计算机网络 HTTP 实验

PB20000180 刘良宇

1. 基本的 HTTP GET 响应

```
No.     Time              Source                Destination           Protocol Length Info
    128 16:20:54.561422481 114.214.255.65       128.119.245.12        HTTP     553    GET /wireshark-labs/HTTP-wireshark-
file1.html HTTP/1.1
Frame 128: 553 bytes on wire (4424 bits), 553 bytes captured (4424 bits) on interface any, id 0
Linux cooked capture v1
Internet Protocol Version 4, Src: 114.214.255.65, Dst: 128.119.245.12
Transmission Control Protocol, Src Port: 48690, Dst Port: 80, Seq: 1, Ack: 1, Len: 485
Hypertext Transfer Protocol
    GET /wireshark-labs/HTTP-wireshark-file1.html HTTP/1.1\r\n
    Host: gaia.cs.umass.edu\r\n
    Connection: keep-alive\r\n
    Cache-Control: max-age=0\r\n
    Upgrade-Insecure-Requests: 1\r\n
    User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/104.0.0.0 Safari/537.36\r\n
    Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,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,en;q=0.9\r\n
    \r\n
    [Full request URI: http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file1.html]
    [HTTP request 1/1]
    [Response in frame: 139]
No.     Time              Source                Destination           Protocol Length Info
    139 16:20:54.942389775 128.119.245.12        114.214.255.65        HTTP     554    HTTP/1.1 200 OK  (text/html)
Frame 139: 554 bytes on wire (4432 bits), 554 bytes captured (4432 bits) on interface any, id 0
Linux cooked capture v1
Internet Protocol Version 4, Src: 128.119.245.12, Dst: 114.214.255.65
Transmission Control Protocol, Src Port: 80, Dst Port: 48690, Seq: 1, Ack: 486, Len: 486
Hypertext Transfer Protocol
    HTTP/1.1 200 OK\r\n
    Date: Mon, 19 Sep 2022 08:20:54 GMT\r\n
    Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips PHP/7.4.30 mod_perl/2.0.11 Perl/v5.16.3\r\n
    Last-Modified: Mon, 19 Sep 2022 05:59:01 GMT\r\n
    ETag: "80-5e901688a434a"\r\n
    Accept-Ranges: bytes\r\n
    Content-Length: 128\r\n
    Keep-Alive: timeout=5, max=100\r\n
    Connection: Keep-Alive\r\n
    Content-Type: text/html; charset=UTF-8\r\n
    \r\n
    [HTTP response 1/1]
    [Time since request: 0.380967294 seconds]
    [Request in frame: 128]
    [Request URI: http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file1.html]
    File Data: 128 bytes
Line-based text data: text/html (4 lines)
```

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

   都是 `HTTP 1.1`

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

   `zh, en`

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

   本机：`114.214.255.65`

   服务器：`128.119.245.12`

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

   `200 OK`

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

   `Mon, 19 Sep 2022 05:59:01 GMT`

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

   `Content-Length: 128`

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.

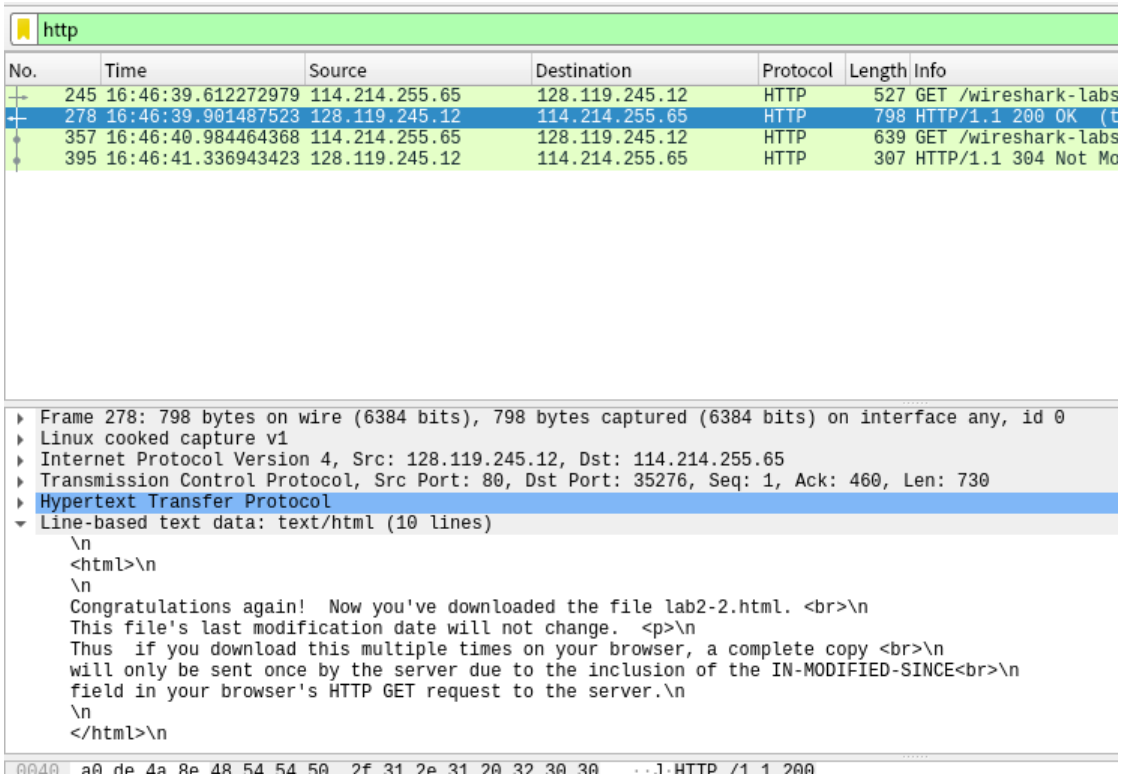
   例如 `Accept-Language`

2. 有条件的 HTTP GET 响应

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?

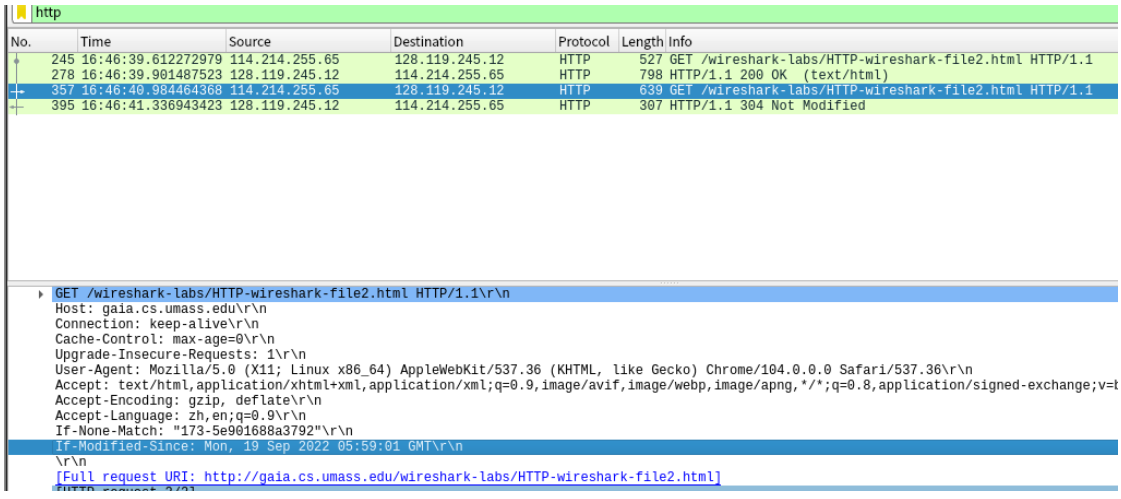
   无

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



   存在 `content`（从头里面的 `content-length` 也可以看出）

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?



存在，和上一条 GET 服务器返回内容的 `Last-Modified` 值 `Mon, 19 Sep 2022 05:59:01 GMT` 相同

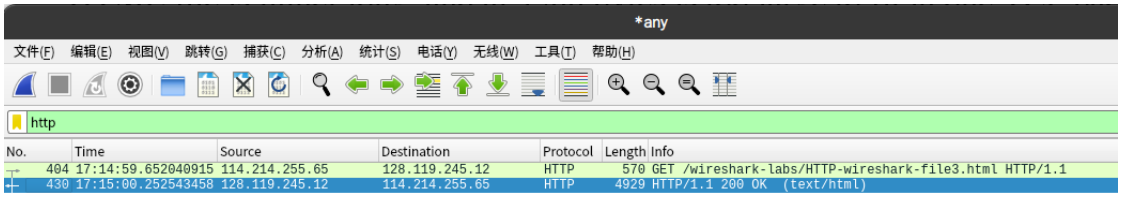
11. What is the HTTP status code and phrase returned from the server in response to this second HTTP GET? Did the server explicitly return the contents of the file? Explain.

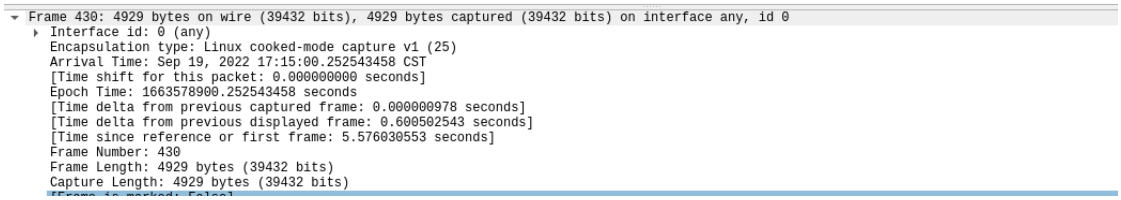
    `304 Not Modified`

    没有返回文件内容（没有 `content`，头里也没有 `content-length`）

3. 检索长文档

12. How many HTTP GET request messages did your browser send? Which packet number in the trace contains the GET message for the Bill of Rights?

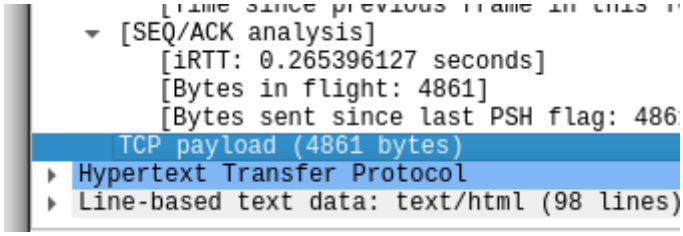




    - 发送了 1 个 GET 请求
    - 430 收到了结果

13. Which packet number in the trace contains the status code and phrase associated with the response to the HTTP GET request?
    - 也为 430
14. What is the status code and phrase in the response?
    - `200 OK`
15. How many data-containing TCP segments were needed to carry the single HTTP response and the text of the Bill of Rights?
    - 1个

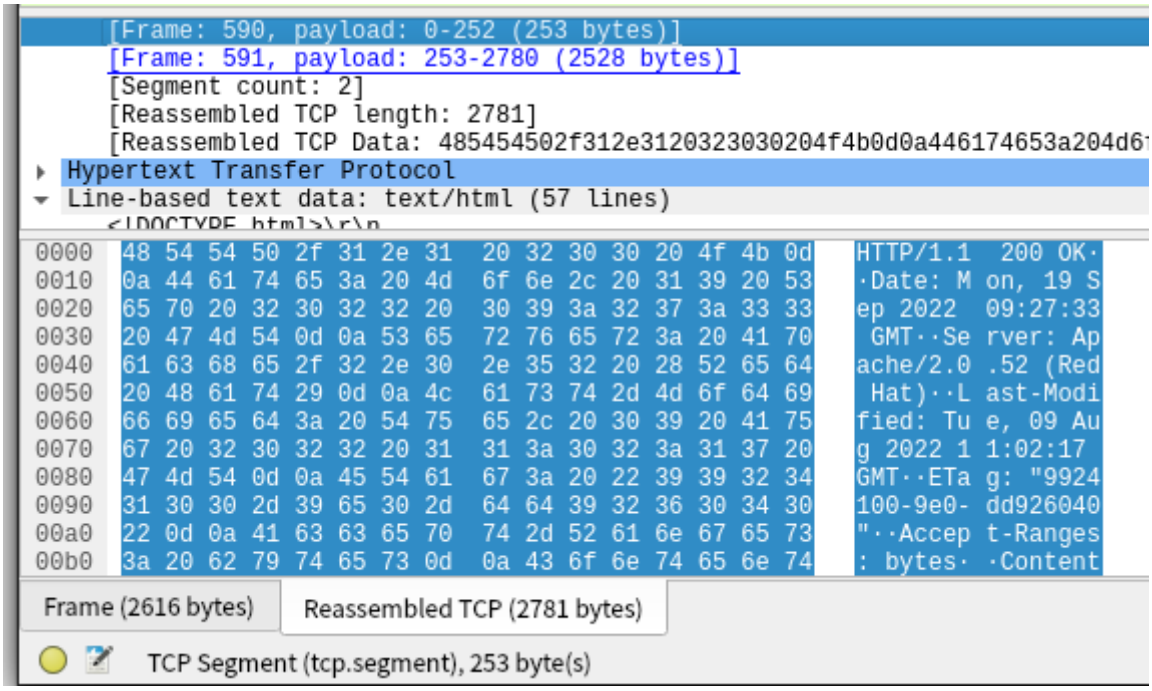
注：以上经过反复实验，均只有一个 packet



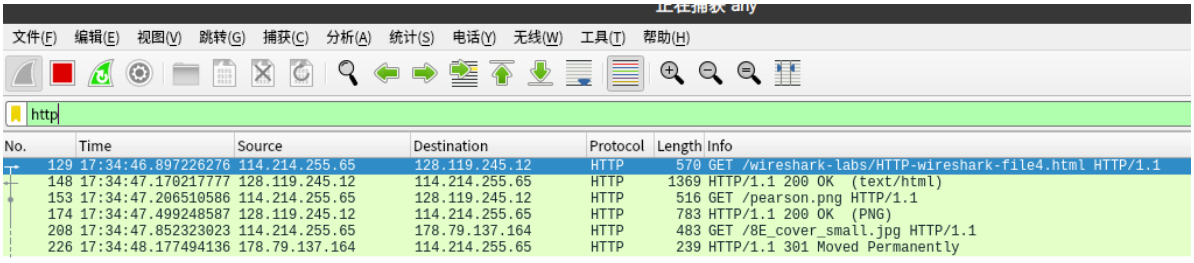
作为对比，这里抓取另外一个网页的 GET 包：

```
        [Bytes in flight: 2781]
        [Bytes sent since last PSK flag: 2528]
    TCP payload (2528 bytes)
    TCP segment data (2528 bytes)
▾ [2 Reassembled TCP Segments (2781 bytes): #590(253), #591(2528)]
    [Frame: 590, payload: 0-252 (253 bytes)]
    [Frame: 591, payload: 253-2780 (2528 bytes)]
    [Segment count: 2]
    [Reassembled TCP length: 2781]
    [Reassembled TCP Data: 485454502f312e3120323030204f4b0d0a446174653a204d6
```

对于这个示例而言，GET 的结果存在了 590 和 591 中，用了两个 packet，返回的 HTTP 状态码信息在第一个包中



4. 内嵌对象的 HTML 文档




```
▾ Frame 129: 570 bytes on wire (4560 bits), 570 bytes captured (4560 bits) on interface any, id 0
  ▸ Interface id: 0 (any)
    Encapsulation type: Linux cooked-mode capture v1 (25)
    Arrival Time: Sep 19, 2022 17:34:46.897226276 CST
    [Time shift for this packet: 0.000000000 seconds]
    Epoch Time: 1663580086.897226276 seconds
    [Time delta from previous captured frame: 0.005405529 seconds]
```

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

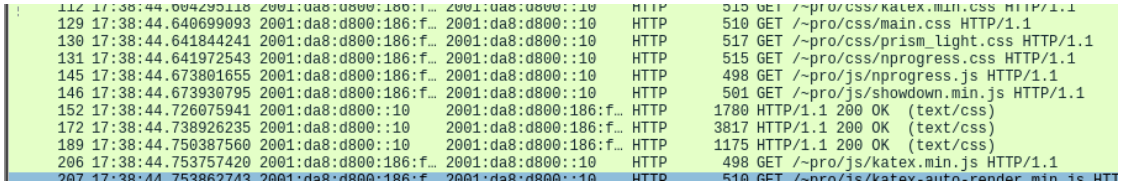
    3 个

    - HTML 文档和第一张图片：`128.119.245.12`
    - 第二张图片：`178.79.137.164`

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.

    从时间上无法辨别（可能是串行，可能是并行）

    对另一个资源比较多的网站进行抓包，可以看出另一个网站是并行的：



5. HTTP 鉴权

```
No.     Time              Source                Destination           Protocol Length Info
    175 17:40:45.409976017 114.214.255.65        128.119.245.12        HTTP     586    GET /wireshark-labs/protected_pages/
HTTP-wireshark-file5.html HTTP/1.1
Frame 175: 586 bytes on wire (4688 bits), 586 bytes captured (4688 bits) on interface any, id 0
Linux cooked capture v1
Internet Protocol Version 4, Src: 114.214.255.65, Dst: 128.119.245.12
Transmission Control Protocol, Src Port: 60808, Dst Port: 80, Seq: 1, Ack: 1, Len: 518
Hypertext Transfer Protocol
No.     Time              Source                Destination           Protocol Length Info
    220 17:40:45.704283677 128.119.245.12        114.214.255.65        HTTP     785    HTTP/1.1 401 Unauthorized  (text/html)
Frame 220: 785 bytes on wire (6280 bits), 785 bytes captured (6280 bits) on interface any, id 0
Linux cooked capture v1
Internet Protocol Version 4, Src: 128.119.245.12, Dst: 114.214.255.65
Transmission Control Protocol, Src Port: 80, Dst Port: 60808, Seq: 1, Ack: 519, Len: 717
Hypertext Transfer Protocol
Line-based text data: text/html (12 lines)
No.     Time              Source                Destination           Protocol Length Info
   1186 17:40:55.227248570 114.214.255.65        128.119.245.12        HTTP     645    GET /wireshark-labs/protected_pages/
HTTP-wireshark-file5.html HTTP/1.1
Frame 1186: 645 bytes on wire (5160 bits), 645 bytes captured (5160 bits) on interface any, id 0
Linux cooked capture v1
Internet Protocol Version 4, Src: 114.214.255.65, Dst: 128.119.245.12
Transmission Control Protocol, Src Port: 53354, Dst Port: 80, Seq: 1, Ack: 1, Len: 577
Hypertext Transfer Protocol
No.     Time              Source                Destination           Protocol Length Info
   1219 17:40:55.544139377 128.119.245.12        114.214.255.65        HTTP     558    HTTP/1.1 200 OK  (text/html)
Frame 1219: 558 bytes on wire (4464 bits), 558 bytes captured (4464 bits) on interface any, id 0
Linux cooked capture v1
Internet Protocol Version 4, Src: 128.119.245.12, Dst: 114.214.255.65
Transmission Control Protocol, Src Port: 80, Dst Port: 53354, Seq: 1, Ack: 578, Len: 490
Hypertext Transfer Protocol
Line-based text data: text/html (6 lines)
```

18. What is the server's response (status code and phrase) in response to the initial HTTP GET message from your browser?

    `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?