中国科学技术大学计算机学院

计算机网络实验报告

实验二

利用 Wireshark 观察 http 报文

学　　号：**PB15111604**

姓　　名：金泽文

专　　业：计算机科学与技术

指导老师：张信明

中国科学技术大学计算机学院

2017 年 11 月 6 日

一、 实验目的

1、 熟悉并掌握 wireshark；

2、 通过捕获观察并分析 http 报文，理解 http；

二、 实验原理

Wireshark 是一个 packet 分析工具，可以抓取 packet，并分析出详细信息。Wireshark 使用 wincap 作为接口，直接与网卡进行 packet 交换，监听共享网络上传送的 packet。

三、 实验条件

1、 硬件条件： 联想 Y700：

i5-6300HQ 2.30GHz

16G 内存

Intel(R) Dual Band Wireless-AC 3165

2、 软件条件： Win10 Professional 1703

Wireshark2.4.0

四、 实验过程

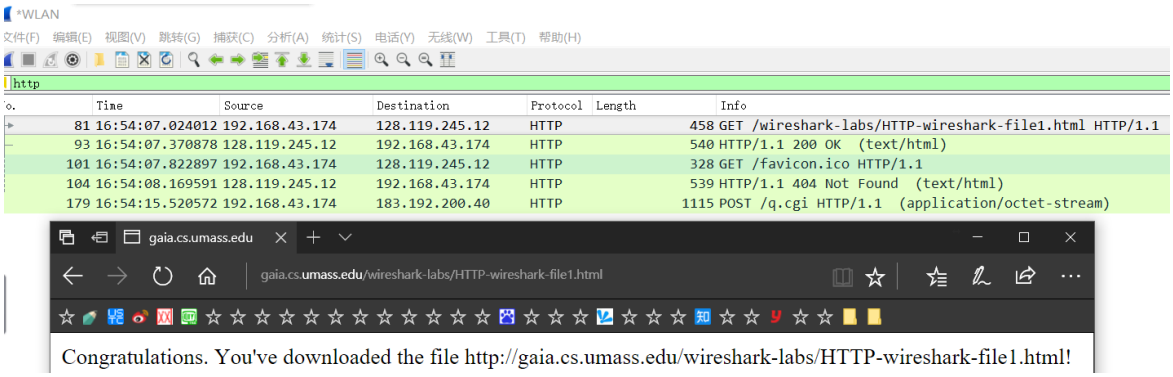
1、Wireshark 的安装

由于之前有分析报文的需要，所以早已装好。

2、 利用 Wireshark 观察 http 报文并回答问题

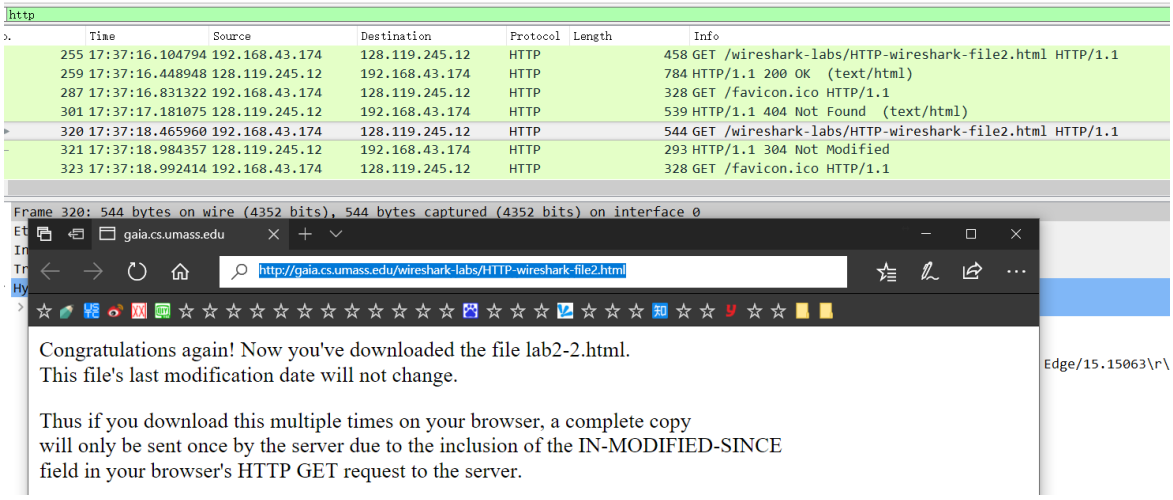
• The Basic HTTP GET/response interaction

打开 edge，打开 wireshark，稍等片刻，开始捕获同时设置过滤为“http”，打开 http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file1.html 停止捕获。得到下图：



• The HTTP CONDITIONAL GET/response interaction

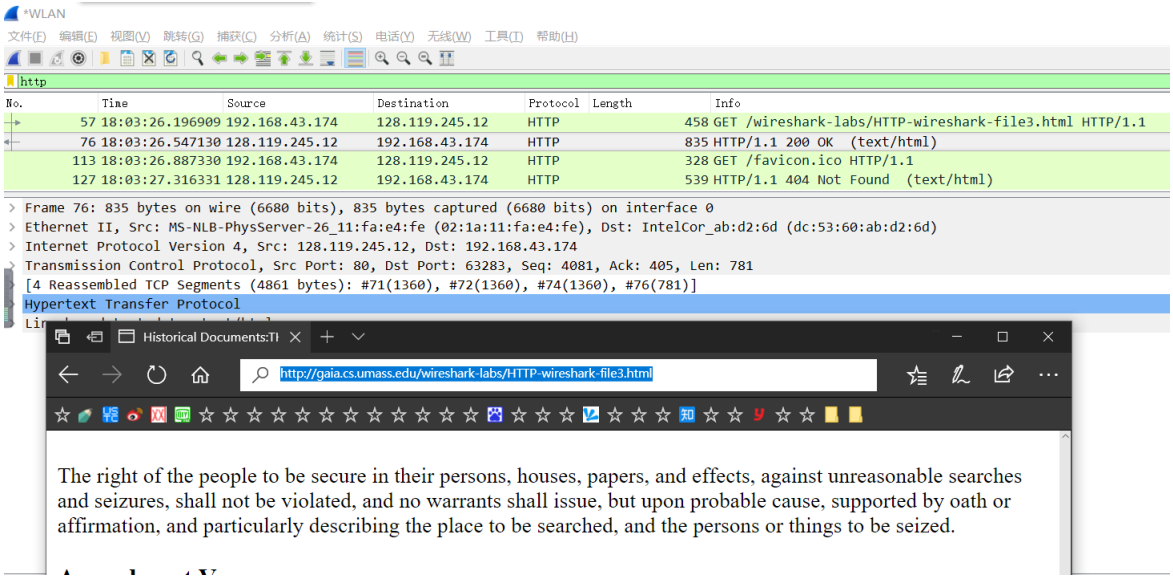
清除浏览器缓存，重新打开 wireshark 开始捕获，重新打开浏览器，打开网页 http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file2.html，并快速刷新一次，停止捕获。观察 wireshark 中 http 报文。得到下图：



• Retrieving Long Documents

清除浏览器缓存，打开 wireshark 开始捕获，打开网页

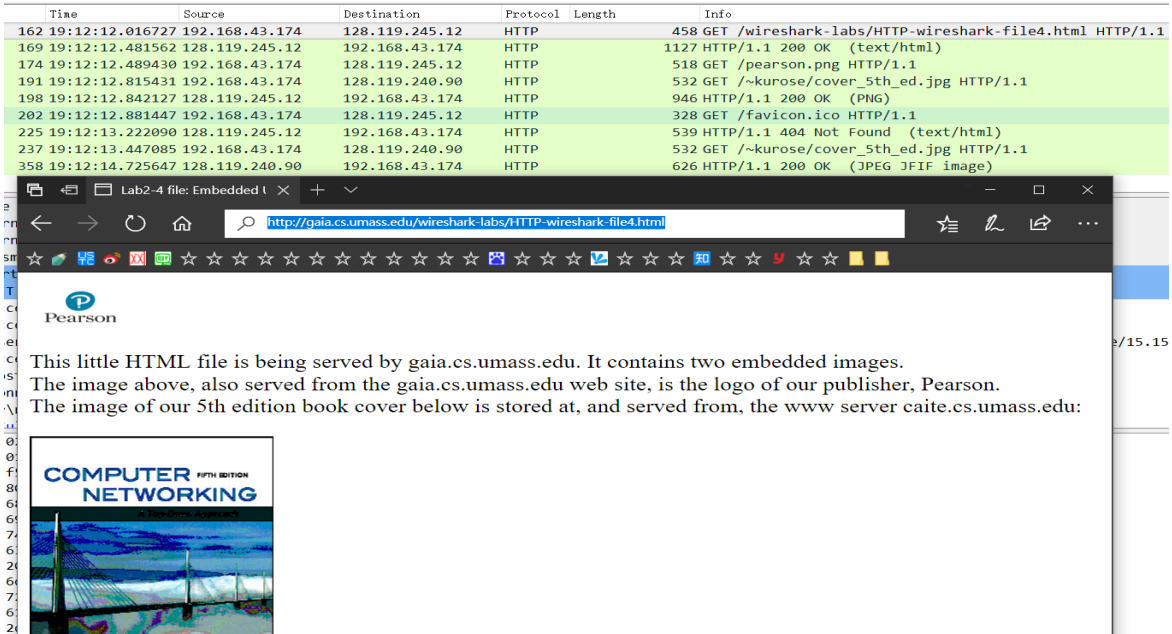
http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file3.html，停止捕获，得到：



• HTML Documents with Embedded Objects

清除浏览器缓存，打开 wireshark 开始捕获，打开网页

http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file4.html 等到两张图片加载完毕，停止捕获。得到下图：

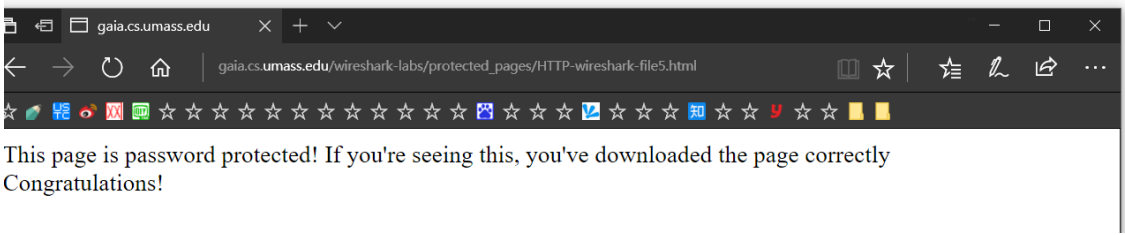


• HTTP Authentication

清除浏览器缓存，开始捕获，打开

http://gaia.cs.umass.edu/wireshark-labs/protected_pages/HTTP-wireshark-file5.html 并输入用户名密码，加载完毕之后停止捕获。得到下图：

	Time	Source	Destination	Protocol	Length	Info
242	20:17:19.711616	192.168.43.174	128.119.245.12	HTTP	474	GET /wireshark-labs/protected_pages/HTTP-wireshark
304	20:17:20.044995	128.119.245.12	192.168.43.174	HTTP	771	HTTP/1.1 401 Unauthorized (text/html)
342	20:17:25.845634	192.168.43.174	128.119.245.12	HTTP	533	GET /wireshark-labs/protected_pages/HTTP-wireshark
345	20:17:26.251336	128.119.245.12	192.168.43.174	HTTP	544	HTTP/1.1 200 OK (text/html)
352	20:17:26.680790	192.168.43.174	128.119.245.12	HTTP	328	GET /favicon.ico HTTP/1.1
355	20:17:27.169447	128.119.245.12	192.168.43.174	HTTP	539	HTTP/1.1 404 Not Found (text/html)

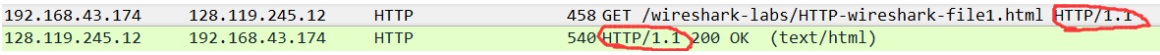


五、 结果分析

以下是《http 报文抓取及分析》对应的回答

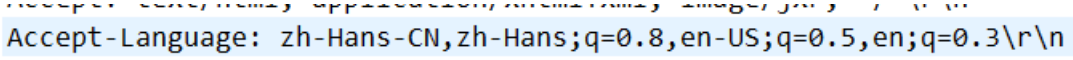
1. Is your browser running HTTP version 1.0 or 1.1? What version of HTTP is the server running?

答：都是 HTTP 1.1，如下图:



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

答：简体中文（大陆使用）、简体中文、英文（美）、英文。如下图：

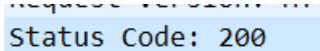


3. What is the IP address of your computer? Of the gaia.cs.umass.edu server?

答：我的是 192.168.43.174，服务器的是 128.119.245.12

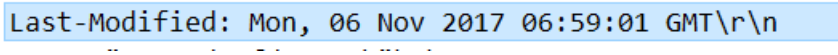
4. What is the status code returned from the server to your browser?

答：200



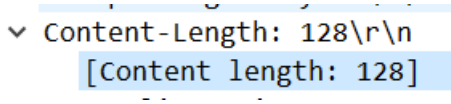
5. When was the HTML file that you are retrieving last modified at the server?

答：



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

答：128 字节，如下图



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.

答：Server，Last-Modified 等

```
Date: Mon, 06 Nov 2017 08:54:07 GMT\r\n
Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips PHP/5.4.16 mod_perl/2.0.10 Perl/v5.16.3\r\n
Last-Modified: Mon, 06 Nov 2017 06:59:01 GMT\r\n
ETag: "80-55d4afd44a53b"\r\n
Accept-Ranges: bytes\r\n
```

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?

答：没有

9. Inspect the contents of the server response. Did the server explicitly return the contents of the file? How can you tell?

答：第一次的回应确实返回了文件内容，因为这次的报文里包含了 Content-Type:text/html 和 Content-Length。

```
> Content-Length: 371\r\n
  Keep-Alive: timeout=5, max=100\r\n
  Connection: Keep-Alive\r\n
  Content-Type: text/html; charset=UTF-8\r\n
```

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?

答：有。

```
Host: gaia.cs.umass.edu\r\n
If-Modified-Since: Mon, 06 Nov 2017 06:59:01 GMT\r\n
```

它后面是上一次 response 时发送的文件 Last-Modified 对应的时间。

```
Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips PHP/5.4.1
Last-Modified: Mon, 06 Nov 2017 06:59:01 GMT\r\n
ETag: "173-55d4afd449d6b"\r\n
```

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.

答：304 Not Modified。没有实际发送文件内容。因为这次的 response

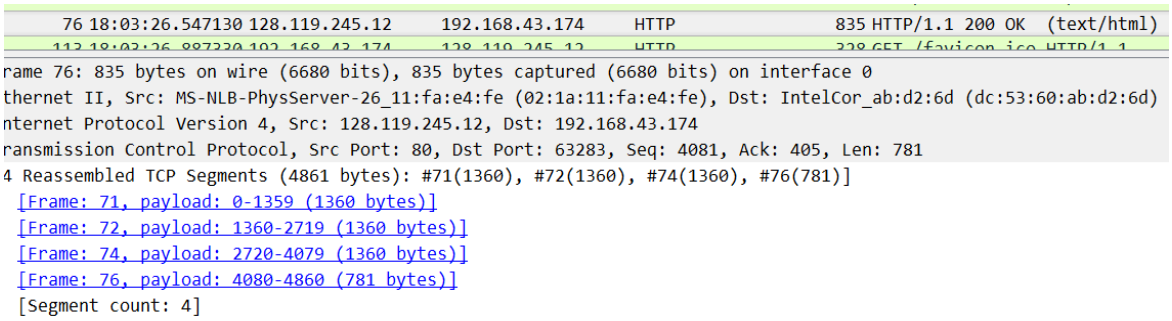
没有 Content-Type，Content-Length，说明没有文件内容发过来

12. How many HTTP GET request messages were sent by your browser?

答：1个

13. How many data-containing TCP segments were needed to carry the single HTTP response?

答：4个。



14. What is the status code and phrase associated with the response to the HTTP GET request?

答：200 OK

15. Are there any HTTP status lines in the transmitted data associated with a TCP induced "Continuation"?

答：没有。正如 lab 的 pdf 里面提到的

blank line, followed by the entity body. In the case of our HTTP GET, the entity body in the response is the *entire* requested HTML file. In our case here, the HTML file is rather long, and at 4500 bytes is too large to fit in one TCP packet. The single HTTP response message is thus broken into several pieces by TCP, with each piece being contained within a separate TCP segment (see Figure 1.20 in the text). Each TCP segment is recorded as a separate packet by Wireshark, and the fact that the single HTTP response was fragmented across multiple TCP packets is indicated by the "Continuation" phrase displayed by Wireshark. We stress here that there is no "Continuation" message in HTTP!

Answer the following questions:

16. How many HTTP GET request messages were sent by your browser? To which Internet addresses were these GET requests sent?

答：一共5个。3个发往gaia.cs.umass.edu（128.119.245.12），2个发往

caite.cs.umass.edu（128.119.240.90）。

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.

答：parallel，因为 person.png 发出后没等到 person.png 的 response 回来，就已经发送了 cover_5th_ed.jpg 的 get 请求了，如下图：

174	19:12:12.489430	192.168.43.174	128.119.245.12	HTTP	518 GET /pearson.png HTTP/1.1
191	19:12:12.815431	192.168.43.174	128.119.240.90	HTTP	532 GET /~kurose/cover_5th_ed.jpg HTTP/1.1
198	19:12:12.842127	128.119.245.12	192.168.43.174	HTTP	946 HTTP/1.1 200 OK (PNG)
202	19:12:12.881447	192.168.43.174	128.119.245.12	HTTP	328 GET /favicon.ico HTTP/1.1
225	19:12:13.222090	128.119.245.12	192.168.43.174	HTTP	539 HTTP/1.1 404 Not Found (text/html)
237	19:12:13.447085	192.168.43.174	128.119.240.90	HTTP	532 GET /~kurose/cover_5th_ed.jpg HTTP/1.1
358	19:12:14.725647	128.119.240.90	192.168.43.174	HTTP	626 HTTP/1.1 200 OK (JPEG JFIF image)

虽然 get person.png 的请求发送过了 0.33s 左右才有 cover_5th_ed.jpg 的 get 请求发送。

174	19:12:12.489430	192.168.43.174	128.119.245.12	HTTP	518 GET /pearson.png HTTP/1.1
175	19:12:12.496202	192.168.43.174	192.168.43.1	DNS	78 Standard query 0x87c1 A manic.cs.umass.edu
176	19:12:12.515413	192.168.43.1	192.168.43.174	DNS	126 Standard query response 0x87c1 A manic.cs.
177	19:12:12.516161	192.168.43.174	128.119.240.90	TCP	66 51070 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS

但是仔细一看（如下图），发送 GET pearson.jpg 请求之后过了 0.007s 左右发送了 dns 查询请求，正是另一张图片地址的 dns 查询。这么看来，get 前后相差的 0.33s 根本不多。

174	19:12:12.489430	192.168.43.174	128.119.245.12	HTTP
175	19:12:12.496202	192.168.43.174	192.168.43.1	DNS
176	19:12:12.515413	192.168.43.1	192.168.43.174	DNS
177	19:12:12.516161	192.168.43.174	128.119.240.90	TCP

```
Frame 175: 78 bytes on wire (624 bits), 78 bytes captured (624 bits)
Ethernet II, Src: IntelCor_ab:d2:6d (dc:53:60:ab:d2:6d), Dst: MS-NLB
Internet Protocol Version 4, Src: 192.168.43.174, Dst: 192.168.43.1
User Datagram Protocol, Src Port: 55863, Dst Port: 53
Domain Name System (query)
  [Response In: 176]
  Transaction ID: 0x87c1
> Flags: 0x0100 Standard query
  Questions: 1
  Answer RRs: 0
  Authority RRs: 0
  Additional RRs: 0
v Queries
  v manic.cs.umass.edu: type A, class IN
      Name: manic.cs.umass.edu
```

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

HTTP GET message from your browser?

答：401 Unauthorized

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

答：Authorizations： Basic

```
Connection: Keep-Alive\r\n
Authorization: Basic d2lyZXNoYXJrLXN0dWRlbnRzOm5ldHdvcms=\r\n
    Credentials: wireshark-students:network
\r\n
```

以下是《Wireshark 简介》对应的回答

1. List up to 10 different protocols that appear in the protocol column in the unfiltered packet-listing window in step 7 above.

答：TCP UDP HTTP TLSv1.2 OICQ ARP DNS RTPproxy NBNS MDNS

2. How long did it take from when the HTTP GET message was sent until the HTTP OK reply was received? (By default, the value of the Time column in the packetlisting window is the amount of time, in seconds, since Wireshark tracing began. To display the Time field in time-of-day format, select the Wireshark *View* pull down menu, then select Time *Display Format*, then select *Time-of-day*.)

答：约 0.47s

270	21:20:24.641385	192.168.43.174	128.119.245.12	HTTP	544	GET /wireshark-labs/INTRO-wireshark-file1.html HTTP/1.1
272	21:20:25.108249	128.119.245.12	192.168.43.174	HTTP	293	HTTP/1.1 304 Not Modified

3. What is the Internet address of the gaia.cs.umass.edu (also known as wwwnet. cs.umass.edu)? What is the Internet address of your computer?

答：128.119.245.12 192.168.43.174（本地）

4. Print the two HTTP messages displayed in step 9 above. To do so, select *Print* from the Wireshark *File* command menu, and select "*Selected Packet Only"* and *"Print as displayed"* and then click OK.

答：助教说不用答啦啦啦

以下是《ppt》对应的回答

1. *分析 HTTP 中 get 和 post 请求方式的区别。*

答：

- 通常意义上的区别：

  GET 一般用于查询信息，

  POST 一般用于改变信息。

- 原理上的区别：

  根据 HTTP 规范，GET 用于信息的获取，应该是安全并且幂等的。

  根据 HTTP 规范，POST 表示可能修改服务器上的资源的请求，是非幂等的。

- 实际中的区别：

  1. GET 请求将请求的数据附在 URL 之后，

     POST 将请求的数据放在 HTTP 包的包体中。而实际中浏览器或者操作系统可能对 url 长度进行限制，所以 GET，POST 此时所能传输的信息有区别。

  2. POST 的安全性别 GET 高。

     对于密码等隐私数据，GET 会放在 url 上，而 POST 不会。ur