Wireshark 菜单栏中的 stop 按钮，

停止分组俘获。之后在显示筛选规则中输入“http”，单击“回车”，让分组列表窗口将只显示HTTP 协议报文，这就是从www.hit.edu.cn服务器收到的HTTP GET报文，接着查看该报文的以太网帧、IP 数据报、TCP 报文段、以及HTTP 报文首部信息。然后保存到use.pcapng文件中。

（2） HTTP分析

1）HTTP GET/response 交互 ″启动 Web browser，然后启动 Wireshark 分组嗅探器。在窗口的显 示过滤说明处输入“http”，分组列表子窗口中将只显示所俘获到的 HTTP 报文。 ″开始 Wireshark 分组俘获。 ″在打开的 Web browser 窗 口 中 输 入 一 下 地 址 ：http://news.hit.edu.cn/停止分组俘获。所有结果保存在 http_get.pcapng 文件中。

2）HTTP 条件 GET/response 交互 《计算机网络》实验指导书 64 ″启动浏览器，清空浏览器的缓存（在浏览器中，选择“工具”菜单 中的“Internet 选项”命令，在出现的对话框中，选择“删除文件”）。 ″启动 Wireshark 分组俘获器。开始 Wireshark 分组俘获。 ″在浏览器的地址栏中输入以下 URL: http://news.hit.edu.cn/,在 你的浏览器中重新输入相同的 URL 或单击浏览器中的“刷新”按 钮。 ″停止 Wireshark 分组俘获，在显示过滤筛选说明处输入“http”,分组 列表子窗口中将只显示所俘获到的 HTTP 报文。所有结果保存在http_get2.pcapng文件中

（3） TCP分析

首先访问http://gaia.cs.umass.edu/wireshark-labs/alice.txt，获得alice.txt文件。然后打开https://gaia.cs.umass.edu/wireshark-labs/TCP-wireshark-file1.html，选择好本地alice.txt文件的位置。然后启动Wireshark开始分组捕获，在浏览器中点击”Upload alice.txt file”按钮上传文件，在文件上传完毕后停止Wireshark分组捕获。在筛选规则中选择“tcp”部分，进行分析即可，将所有分组保存在文件tcp.pcapng中。

（4） IP分析

首先，先下载并安装pingplotter，然后进行实验，使用的网站为www.baidu.com，启动Wireshark开始分组捕获，首先发送56字节的包；再发送2000字节的包,；再发送3500字节的包，然后停止Wireshark捕获。将所有分组保存在ip.pcapng中。

（5） 抓取ARP数据包

在cmd中输入 arp –a 命令，查看主机上ARP缓存的内容。在命令行模式下输入：ping 192.168.1.82。然后启动Wireshark进行捕获。将所有分组保存在arp.pcapng中。

（6） 抓取UDP数据包

启动 Wireshark 分组捕获，在 QQ 中给好友发送消息，然后停止分组捕获。将结果保存在 udp.pcapng 中。

（7） 利用Witeshark进行DNS协议分析

在浏览器中访http://www.baidu.com，进行Wireshark抓包。将所有分组保存在dns.pcapng中。

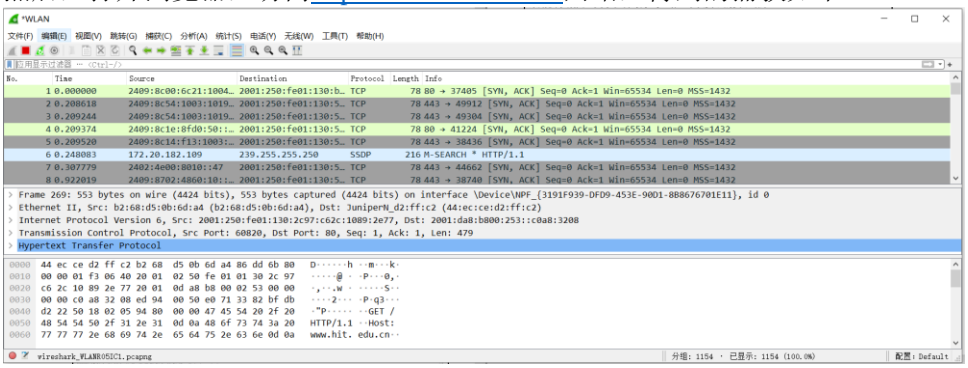
实验结果：

采用演示截图、文字说明等方式，给出本次实验的实验结果。

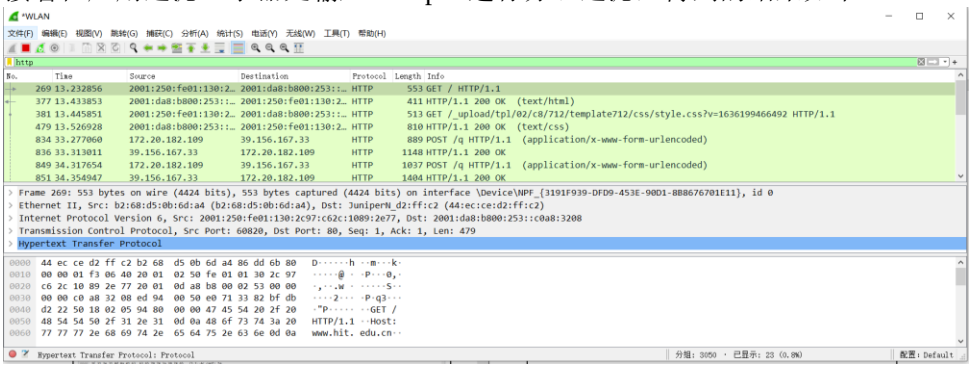
（1）Wireshark的使用

首先点击图中红圈部分，捕获win10无线网卡

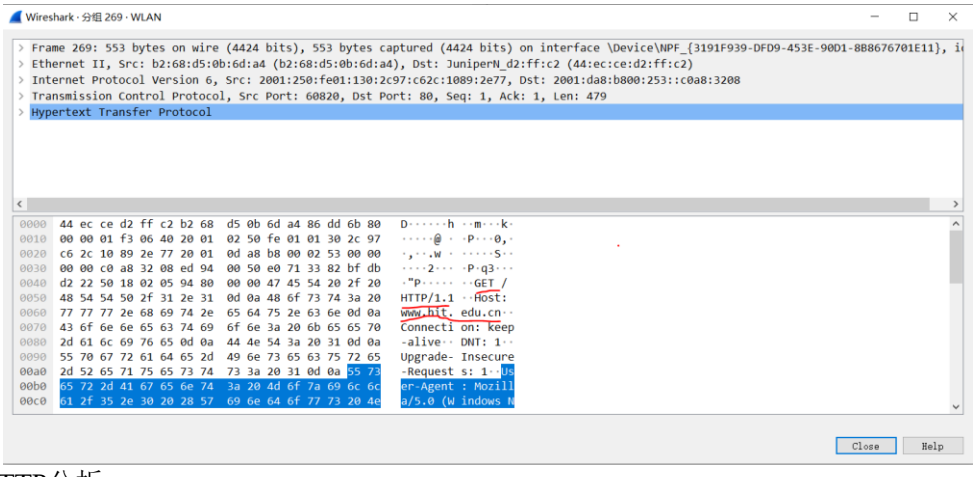
然后，打开浏览器，访问http://www.hit.edu.cn/网站，得到的捕获如下：



接着在应用过滤显示器处输入“http”进行分组过滤，得到的结果如下：



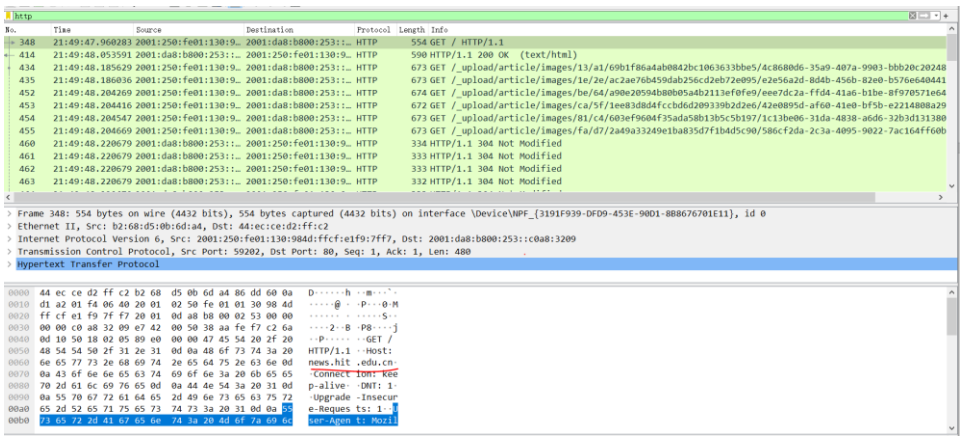
然后点击第一条HTTP报文，可以看到如下的信息：



（2）HTTP分析

1. **HTTP GET/response交互**

首先在过滤显示器处输入**http**，然后点击**WLAN**进行捕获，然后访问http://news.hit.edu.cn/网站，可以看到如下信息：



思考问题：

1.″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？

我的浏览器和访问的服务器运行的HTTP协议的版本号均为：HTTP 1.1

2.″你的浏览器向服务器指出它能接收何种语言版本的对象？

如图所示：

3.″你的计算机的 IP 地址是多少？服务器 http://hitgs.hit.edu.cn/news 的 IP 地址是多少？

我的计算机的IPv6地址为：2001:250：fe01:130:984d：ffcf：e1f9:7ff7

服务器的IPv6地址为：2001：da8:b800:253:：c0a8:3209

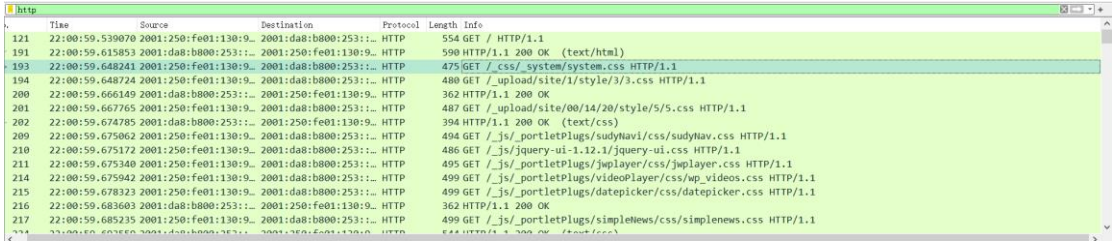
4.″从服务器向你的浏览器返回的状态代码是多少？

从服务器向你的浏览器返回的状态代码是：200

如上面的图所示；

2. HTTP 条件GET/response交互

清除浏览器缓存后再次实验，如下：

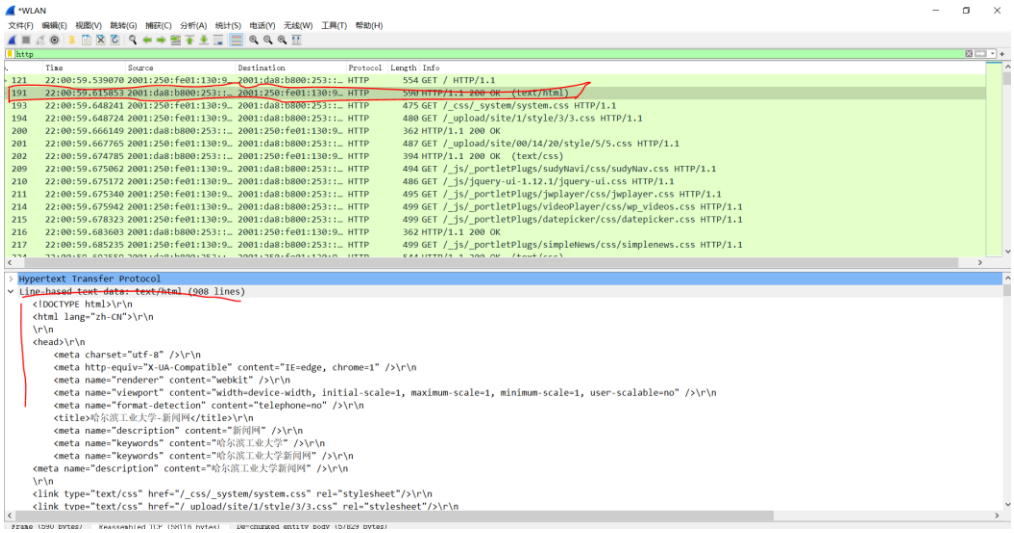


问题1：分析你的浏览器向服务器发出的第一个 HTTP GET 请求的内容， 在该请求报文中，是否有一行是：IF-MODIFIED-SINCE？

第一个HTTP GET请求没有IF-MODIFIED-SINCE头部。

问题2：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？

在第一个GET中明确了返回文件的内容，该内容是用来组成主页HTML的其他元素，如下图所示：



问题3：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。该行后面跟着的是最后修改日期：

```
If-Modified-Since: Wed, 16 Oct 2019 13:15:16 GMT\r\n
If-None-Match: "6393-59506e45799b6"\r\n
```

问题4：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。

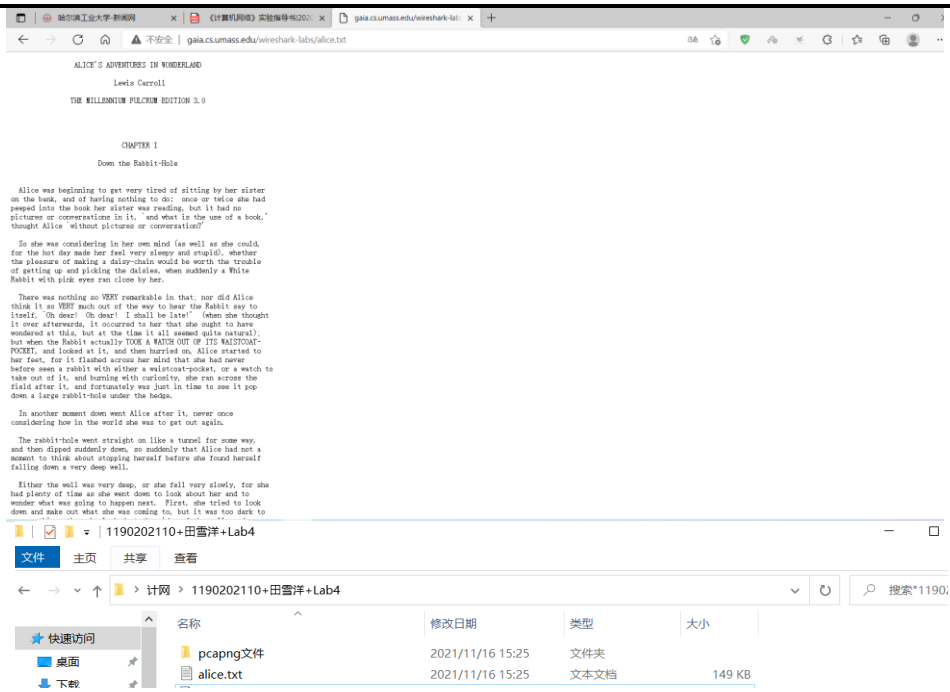
较晚的状态码是304。没有明确返回文件内容。

原因：此时本地有了缓存，且未改变，所以，浏览器直接从本地缓存读取内容。

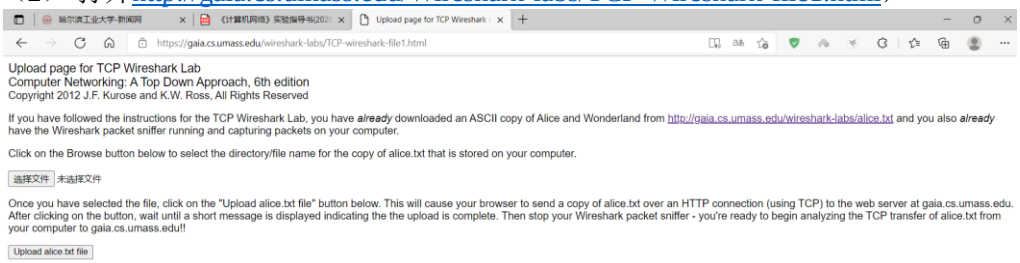
（3）TCP分析

A. 俘获大量的由本地主机到远程服务器的 TCP 分组

（1） 启动浏览器，打开http://gaia.cs.umass.edu/Wireshark-labs/alice.txt网页，得到ALICE'S ADVENTURES IN WONDERLAND文本，将该文件保存到你的主机上。

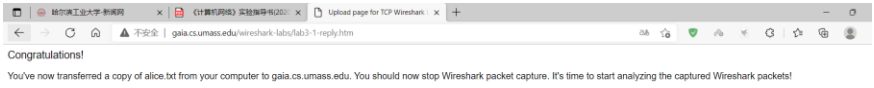


（2） 打开http://gaia.cs.umass.edu/Wireshark-labs/TCP-Wireshark-file1.html，



（3） 启动Wireshark，开始分组俘获。

（4） 在浏览器中，单击“Upload alice.txt file”按钮，将文件上传到gaia.cs.umass.edu服务器，一旦文件上传完毕，一个简短的贺词信息将显示在你的浏览器窗口中。



（5） 停止俘获。得到的结果保存到tcp.pcapng文件中。

在Wireshark筛选TCP报文后如下：

34	172.20.37.57	128.119.245.12	TCP	1514	51399 → 80 [ACK] Seq=12343 Ack=1 Win=131328 Len=1460 [TCP segment of a reassembled PDU]
35	128.119.245.12	172.20.37.57	TCP	56	80 → 51399 [ACK] Seq=1 Ack=663 Win=30592 Len=0
36	128.119.245.12	172.20.37.57	TCP	56	80 → 51399 [ACK] Seq=1 Ack=5043 Win=39296 Len=0
37	128.119.245.12	172.20.37.57	TCP	56	80 → 51399 [ACK] Seq=1 Ack=6503 Win=42240 Len=0
38	128.119.245.12	172.20.37.57	TCP	56	80 → 51399 [ACK] Seq=1 Ack=9423 Win=48128 Len=0
39	128.119.245.12	172.20.37.57	TCP	56	80 → 51399 [ACK] Seq=1 Ack=10883 Win=51072 Len=0
40	128.119.245.12	172.20.37.57	TCP	56	80 → 51399 [ACK] Seq=1 Ack=13803 Win=56832 Len=0
41	172.20.37.57	128.119.245.12	TCP	1514	51399 → 80 [ACK] Seq=13803 Ack=1 Win=131328 Len=1460 [TCP segment of a reassembled PDU]
42	172.20.37.57	128.119.245.12	TCP	1514	51399 → 80 [ACK] Seq=15263 Ack=1 Win=131328 Len=1460 [TCP segment of a reassembled PDU]
43	172.20.37.57	128.119.245.12	TCP	1514	51399 → 80 [PSH, ACK] Seq=16723 Ack=1 Win=131328 Len=1460 [TCP segment of a reassembled PDU]
44	172.20.37.57	128.119.245.12	TCP	1514	51399 → 80 [ACK] Seq=18183 Ack=1 Win=131328 Len=1460 [TCP segment of a reassembled PDU]
45	172.20.37.57	128.119.245.12	TCP	1514	51399 → 80 [ACK] Seq=19643 Ack=1 Win=131328 Len=1460 [TCP segment of a reassembled PDU]

问题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？

客户主机的IP地址：172.20.37.57, TCP的端口为：51399

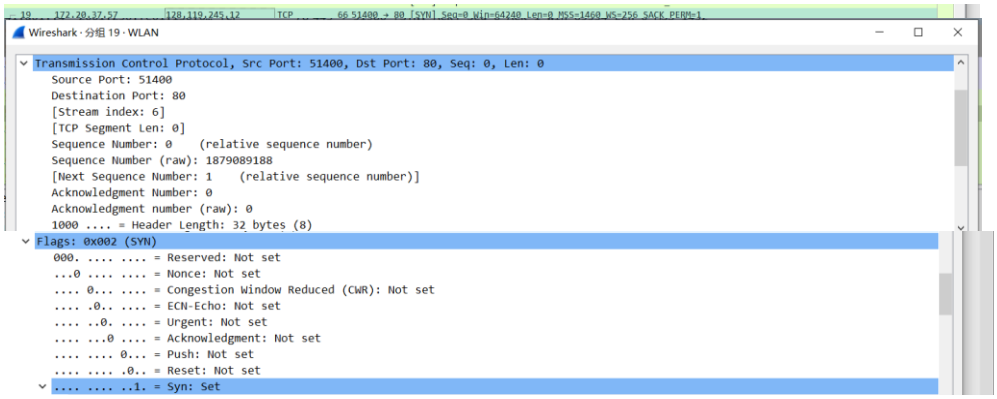
34	172.20.37.57	128.119.245.12	TCP	1514 51399 → 80 [ACK] Seq=12343 Ack=1 Win=131328 Len=1460 [TCP segment of a reassembled PDU]
35	128.119.245.12	172.20.37.57	TCP	56 80 → 51399 [ACK] Seq=1 Ack=663 Win=30592 Len=0
36	128.119.245.12	172.20.37.57	TCP	56 80 → 51399 [ACK] Seq=1 Ack=5043 Win=39296 Len=0
37	128.119.245.12	172.20.37.57	TCP	56 80 → 51399 [ACK] Seq=1 Ack=6503 Win=42240 Len=0
38	128.119.245.12	172.20.37.57	TCP	56 80 → 51399 [ACK] Seq=1 Ack=9423 Win=48128 Len=0
39	128.119.245.12	172.20.37.57	TCP	56 80 → 51399 [ACK] Seq=1 Ack=10883 Win=51072 Len=0
40	128.119.245.12	172.20.37.57	TCP	56 80 → 51399 [ACK] Seq=1 Ack=13803 Win=56832 Len=0
41	172.20.37.57	128.119.245.12	TCP	1514 51399 → 80 [ACK] Seq=13803 Ack=1 Win=131328 Len=1460 [TCP segment of a reassembled PDU]
42	172.20.37.57	128.119.245.12	TCP	1514 51399 → 80 [ACK] Seq=15263 Ack=1 Win=131328 Len=1460 [TCP segment of a reassembled PDU]
43	172.20.37.57	128.119.245.12	TCP	1514 51399 → 80 [PSH, ACK] Seq=16723 Ack=1 Win=131328 Len=1460 [TCP segment of a reassembled PDU]
44	172.20.37.57	128.119.245.12	TCP	1514 51399 → 80 [ACK] Seq=18183 Ack=1 Win=131328 Len=1460 [TCP segment of a reassembled PDU]
45	172.20.37.57	128.119.245.12	TCP	1514 51399 → 80 [ACK] Seq=19643 Ack=1 Win=131328 Len=1460 [TCP segment of a reassembled PDU]

问题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？

服务器的IP地址为：128.119.245.12。对这一连接，它用来发送和接收TCP报文的端口为：80

问题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？

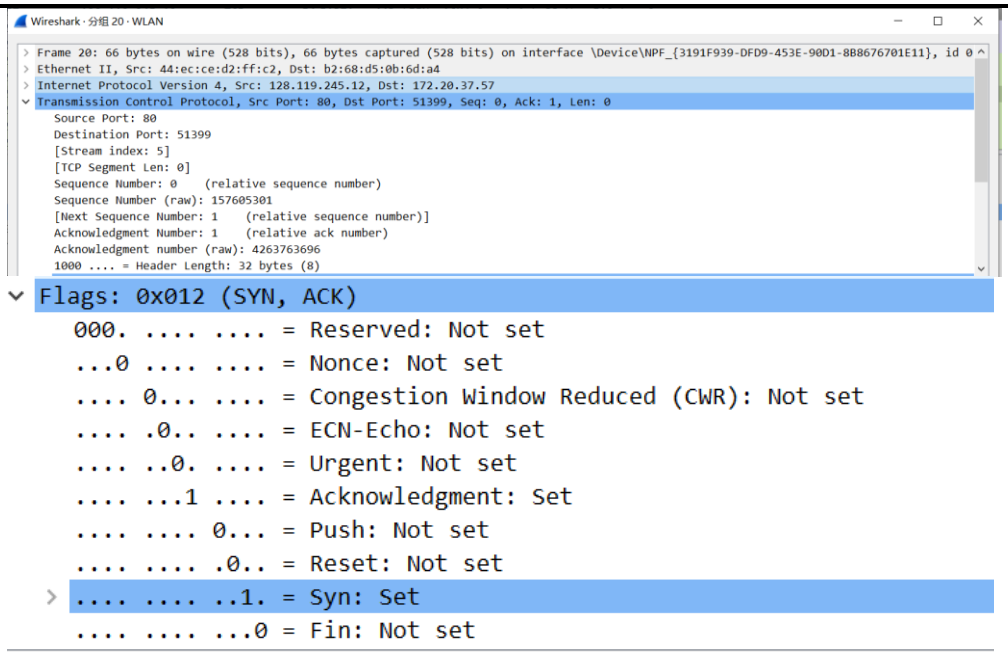
客户端和服务器之间用于初始化TCP 连接的TCP SYN 报文段的序号是0，该报文段将SYN置为1，表示该报文段用于tcp建立连接。如下图：



问题4：服务器向客户端发送的 SYNACK 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？

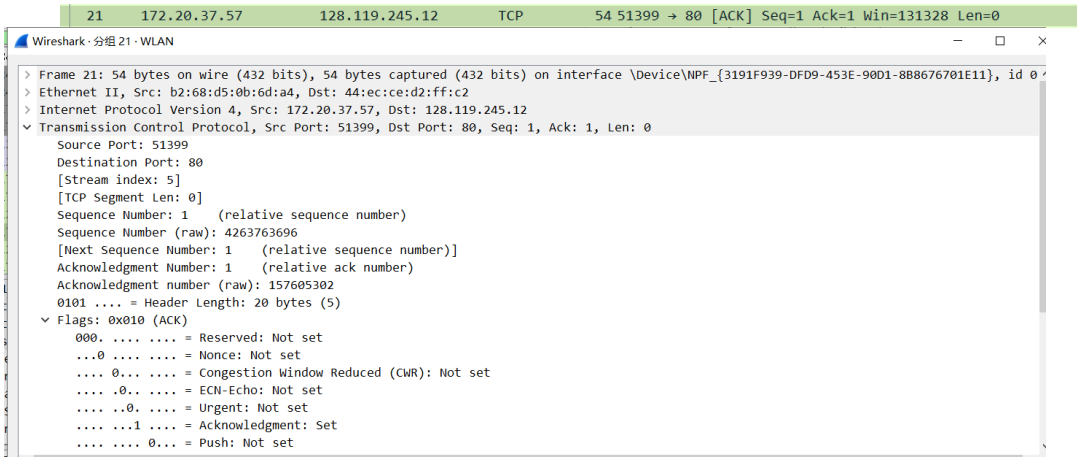
服务器向客户端发送的 SYNACK 报文段序号是0。该报文段中，Acknowledgement 字段的值是1。Gaia.cs.umass.edu服务器通过SYN请求报文段的seq序号加1来决定此值。在该报文段中，是使用flag部分的ack以及SYN标记为1来标示该报文段是SYNACK报文段的。如下图：

20	128.119.245.12	172.20.37.57	TCP	66 80 → 51399 [SYN, ACK] Seq=0 Ack=1 Win=29200 Len=0 MSS=1460 SACK_PERM=1 WS=128



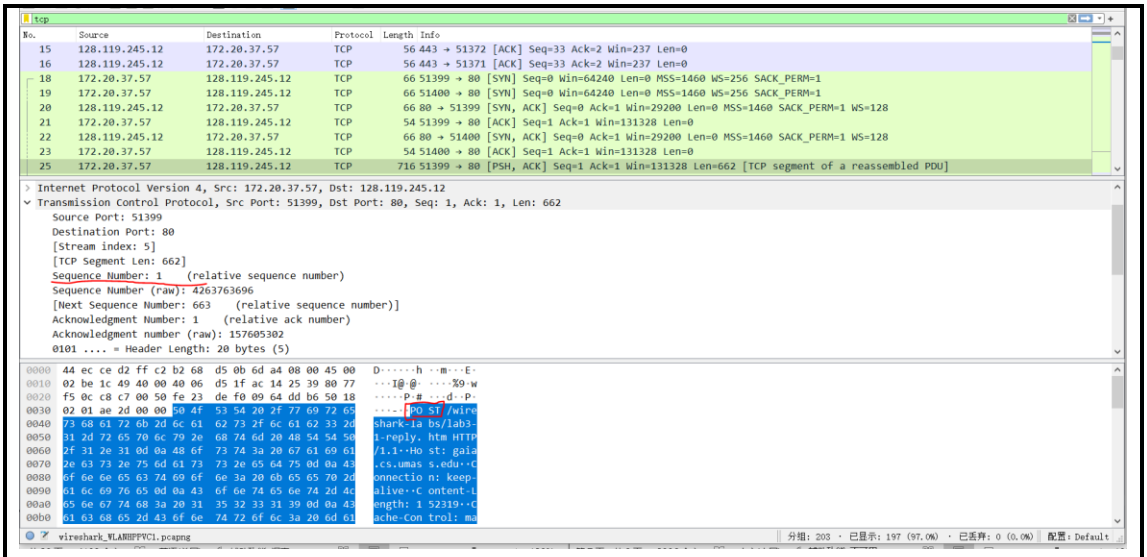
问题5：你能从捕获的数据包中分析出 tcp 三次握手过程吗？

能，上面两图是tcp三次握手的前两次握手，完成了客户端向服务器发送SYN请求报文，服务器向客户端回复SYNACK报文，第三次握手就是客户端像服务器回复ack报文，此时，ack=1（为SYNACK报文段序号+1），前两次握手过程前文已叙述完毕，此段只叙述第三次握手过程：



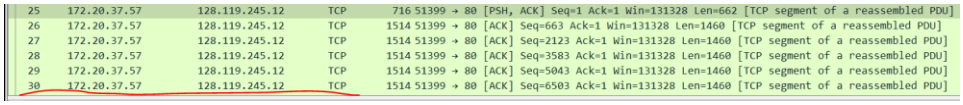
问题6：包含 HTTP POST 命令的 TCP 报文段的序号是多少？

该TCP报文段的序号是25，如图所示：

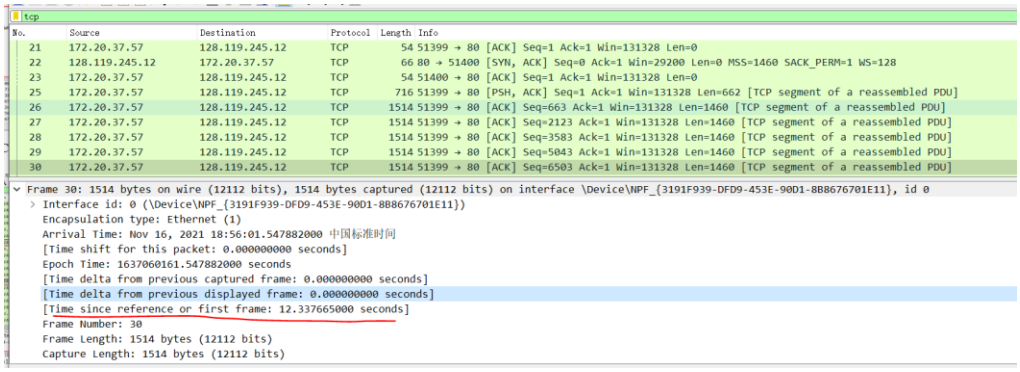


问题7：如果将包含 HTTP POST 命令的 TCP 报文段看作是 TCP 连接上的 第一个报文段，那么该 TCP 连接上的第六个报文段的序号是多 少？是何时发送的？该报文段所对应的 ACK 是何时接收的？

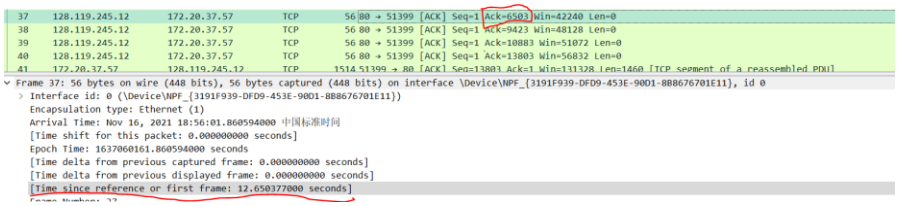
第六个报文段的序号为：30



发送时间是在第一帧发送后 12.337665s发送的该报文段。



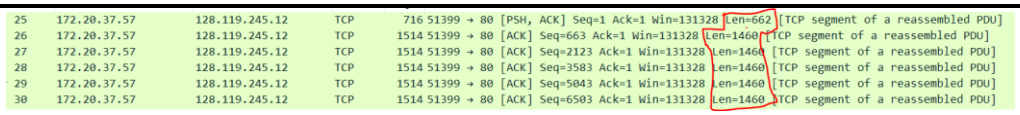
该报文段对应的ACK=6503报文接收如下：



可以看出接收时间为第一帧发送后的12.650377s

问题8：前六个 TCP 报文段的长度各是多少？

如图所示：



问题9：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？

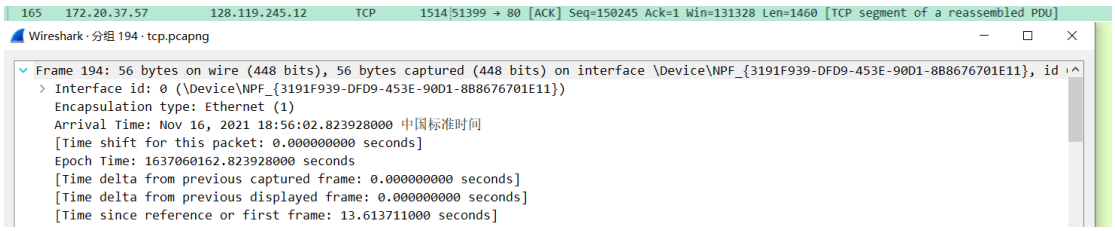
接收端公示的最小的可用缓存空间为29200字节。在整个过程中接收端并没有对发送端的传输进行限制。

问题10：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？

没有。

依据：在整个过程中，序列号随时间一直增长，而若有重传的报文段，会出现序列号减小的情况。但实际没有。

问题11：TCP 连接的 throughput (bytes transferred per unit time)是多少？请 写出你的计算过程。

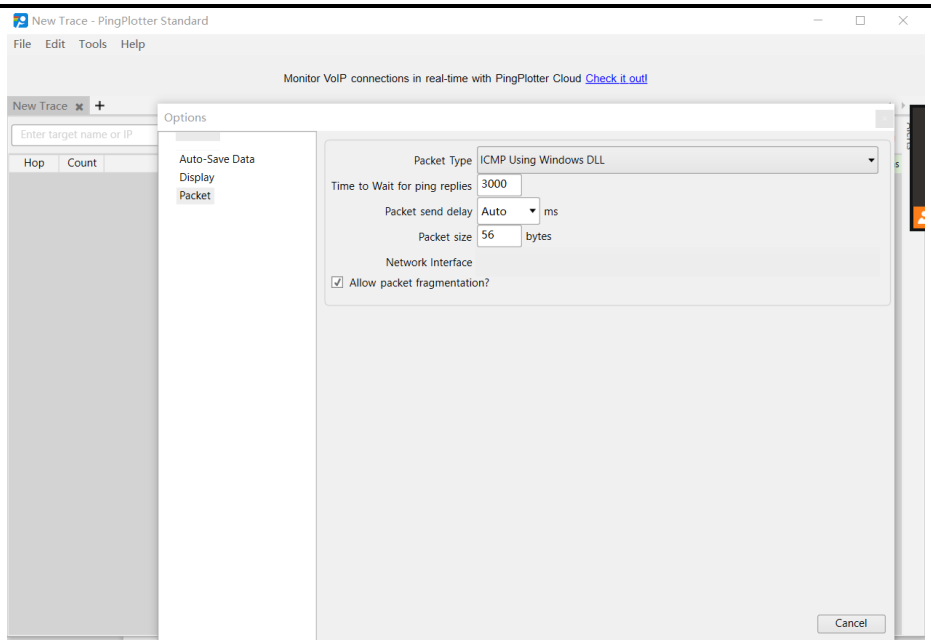


如图，最后一个seq为150245，则共传送150244个字节，用时：13.61371-12.650377=0.96333s

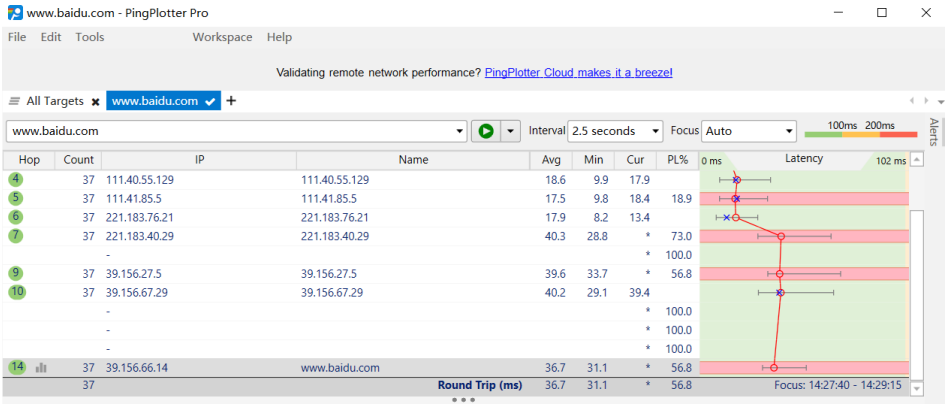
吞吐量为：1.56MB/S

（4）IP分析

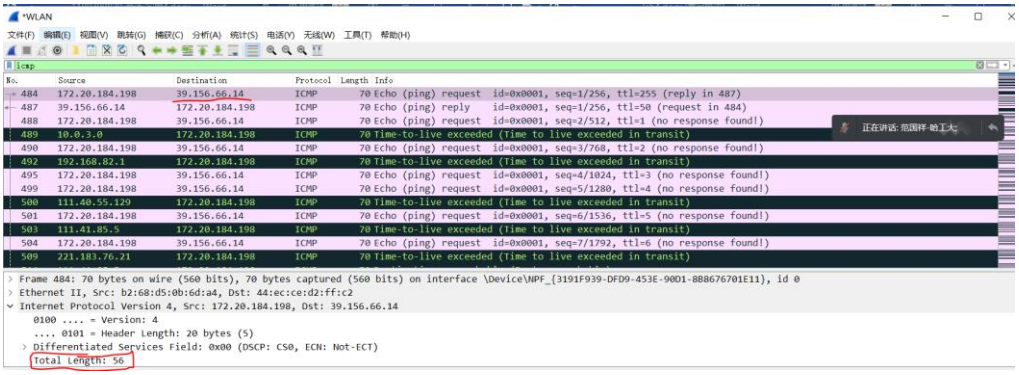
1.下载并安装 pingplotter。ICMP echo 请求消息的大小可以 通过下面方法在 pingplotter 中进行设置。Edit->Options->Packet，然后填 写 Packet Size(in bytes，default=56)域。

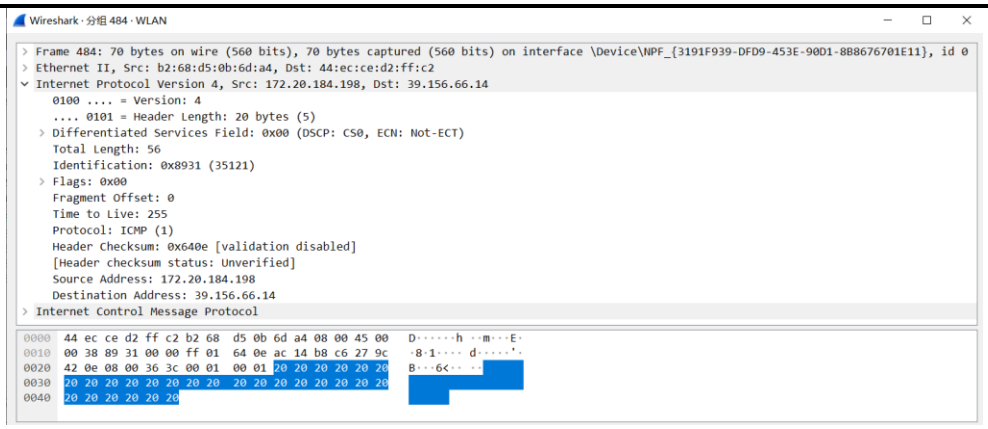


2. 启动 pingplotter 并“Address to Trace Window”域中输入目的地址。在“# of times to Trace”域中输入“3”，这样就不过采集过多的数据。Edit->Options->Packet，将 Packet Size(in bytes,default=56)域设为 56，这样将发送一系列大小为 56 字节的包。然后按下“Trace”按钮。



使用wireshark进行捕获结果如下：





问题1：你主机的IP地址是什么？

172.20.184.198

问题2：在IP数据包头中，上层协议（upper layer）字段的值是什么？

ICMP(1)

问题3： 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

IP头有20个字节，该IP数据包的净载为36个字节。（IP数据包总大小为56字节，头部有20字节，则净载为56-20=36字节）

问题4： 该IP数据包的净载大小的？

36个字节，见上问。

问题5： 该IP数据包分片了吗？解释你是如何确定该IP数据包是否进行了分片

没有分片，观察flag区如下：

```
Flags: 0x00
    0... .... = Reserved bit: Not set
    .0.. .... = Don't fragment: Not set
    ..0. .... = More fragments: Not set
```

从上图可以看到没有其余的帧且偏移为0，MF=0.说明没有分片。

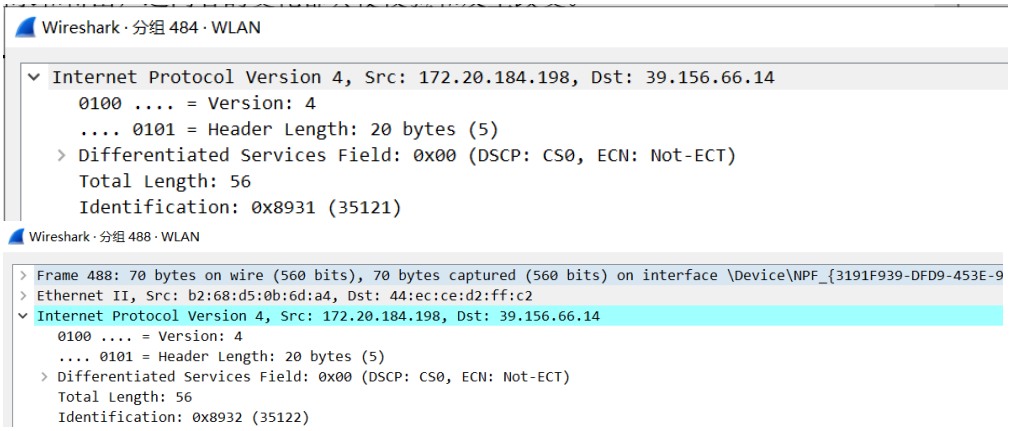
问题6：你主机发出的一系列ICMP消息中IP数据报中哪些字段总是发生 改变？

总是在发生改变的：Identification，TTL以及checknum。
Identification就是IP数据包的序号，每个包的序号都不同。
根据traceroute的工作原理，每次主机发送的IP数据包的TTL都加一。
checksum为头部数据求和得出，这两者的变化都会使校验和发生改变。

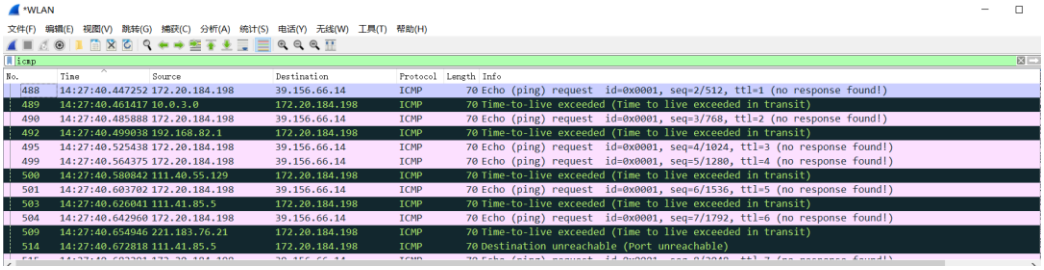
问题7： 哪些字段必须保持常量？哪些字段必须改变？为什么？

必须保持常量：版本号，上层协议，源IP地址，目的IP地址。
总是发生改变的： Identification，TTL以及checknum。

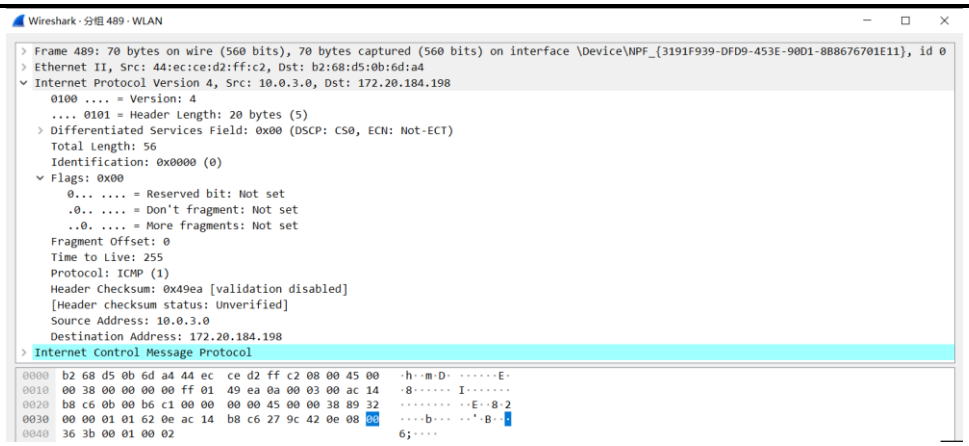
问题8： 描述你看到的IP数据包Identification字段值的形式。
Identification大小为2字节，第一个IP数据包的Identification是随机产生的值，之后的IP数据包的Identification每次增加1。
如图：



问题9：Identification字段和TTL字段的值是什么？
按时间排序报文：



第2个报文就是最近的路由器返回的ICMP Time-to-lice exceeded消息，查看该报文如下：



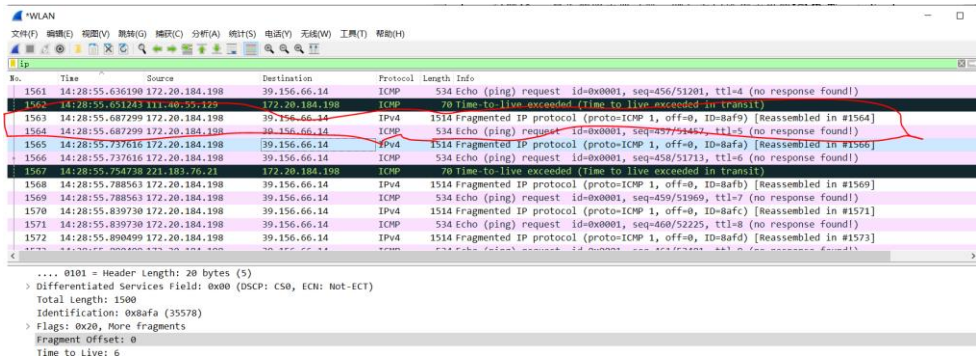
可以看到Identification字段为0，TTL字段为255

问题10： 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字段的值会改变，但是TTL不变.

原因：Identification是用来区分不同的IP数据包的，数据包不同，Identification不一样。TTL是默认设置为255的。

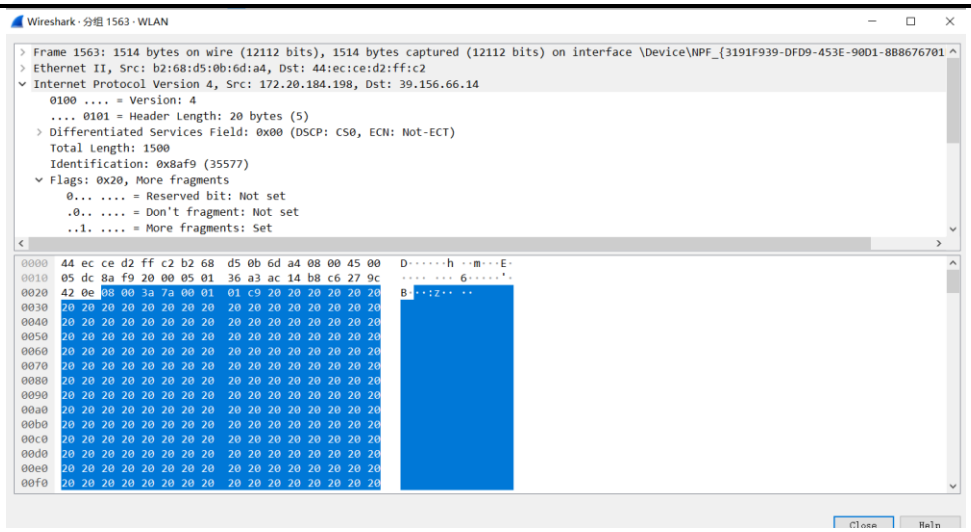
问题11：该消息是否被分解成不止一个IP数据报？



如图，可以看到，包的大小改为2000字节后，主机发送的第一个ICMP Echo 消息被切片成了两片。

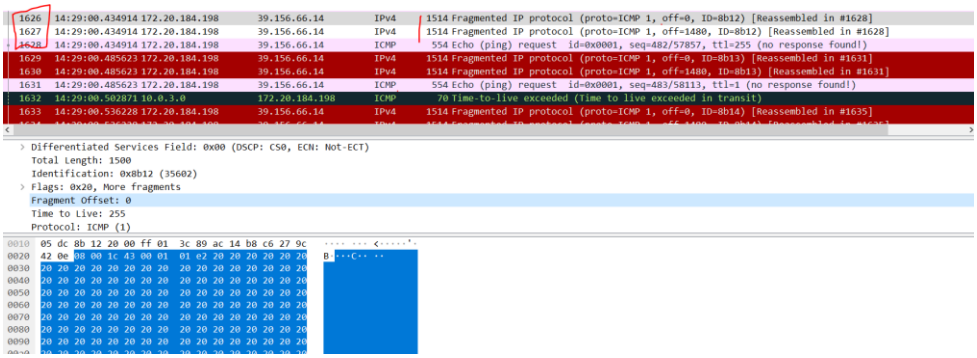
问题12： 观察第一个IP分片，IP头部的哪些信息表明数据包被进行了分片？IP头部的哪些信息表明数据包是第一个而不是最后一个分 片？该分片的长度是多少

DF=0,MF=1,说明了该数据包进行了切片，并且该片不是最后一片。该数据包的片偏移=0，说明为第一个包。该分片的长度为1500字节。



问题13：原始数据包被分成了多少片？

三片。

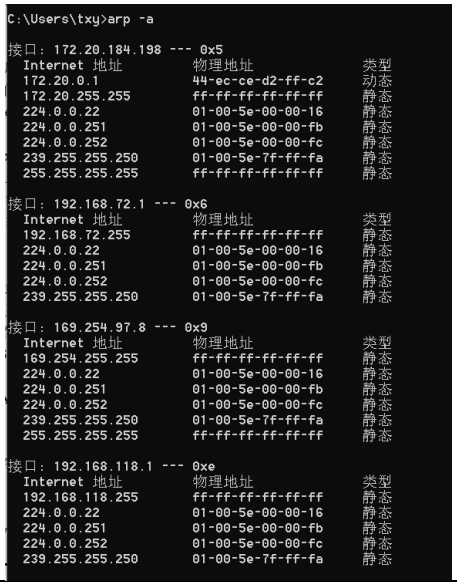


问题14：这些分片中IP数据报头部哪些字段发生了变化？

这些分片的头部长度，片偏移，标志位，校验和字段发生了变化

（5）抓取ARP数据包

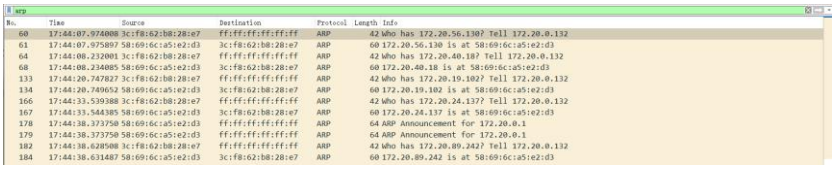
1. 利用 MS-DOS 命令：arp -a 或 c:\windows\system32\arp 查看主机上 ARP 缓存的内容。



问题1：利用 MS-DOS 命令：arp -a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?

第一列为为接口的IP地址；第二列为接口的MAC地址；第三列为地址的类型，其中动态条目是通过ARP协议学来，而静态条目是由网卡自己生成或者是通过手工配置的。

2.清空arp缓存 在命令行模式下输入：ping 192.168.1.82（或其他 IP 地址）启动 Wireshark，开始分组俘获，得到的数据包如下：



问题2：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？

由9部分构成：

硬件类型：2字节
协议类型：2字节
硬件大小：1字节
协议地址大小：1字节
Opcode：2字节
发送方MAC地址：6字节
发送方IP地址：4字节
接收方MAC地址：6字节
接收方IP地址：4字节

```
Address Resolution Protocol (request)
  Hardware type: Ethernet (1)
  Protocol type: IPv4 (0x0800)
  Hardware size: 6
  Protocol size: 4
  Opcode: request (1)
  Sender MAC address: 3c:f8:62:b8:28:e7
  Sender IP address: 172.20.0.132
  Target MAC address: 00:00:00:00:00:00
  Target IP address: 172.20.56.130
```

问题3：如何判断一个ARP数据是请求包还是应答包？

若Opcode=1，为request包

若Opcode=2，为reply包

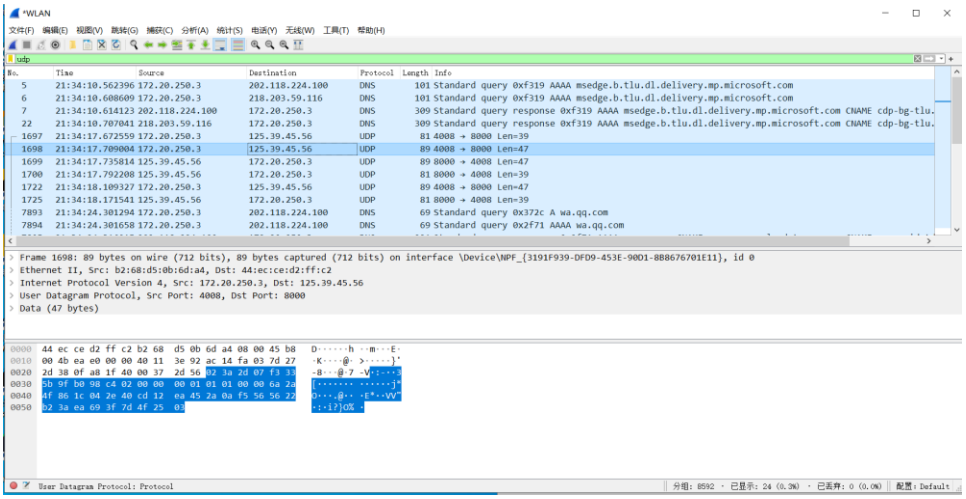
```
Opcode: request (1)
Opcode: reply (2)
```

问题4：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，在查询MAC地址时主机不知道目的IP的MAC地址，因此需要在自己的局域网

中进行广播查询。而ARP响应只需要发给提出目的主机即可，故ARP响应要在一个有着明确目的局域网地址的帧中传送。

（6）抓取UDP数据包



问题 1：消息是基于 UDP 的还是 TCP 的？

消息是基于 UDP 的。

问题 2：你的主机 ip 地址是什么？目的主机 ip 地址是什么？

如上图所示：我的主机 ip 地址是：172.20.250.3
目的主机的 IP 地址是：202.118.224.100

问题 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？

```
User Datagram Protocol, Src Port: 54263, Dst Port: 53
    Source Port: 54263
    Destination Port: 53
```

我的主机发送 QQ 消息的端口号是：54263
QQ 服务器的端口号是：53

问题 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？

```
User Datagram Protocol, Src Port: 54263, Dst Port: 53
    Source Port: 54263
    Destination Port: 53
    Length: 67
    Checksum: 0x80f2 [unverified]
    [Checksum Status: Unverified]
    [Stream index: 0]
  > [Timestamps]
    UDP payload (59 bytes)
> Domain Name System (query)
```

如上图所示：数据包的格式如上图所示，字段为：
源端口号：2 字节
目的端口号：2 字节
报文长度：2 字节

校验和：2 字节

问题 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？

原因：UDP是不可靠数据传输，服务器又返回给你的主机一个 ICQ数据包进行确认。

有联系。

能，因为UDP在发送报文前没有握手的过程。

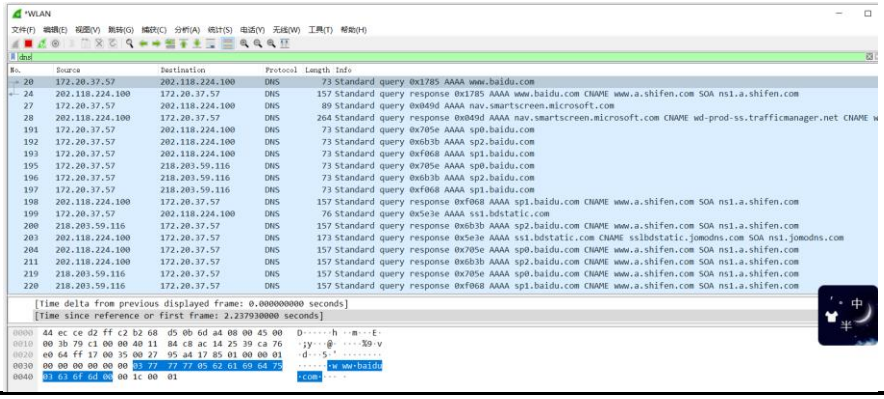
（7）利用Witeshark进行DNS协议分析

（1）打开浏览器键入:www.baidu.com

（2）打开 Wireshark,启动抓包.

（3）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：



问题讨论：

对实验过程中的思考问题进行讨论或回答。

所有的思考问题都在上面的实验结果部分进行了回答。详细内容见前文实验内容部分。

心得体会：

结合实验过程和结果给出实验的体会和收获。

通过本次实验，初步掌握了使用Wireshark分析TCP,HTTP,UDP,IP,ARP等网络协议的执行，并且通过这些分析，对这些协议的各个字段的理解更加深入，为以后的计算机网络的学习打下了基础。