《计算机网络协议开发》实验报告

第二次实验：Wireshark 数据包嗅探实验

姓名：吕玉龙

学号：191220076

2019 级计算机科学与技术院/系

邮箱：1931015836@qq.com

时间：2022.3.6

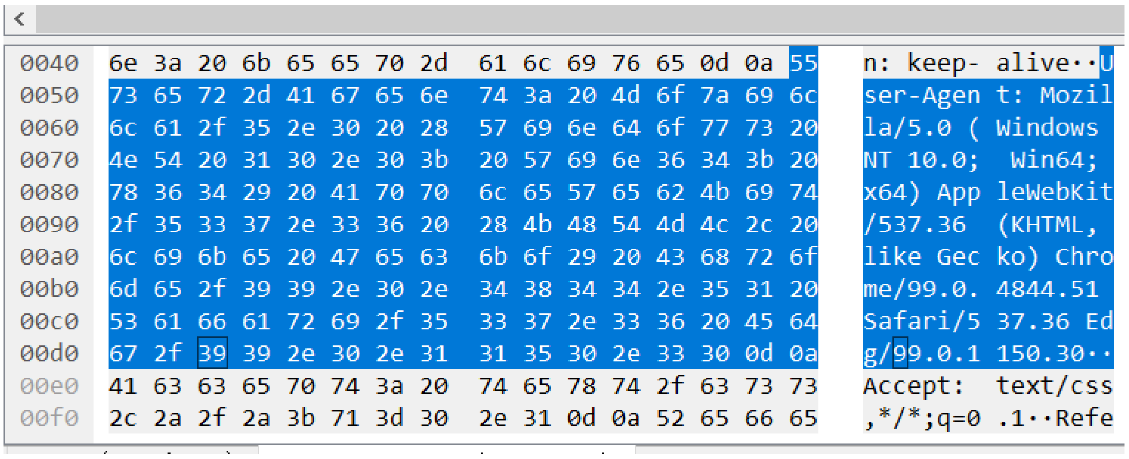
一、 实验目的

熟悉 Wireshark 和数据包嗅探。

二、 实验内容

问题 1：访问 www.4399.com 网站，保留 trace 文件

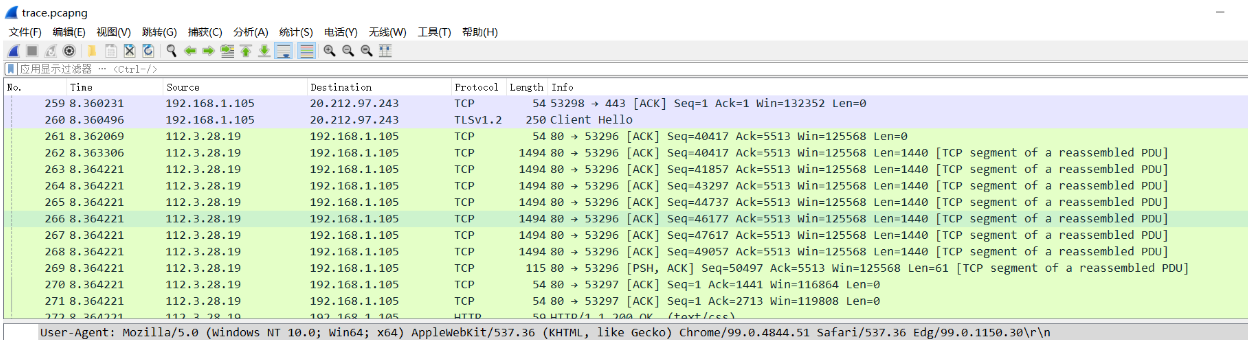
问题 2：包含我所使用的浏览器的详细信息，为 Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/99.0.4844.51 Safari/537.36 Edg/99.0.1150.30（即浏览器为 EDGE）对应的值为



```
0040  6e 3a 20 6b 65 65 70 2d  61 6c 69 76 65 0d 0a 55   n: keep- alive··U
0050  73 65 72 2d 41 67 65 6e  74 3a 20 4d 6f 7a 69 6c   ser-Agen t: Mozil
0060  6c 61 2f 35 2e 30 20 28  57 69 6e 64 6f 77 73 20   la/5.0 ( Windows
0070  4e 54 20 31 30 2e 30 3b  20 57 69 6e 36 34 3b 20   NT 10.0;  Win64;
0080  78 36 34 29 20 41 70 70  6c 65 57 65 62 4b 69 74   x64) App leWebKit
0090  2f 35 33 37 2e 33 36 20  28 4b 48 54 4d 4c 2c 20   /537.36  (KHTML,
00a0  6c 69 6b 65 20 47 65 63  6b 6f 29 20 43 68 72 6f   like Gec ko) Chro
00b0  6d 65 2f 39 39 2e 30 2e  34 38 34 34 2e 35 31 20   me/99.0. 4844.51
00c0  53 61 66 61 72 69 2f 35  33 37 2e 33 36 20 45 64   Safari/5 37.36 Ed
00d0  67 2f 39 39 2e 30 2e 31  31 35 30 2e 33 30 0d 0a   g/99.0.1 150.30··
00e0  41 63 63 65 70 74 3a 20  74 65 78 74 2f 63 73 73   Accept:  text/css
00f0  2c 2a 2f 2a 3b 71 3d 30  2e 31 0d 0a 52 65 66 65   ,*/*;q=0 .1··Refe
```

从方框中的 00d0 那行的 39 开始到这一行的 30 结束
其它属性的定义例如 accept-encoding，accept-language 等等
在页面源代码中并没有这些属性

问题 3：
使用的是 tcp 协议，图如下：



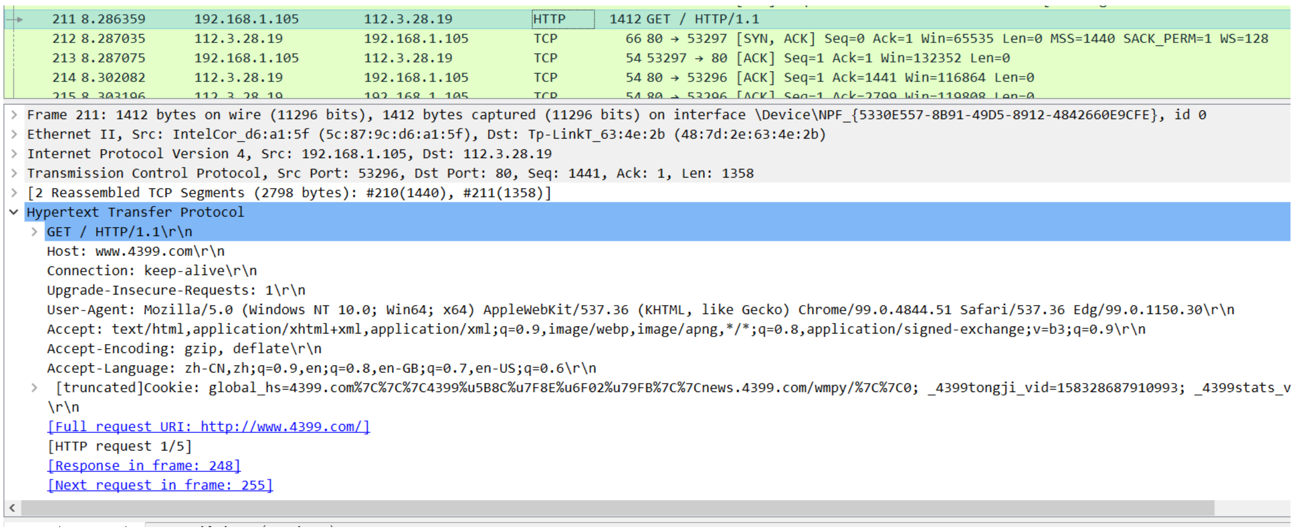
trace.pcapng
文件(F) 编辑(E) 视图(V) 跳转(G) 捕获(C) 分析(A) 统计(S) 电话(Y) 无线(W) 工具(T) 帮助(H)
应用显示过滤器 … <Ctrl-/>

```
No.     Time      Source          Destination     Protocol Length Info
    259 8.360231  192.168.1.105   20.212.97.243   TCP        54 53298 → 443 [ACK] Seq=1 Ack=1 Win=132352 Len=0
    260 8.360496  192.168.1.105   20.212.97.243   TLSv1.2   250 Client Hello
    261 8.362069  112.3.28.19     192.168.1.105   TCP        54 80 → 53296 [ACK] Seq=40417 Ack=5513 Win=125568 Len=0
    262 8.363306  112.3.28.19     192.168.1.105   TCP      1494 80 → 53296 [ACK] Seq=40417 Ack=5513 Win=125568 Len=1440 [TCP segment of a reassembled PDU]
    263 8.364221  112.3.28.19     192.168.1.105   TCP      1494 80 → 53296 [ACK] Seq=41857 Ack=5513 Win=125568 Len=1440 [TCP segment of a reassembled PDU]
    264 8.364221  112.3.28.19     192.168.1.105   TCP      1494 80 → 53296 [ACK] Seq=43297 Ack=5513 Win=125568 Len=1440 [TCP segment of a reassembled PDU]
    265 8.364221  112.3.28.19     192.168.1.105   TCP      1494 80 → 53296 [ACK] Seq=44737 Ack=5513 Win=125568 Len=1440 [TCP segment of a reassembled PDU]
    266 8.364221  112.3.28.19     192.168.1.105   TCP      1494 80 → 53296 [ACK] Seq=46177 Ack=5513 Win=125568 Len=1440 [TCP segment of a reassembled PDU]
    267 8.364221  112.3.28.19     192.168.1.105   TCP      1494 80 → 53296 [ACK] Seq=47617 Ack=5513 Win=125568 Len=1440 [TCP segment of a reassembled PDU]
    268 8.364221  112.3.28.19     192.168.1.105   TCP      1494 80 → 53296 [ACK] Seq=49057 Ack=5513 Win=125568 Len=1440 [TCP segment of a reassembled PDU]
    269 8.364221  112.3.28.19     192.168.1.105   TCP       115 80 → 53296 [PSH, ACK] Seq=50497 Ack=5513 Win=125568 Len=61 [TCP segment of a reassembled PDU]
    270 8.364221  112.3.28.19     192.168.1.105   TCP        54 80 → 53297 [ACK] Seq=1 Ack=1441 Win=116864 Len=0
    271 8.364221  112.3.28.19     192.168.1.105   TCP        54 80 → 53297 [ACK] Seq=1 Ack=2713 Win=119808 Len=0
    272 8.364221  112.3.28.19     192.168.1.105   HTTP       59 HTTP/1.1 200 OK  (text/css)

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/99.0.4844.51 Safari/537.36 Edg/99.0.1150.30\r\n
```

tcp 和 http 的关系：TCP 对应于传输层，定义的是数据传输和连接方式的规范。HTTP 对应于应用层，定义的是传输数据的内容的规范。。Http 协议是建立在 TCP 协议基础之上的，当浏览器需要从服务器获取网页数据的时候，会发出一次 Http 请求。Http 会通过 TCP 建立起一个到服务器的连接通道，当本次请求需要的数据完毕后，Http 会立即将 TCP 连接断开，这个过程是很短的。所以 Http 连接是一种短连接，是一种无状态的连接。
请求 web 页面需要使用 http 协议，而 http 协议需要通过底层的 tcp 协议来建立。

问题 4：



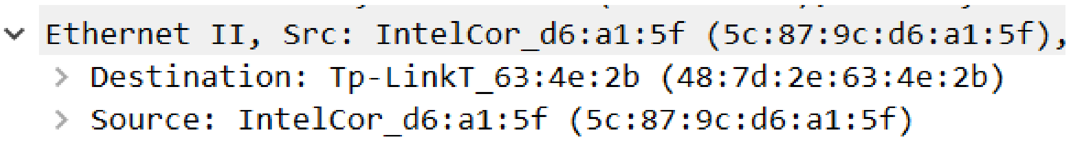
```
  211 8.286359    192.168.1.105     112.3.28.19       HTTP     1412 GET / HTTP/1.1
  212 8.287035    112.3.28.19       192.168.1.105     TCP        66 80 → 53297 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1440 SACK_PERM=1 WS=128
  213 8.287075    192.168.1.105     112.3.28.19       TCP        54 53297 → 80 [ACK] Seq=1 Ack=1 Win=132352 Len=0
  214 8.302082    112.3.28.19       192.168.1.105     TCP        54 80 → 53296 [ACK] Seq=1 Ack=1441 Win=116864 Len=0

> Frame 211: 1412 bytes on wire (11296 bits), 1412 bytes captured (11296 bits) on interface \Device\NPF_{5330E557-8B91-49D5-8912-4842660E9CFE}, id 0
> Ethernet II, Src: IntelCor_d6:a1:5f (5c:87:9c:d6:a1:5f), Dst: Tp-LinkT_63:4e:2b (48:7d:2e:63:4e:2b)
> Internet Protocol Version 4, Src: 192.168.1.105, Dst: 112.3.28.19
> Transmission Control Protocol, Src Port: 53296, Dst Port: 80, Seq: 1441, Ack: 1, Len: 1358
> [2 Reassembled TCP Segments (2798 bytes): #210(1440), #211(1358)]
v Hypertext Transfer Protocol
  > GET / HTTP/1.1\r\n
    Host: www.4399.com\r\n
    Connection: keep-alive\r\n
    Upgrade-Insecure-Requests: 1\r\n
    User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/99.0.4844.51 Safari/537.36 Edg/99.0.1150.30\r\n
    Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9\r\n
    Accept-Encoding: gzip, deflate\r\n
    Accept-Language: zh-CN,zh;q=0.9,en;q=0.8,en-GB;q=0.7,en-US;q=0.6\r\n
  >  [truncated]Cookie: global_hs=4399.com%7C%7C%7C4399%u5B8C%u7F8E%u6F02%u79FB%7C%7Cnews.4399.com/wmpy/%7C%7C0; _4399tongji_vid=158328687910993; _4399stats_v
    \r\n
    [Full request URI: http://www.4399.com/]
    [HTTP request 1/5]
    [Response in frame: 248]
    [Next request in frame: 255]
```

我的机器的 ip 地址为 192.168.1.105，这个报文是向 4399 的网页的服务器发送申请，请求获取主页信息

问题 5：
网络层
网络层主要复制主机之间 packet 的传输，而应用层则是希望能够达到一个进程之间通信的效果，显然中间还差一层端到端的传输，所以网络层无法跟应用层相关联。

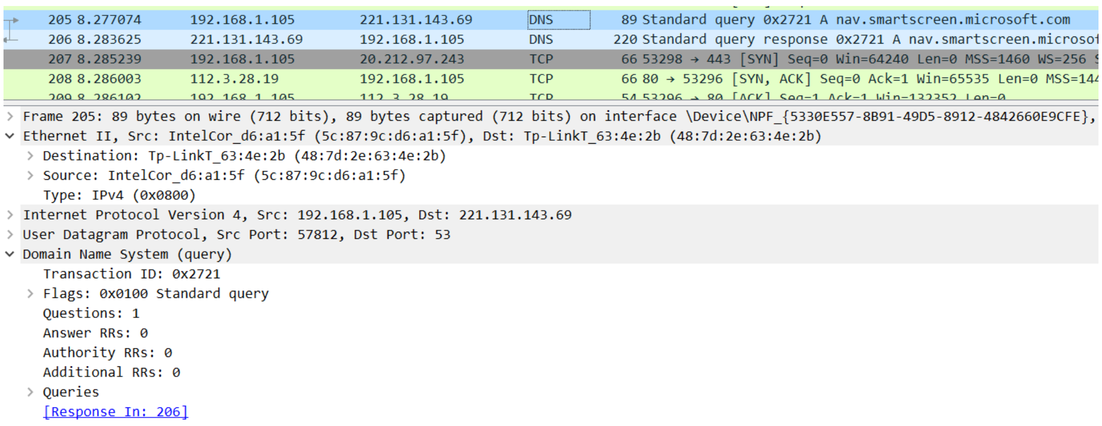
问题 6：
MAC 地址如下：5c879cd6a15f



```
v Ethernet II, Src: IntelCor_d6:a1:5f (5c:87:9c:d6:a1:5f),
   > Destination: Tp-LinkT_63:4e:2b (48:7d:2e:63:4e:2b)
   > Source: IntelCor_d6:a1:5f (5c:87:9c:d6:a1:5f)
```

MAC 地址的作用：现在 Internet 的方式是把主机通过局域网组织在一起，然后再通过交换机和 Internet 相连接，所以需要区分具体用户。故 MAC 地址是用来区分局域网内（交换机内部）具体用户的地址，具有唯一性。

问题 7：
DNS 协议。
DNS 协议可以将域名转换为 IP 地址。
RFC974，讲述了 mail routing and domain system。该篇 RFC 文档主要内容是解释邮件发送者如何决定如何将邮件发送到给定的 Internet 域名。 这涉及到邮件程序如何解释用于消息路由的 MX RR 的讨论。但它没有说明邮件程序如何处理 MB 和 MG RR（用于解释邮箱名称）



```
205 8.277074   192.168.1.105    221.131.143.69   DNS    89 Standard query 0x2721 A nav.smartscreen.microsoft.com
206 8.283625   221.131.143.69   192.168.1.105    DNS   220 Standard query response 0x2721 A nav.smartscreen.microsof
207 8.285239   192.168.1.105    20.212.97.243    TCP    66 53298 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 S
208 8.286003   112.3.28.19      192.168.1.105    TCP    66 80 → 53296 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=144
209 8.286102   192.168.1.105    112.3.28.19      TCP    54 53296 → 80 [ACK] Seq=1 Ack=1 Win=132352 Len=0
> Frame 205: 89 bytes on wire (712 bits), 89 bytes captured (712 bits) on interface \Device\NPF_{5330E557-8B91-49D5-8912-4842660E9CFE},
v Ethernet II, Src: IntelCor_d6:a1:5f (5c:87:9c:d6:a1:5f), Dst: Tp-LinkT_63:4e:2b (48:7d:2e:63:4e:2b)
  > Destination: Tp-LinkT_63:4e:2b (48:7d:2e:63:4e:2b)
  > Source: IntelCor_d6:a1:5f (5c:87:9c:d6:a1:5f)
    Type: IPv4 (0x0800)
> Internet Protocol Version 4, Src: 192.168.1.105, Dst: 221.131.143.69
> User Datagram Protocol, Src Port: 57812, Dst Port: 53
v Domain Name System (query)
    Transaction ID: 0x2721
  > Flags: 0x0100 Standard query
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 0
  > Queries
    [Response In: 206]
```

问题 8：
什么是协议：在计算机网络中，把用于规定数据的交换规则、报文格式，以及如何发送和接收数据的一套规则称为网络协议
为什么要设计协议：因为需要协议使得发送方与接收方能够正确地通信，能够知晓发送的是何种信息
协议是否需要标准化：显然需要，因为不标准化的话，仍然会出现无法解包的情况
协议的实现是否需要标准化：需要，有一个统一的协议栈利于主机通信，同时标准的协议实现更有利于各层之间独立而协调地工作。
TCP/IP 协议栈为什么要分层：首先可以简化网络协议的复杂性，同时各层之间能够相互独立又高效地协调工作，灵活性更好。

问题 9：经历过一系列的演变之后，因特网架构已经被证明非常的成功。设计师不仅需要考虑协议栈设计的优先级，用户的体验也要被充分考虑。如果考虑继续演变的话，新的构建块的需要它可以识别从源到目的地的一系列数据包，而不假设该服务具有任何特定类型的服务，才能够拥有更好的生存性以及灵活性。这个演变的过程是复杂而长期的，无数的技术人员为此奉献了他们的精力，我们却不能一一记住他们的名字，但是这不能否认他们的功绩，使用现在因特网架构的我们仍要感谢他们。

三、实验中遇到的问题及解决方案

第二题中，我对于浏览器属性是什么仍然不是很清楚，以及网页的 html 代码也不是很了解。对于 wireshark 的应用仍然不是非常熟练，其中的某些数据包用的协议不了解。

四、实验的启示/意见和建议

附：总用时：大概 4.5h，包括看各种文档以及撰写报告