**电子科技大学**

**实**

**验**

**报**

**告**

学生姓名：李昊霖

学号：2019270102007

课程名称：计算机网络基础

指导教师：杨宁

日期： 2021 年 3 月 12 日

实验项目名称： 2-1wireshark-http实验

报告评分： 教师签字：

**一、实验概要**

**实验目的:**

通过实验2-1学习wireshark抓包软件的基本使用和注意事项以及HTTP报文的相关知识。

**实验内容**：

依次进行五个实验，学习基本的GET /响应交互，HTTP消息格式，检索大型HTML文件，检索具有嵌入式对象的HTML文件以及HTTP身份验证和安全性的内容，并分析相应的HTTP报文，回答指导书中的相关问题。

**二、实验步骤、数据及分析结果**

**实验步骤:**

**1. The Basic HTTP GET/response interaction**

1. Start up your web browser.

2. Start up the Wireshark packet sniffer, as described in the Introductory lab (but

don’t yet begin packet capture). Enter “http” (just the letters, not the quotation

marks) in the display-filter-specification window, so that only captured HTTP

messages will be displayed later in the packet-listing window. (We’re only

interested in the HTTP protocol here, and don’t want to see the clutter of all

captured packets).

3. Wait a bit more than one minute (we’ll see why shortly), and then begin

Wireshark packet capture.

4. Enter the following to your browser

http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file1.html

Your browser should display the very simple, one-line HTML file.

5. Stop Wireshark packet capture.

**2. The HTTP CONDITIONAL GET/response interaction**

1.Start up the Wireshark packet sniffer

Enter the following URL into your browser

http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file2.html

Your browser should display a very simple five-line HTML file.

3.Quickly enter the same URL into your browser again (or simply select the refresh button on your browser)

Stop Wireshark packet capture, and enter “http” in the display-filter-specification

window, so that only captured HTTP messages will be displayed later in the packet-listing window.

(*Note:* If you are unable to run Wireshark on a live network connection, you can use the http-ethereal-trace-2 packet trace to answer the questions below; see footnote 1. This trace file was gathered while performing the steps above on one )

**3. Retrieving Long Documents**

Start up your web browser, and make sure your browser’s cache is cleared, as

discussed above.

Start up the Wireshark packet sniffer

Enter the following URL into your browser

http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file3.html

Your browser should display the rather lengthy US Bill of Rights.

Stop Wireshark packet capture, and enter “http” in the display-filter-specification

window, so that only captured HTTP messages will be displayed.

(*Note:* If you are unable to run Wireshark on a live network connection, you can use the http-ethereal-trace-3 packet trace to answer the questions below; see footnote 1. This trace file was gathered while performing the steps above on one of the author’s computers.)

**4. HTML Documents with Embedded Objects**

Start up your web browser, and make sure your browser’s cache is cleared, as

discussed above.

Start up the Wireshark packet sniffer

Enter the following URL into your browser

http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file4.html

Your browser should display a short HTML file with two images. These two

images are referenced in the base HTML file. That is, the images themselves are

not contained in the HTML; instead the URLs for the images are contained in the

downloaded HTML file. As discussed in the textbook, your browser will have to

retrieve these logos from the indicated web sites. Our publisher’s logo is retrieved from the gaia.cs.umass.edu web site. The image of the cover for our 5th edition (one of our favorite covers) is stored at the caite.cs.umass.edu server. (These are two different web servers inside cs.umass.edu).

Stop Wireshark packet capture, and enter “http” in the display-filter-specification window, so that only captured HTTP messages will be displayed.

(*Note:* If you are unable to run Wireshark on a live network connection, you can use the http-ethereal-trace-4 packet trace to answer the questions below; see footnote 1. This trace file was gathered while performing the steps above on one of the author’s computers.)

5 HTTP Authentication

Make sure your browser’s cache is cleared, as discussed above, and close down

your browser. Then, start up your browser

Start up the Wireshark packet sniffer

Enter the following URL into your browser

http://gaia.cs.umass.edu/wireshark-labs/protected\_pages/HTTP-wireshark

file5.html Type the requested user name and password into the pop up box.

Stop Wireshark packet capture, and enter “http” in the display-filter-specification window, so that only captured HTTP messages will be displayed later in the packet-listing window.

(*Note:* If you are unable to run Wireshark on a live network connection, you can use the http-ethereal-trace-5 packet trace to answer the questions below; see footnote 2. This trace file was gathered while performing the steps above on one of the author’s computers.)

**实验数据及分析:**

1. **回答指导书问题**

**问题1-7：**

1. Is your browser running HTTP version 1.0 or 1.1? What version of HTTP is the

server running?

2. What languages (if any) does your browser indicate that it can accept to the

server?

3. What is the IP address of your computer? Of the gaia.cs.umass.edu server?

4. What is the status code returned from the server to your browser?

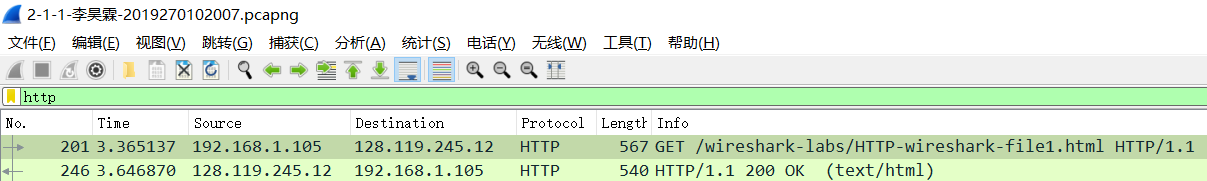
5. When was the HTML file that you are retrieving last modified at the server?

6. How many bytes of content are being returned to your browser?

7. By inspecting the raw data in the packet content window, do you see any headers

within the data that are not displayed in the packet-listing window? If so, name one.

1. 均为HTTP1.1，如下图



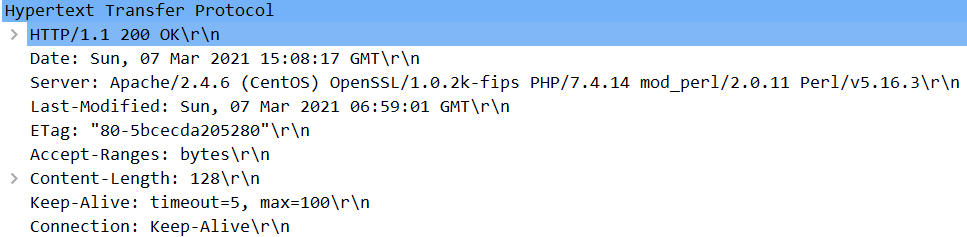
2.中文，英文



分析报文浏览器可知支持简体中文和中文（简体优先），其权值为0.9，优先级最高；并支持英语，权值0.8；支持英式英语，权值0.7；支持美式英语，权值0.6。

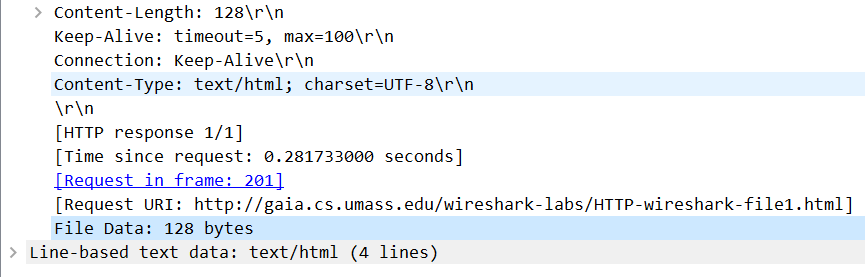
3.PC的IP地址：192.168.1.105；服务区IP地址：128.119.245.12

4.status code:200，如下图



5.2021年3月7号06:59:01

6.128bytes，如下图



7. 请求报文中还有Host字段、connection字段、Accept字段、User-agent字段 ；响应报文中还有server字段、connection字段等。

**问题8-11**：

8. Inspect the contents of the first HTTP GET request from your browser to the

server. Do you see an “IF-MODIFIED-SINCE” line in the HTTP GET?

9. Inspect the contents of the server response. Did the server explicitly return the

contents of the file? How can you tell?

10. Now inspect the contents of the second HTTP GET request from your browser to

the server. Do you see an “IF-MODIFIED-SINCE:” line in the HTTP GET? If

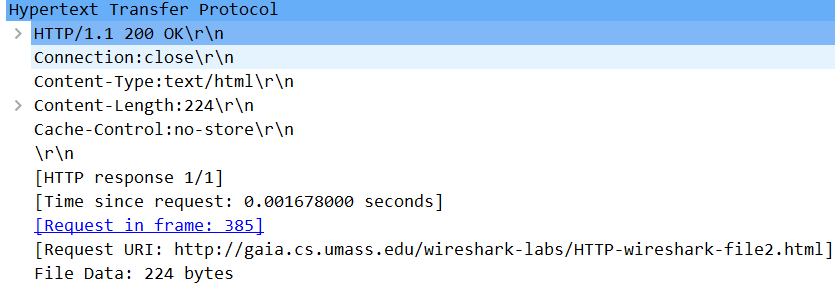
so, what information follows the “IF-MODIFIED-SINCE:” header?

11. What is the HTTP status code and phrase returned from the server in response to this second HTTP GET? Did the server explicitly return the contents of the file? Explain.

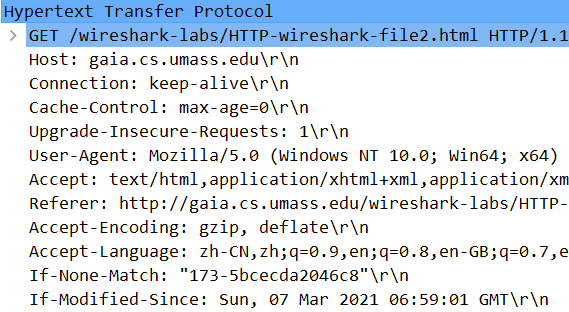
8.第一份报文没有“IF-MODIFIED-SINCE”，如下图



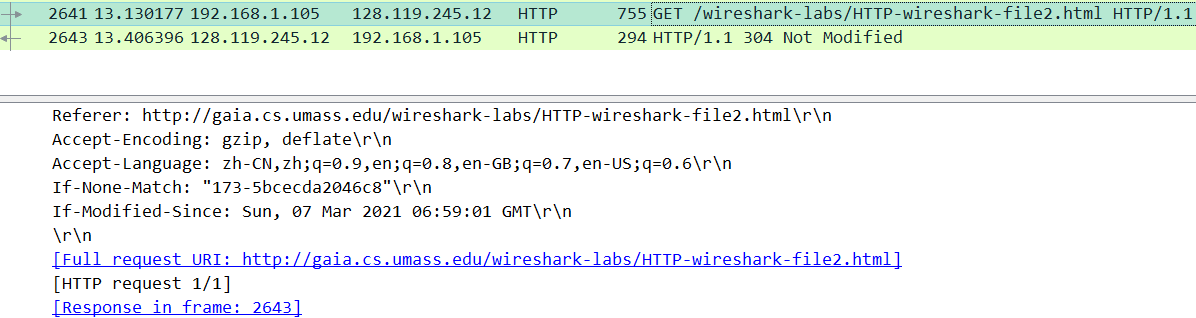
9.已明确返回了文件内容，状态码与短语：200 OK表示请求成功，信息在返回的响应报文中。



10. 第二份报文有“IF-MODIFIED-SINCE”,时间为2021.3.7 06:59:01



11.304 not modified。服务器没有返回请求对象。



第二次发送的请求报文是条件GET报文，包含if-modified-since，而服务器返回的响应报文表明在指定日期后不曾修改过，可以使用缓存器中的副本。

**问题12-15**：

12. How many HTTP GET request messages did your browser send? Which packet

number in the trace contains the GET message for the Bill or Rights?

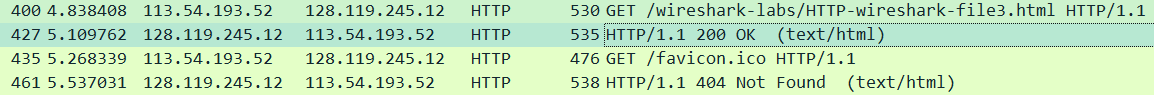
13. Which packet number in the trace contains the status code and phrase associated

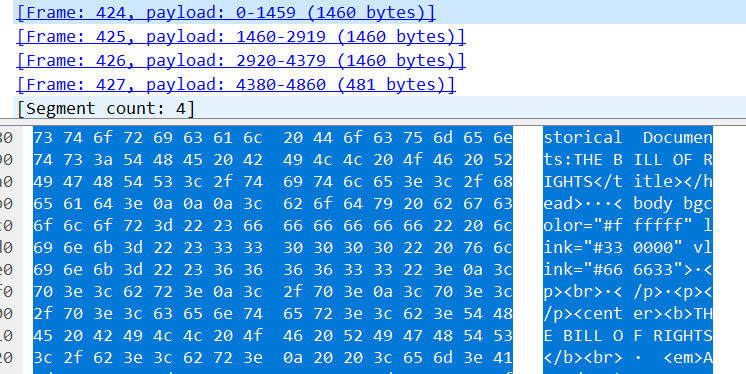
with the response to the HTTP GET request?

14. What is the status code and phrase in the response?

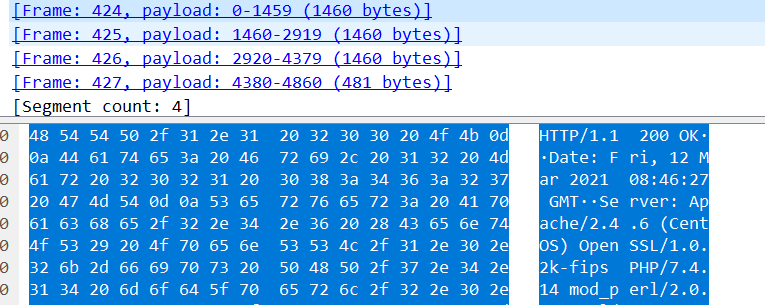
15. How many data-containing TCP segments were needed to carry the single HTTP response and the text of the Bill of Rights?

12.发送了两条请求报文。第一个数据包括“Bill or Rights”，如图第一至三行“documents：BILL OF RIGHTS”





13.第一个数据包包括状态码和短语，如图第一个数据包的第一行。



14.200 OK

15.4个TCP段

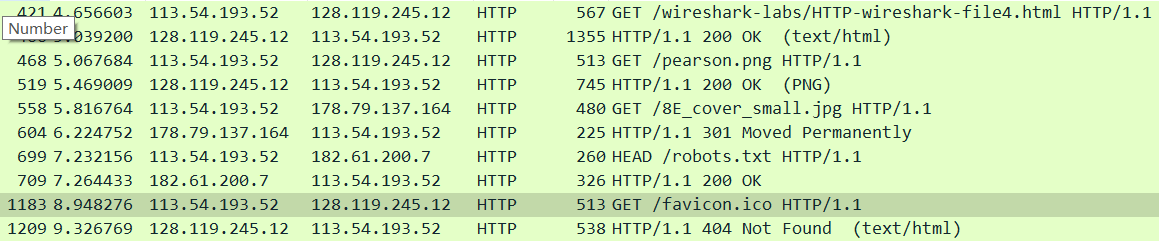
**问题16,17**：

16. How many HTTP GET request messages did your browser send? To which

Internet addresses were these GET requests sent?

17. Can you tell whether your browser downloaded the two images serially, or whether they were downloaded from the two web sites in parallel? Explain.

16.发送了4条GET请求报文，目的地址为128.119.245.12和178.79.137.164



17.是逐个下载图片的。因为第一个png图片已经下载完成后才请求下载第二个jpg图片。

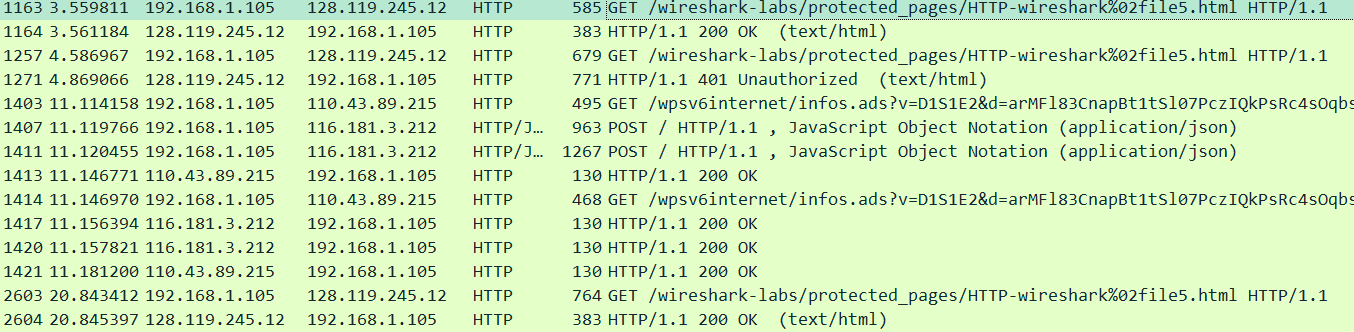
问题18,19：

18. What is the server’s response (status code and phrase) in response to the initial

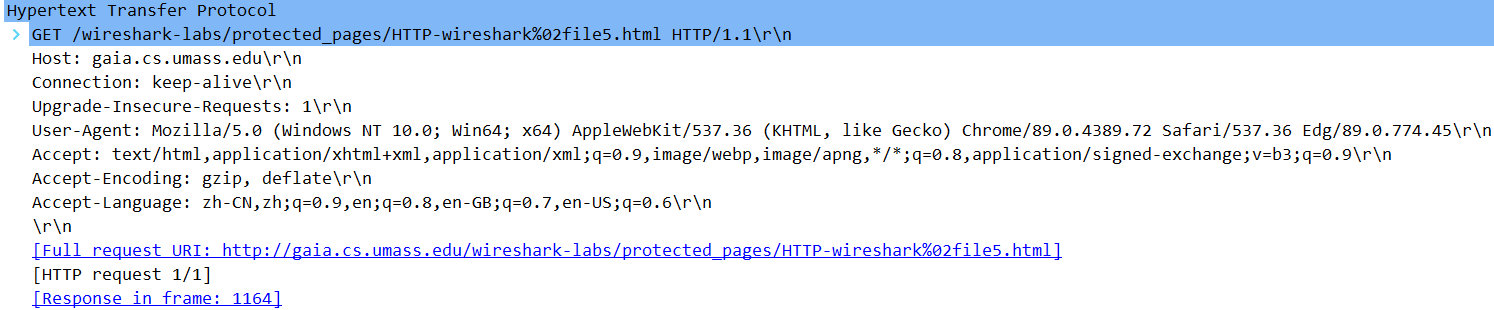
HTTP GET message from your browser?

19. When your browser’s sends the HTTP GET message for the second time, what new field is included in the HTTP GET message?

18．200 OK



19.出现了新的首部行cache\_control，authorization和referer



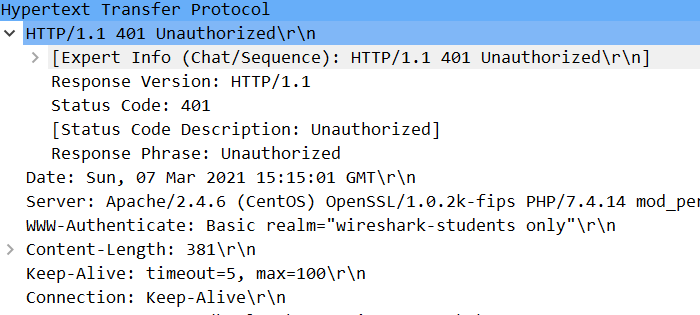
第一份请求报文



第二份请求报文

通用首部字段cache\_control操作缓存的工作机制。max-age指定:响应的最大Age值，如果判定缓存资源的缓存时间数值比指定时间的数值更小，那么客户端就接收缓存的资源。该报文中max-age值为0，那么缓存服务器需要将请求转发给源服务器。

安全请求首部authorization：验证用户身份。根据之前的响应报文中的“401 unauthorized”，请求报文加入“authorization”首部行。这里权限要求仅限于wireshark students，相应的用户名和密码分别为：wireshark-students，network。

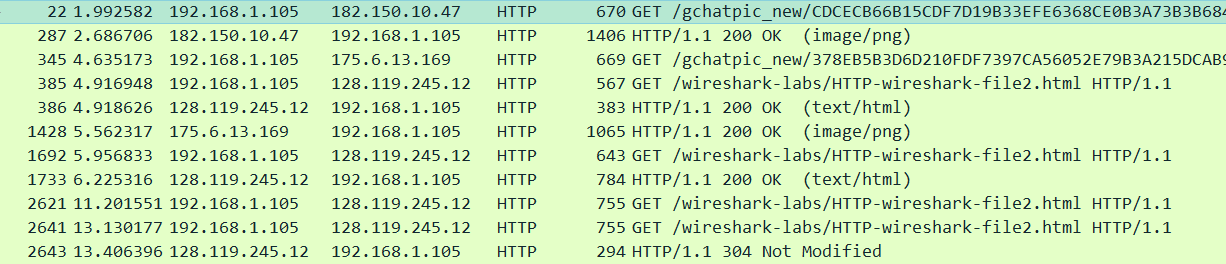


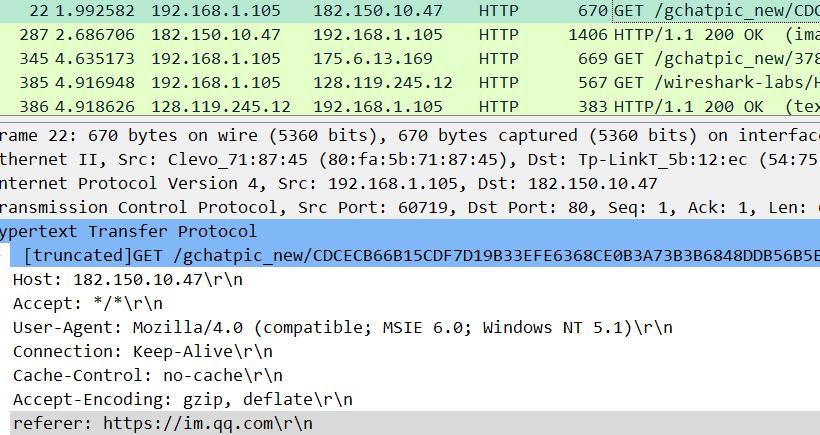
要求权限的响应报文

Referer首部：告知服务器用户是从哪个页面链接过来的，地址如图。

1. **其它报文分析**

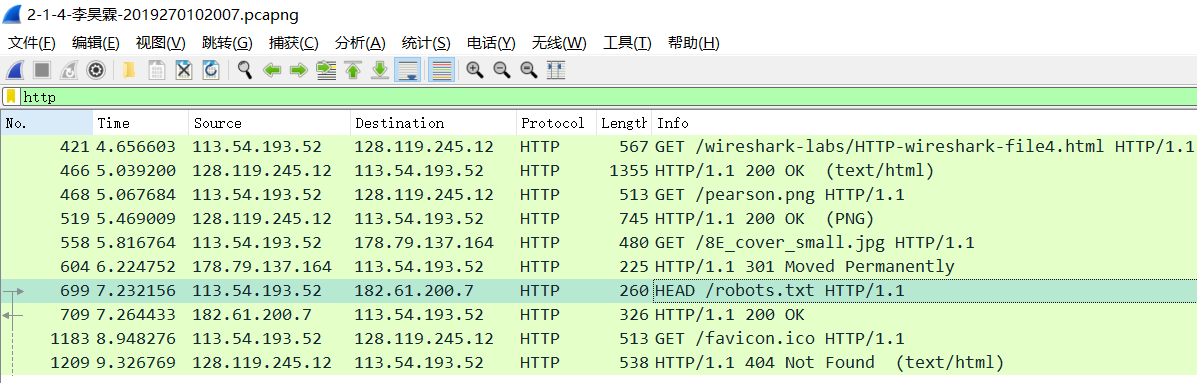
1.实验2中多余报文

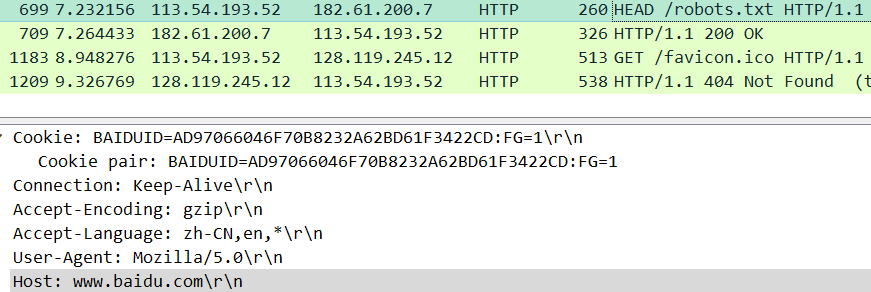




实验二中前三条及第六条报文均为无关报文，访问referer后发现，原因应为QQ更新。

1. 实验4中无关报文





实验4中后四份报文中的前两份为机器人想要访问百度，在访问之前查看robot.txt文件，查看是否有访问权限；后两份为请求实验中网站的图标，而图标不在服务器上。

**三、总结及心得体会**

本次实验让我学到了wireshark软件的基本使用方法，并在实验过程中认识到了使用不同的浏览器，是否清楚缓存，是否使用了vpn等代理软件对http报文的影响。

通过分析各个实验的http报文，我对http的GET请求，响应报文有了更深刻的认识，也学习了条件GET请求，HEAD请求以及其它课本上不曾出现过的诸如authorization等首部行的相关知识。

**四、对本实验过程及方法、手段的改进建议**

在实验时使用不同浏览器访问，挂 vpn后访问并分析相关报文可以帮助学生更深刻地认识http报文。

希望指导书中难懂的问题可以稍加中文注释，便于理解，谢谢老师。