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	王丙昊		院系	计算机科学与技术	
班级	1603108		学号	1160300302	
任课教师	聂兰顺		指导教师	聂兰顺	
实验地点	格物 207		实验时间	2018.11.21	
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

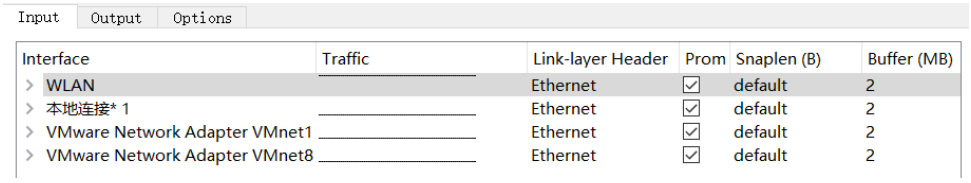
选做内容：

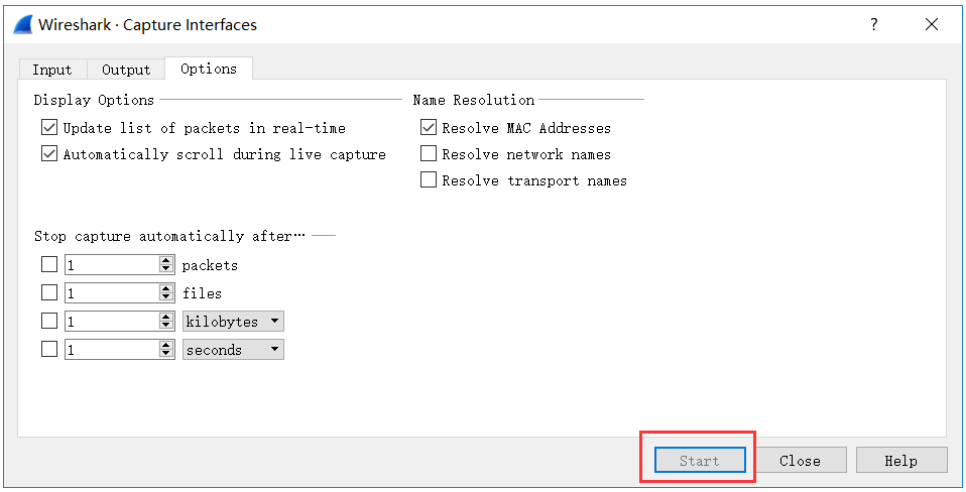
a) 利用Wireshark分析DNS协议
b) 利用Wireshark分析UDP协议
c) 利用Wireshark分析ARP协议

实验过程、结果以及思考问题：

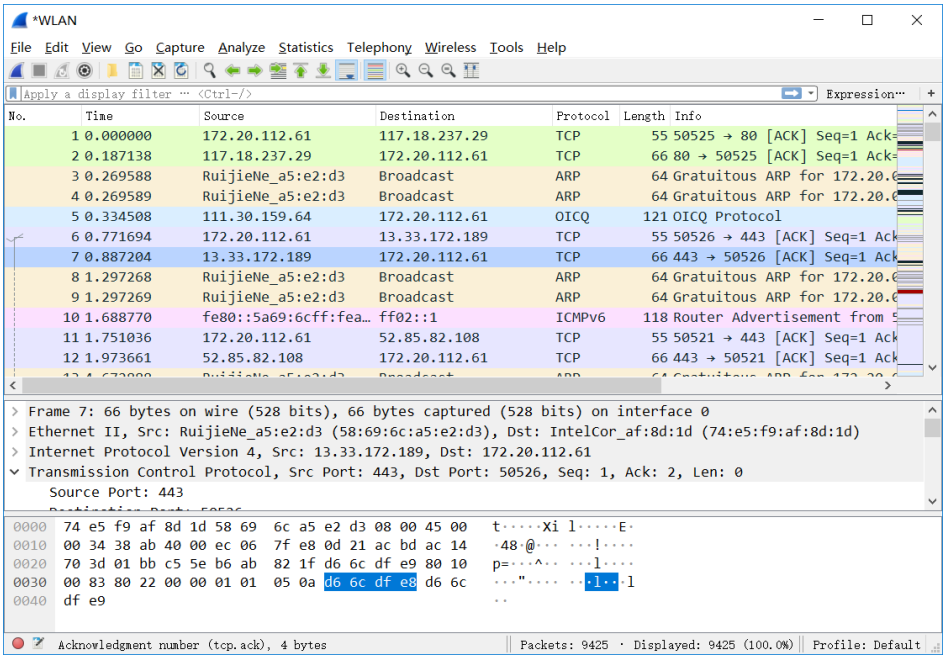
一、 Wireshark的使用

1. 启动web浏览器
2. 启动Wireshark
3. 开始分组俘获：**Capture -> Options**，采用默认设置，点击**start**开始进行分组俘获，所有由默认网卡发送和接收的分组都将被俘获。





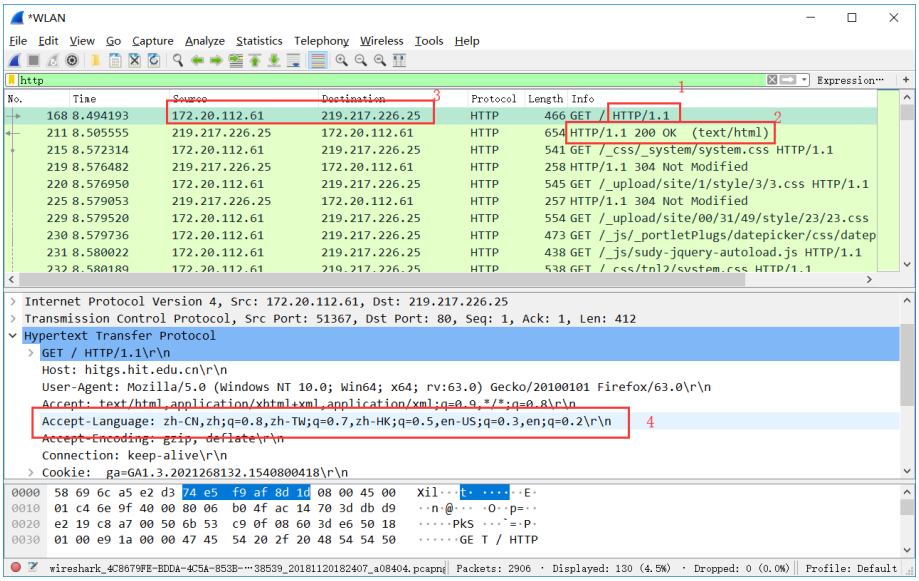
4. 开始俘获后，窗口会显示各类已经俘获的数据包。俘获过程中，可以通过Web浏览器访问网页。通过在显示**筛选**规则中输入不同的协议类型（如”http”）来查看不同类型协议的报文。抓包界面如下所示：



二、 HTTP协议分析

(1) HTTP GET/response 交互

启动web浏览器以及Wireshark，访问http://hitgs.hit.edu.cn，在筛选处输入http



思考问题：

1） HTTP版本

我的浏览器HTTP版本：HTTP 1.1

所访问的服务器HTTP版本：HTTP 1.1

2） 浏览器向服务器指出能接收何种语言版本的对象

Accept-Language: zh-CN，即简体中文

3） 计算机IP地址以及服务器IP地址

本机IP地址：172.20.112.61

所访问服务器IP地址：219.217.226.25

4） 从服务器向浏览器返回的状态代码

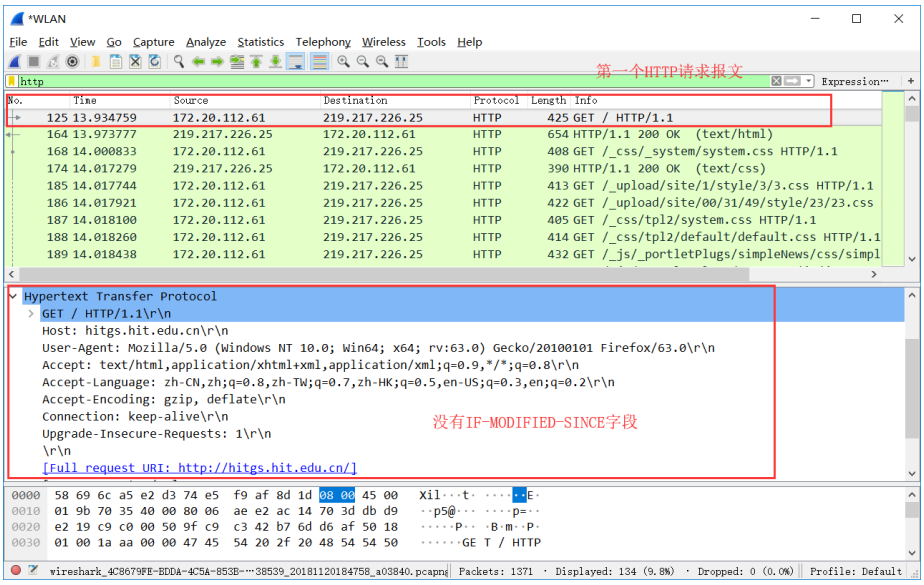
200

(2) HTTP 条件GET/response交互

清空浏览器的缓存，启动Wireshark，访问http://hitgs/hit.edu.cn，再点击刷新**再次访问**该网页，然后在筛选处输入http：

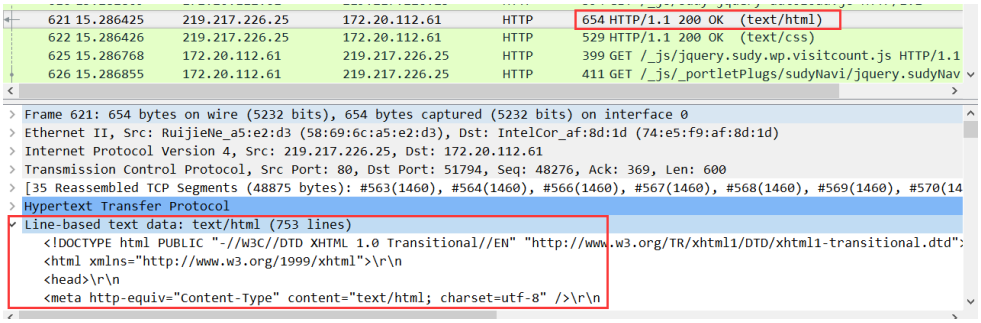
思考问题：

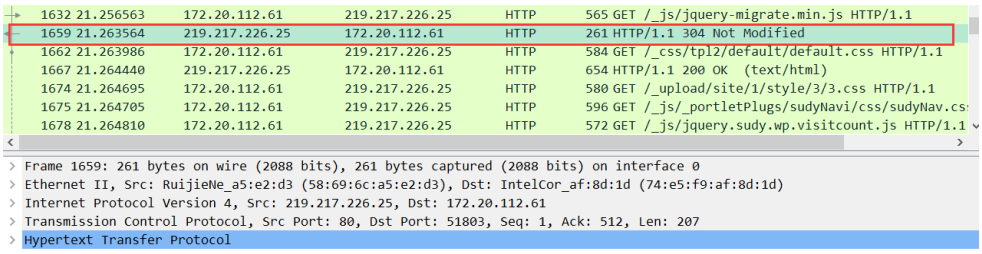
1） 在第一个HTTP GET请求中，请求报文中是否有 IF-MODIFIED-SINCE行？



可以看到在第一个HTTP请求报文中，没有IF_MODIFIED-SINCE字段

2） 服务器是否明确返回了文件的内容？如何获知？



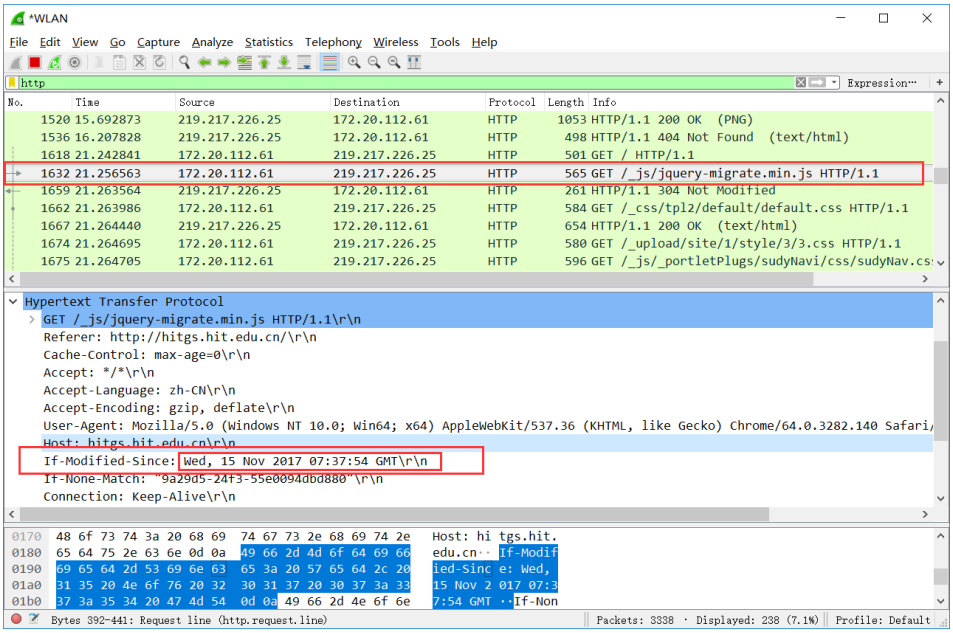


当响应报文状态代码为 **200**时，服务器明确返回了文件的内容

当响应报文状态代码为 **304**时，服务器不明确返回文件的内容

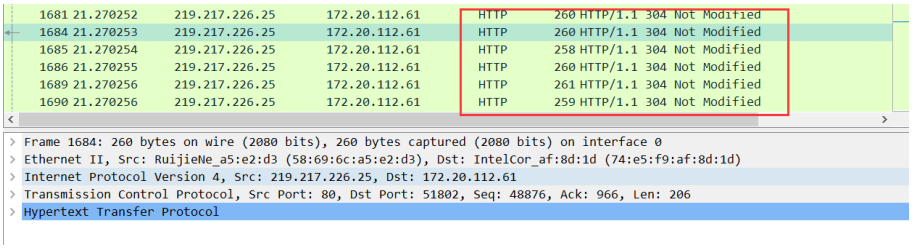
如何获知：可以从响应报文中是否有data字段，来确定有没有明确返回文件内容。

3） 较晚的HTTP GET请求中，有没有IF-MODIFIED-SINCE行？有的话，该行后面的信息是什么？



在较晚的HTTP请求报文中，可以看到**有IF-MODIFIED-SINCE首部行**，后面跟着的信息是**时间**，表示询问服务器在改时间后所请求的内容是否有更新。

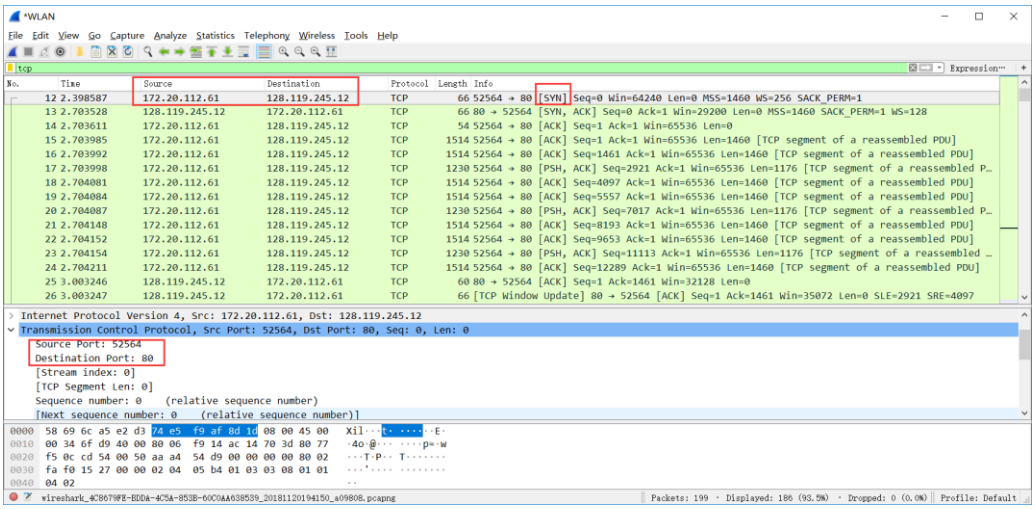
4） 服务器对较晚的HTTP GET请求响应的HTTP状态代码是多少？是否明确返回了文件的内容？



对于较晚的HTTP请求的响应报文，由于请求内容并**没有被修改**，所以响应报文中的**状态代码为304**，从响应报文中没有data字段可以看出，**没有明确返回文件的内容**。

三、 TCP协议分析

访问实验网址，启动Wireshark，上传文件并开始分组俘获，直到上传完毕，停止俘获。



思考问题：

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

172.20.112.61

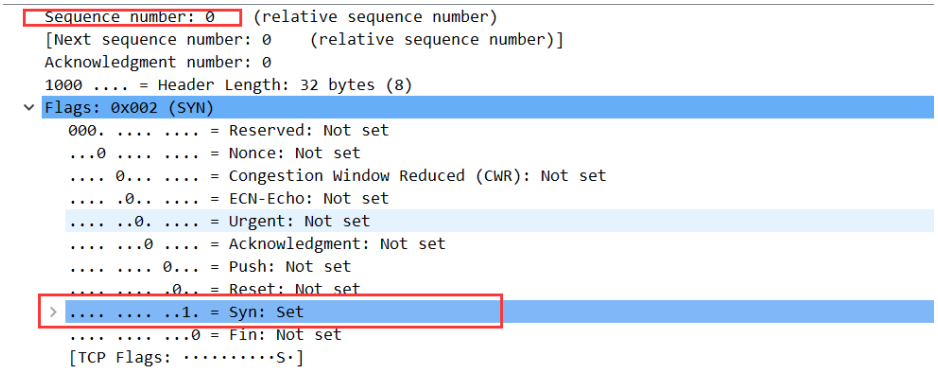
52564

2） Gaia.cs.umass.edu服务器的IP地址是多少？对这一链接，用来发送和接收TCP报文的端口号是多少

128.119.245.12

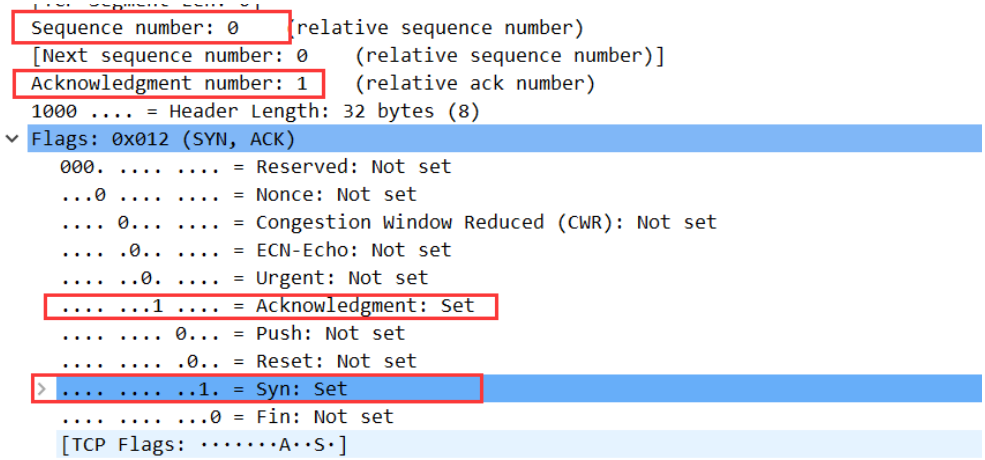
80

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？



上图可以看出，初始化TCP连接的报文段的**seq为0**，通过**将其SYN段置1**来表示这是SYN报文段。

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？

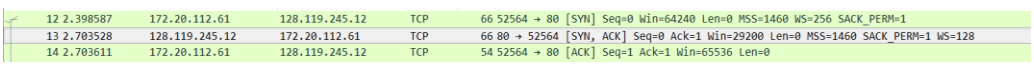


SYNACK报文段序号为0

Acknowledgement字段的值为1，是根据上一次客户端发给服务器的SYN段的**seq + 1**得到的

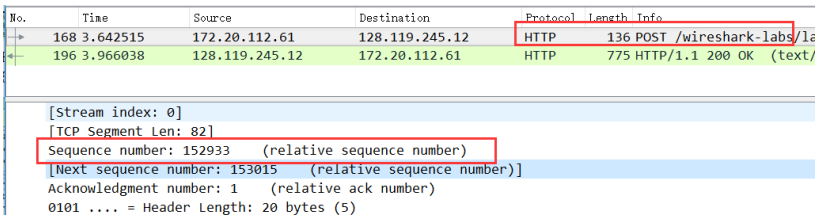
通过**将SYN段置1，将ACK段置1**来标识该报文段是SYNACK报文段。

5） 你能从捕获的数据包中分析出TCP三次握手过程吗？



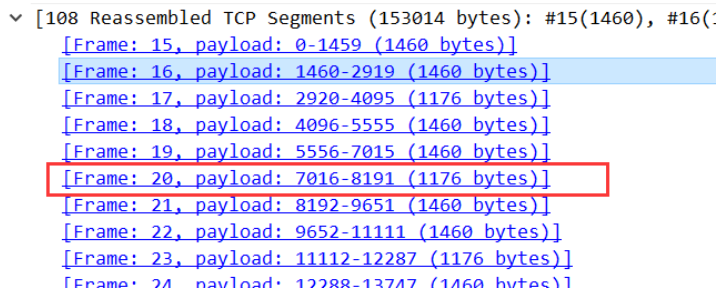
上图所示为TCP连接建立的三次握手过程。

6） 包含HTTP POST 命令的TCP报文段的序号是多少？

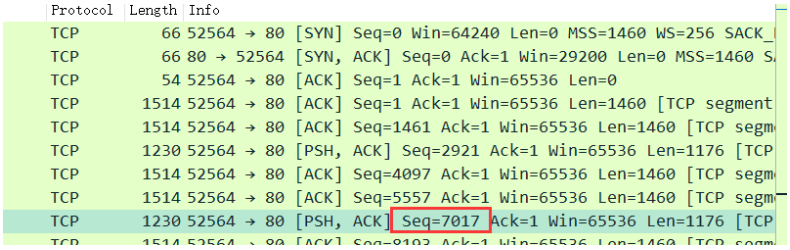


包含HTTP POST 命令的TCP报文段的序号为 152933.

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？

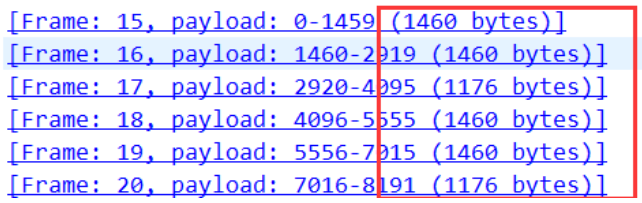


第六个报文段 seq = 7017



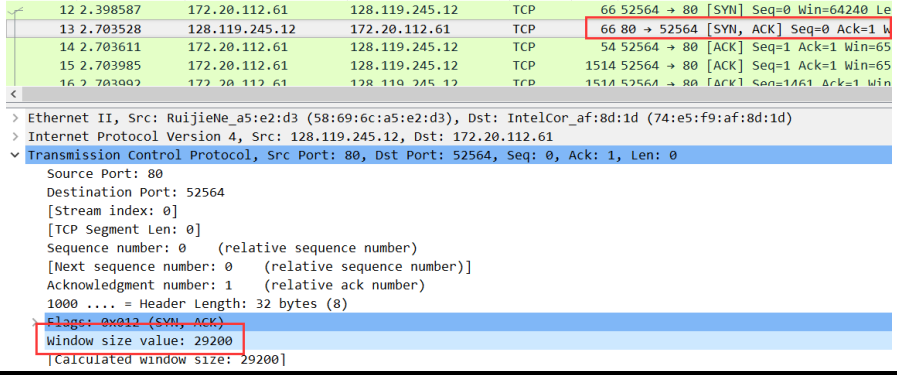
由上图可以看出，是在HTTP POST（seq=152933）发送之前发送的，所对应的ACK为服务器返回的第六个ACK

8） 前六个TCP报文段的长度各是多少？



1460 1460 1176 1460 1460 1176

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？

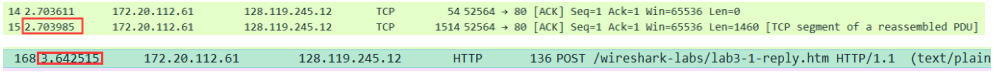


接收端公示的最小的可用缓存空间为29200，不会出现仍然不够用的情况，因为该窗口大小一直增加

10） 在跟踪文件中是否有重传的报文段？判断的依据？

没有，依据：客户端发送的seq没有出现重复

11） TCP连接的throughput是多少？计算过程？



传输的数据总量为 153014 + 108 * 54 = 158846B

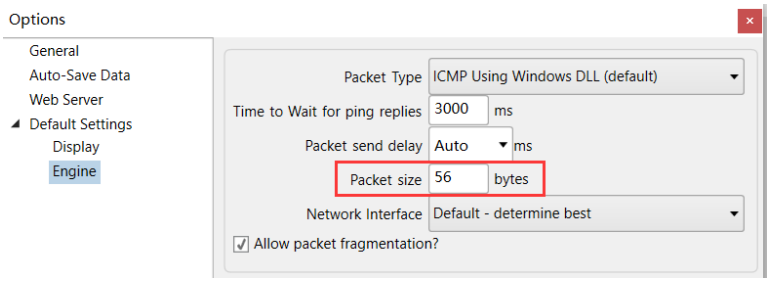
传输时间为 3.642515 – 2.703985 = 0.93853s

Throughput = 158846B / 0.93853s = 169249.784bps

四、 IP协议分析

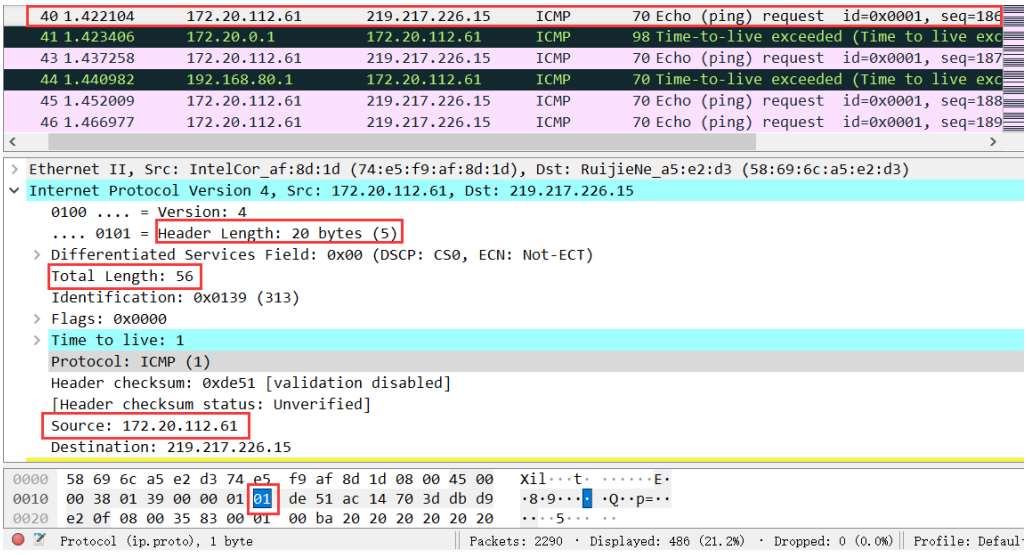
a) 通过pingplotter程序来发送数据包

启动Wireshark，利用pingplotter发送具有**不同大小的数据包**给目的主机，分析程序发送和接收到的IP数据包。



b) 对捕获的数据包进行分析

(1) 在捕获窗口中，选择主机收到的第一个主机发出的ICMP Echo Request消息，在pachet details窗口展开数据包的Internet Protocol部分。



思考问题：

1） 你主机的IP地址是什么？

172.20.112.61

2） 在IP数据包头中，上层协议(upper layer)字段的值是什么？

```
0000  58 69 6c a5 e2 d3 74 e5  f9 af 8d 1d 08 00 45 00   Xil···t· ·····E·
0010  00 38 01 39 00 00 01 01  de 51 ac 14 70 3d db d9   ·8·9···· ·Q··p=··
0020  e2 0f 08 00 35 83 00 01  00 ba 20 20 20 20 20 20   ····5··· ··
```

01

3） IP头有多少字节？该IP数据包的净载为多少字节？并解释你是怎么确定的

IP头为20字节，数据包的总长度为56字节，所以净载为36字节。

4） 该IP数据包的净载大小？

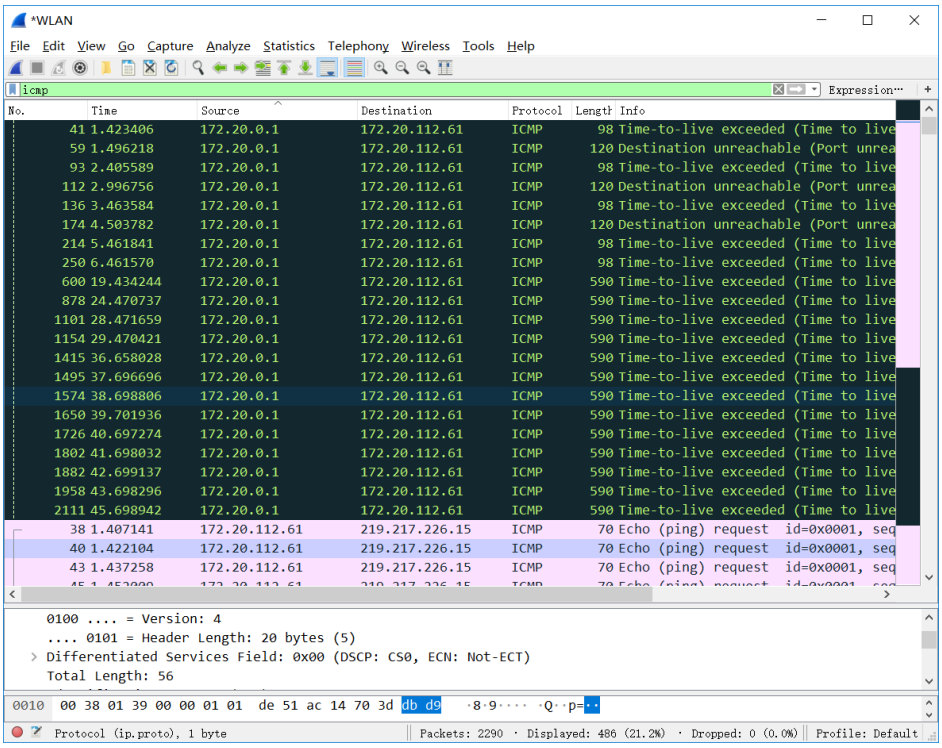
36字节

5） 该IP数据包分片了吗？解释你是如何确定的。

```
Flags: 0x0000
    0... .... .... .... = Reserved bit: Not set
    .0.. .... .... .... = Don't fragment: Not set
    ..0. .... .... .... = More fragments: Not set
    ...0 0000 0000 0000 = Fragment offset: 0
```

没有分片，由于Flag字段全为0，并且片偏移为0

(2) 单机Source列按钮，将捕获的数据包按源IP地址排序。选择第一个主机发出的ICMP Echo Request消息，在pachet details窗口展开数据包的Internet Protocol部分。在“listing of captured packets”窗口，会看到许多后续的ICMP消息。



思考问题：

1） 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

2） 哪些字段必须保持常量？哪些字段必须改变？为什么？

```
> Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
  Total Length: 56
  Identification: 0x0138 (312)
v Flags: 0x0000
    0... .... .... .... = Reserved bit: Not set
    .0.. .... .... .... = Don't fragment: Not set
    ..0. .... .... .... = More fragments: Not set
    ...0 0000 0000 0000 = Fragment offset: 0
  Time to live: 255
  Protocol: ICMP (1)
  Header checksum: 0xe051 [validation disabled]
  [Header checksum status: Unverified]
  Source: 172.20.112.61
  Destination: 219.217.226.15
```

可以看出，除了ID、TTL、Header checksum之外的字段**保持常量**，而ID、TTL、Header checksum必须改变。

3） 描述你看到的IP数据包Identification字段值的形式。

为4位16进制的数，也就是16位。

(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。

思考问题：

No.	Time	Source	Destination	Protocol	Length	Info
44	1.440982	192.168.80.1	172.20.112.61	ICMP	70	Time-to-live exceeded (Time to live
69	1.555488	192.168.80.1	172.20.112.61	ICMP	70	Destination unreachable (Port unrea
95	2.420144	192.168.80.1	172.20.112.61	ICMP	70	Time-to-live exceeded (Time to live
119	3.056768	192.168.80.1	172.20.112.61	ICMP	70	Destination unreachable (Port unrea
138	3.511950	192.168.80.1	172.20.112.61	ICMP	70	Time-to-live exceeded (Time to live
176	4.515875	192.168.80.1	172.20.112.61	ICMP	70	Time-to-live exceeded (Time to live
180	4.561694	192.168.80.1	172.20.112.61	ICMP	70	Destination unreachable (Port unrea
216	5.511844	192.168.80.1	172.20.112.61	ICMP	70	Time-to-live exceeded (Time to live

```
> Frame 44: 70 bytes on wire (560 bits), 70 bytes captured (560 bits) on interface 0
> Ethernet II, Src: RuijieNe_a5:e2:d3 (58:69:6c:a5:e2:d3), Dst: IntelCor_af:8d:1d (74:e5:f9:af:8d:1d)
v Internet Protocol Version 4, Src: 192.168.80.1, Dst: 172.20.112.61
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 56
    Identification: 0x0000 (0)
  > Flags: 0x0000
    Time to live: 254
    Protocol: ICMP (1)
    Header checksum: 0x8fc9 [validation disabled]
    [Header checksum status: Unverified]
    Source: 192.168.80.1
    Destination: 172.20.112.61
```

1） Identification字段和TTL字段的值是什么？

Identification：0x0000

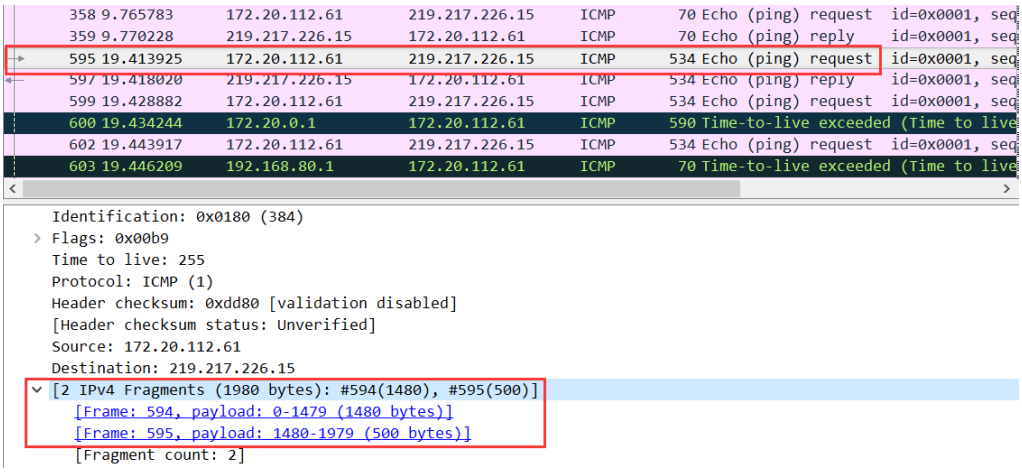
TTL：254

2） 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。因为都是第一跳路由器返回的数据包，所以TTL字段都是可取的最大值；相同的标识是为了分段后进行重组，给同一个主机返回的标识不代表序号，因此Identification字段也不变。

(4) 对捕获的数据包按时间排序。找到在将包大小改为2000字节后主机发送的第一个ICMP Echo Request消息。

下图为改为2000字节后，主机发送的第一个ICMP Echo Request消息：

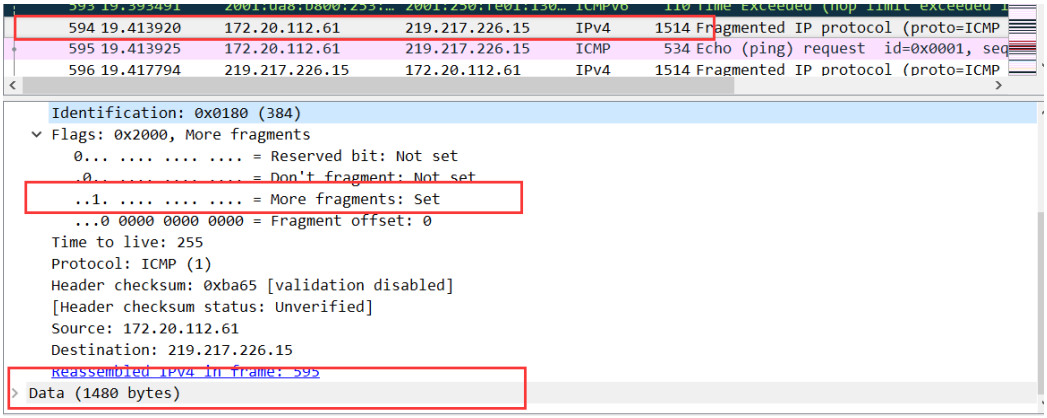


思考问题：

1） 该消息是否被分解成不止一个IP数据报？

可以看到，该消息被分解成不止一个IP数据报。

2） 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头部的那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？

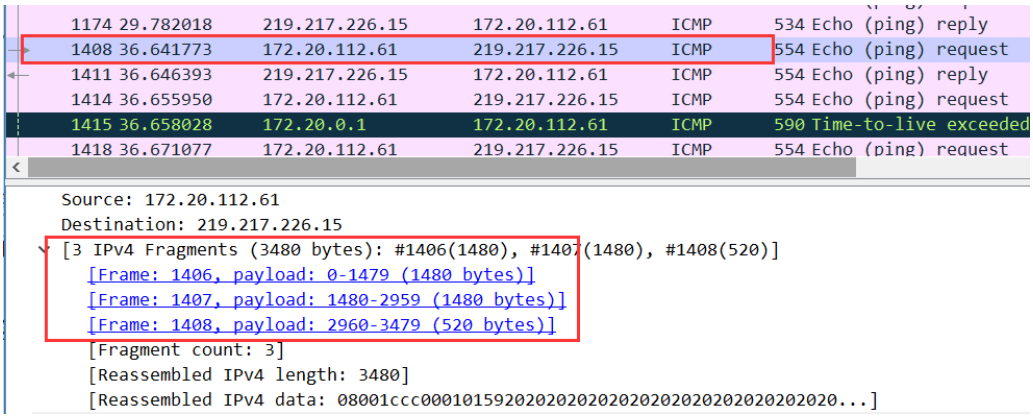


发生了分片，找到No.594的报文，由flag中的 **MF=1** 可知，该分片是第一个而不是最后一个分片。由Data字段为1480字节可知，该分片的长度为**1500字节**。

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。

思考问题：

1） 原始数据包被分成了多少片？



被分成了三片。

2） 这些分片中IP数据报头部哪些字段发生了变化？

```
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
```

1

```
0... .... .... .... = Reserved bit: Not set
.0.. .... .... .... = Don't fragment: Not set
..1. .... .... .... = More fragments: Set
...0 0000 1011 1001 = Fragment offset: 185
```

2

```
Identification: 0x01d8 (472)
Flags: 0x0172
    0... .... .... .... = Reserved bit: Not set
    .0.. .... .... .... = Don't fragment: Not set
    ..0. .... .... .... = More fragments: Not set
    ...0 0001 0111 0010 = Fragment offset: 370
```

3

前两个分片的 **MF段**为1，后两个分片的**片偏移**发生了变化。

五、 Ethernet数据帧分析

```
Ethernet II, Src: RuijieNe_a5:e2:d3 (58:69:6c:a5:e2:d3), Dst: IntelCor_af:8d:1d (74:e5:f9:af:8d:1d)
  Destination: IntelCor_af:8d:1d (74:e5:f9:af:8d:1d)
    Address: IntelCor_af:8d:1d (74:e5:f9:af:8d:1d)
    .... ..0. .... .... .... .... = LG bit: Globally unique address (factory default)
    .... ...0 .... .... .... .... = IG bit: Individual address (unicast)
  Source: RuijieNe_a5:e2:d3 (58:69:6c:a5:e2:d3)
    Address: RuijieNe_a5:e2:d3 (58:69:6c:a5:e2:d3)
    .... ..0. .... .... .... .... = LG bit: Globally unique address (factory default)
    .... ...0 .... .... .... .... = IG bit: Individual address (unicast)
  Type: IPv4 (0x0800)
```

上图为ICMP报文的Ethernet Ⅱ信息。**源MAC地址**：58:59:6c:a5:e2:d3，**目的MAC地址**：74:e5:f9:af:8d:1d，Type : IPv4(0x0800)

对于其他类型的报文Ethernet数据帧分析与上面类似。

六、 DNS协议分析

实验步骤：

1） 打开Wireshark，启动抓包

2） 打开浏览器键入：www.baidu.com

3） 在控制台回车执行完毕后停止抓包。

1） DNS查询消息如下所示：

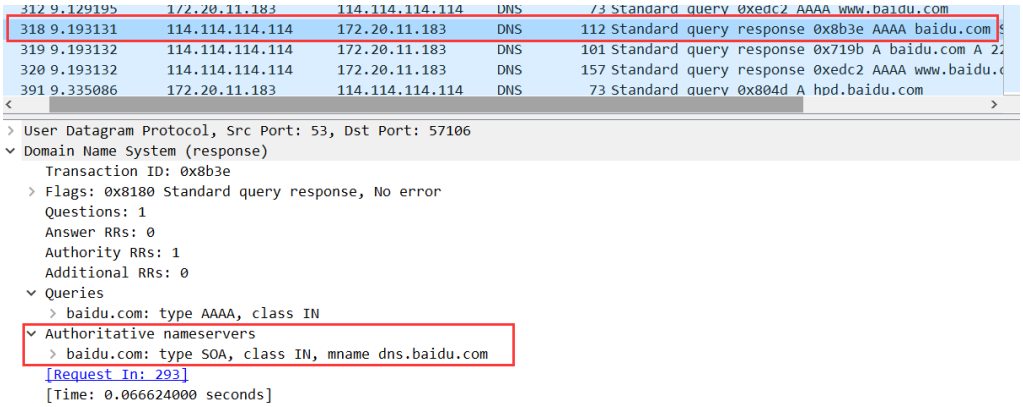
	Time	Source	Destination	Protocol	Length	Info
265	8.814690	172.20.11.183	114.114.114.114	DNS	77	Standard query 0x5138 A dd.browser.360.cn
266	8.815025	172.20.11.183	114.114.114.114	DNS	77	Standard query 0xe062 AAAA dd.browser.360.cn
267	8.844229	114.114.114.114	172.20.11.183	DNS	109	Standard query response 0x5138 A dd.browser.3
268	8.844230	114.114.114.114	172.20.11.183	DNS	141	Standard query response 0xe062 AAAA dd.browse
289	9.126154	172.20.11.183	114.114.114.114	DNS	69	Standard query 0x719b A baidu.com
293	9.126507	172.20.11.183	114.114.114.114	DNS	69	Standard query 0x8b3e AAAA baidu.com
312	9.129195	172.20.11.183	114.114.114.114	DNS	73	Standard query 0xedc2 AAAA www.baidu.com

2） 由上图可以看出，本机的IP地址为172.20.11.183，**本地域名服务器的IP地址为114.114.114.114**

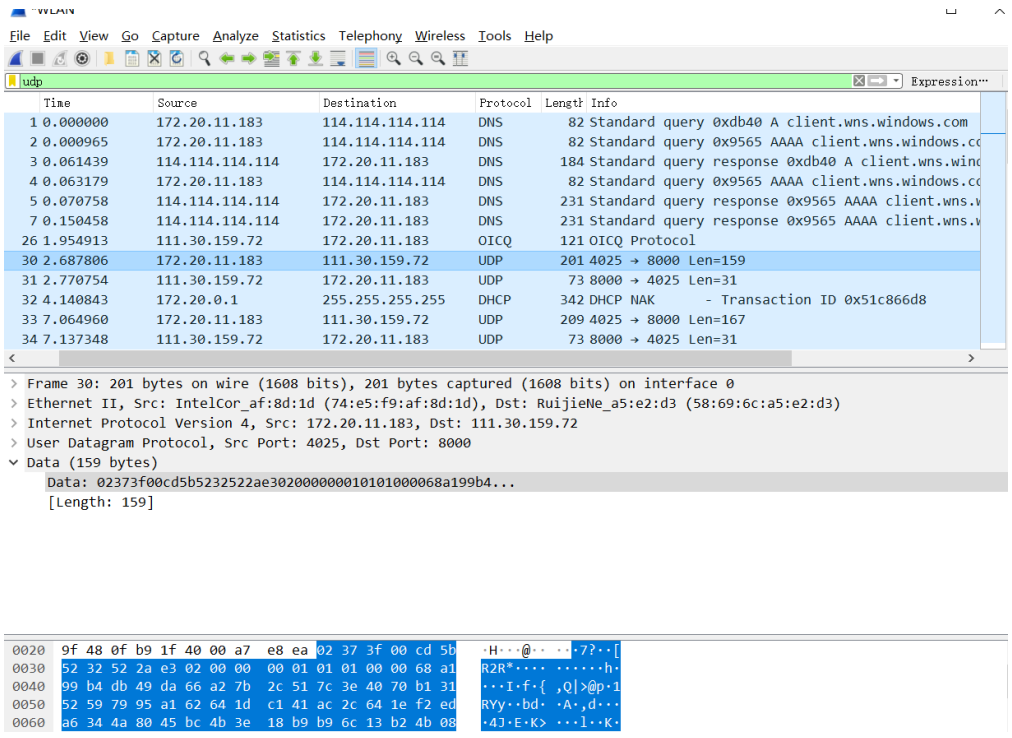
3） DNS查询报文的内容如下所示：

```
Domain Name System (query)
    Transaction ID: 0x719b
  > Flags: 0x0100 Standard query
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 0
  Queries
    > baidu.com: type A, class IN
    [Response In: 319]
```

4） DNS回复消息：



七、 UDP协议分析



思考问题：

1） 消息是基于UDP的还是TCP的？

基于UDP的。

2） 你的主机IP地址是什么？目的主机IP地址是什么？

```
30 2.687806    172.20.11.183    111.30.159.72    UDP    201 4025 → 8000 Len=159
```

主机的IP地址为172.20.11.183，目的主机的IP地址为111.20.159.72

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？

我的端口：4025

服务器端口：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数据报包括**首部字段和数据字段**，其中首部字段包括：

源端口号：16位

目的端口号：16位

长度：16位

校验和字段：16位

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？这与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？

因为服务器应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，并不能保证数据一定送达。

可以看出。可以看出在传输开始时，没有连接建立的过程，也就没有初始序列号的交换，因此发送的数据也是乱序的。

八、 ARP协议分析

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

```
C:\Users\W24>arp -a

接口: 192.168.62.1 --- 0xa
  Internet 地址         物理地址              类型
  192.168.62.255        ff-ff-ff-ff-ff-ff     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
  239.255.255.250       01-00-5e-7f-ff-fa     静态
  255.255.255.255       ff-ff-ff-ff-ff-ff     静态

接口: 172.20.11.142 --- 0xb
  Internet 地址         物理地址              类型
  172.20.0.1            58-69-6c-a5-e2-d3     动态
  172.20.127.255        ff-ff-ff-ff-ff-ff     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
  255.255.255.255       ff-ff-ff-ff-ff-ff     静态

接口: 192.168.163.1 --- 0xf
  Internet 地址         物理地址              类型
  192.168.163.254       00-50-56-ed-38-aa     动态
  192.168.163.255       ff-ff-ff-ff-ff-ff     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
  239.255.255.250       01-00-5e-7f-ff-fa     静态
  255.255.255.255       ff-ff-ff-ff-ff-ff     静态
```

思考问题：

1） 说明ARP缓存中的每一列的含义是什么？

第一列：IP地址

第二列：物理地址（MAC地址）

第三列：类型，如果是动态类型，过一段时间会被删除。

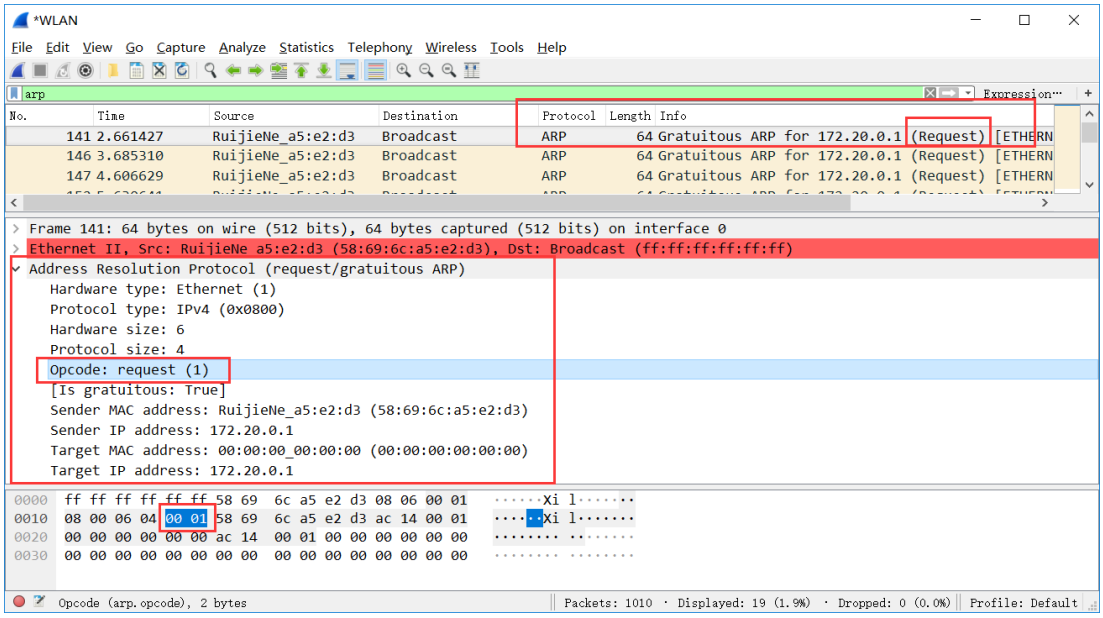
（2） 在命令行模式下输入：ping 192.168.1.82。

```
C:\Users\W24>ping 219.217.226.15

正在 Ping 219.217.226.15 具有 32 字节的数据:
来自 219.217.226.15 的回复: 字节=32 时间=4ms TTL=59
来自 219.217.226.15 的回复: 字节=32 时间=2ms TTL=59
来自 219.217.226.15 的回复: 字节=32 时间=3ms TTL=59
来自 219.217.226.15 的回复: 字节=32 时间=2ms TTL=59

219.217.226.15 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 = 2ms, 最长 = 4ms, 平均 = 2ms
```

思考问题：



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

由上图可以看出**ARP数据包的格式**：

硬件格式：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

目标MAC地址：6字节

目标IP地址：4字节

2） 如何判断一个ARP数据是请求包还是应答包？

如果**OP字段**为1(0x0001)，说明为请求包；如果OP字段为2(0x0002)，说明为应答包。

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，并不知道其所在主机的MAC地址，因此采用广播的方式；应答时，主机可以从查询报文中获得查询主机的MAC地址，并且局域网中的其他主机不需要该响应信息，所以在有明确的局域网地址的帧中传送。

心得体会：

1. 巩固了本学期所学习的知识，并有了进一步的了解；
2. 掌握了Wireshark工具进行抓包并分析的方法；
3. 了解了各网络协议之间进行报文交换的情况。