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	典型协议的抓包分析——利用 Wireshark 进行协议分析				
姓名	杨明达		院系	未来技术学院	
班级	22R0311		学号	2022110829	
任课教师	李全龙		指导教师	李全龙、郭勇	
实验地点	G001		实验时间	周六 5、6 节	
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：

必做内容：

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

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：

（1）学习 Wireshark 的使用

Wireshark 是一种可以运行在 Windows, UNIX, Linux 等操作系统上的分组分析器。可以截取各种网络数据包，并显示数据包详细信息。常用于开发测试过程中各种问题定位。适用于 UNIX 和 Windows，从网络接口捕获实时数据包数据，保存捕获的数据包数据，以多种捕获文件格式 导出部分或全部数据包，根据多种标准 过滤数据包，根据多种标准 搜索数据包等。其可以截取网络封包，并尽可能的显示出最为详细的网络封包资料。Wireshark 使用 WinPCAP 接口，直接与网卡进行数据报文交换。

1）启动主机上的 web 浏览器

2）启动 Wireshark，如图 1-1 所示，窗口没有任何分组列表。

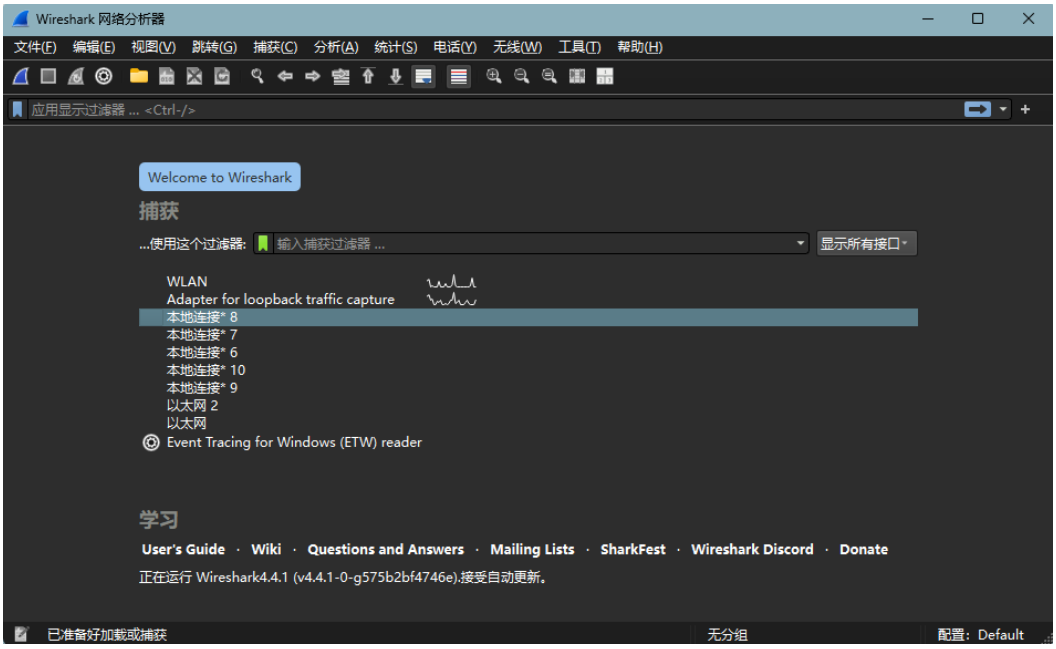

图 1-1 Wireshark 初始窗口

3）开始分组俘获：选择“捕获”下拉菜单中的“选项”命令，会出现如图 1-1 菜单栏上所示的“Wireshark:捕获选项”窗口，可以设置分组俘获的选项。

4）使用窗口中显示的默认值。见图 1-2，在“Wireshark: 捕获选项”窗口的 “接口”下拉菜单，其中显示计算机所具有的网络接口（即网卡）。当计算机具有多个活动网卡时，需要选择“WLAN”。随后，单击“Start”开始进行分组俘获，所有由选定网卡发送和接收的分组都将被俘获。

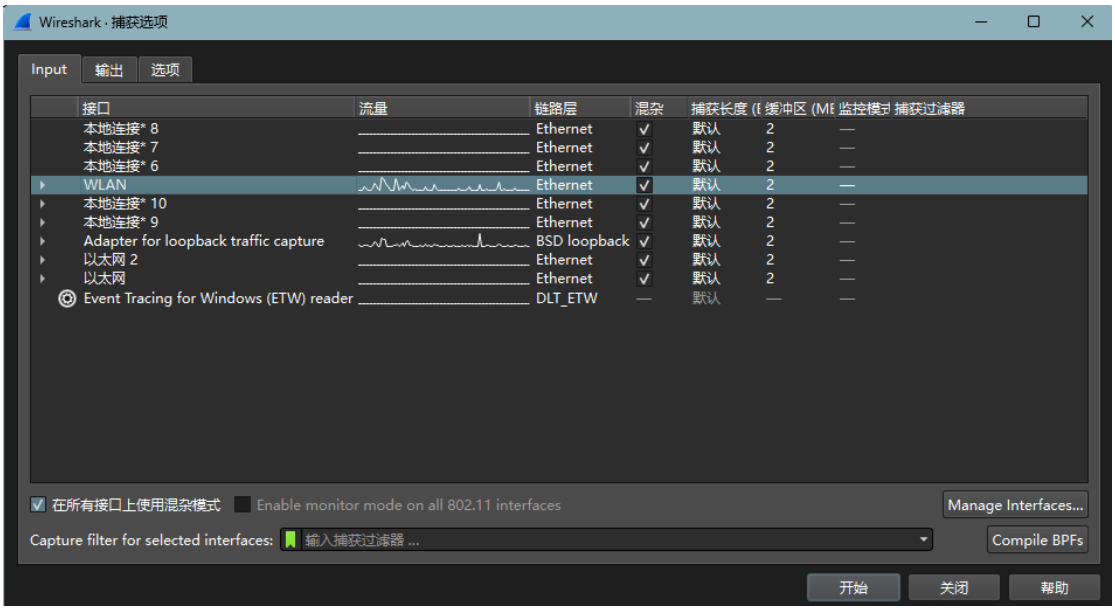

图 1-2 Wireshark 捕获选项窗口

5）开始分组俘获后，会出现如图 1-3 所示的窗口。该窗口统计显示各类已俘获数据包。在该窗口的工具栏中有一个“stop”按钮，可以停止分组的俘获。

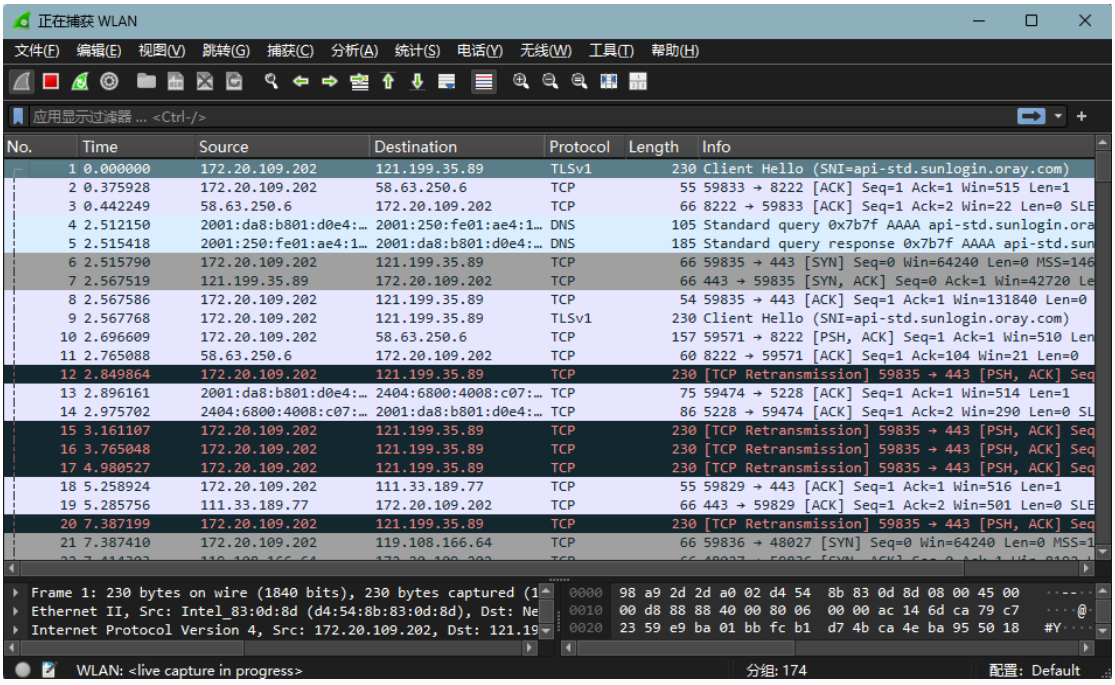

图 1-3 Wireshark 捕获信息窗口

6）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，如： http://www.hit.edu.cn 。为显示该网页，浏览器需要连接 www.hit.edu.cn 的服务器，并与之交换 HTTP 消息，以下载该网页。包含这些 HTTP 报文的以太网帧将被 Wireshark 俘获。浏览器界面如图 1-4 所示。

图 1-4 浏览器访问界面

7）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。Wireshark 主窗口显示已俘获的计算机与其他网络实体交换的所有协议报文，其中一部分就是与 www.hit.edu.cn 服务器交换的 HTTP 报文。此时主窗口如图 1-5。

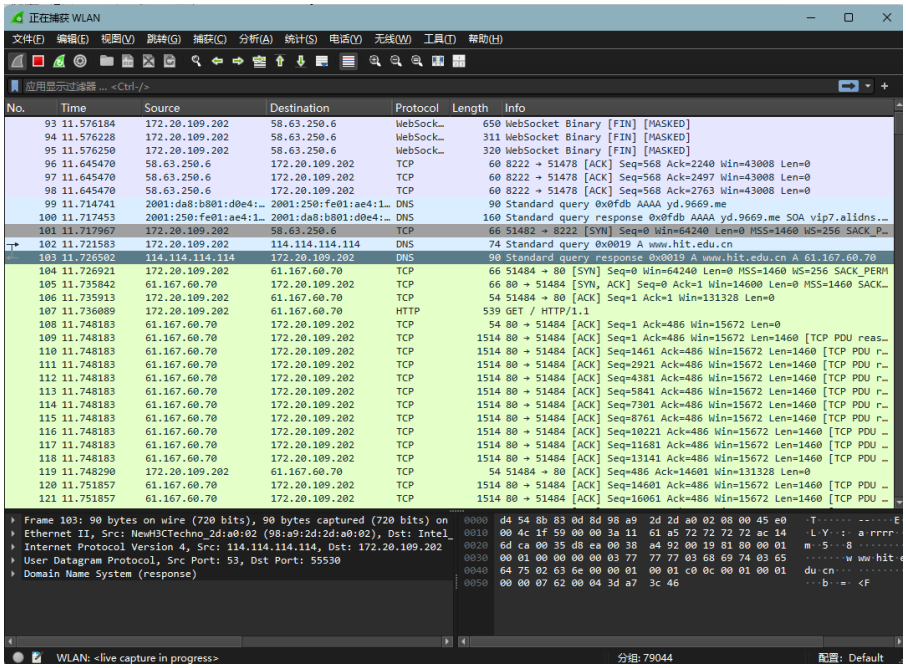

图 1-5 Wireshark 主窗口（网站访问）

8）在显示筛选规则中输入“http”，单击“回车”，分组列表窗口将只显示 HTTP 协议报文。如图 1-6 所示。

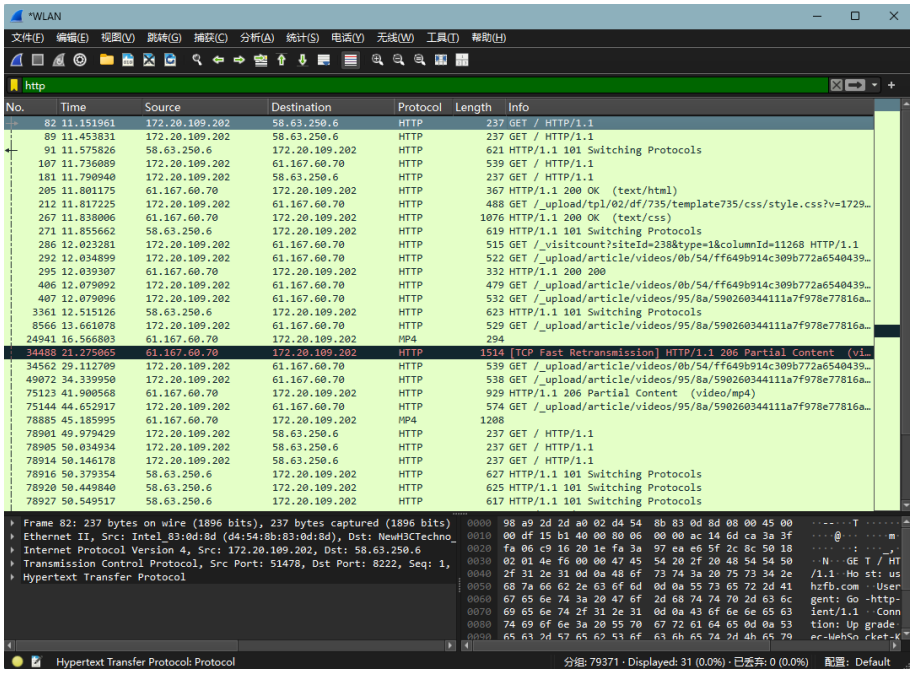

图 1-6 HTTP 协议报文

9）选择分组列表窗口中的第一条 http 报文。它应该是计算机发向 www.hit.edu.cn 服务器的 HTTP GET 报文。当你选择该报文后，以太网帧、 IP 数据报、 TCP 报文段、以及 HTTP 报文首部信息都将显示在分组首部子窗口中。如图 1-7，单击分组首部详细信息子窗口中向右和向下箭头，可以最小化帧、以太网、 IP、 TCP 信息显示量，可以最大化 HTTP 协议相关信息的显示量。

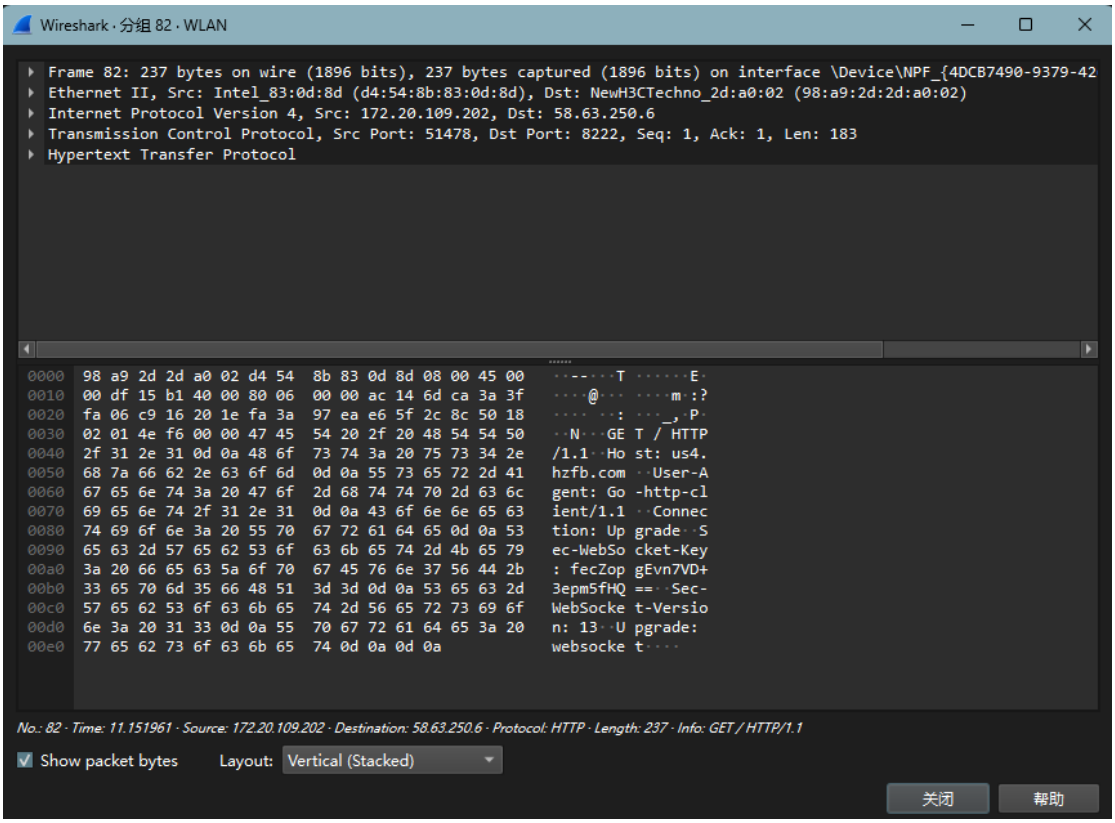

图 1-7 分组首部详细信息

（2）利用 Wireshark 分析 HTTP 协议

1）HTTP GET/response 交互

a）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。

b）开始 Wireshark 分组俘获。

c）在打开的 Web browser 窗口中输入一下地址：http://today.hit.edu.cn

图 2-1 浏览器访问

d）停止分组俘获。

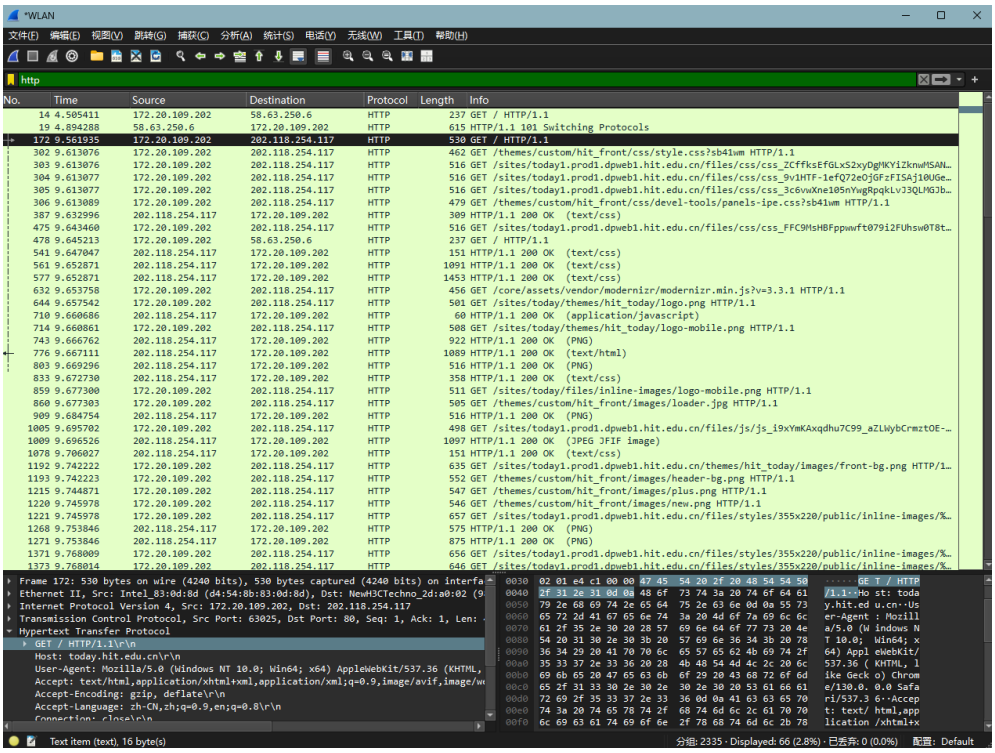

图 2-2 Wireshark 主窗口（网站访问）

根据俘获窗口内容，思考以下问题：

➢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？

浏览器运行的是 HTTP 1.1，服务器运行的 HTTP 协议版本号为 HTTP 1.1。

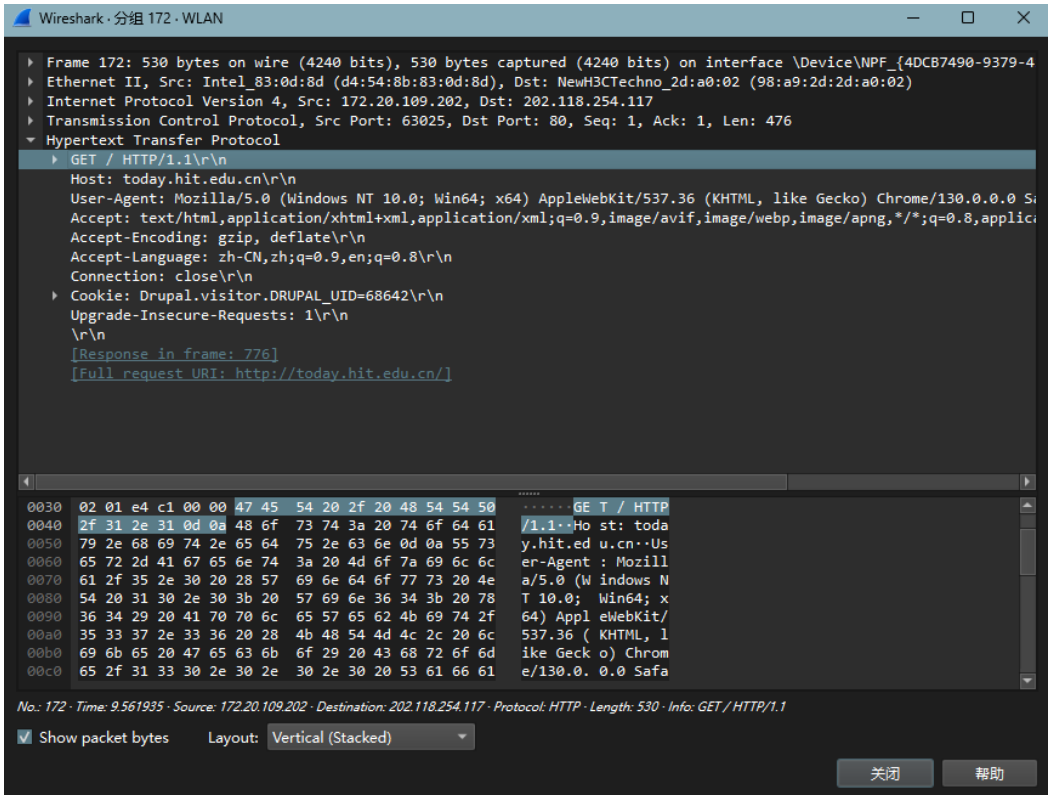

图 2-3 浏览器 HTTP 版本

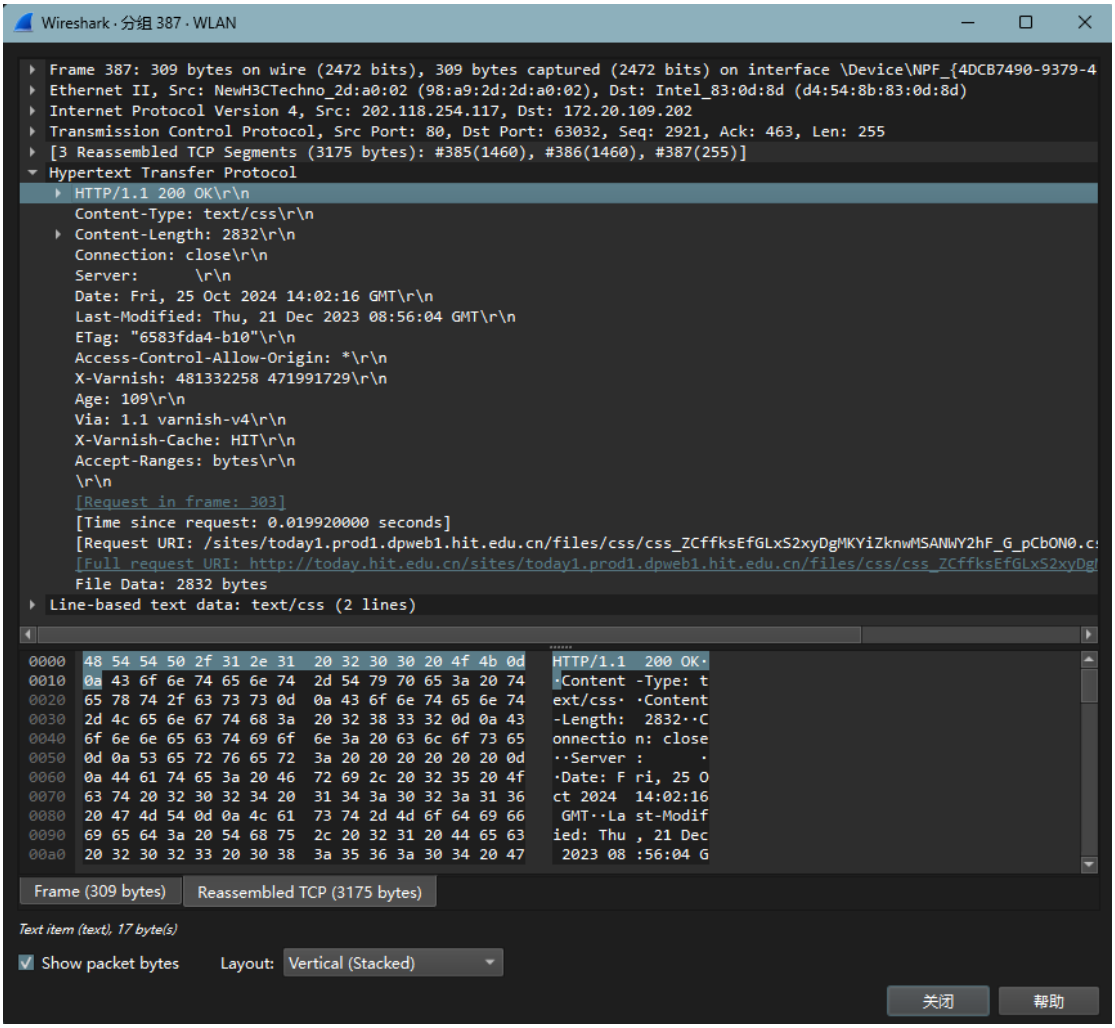

图 2-4 服务器 HTTP 版本

➢你的浏览器向服务器指出它能接收何种语言版本的对象？

浏览器向服务器指出可接受的语言版本为 zh-CN。

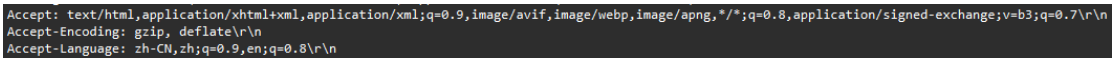

图 2-5 语言版本

➢你的计算机的 IP 地址是多少？服务器 http://today.hit.edu.cn 的 IP 地址是多少？

计算机的 IP 地址为：172.20.109.202，服务器的 IP 地址为：202.118.254.117。

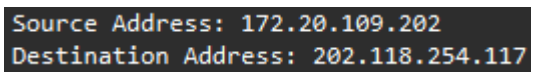

图 2-6 计算机和服务器 IP 地址

➢从服务器向你的浏览器返回的状态代码是多少？

从服务器向浏览器返回的状态代码为 200。

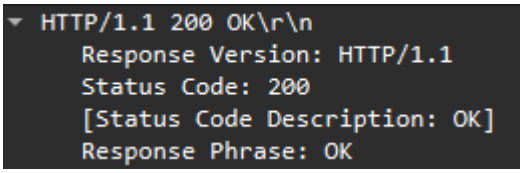

图 2-7 返回的状态代码

2）HTTP 条件 GET/response 交互

a）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”）。

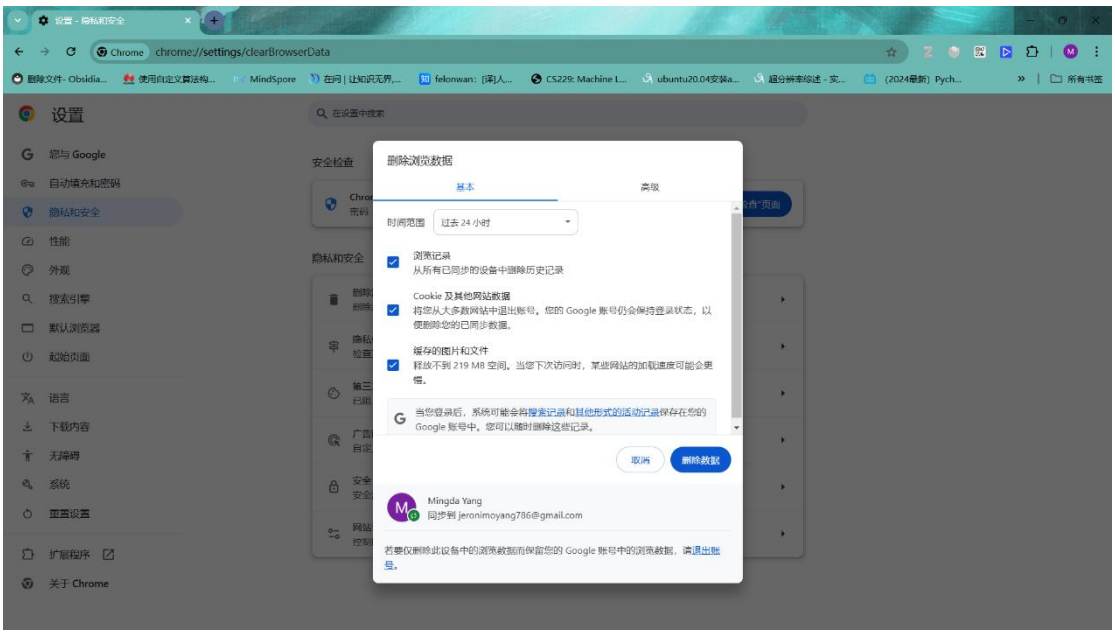

图 2-8 清空缓存

b）启动 Wireshark 分组俘获器。开始 Wireshark 分组俘获。

c）在浏览器的地址栏中输入以下 URL: http://today.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。

d）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。

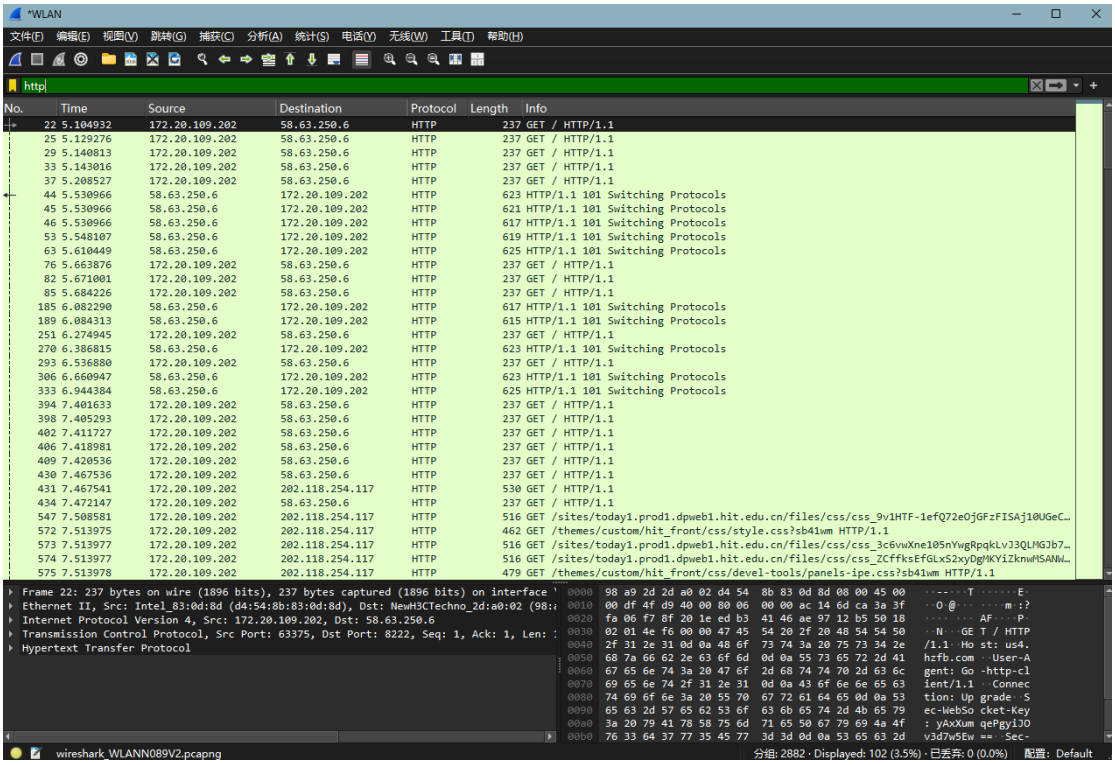

图 2-8 Wireshark 主窗口（网站访问）

根据俘获窗口内容，思考以下问题：

➢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 GET 请求在请求报文中没有 IF-MODIFIED-SINCE

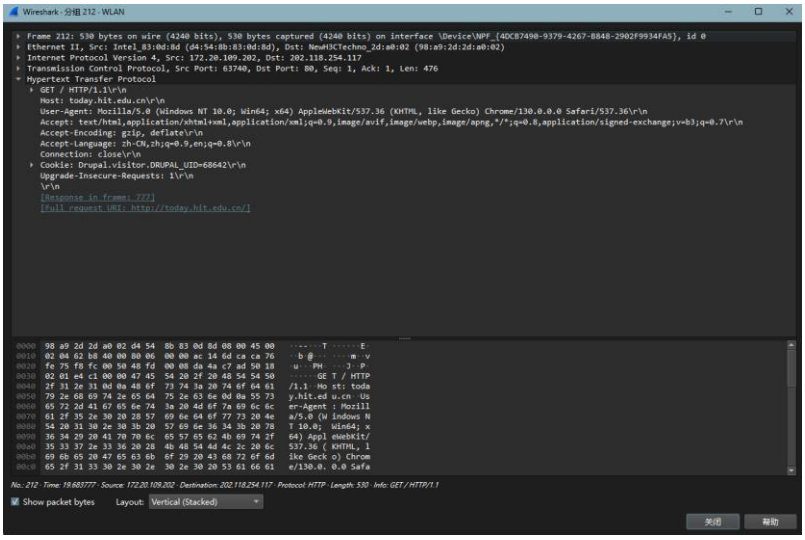

图 2-9 第一个 GET 请求

➢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？

服务器返回状态码和数据部分等文件内容。

图 2-10 服务器返回文件内容

➢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？

向服务器发出较晚的“HTTP GET”请求，请求报文中有 IF-MODIFIED-SINCE。在改行后的信息是本地缓存文件中 Last-Modifined 字段最后修改时间。

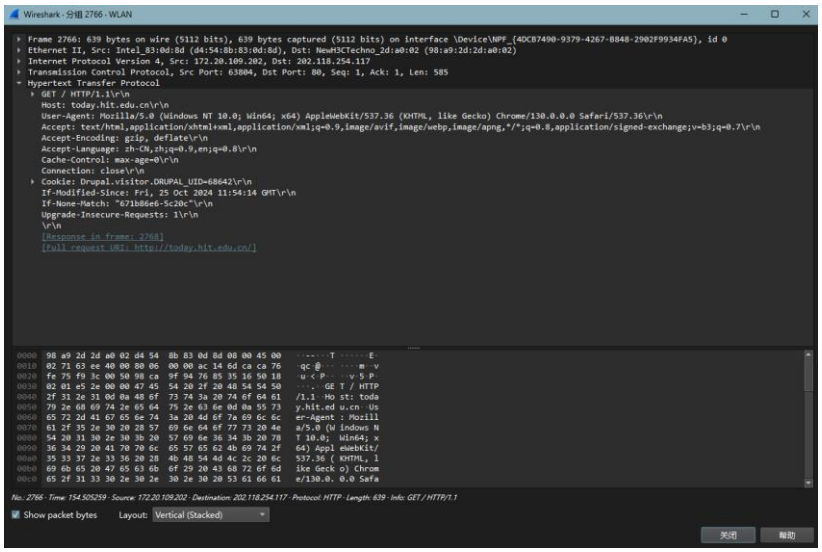

图 2-11 较晚的 GET 请求

➢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。

服务器对较晚的 HTTP GET 请求的响应中的 HTTP 状态代码是 304。服务器返回了文件的内容，因为客户端找到本地缓存后，经过请求报文向服务器端确定这一份缓存时最新的，所以服务器端不会向客户端发送数据，而是客户端直接选择这个缓存的数据段。

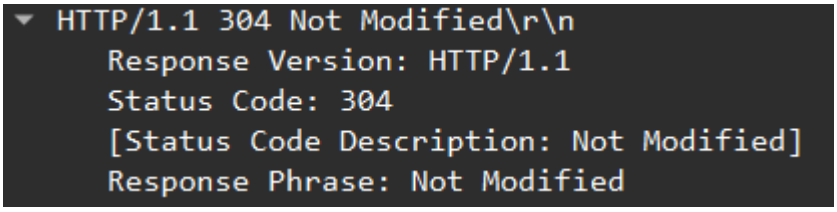

图 2-12 较晚的 HTTP 状态代码

（3）利用 Wireshark 分析 TCP 协议

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。

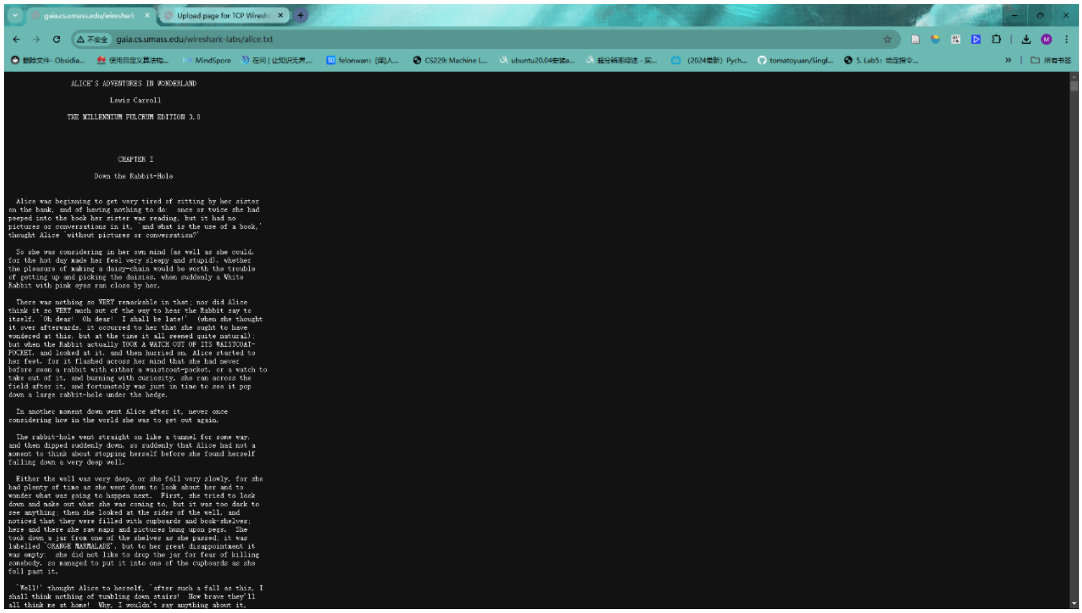

图 3-1 alice.txt 网页

2）打开 http://gaia.cs.umass.edu/wireshark-labs/TCP-Wireshark-file1.html ，如图 6-6 所示， 窗口如下图所示。在 Browse 按钮旁的文本框中输入保存在你的主机上的文件 ALICE'S ADVENTURES IN WONDERLAND 的全名（含路径），此时不要按“Upload alice.txt file”按钮。

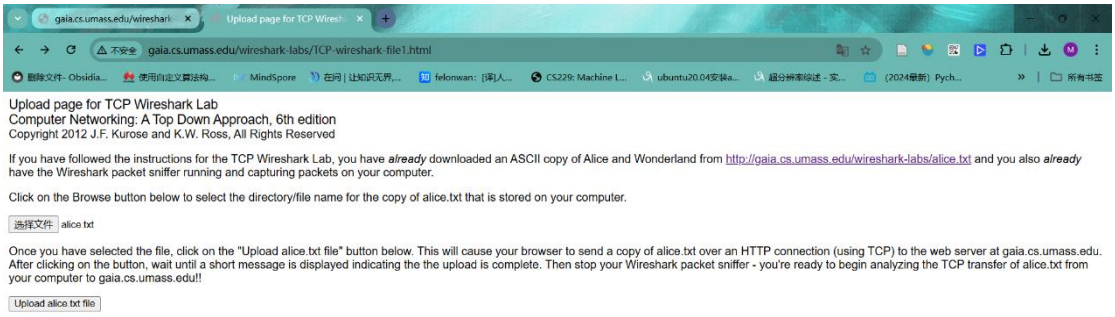

图 3-2 上传网页

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。

图 3-3 上传成功网页

5）停止俘获。

B. 浏览追踪信息

在显示筛选规则中输入“tcp”,可以看到在本地主机和服务器之间传输的一系列 tcp 和 http 报文，你应该能看到包含 SYN 报文的三次握手。也可以看到有主机向服务器发送的一个 HTTP POST 报文和一系列的“http continuation” 报文。

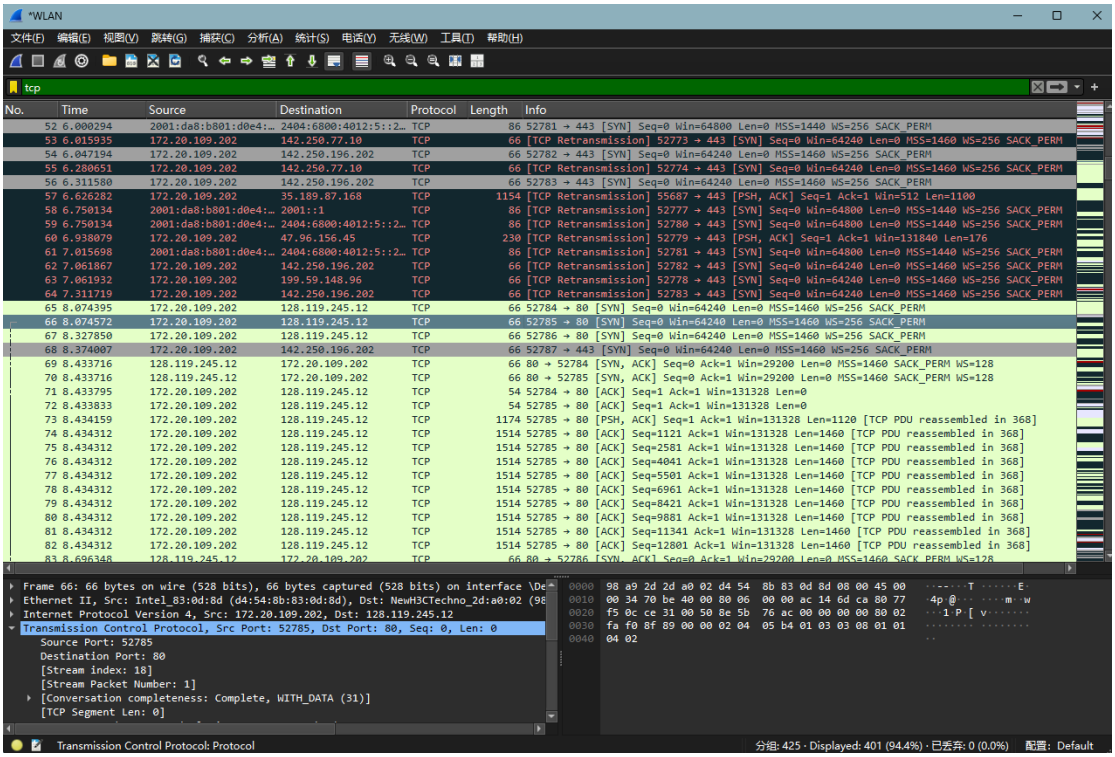

图 3-4 Wireshark 捕获

根据操作思考一下问题：

➢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？

向 gaia.cs.umass.edu 服务器传送文件的客户端主机的 IP 地址是 172.20.109.202，TCP 端口号是 52684。

➢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？

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。

C. TCP 基础

根据操作思考一下问题：

➢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？

客户服务器之间用于初始化 TCP 连接的 TCP SYN 报文段的序号是 0xd0f36fe2，在该报文段中，是用 SYN 标志位是否为 1 标示该报文段是 SYN 报文段的。

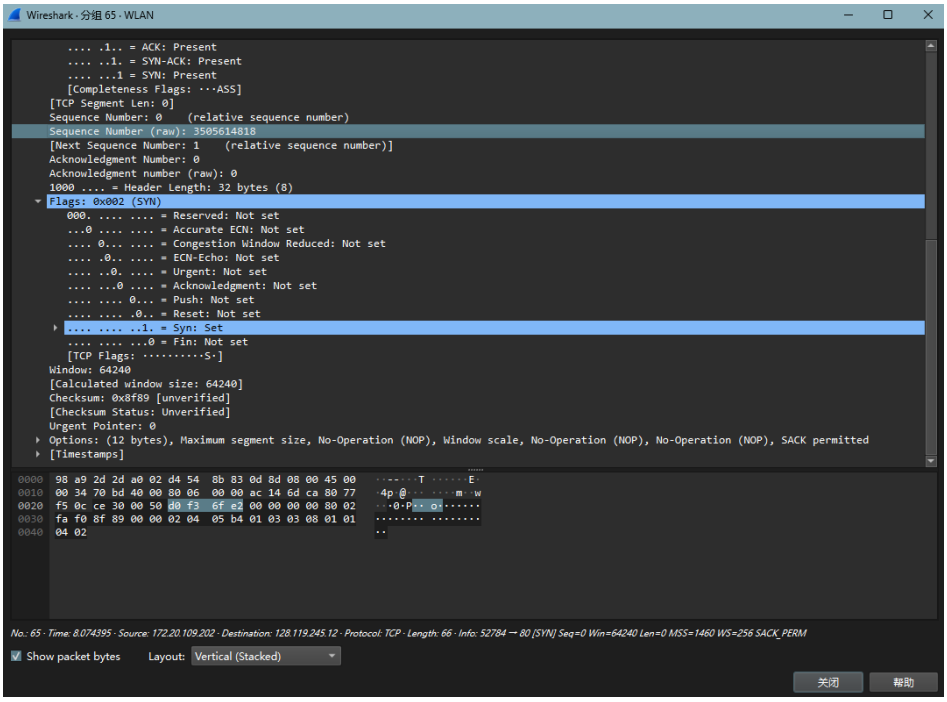

图 3-5 TCP SYN 报文段

➢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 0x2099bdcd;该报文段中，Acknowledgement 字段的值是 0xd0f36fe3;gaia.cs.umass.edu 服务器是将 SYN 报文段序号+1 确定的这个值。在该报文段中，可以通过 SYN 和 ACK 标志位都为 1 标识该报文段。

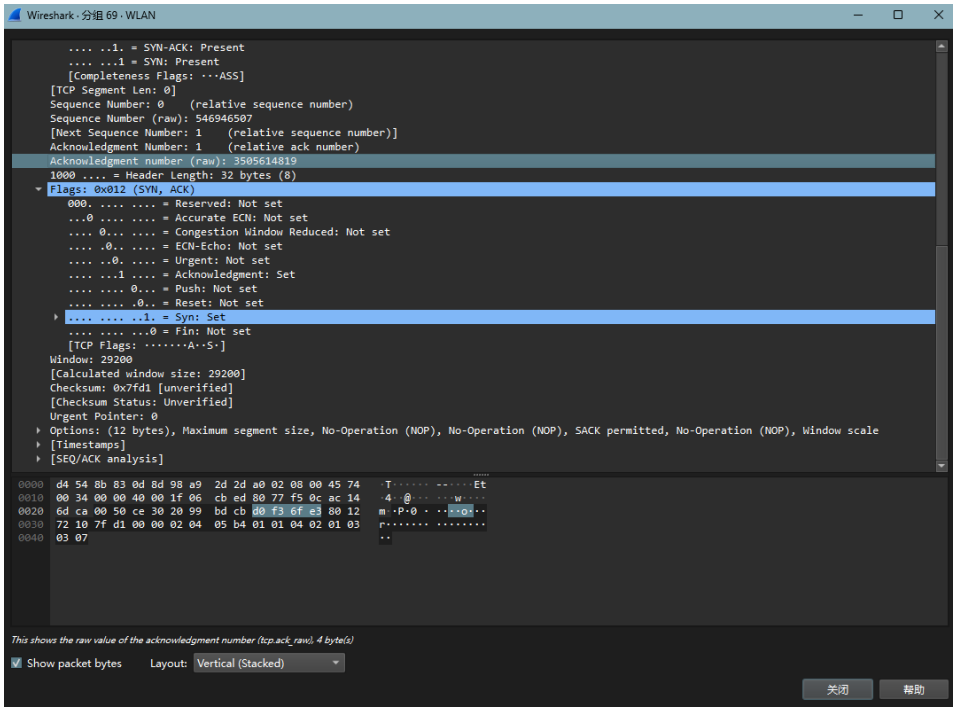

图 3-6 SYNACK 报文段

➢你能从捕获的数据包中分析出 tcp 三次握手过程吗？

可以分析出三次握手

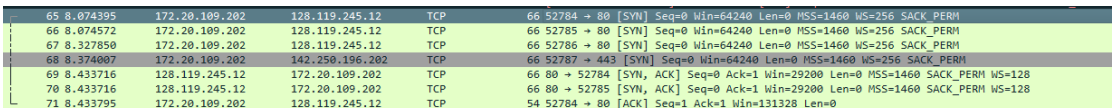

图 3-7 三次握手

➢包含 HTTP POST 命令的 TCP 报文段的序号是多少？

包含 POST 命令的 TCP 报文段序号为 8e5b765d

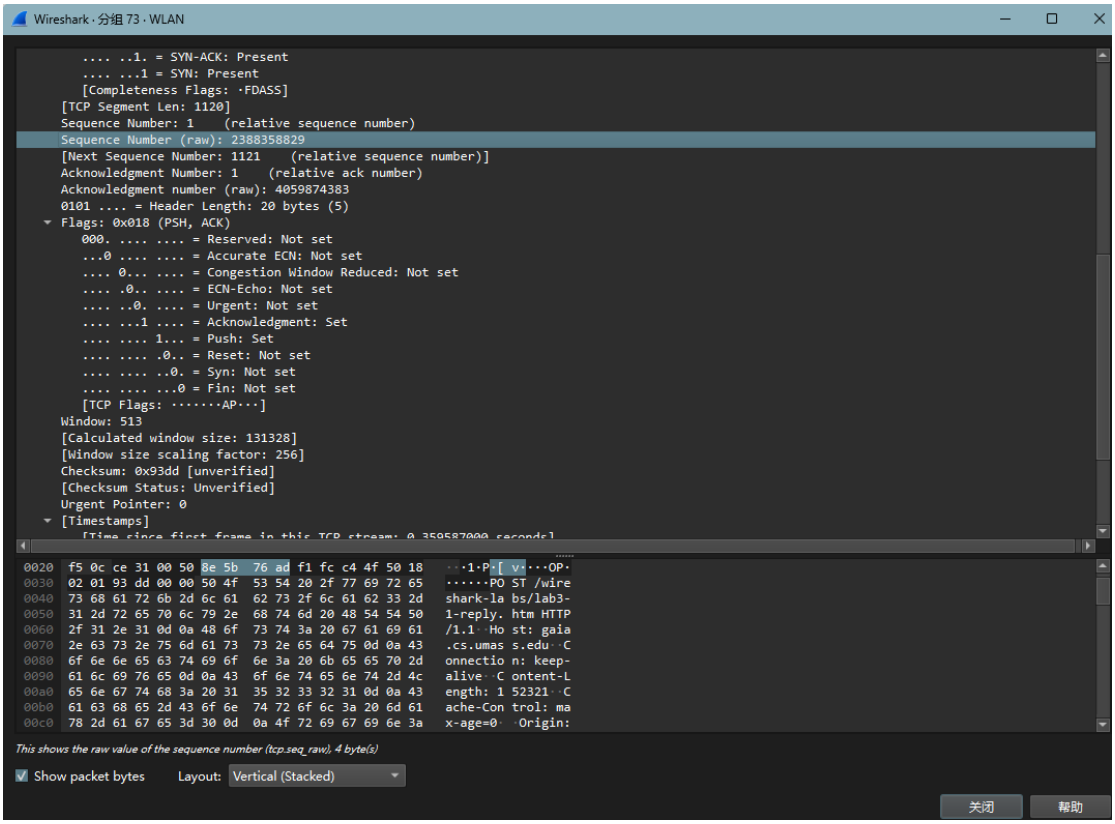

图 3-8 POST 命令的 TCP 报文段

➢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？

那么该 TCP 连接上的第六个报文段的序号是 8e5b91dd；发送时间：该报文段于 TCP 三次握手之后，第四次挥手之前发送的；该报文段所对应的 ACK 是在第三次握手的时候接受的。

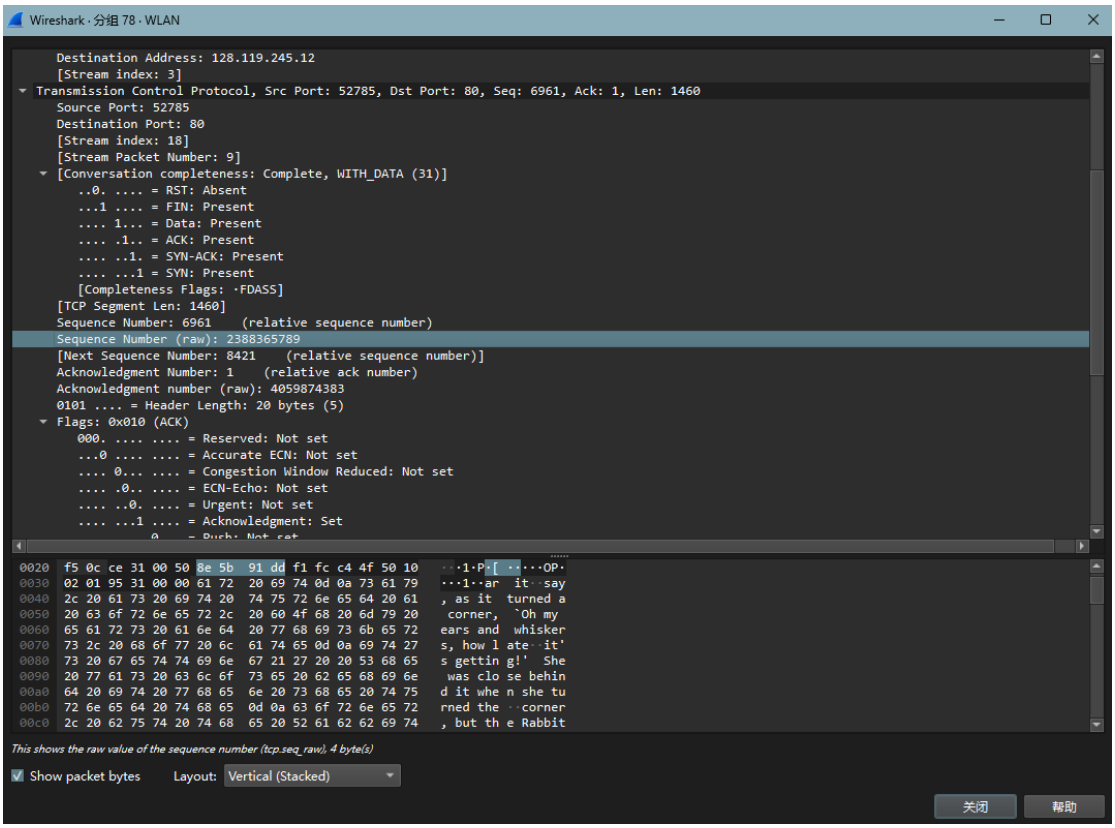

图 3-9 第六个报文段

➢前六个 TCP 报文段的长度各是多少？

前六个 TCP 报文段的长度各是 1174、1514、1514、1514、1514、1514。

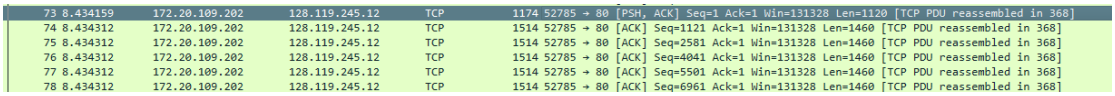

图 3-10 前六个 TCP 报文段长度

➢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？

在整个跟踪过程中，接收端公示的最小的可用缓存空间是 513。限制发送端的传输以后，发现接收端的可用缓存空间在很多时候都是在递增的，最终可用缓存为 131328。由此可见，接收端的缓存是足够的。

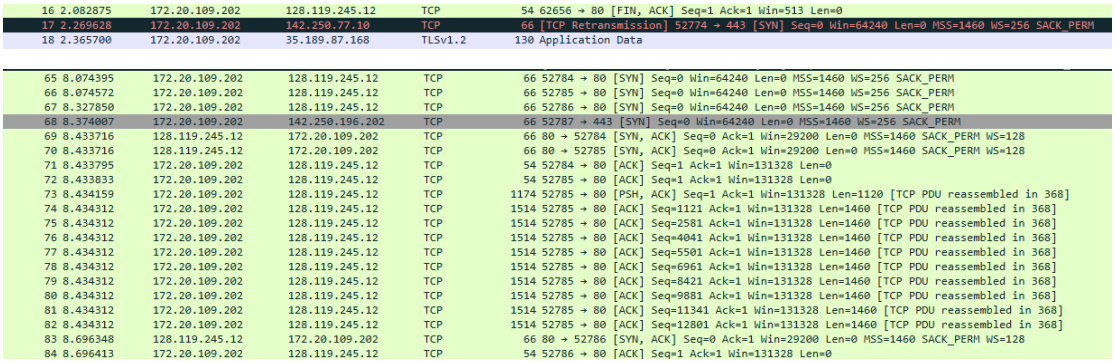

图 3-11 最小缓存空间（上）和最大缓存空间（下）

➢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？

在跟踪文件中没有重传的报文段，进行判断的依据是通过观察客户端的分组序号，可以发现分组序号 Frame 是一直在增长，没有出现过重复的序号的，因此可以判断没有重传的报文段。

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

TCP 连接的 throughput 是 153442B/21.119336 = 7265.47Bps。

```
16 2.082875     172.20.109.202     128.119.245.12     TCP          54 62656 → 80 [FIN, ACK] Seq=1 Ack=1 Win=513 Len=0

397 23.202211   172.20.109.202     128.119.245.12     TCP          54 52785 → 80 [ACK] Seq=153442 Ack=779 Win=130560 Len=0
```

图 3-12 throughout 计算

（4）利用 Wireshark 分析 IP 协议

A. 通过执行 traceroute 执行捕获数据包

1）启动 Wireshark 并开始数据包捕获

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。得到的 pingplotter 窗口如图 4-1 所示。

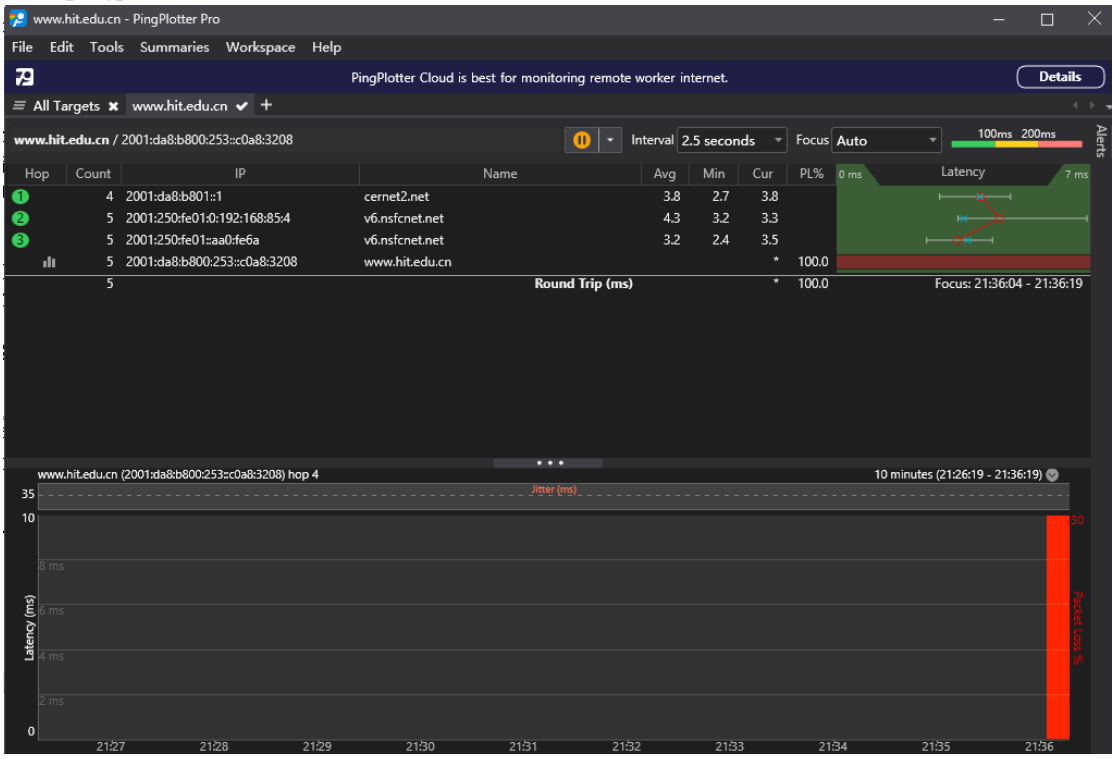

图 4-1 pingplotter 窗口

3）Edit->Options->Packet，然后将 Packet Size(in bytes,default=56)域改为 2000，这样将发送一系列大小为 2000 字节的包。然后按下“Resume”按钮。

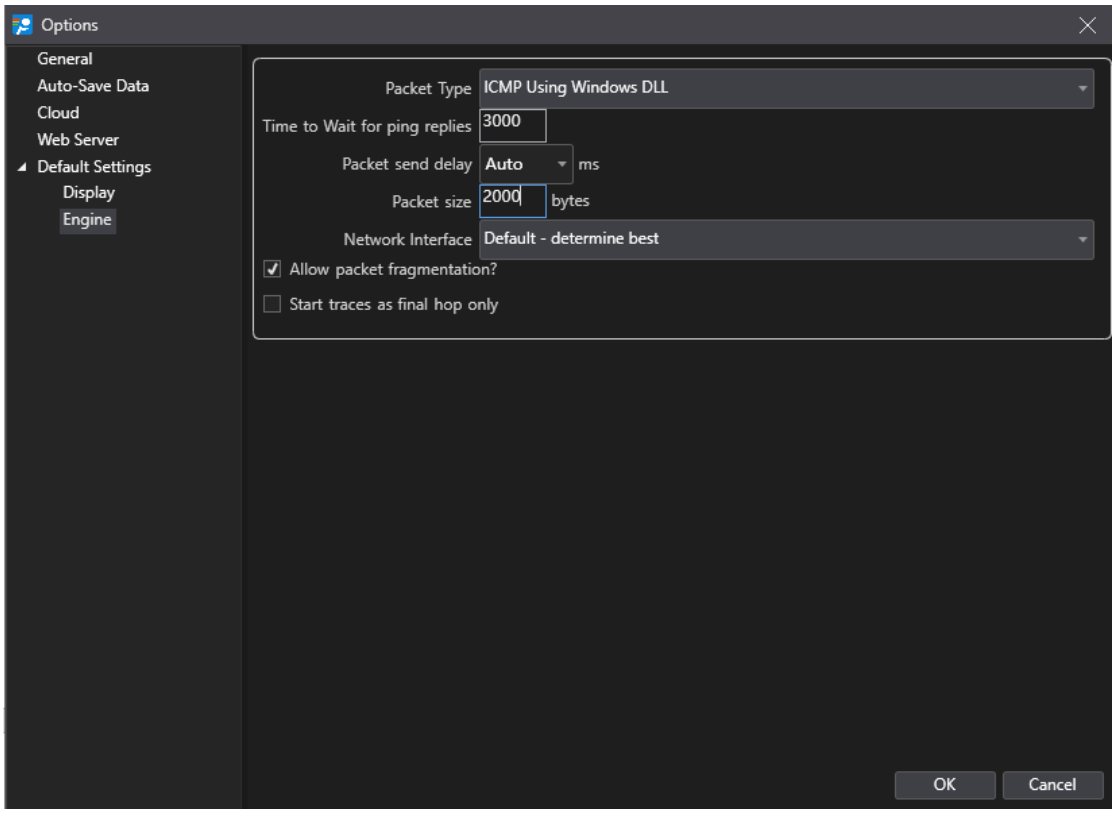

图 4-2 Packet Size 为 2000（设置）

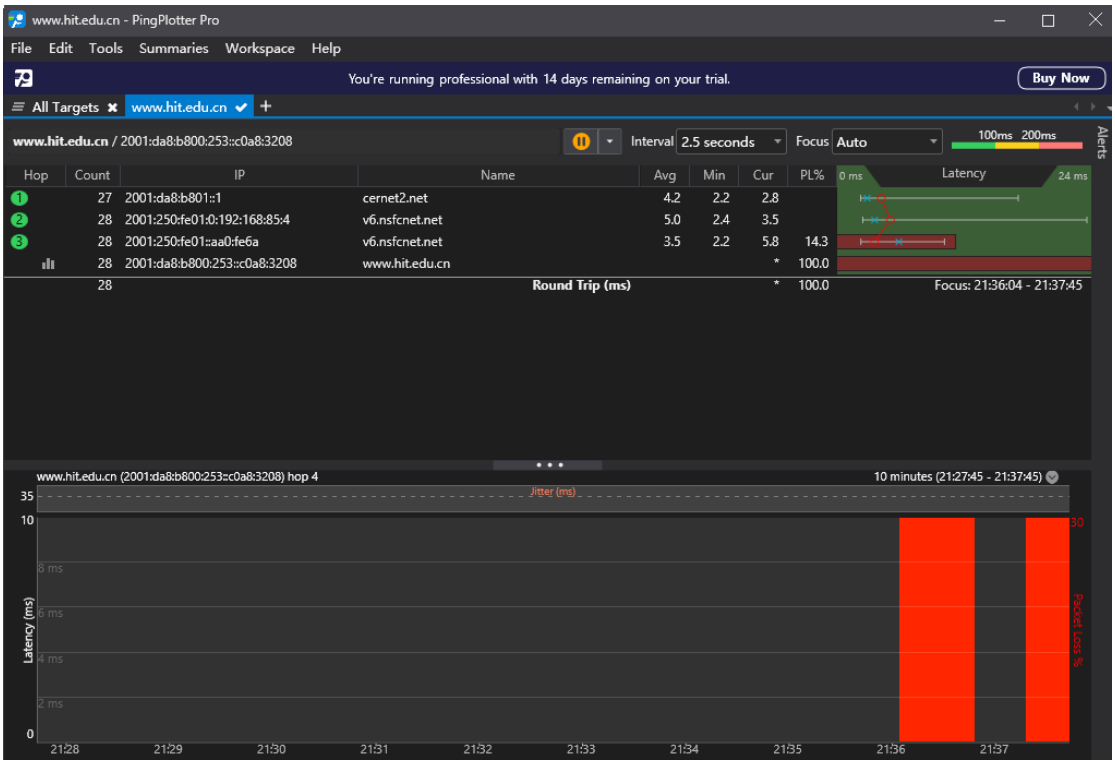

图 4-3 Packet Size 为 2000（运行）

4）最后，将 Packet Size(in bytes,default=56)域改为 3500，发送一系列大小为 3500 字节的包。然后按下“Resume”按钮。

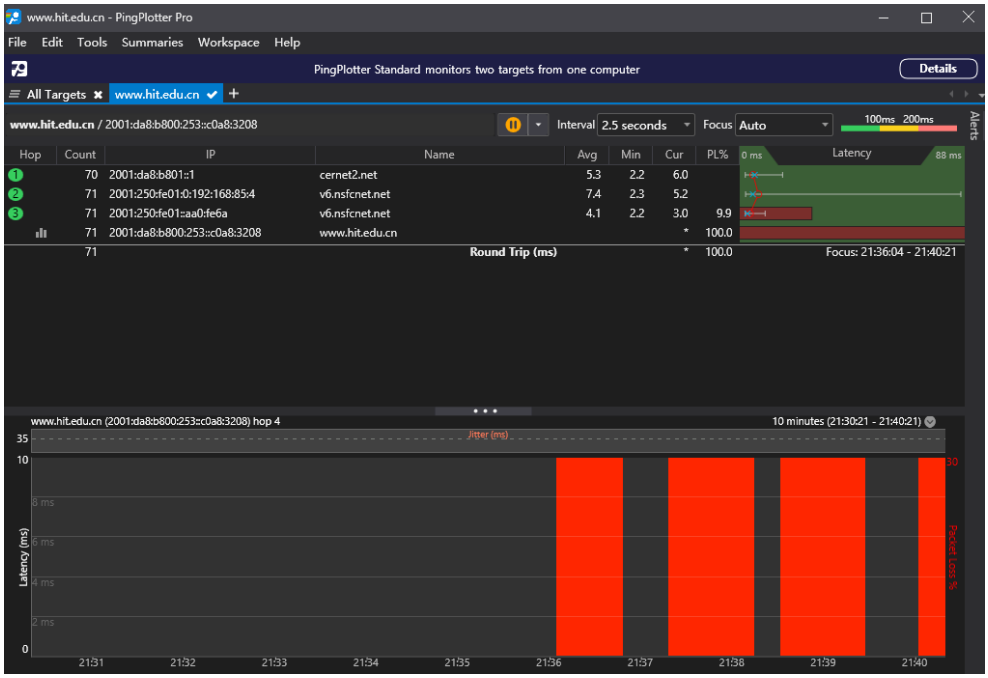

图 4-4 Packet Size 为 3500（运行）

5）停止 Wireshark 的分组捕获。

B. 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，如图 4-5 所示。

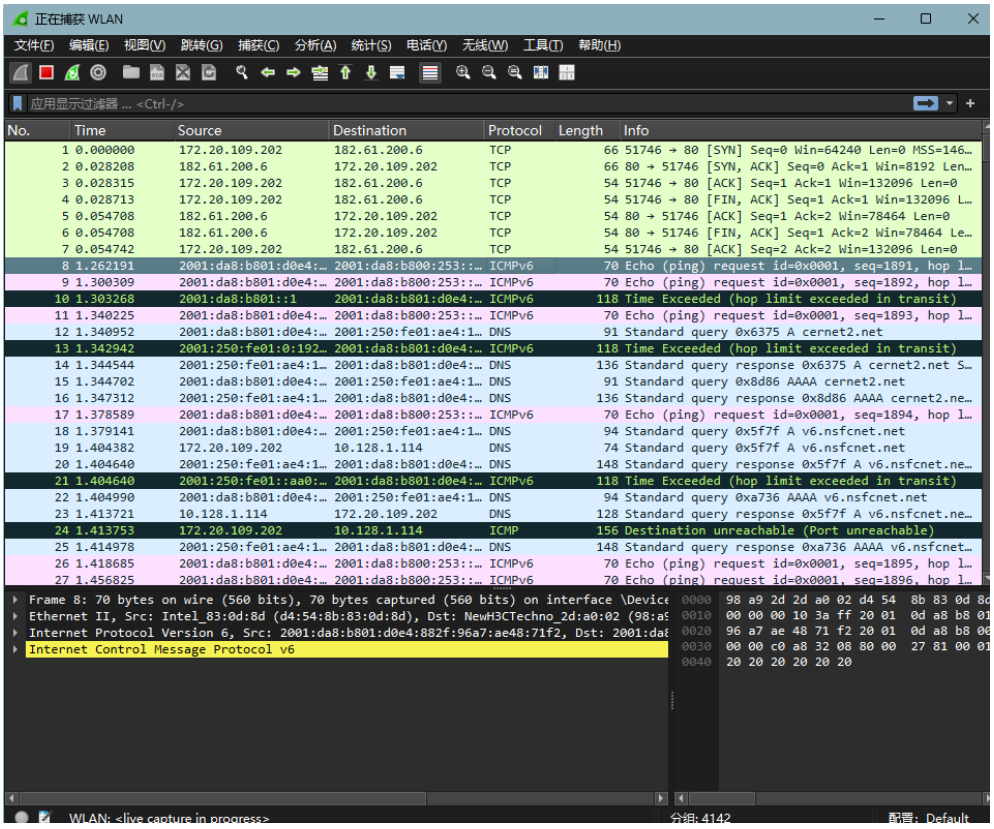

图 4-5 Wireshark 捕获 ICMP

思考下列问题：

➢ **你主机的 IP 地址是什么？**

主机地址为：2001:da8:b801:d0e4:882f:96a7:ae48:71f2

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

上层字段的值为 58。

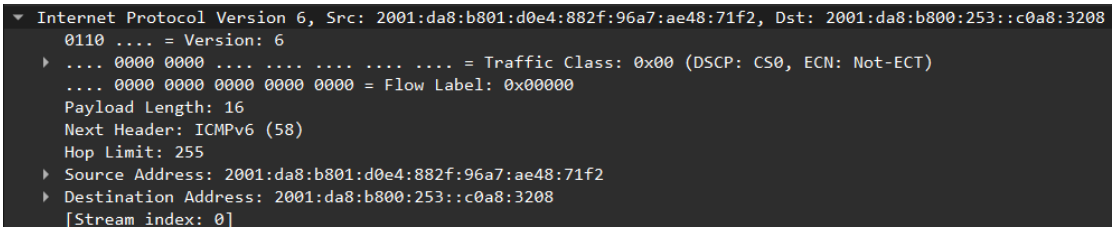

图 4-6 上层协议字段

➢ **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 字节。IP 包的净载为 Total Length-Header Length=142B-20B=122B

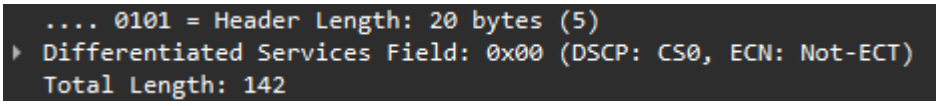

图 4-7 IP 数据包

➢ **该 IP 数据包的净载大小的？**

IP 包的净载为 Total Length-Header Length=142B-20B=122B

➢ **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，因为 More fragments 等于 0 表示后面无分段，所以分片位移为 0。

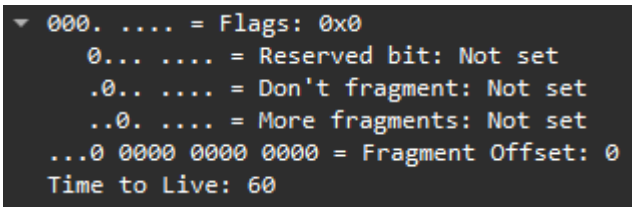

图 4-8 IP 分片

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）

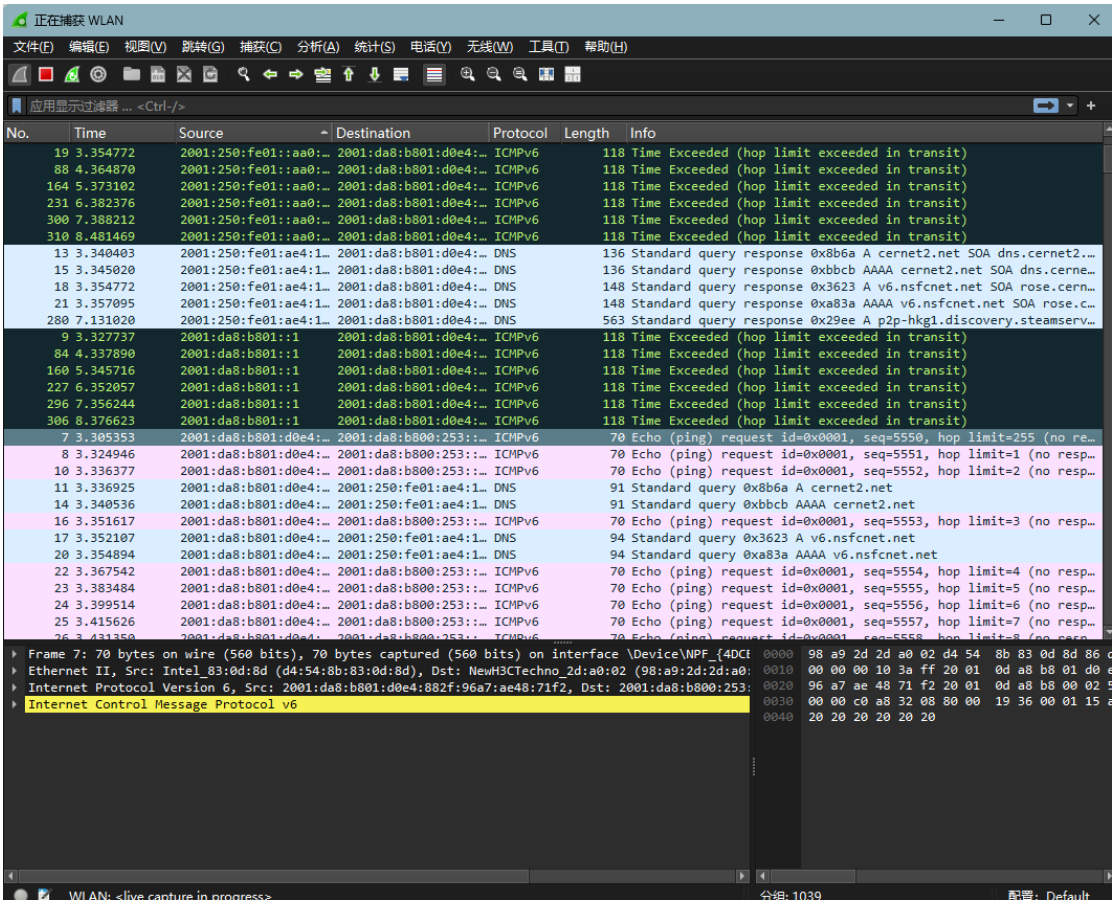

图 4-9 数据包按源 IP 地址排序

思考下列问题：

➢ **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？**

ID、TTL、Header checksum 三个字段总在改变。

➢ **哪些字段必须保持常量？哪些字段必须改变？为什么？**

ID 是鉴别码，用于区分不同的数据包，必须改变；TTL 来自于 traceroute 的要求，用来测试路径上的路由信息，必须改变；Header Checksum 是首部校验和，前面的字段改变时，也必须改变。除了上述 3 个字段其余均可以保持常量。

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

16 位，在固定范围中加一递增。

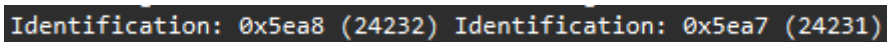

图 4-10 IP 数据包 Indentification 字段值

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。

思考下列问题：

➢ **Identification 字段和 TTL 字段的值是什么？**

Identification字段值为0xda33，TTL字段为193。

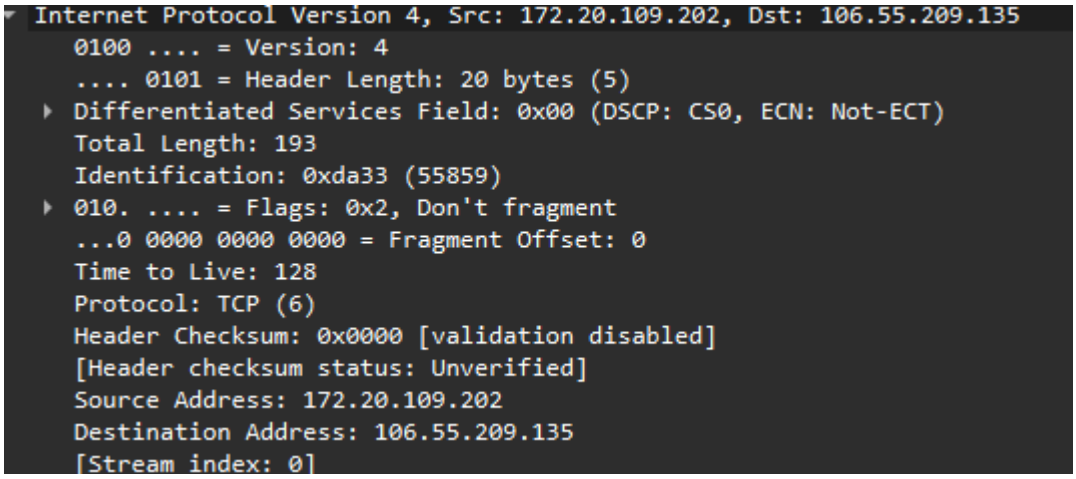

图 4-11 Identification 字段和 TTL 字段

➢ 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？

最近的路由器（第一跳）返回给你主机的ICMP Time-to-live exceeded消息中TTL保持不点，ID 字段不断改变：第一跳路由器设置TTL字段为RFC指定的值，因此始终保持不变；而ID值标识每一个IP字段，是唯一的，因此不断改变。

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？

该消息被分解成两片。

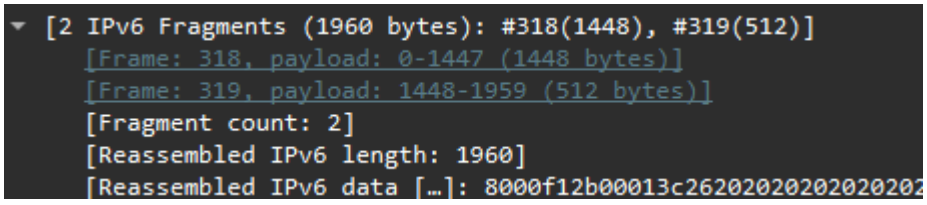

图 4-12 消息两片分解

➢ 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

More fragments 为 1 表示分片且不是最后一个片，该分片的长度是 1448bytes。

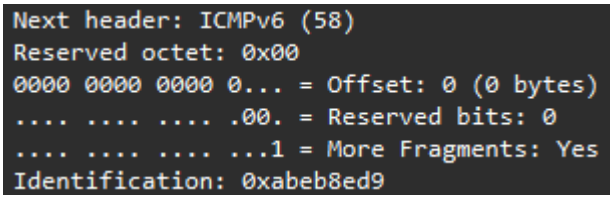

图 4-13 第一个 IP 分片信息

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？

原始数据包被分成了 3 片。

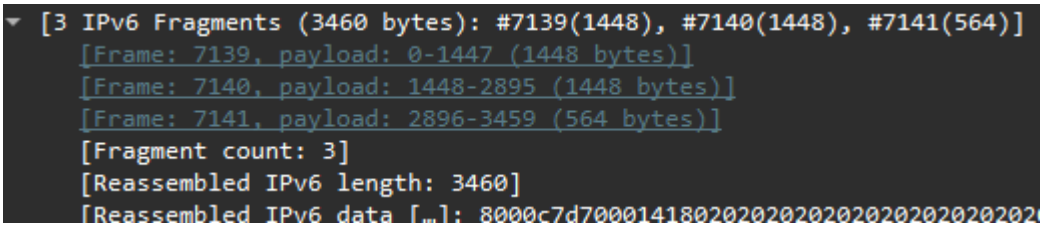

图 4-14 原始数据包 3 片分解

➢这些分片中 IP 数据报头部哪些字段发生了变化？

这些分片中 IP 数据报头部标志位 More fragments 变化、片偏移变化。第一个和第二个分片标志位 More fragments 为 1 标识后面还有分片。后两个分片 offset 变为 1448 和 2896。

（5）利用 Wireshark 分析 DNS 协议

1）打开浏览器键入:www.baidu.com

图 5-1 浏览器访问 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，捕获的 DNS 报文如图 5-2 所示。

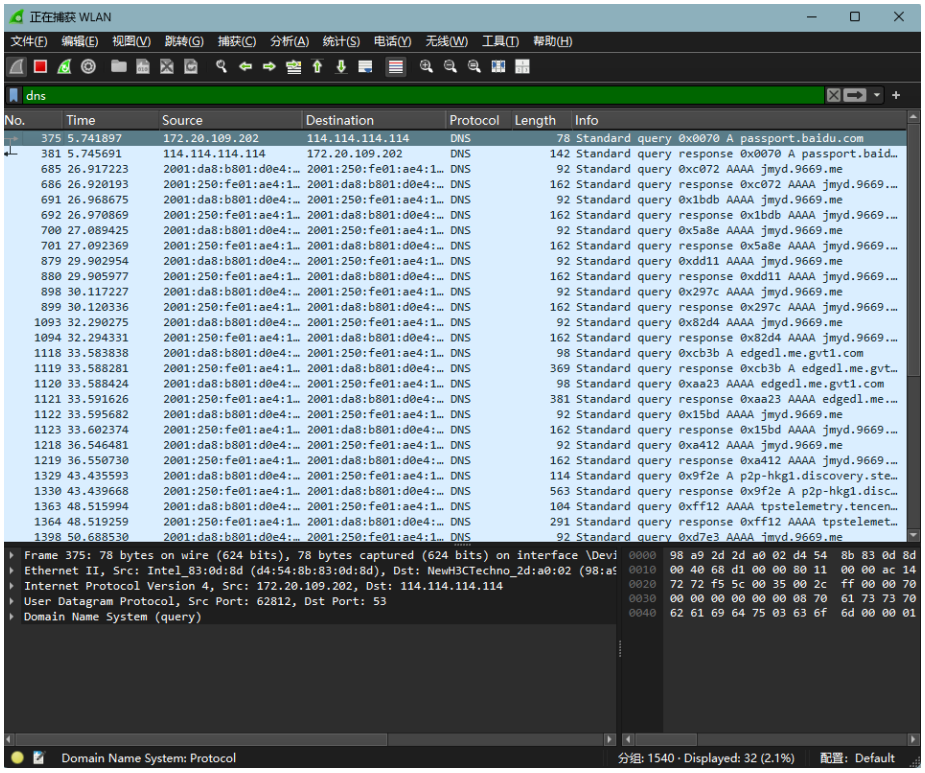

图 5-2 Wireshark 捕获 DNS

（6）利用 Wireshark 分析 UDP 协议

1）启动 Wireshark，开始分组捕获

2）发送 QQ 消息给你的好友

3）停止 Wireshark 组捕获

4）在显示筛选规则中输入“udp”并展开数据包的细节

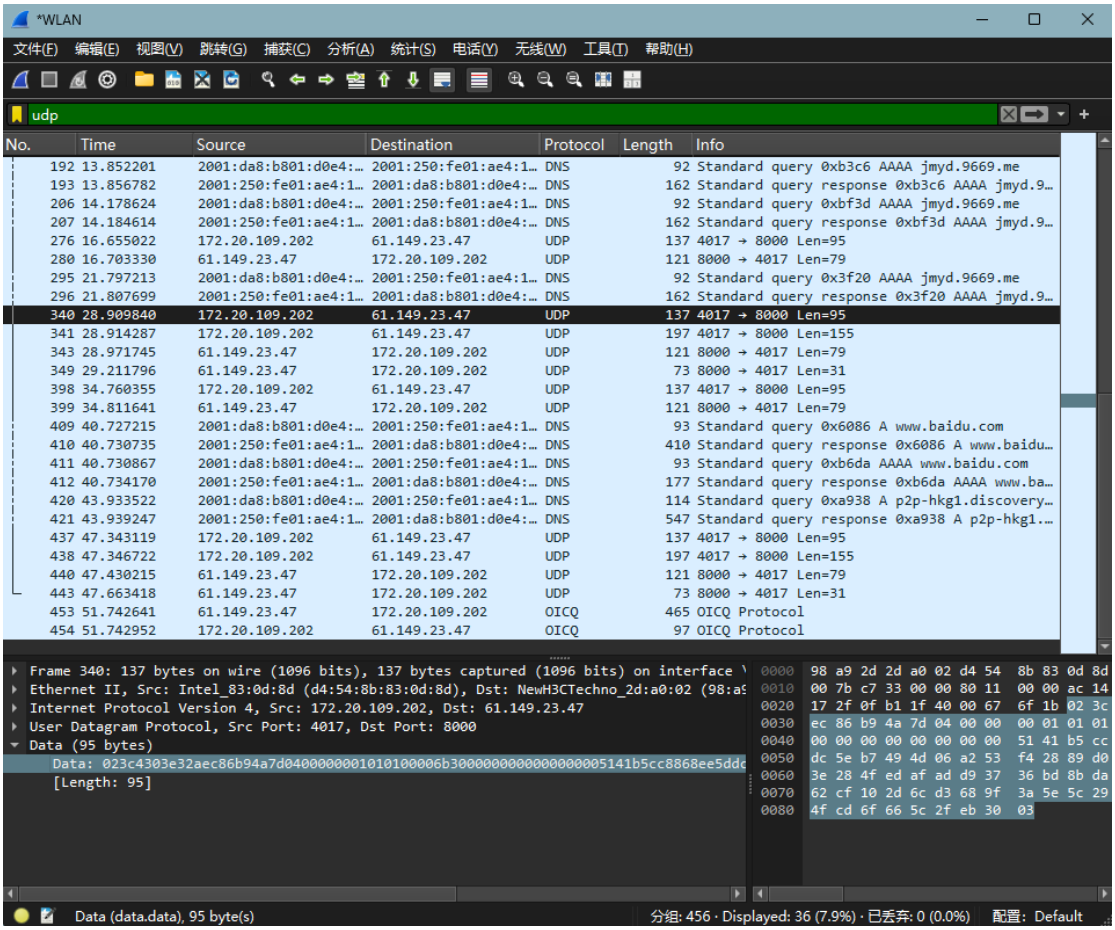

图 6-1 Wireshark 捕获 UDP

分析 QQ 通讯中捕获到的 UDP 数据包。根据操作思考以下问题：

➢消息是基于 UDP 的还是 TCP 的？

消息是基于 UDP 的。

➢你的主机 ip 地址是什么？目的主机 ip 地址是什么？

主机 IP 地址是 127.20.109.202，目的主机 IP 地址是 61.149.23.47。

➢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

主机发送 QQ 消息的端口号是 4017,QQ 服务器的端口号是 8000。

➢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？

数据报格式：源端口号 2B，目的端口号 2B，UDP 段长度 2B，校验和 2B

➢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？

因为服务器需要返回接收的结果给客户端。因为服务器只能提供一次返回的 ACK，不保证数据一定送达。可以看出，因为 UDP 数据包没有序列号，不能像 TCP 协议先握手在发送数据，因为每次只能发送一个数据报，然后等待服务器响应。

（7）利用 Wireshark 分析 ARP 协议

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

```
管理员: Windows PowerShell
Windows PowerShell
版权所有 (C)  Microsoft Corporation。保留所有权利。

安装最新的 PowerShell，了解新功能和改进！ https://aka.ms/PSWindows

PS C:\Users\Administrator> arp -g

接口： 172.20.109.202 --- 0x9
  Internet 地址         物理地址              类型
  172.20.0.1            98-a9-2d-2d-a0-02     动态
  172.20.255.255        ff-ff-ff-ff-ff-ff     静态
  224.0.0.2             01-00-5e-00-00-02     静态
  224.0.0.22            01-00-5e-00-00-16     静态
  224.0.0.251           01-00-5e-00-00-fb     静态
  224.0.0.252           01-00-5e-00-00-fc     静态
  239.192.152.143       01-00-5e-40-98-8f     静态
  239.255.255.250       01-00-5e-7f-ff-fa     静态
  255.255.255.255       ff-ff-ff-ff-ff-ff     静态
PS C:\Users\Administrator>
```

图 7-1 查看 ARP 缓存

2）在命令行模式下输入： ping 172.20.0.1（或其他 IP 地址）

```
PS C:\Users\Administrator> ping 172.20.0.1

正在 Ping 172.20.0.1 具有 32 字节的数据：
来自 172.20.0.1 的回复：字节=32 时间=67ms TTL=255
来自 172.20.0.1 的回复：字节=32 时间=24ms TTL=255
来自 172.20.0.1 的回复：字节=32 时间=64ms TTL=255
来自 172.20.0.1 的回复：字节=32 时间=24ms TTL=255

172.20.0.1 的 Ping 统计信息：
    数据包：已发送 = 4，已接收 = 4，丢失 = 0 (0% 丢失)，
往返行程的估计时间(以毫秒为单位)：
    最短 = 24ms，最长 = 67ms，平均 = 44ms
```

图 7-2 ping 地址

3）启动 Wireshark，开始分组俘获。抓取的数据包大致如下图 7-3 所示。

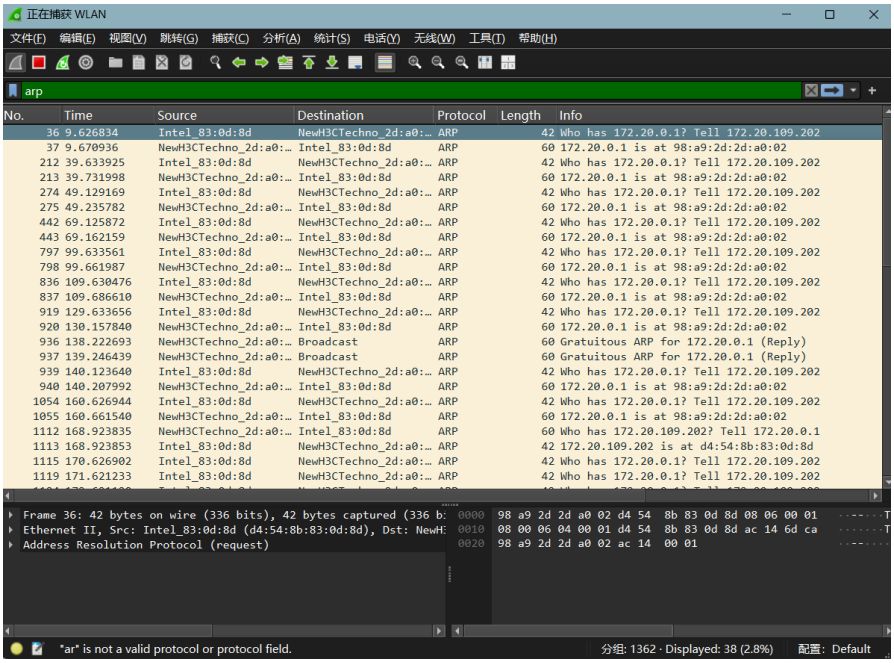

图 7-3 Wireshark 捕获

思考下面问题：

1）说明 ARP 缓存中的每一列的含义。

ARP 缓存中第一列指的是 ARP 协议的缓存的 Internet 地址，第二列是物理地址，第三列是类型，表示是动态类型还是静态类型。

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。

➢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 格式：由 9 部分组成：硬件类型（2B）、协议类型（2B）、硬件地址长度（1B）、协议地址长度（1B）、OP（2B）、发送端 MAC 地址（6B）、发送端 IP 地址（4B）、目的 MAC 地址（6B）、目的 IP 地址（4B）

图 7-4 ARP 格式

➢如何判断一个 ARP 数据是请求包还是应答包？

判断 APR 数据：当 OP 为 1 时是请求包；当 OP 为 2 时是答应包。

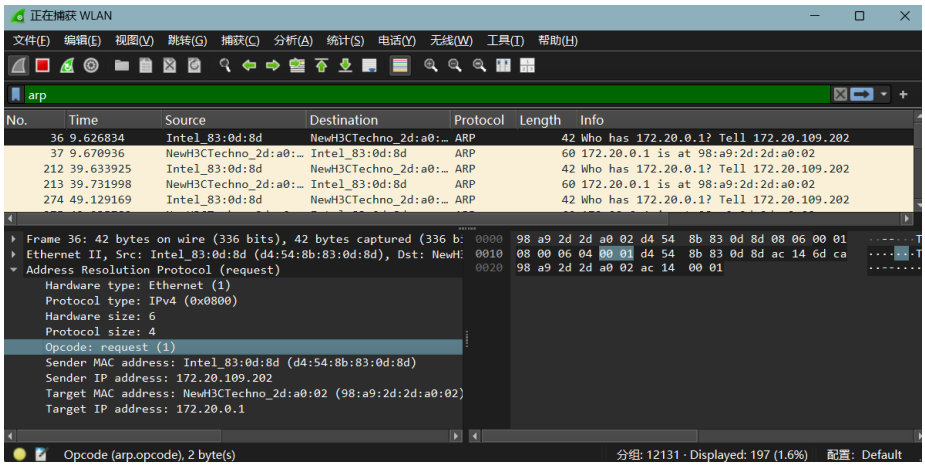

图 7-5 请求包

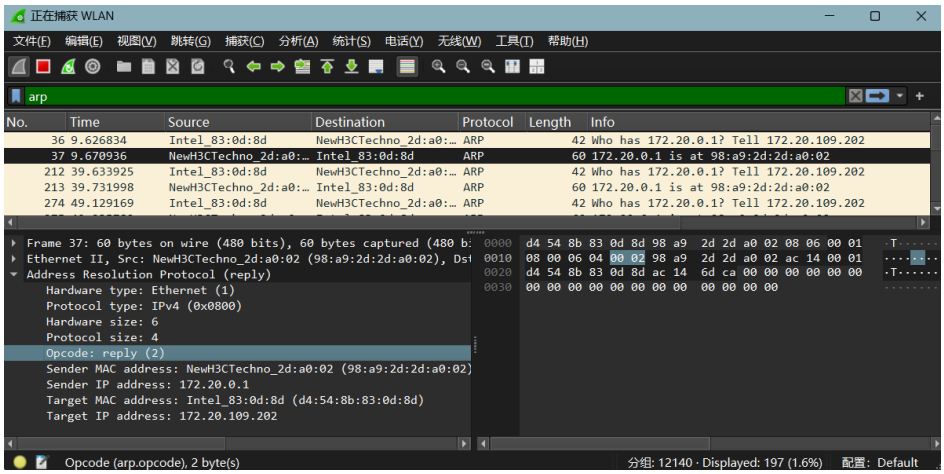

图 7-6 答应包

➢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？

因为查询 ARP 不知道目的 IP 对应的 MAC 地址，所以要广播查询。ARP 响应的时候已经从查询 ARP 中找到发送端 MAC 地址，所以 ARP 响应有一个明确的目的地址。

实验结果：

大部分实验截图在实验过程中已经体现。下面是每个协议的分析。

（1）HTTP协议

1）基本信息

HTTP（超文本传输协议）是一种基于 TCP 的应用层传输协议，旨在规范客户端与服务器之间的数据传输。设计目标是提供一种简单灵活的方式，使用户能够在互联网上访问各种资源。

2）HTTP协议特点

a）无连接：在每次 HTTP 通信中，客户端和服务器会建立并断开 TCP 连接。这意味着每个请求都需要重新建立连接，虽然这样可能导致延迟，但简化了服务器的资源管理。

b）无状态：HTTP 是一种无状态协议，意味着每个请求都是独立的，服务器不会保留之前请求的状态。这种设计有助于提高效率，但也意味着如果需要维护会话状态，开发者需采用其他技术（如 cookies 或 session）。

c）灵活性：HTTP 协议支持多种类型的数据传输，通过 Content-Type 头部字段标识数据类型。这使得它能够传输文本、图像、视频等多种格式的内容，适应不同的应用需求。

d）简单快速：HTTP 请求通常由请求方法、资源路径和协议版本构成。常用的方法包括 GET、HEAD 和 POST，不同方法规定了客户端与服务器的交互方式。由于 HTTP 协议的简洁性，开发者能够快速实现服务器功能，提升了通信速度。

3）HTTP请求状态行

a）请求方法：指明客户端希望对指定资源进行的操作，如 GET（请求资源）、POST（提交数据）等。

b）URL 字段：指定请求的资源地址。

c）HTTP 版本：表明所使用的 HTTP 协议版本，如 HTTP/1.1。

4）HTTP响应状态行

a）HTTP 版本：表明所遵循的 HTTP 协议版本。

b）状态码：一个三位数字，表示请求的处理结果，例如 200（成功）、404（未找到）、500（服务器错误）等。

c）状态描述：对状态码的文本描述，提供更详细的上下文信息。

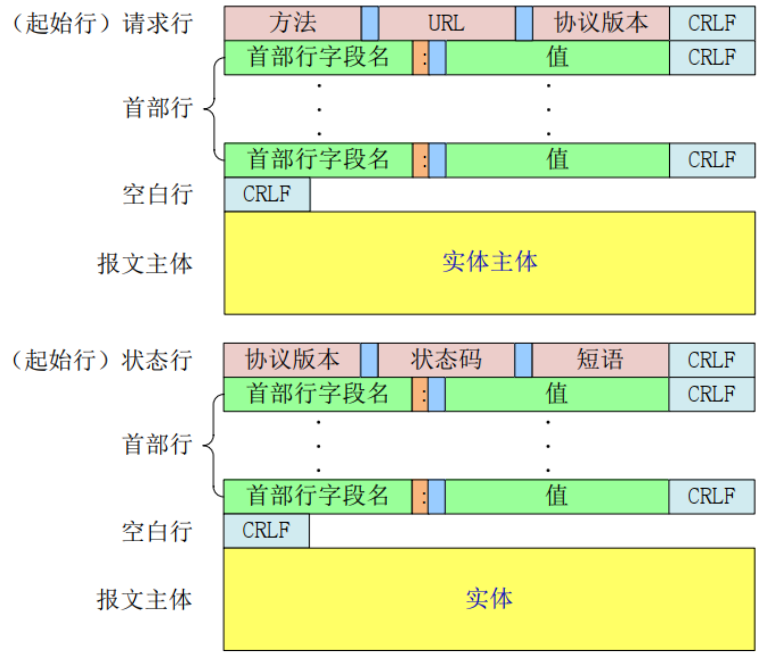

图 1 HTTP 协议请求消息（上）和响应消息（下）格式

（2）TCP协议

1）基本信息

TCP（传输控制协议）是一种面向连接的运输层协议。在应用程序使用 TCP 进行数据传输之前，必须首先建立连接；数据传输完成后，连接也必须被释放。这种连接的建立和释放机制确保了数据传输的可靠性和有效性。

2）TCP协议特点

a）点对点连接：每条 TCP 连接只能有两个端点，形成一对一的连接。这意味着在一个 TCP 连接中，只有两个通信方能够直接进行数据交换，避免了多点通信带来的复杂性和潜在问题。

b）可靠交付：TCP 提供可靠的数据传输服务，确保通过 TCP 连接传送的数据是无差错、不丢失、不重复，并且按序到达。TCP 通过序列号、确认应答、重传机制和流量控制等手段，实现了对数据传输的高可靠性保障。

c）全双工：TCP 协议支持全双工通信，即通信双方的应用程序可以在任意时刻发送和接收数据。这种特性使得 TCP 适用于需要实时交互的应用，如视频会议和在线游戏。每个端点都配有发送缓存和接收缓存，确保数据的高效存储与转发。

d）字节流特性：TCP 被视为一个面向字节流的协议，意味着它将数据视为一个连续的字节序列，而不是一个个独立的数据包。这一特性使得应用程序可以灵活处理数据，无需关心数据的边界，从而简化了编程复杂性。

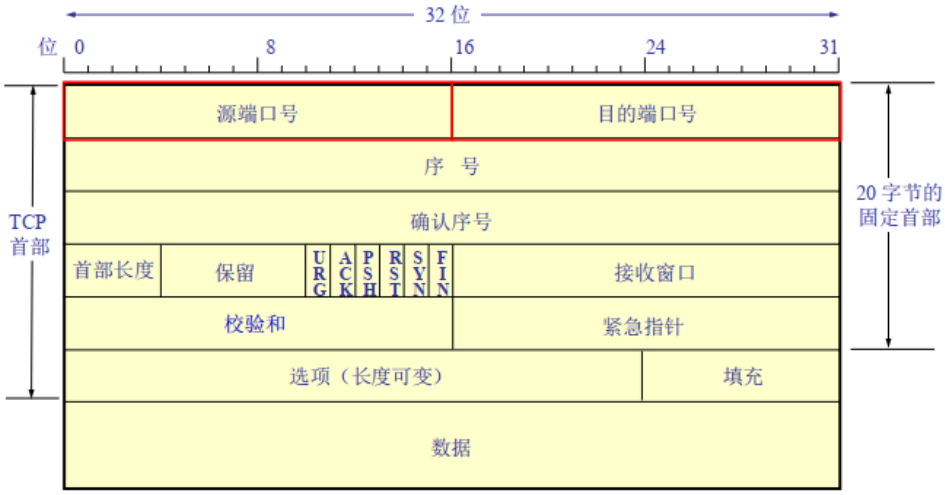

图 2 TCP 协议段结构

（3）IP协议

1）基本信息

IP（互联网协议）是 TCP/IP 协议族的核心组成部分，负责数据在网络中的传输和路由。IP 协议主要涵盖两个方面：

a）IP 头部信息：每个 IP 数据报都包含头部信息，记录源 IP 地址和目的 IP 地址。这些信息不仅用于标识数据包的发送和接收方，还指导数据的分片与重组，并指定部分通信行为。

b）IP 数据报的路由和转发：数据报的路由和转发过程发生在目标机器之外的所有主机和路由器上。这些设备根据路由算法判断数据报是否应当转发以及如何进行转发。

2）IP协议特点

a）无状态：IP 协议不保留任何会话信息。每个数据包都是独立处理的，这意味着网络中的每个设备不需要知道数据包的历史。

b）无连接：IP 协议不建立端到端的连接，数据包的发送不需要在发送方和接收方之间进行任何的预先协商。

c）不可靠：IP 协议本身不保证数据包的送达、顺序或完整性。这些功能通常由更高层的协议（如

TCP）来实现。

3）IP应用

a）路由选择：根据目标 IP 地址选择合适的路径，将数据包发送到目的地。

b）存储转发：在网络设备（如路由器）中临时存储数据包，并根据路由信息决定是否转发。

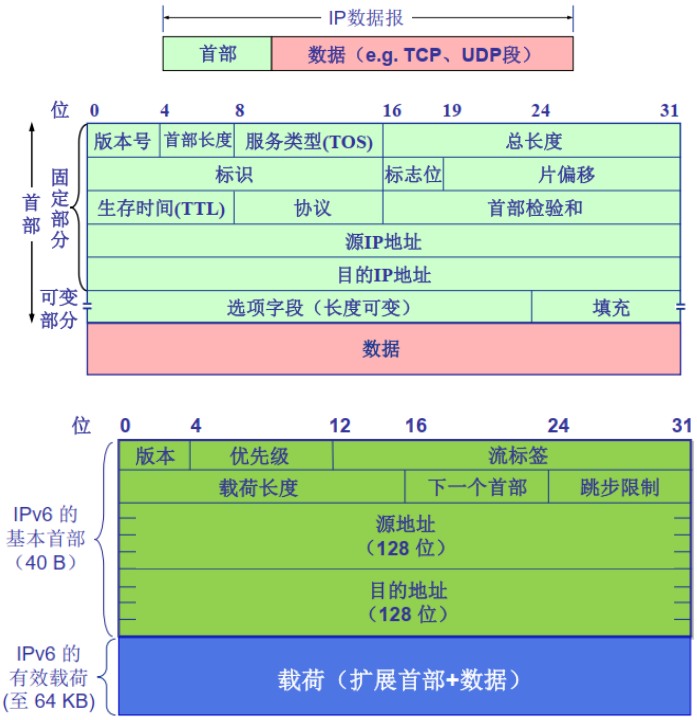

图 3 IPV4（上）和 IPV6（下）数据报格式

（4）DNS协议

1）基本信息

DNS（域名系统）是一个应用层协议，其主要功能是将人类可读的域名转换为机器可读的 IP 地址（例如，192.0.2.44）。DNS 使得用户能够使用易记的域名访问网站，而无需记住复杂的数字 IP 地址。

2）工作原理

a）名称解析：当用户在浏览器中输入域名时，DNS 协议负责解析这个域名，并将其转换为相应的 IP 地址。这个过程通常包括多个步骤，例如查询本地 DNS 缓存、递归查询和授权 DNS 服务器查询。

b）反向解析：除了将域名转换为 IP 地址外，DNS 还支持反向解析，即将 IP 地址转换回域名。这一功能在某些网络安全和管理应用中非常有用。

3）DNS协议特点

a）基于 UDP 和 TCP：DNS 协议主要基于 UDP（用户数据报协议），因为 UDP 的无连接特性使得查询快速且高效。但在某些情况下（如数据量较大或需要保证可靠性时），DNS 也可以通过 TCP（传输控制协议）进行传输。

b）默认端口：DNS 服务的默认端口为 53。这一端口用于监听和响应 DNS 查询请求。

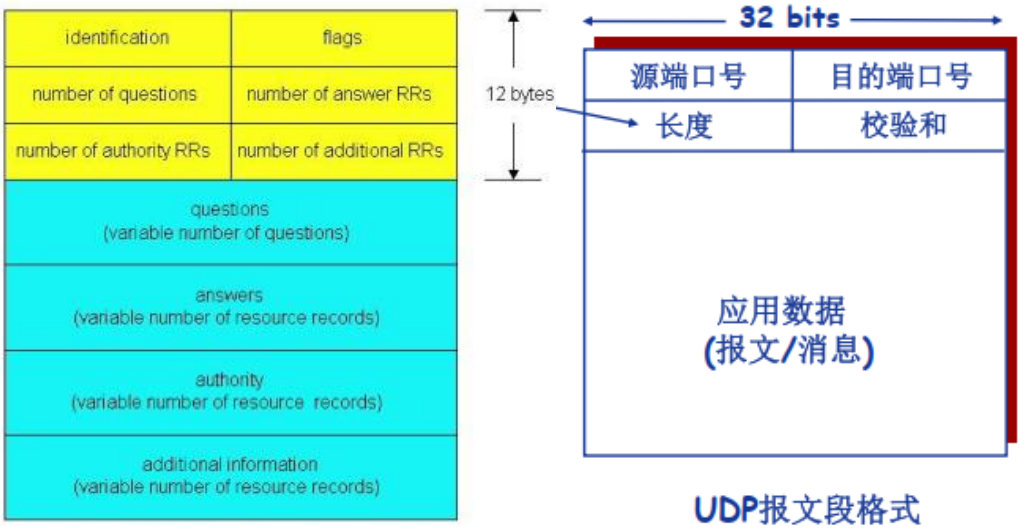

图 4 DNS 协议格式（左）和 UDP 报文段格式（右）

（5）UDP协议

1）基本信息

UDP（用户数据报协议）是一个简单的面向消息的传输层协议。它提供了基本的功能，如标头和有效负载的完整性验证（通过校验和），但并不保证消息的可靠传递。因此，UDP 有时被称为不可靠的数据报协议。发送数据后，UDP 不会维护任何消息状态，适用于对实时性要求高但对可靠性要求不高的应用场景。

2）工作原理

a）无连接特性：UDP 是一种无连接协议，意味着在数据传输之前不需要建立连接。每个数据包（称为数据报）都是独立的，发送方和接收方之间没有建立会话的必要。

b）数据包发送：UDP 将数据划分为数据报，每个数据报在传输过程中不会进行重组，接收方可能会收到重复或错序的数据报。这使得应用层必须自行处理这些问题。

3）UDP协议特点

a）简单轻量：UDP 协议的设计相对简单，头部开销小（仅 8 字节），因此适用于需要快速传输且对带宽敏感的应用（如视频流、在线游戏和实时通信）。

b）低延迟：由于没有连接建立、确认应答和重发机制，UDP 能够实现低延迟的数据传输。

c）不可靠：UDP 不保证数据的送达，不会对丢失、重复或错序的数据报进行处理。

d）缺乏流控制：UDP 不提供流量控制机制，这意味着在网络拥堵时，数据包可能会丢失而不进行重传。

（6）ARP协议

1）基本信息

ARP（地址解析协议）是用于将 IP 地址解析为 MAC 地址的重要网络传输协议。它在 TCP/IP 分层结构中位于网络层，负责在网络通信中实现 IP 地址与链路层地址（MAC 地址）之间的转换。ARP 确保数据能够在以太网等链路层网络中正确寻址，从而实现源主机与目标主机之间的有效通信。

2）工作原理

a）IP 到 MAC 的映射：当一台主机需要发送数据到另一个主机时，它首先需要知道目标主机的 MAC 地址。主机会通过 ARP 请求广播其 IP 地址，询问网络中哪个主机对应这个 IP 地址。

b）ARP 请求和响应：其他主机接收到这个请求后，如果识别出自己的 IP 地址，就会以 ARP 响应的形式返回其 MAC 地址。这样，源主机便能获取目标主机的 MAC 地址，完成数据包的发送。

3）ARP协议特点

a）无连接特性：ARP 协议本身是无连接的，这意味着它在发送请求时不需要建立连接，能够快速进行地址解析。

b）广播与单播：ARP 请求通常以广播的方式发送，以确保网络中的所有设备都能接收到该请求；而 ARP 响应则是单播，直接发送给请求的主机。

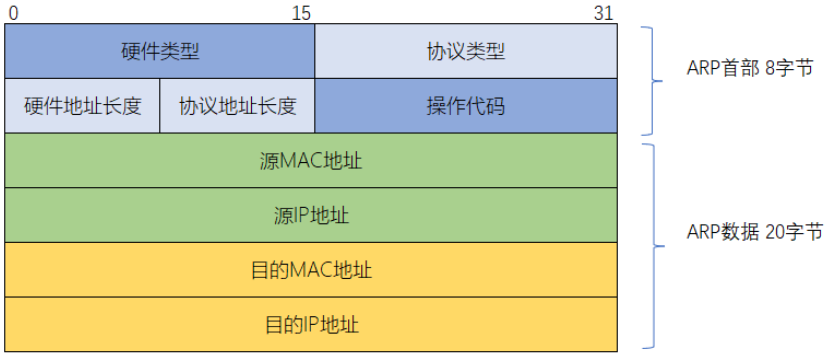

图 5 ARP 协议报文格式

问题讨论：

（1）TCP进行分析时的网址 http://gaia.cs.umass.edu/Wireshark-labs/alice.txt 和 http://gaia.cs.umass.edu/Wireshark-labs/TCP-Wireshark-file1.html 网址打不开，将以上网址中的Wireshark的首字母换成小写wireshark问题解决。

（2）在UDP协议分析中，用到的QQ发送消息时，要事先在QQ登录界面进行设置，在“登录服务器”栏中选择UDP协议，并选好地址和端口号，这样才能在发送消息时，Wireshark 捕获UDP数据包成功。否则发送消息后产生的不是UDP数据包，从而捕获不了。提示：QQ最新版QQ9登录设置栏中没有服务器设置，需要下载老版本的QQ。

（3）ICMP协议是 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（网络通不通、主机是否可达、路由是否可用等）这些控制消息并不传输用户数据，但对于用户数据的传递起着重要的作用。 ICMP 是 IP 的一个组成部分，与 IP 协议、ARP 协议、RARP 协议及 IGMP 协议共同构成 TCP/IP 模型中的网络层。

心得体会：

通过本次实验，我完成了利用 Wireshark 的使用，并且利用 Wireshark 分析 HTTP 协议、TCP 协议、IP 协议、DNS 协议、UDP 协议、ARP 协议，进而熟悉并掌握 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。对应用层、传输层、网络层、链路层的协议有了更加深入的了解，并且对于其结构与工作原理的认识更加深刻。