

Andre Hei Wang Law

4017 5600

COEN 366 – FL-X

Socket Programming + Wireshark Assignment 1

1. “Socket programming assignment.pdf” Assignment

HTML Code - “coen366.html”:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta content="IE=edge" http-equiv="X-UA-Compatible">
  <meta content="width=device-width, initialscale=1.0"
name="viewport">
  <title>Document</title>
</head>
<body>
<h1>Welcome to COEN 366</h1>
<p>Course info:</p>
<ul>
  <li>Given by Prof. Chadi Assi</li>
  <li>Prof's email: chadi.assi@concordia.ca</li>
  <hr>
  <li>There are three TAs for this course:<br/>
    <ul>
      <li>Shreya Khisa</li>
      <li>Ali Amhaz</li>
      <li>Y A Joarder</li>
    </ul>
  </li>
  <hr>
</ul>
</body>
</html>
```

Python Code - "http_server.py":

```
# Andre Hei Wang Law
# 4017 5600
# Socket Programming Assignment 1

# Code based on "Module 2" Moodle slides, page.104

# URL link: http://localhost:12000/coen366.html

from socket import *

serverPort = 12000

# create TCP welcoming socket
serverSocket = socket(AF_INET, SOCK_STREAM)
serverSocket.bind(('', serverPort))

# server begins listening for incoming TCP requests
serverSocket.listen(1)
print("The server is ready to receive")

# loop forever
while True:
    # server waits on accept() for incoming
    # requests, new socket created on return
    connectionSocket, addr = serverSocket.accept()

    # read bytes from socket (but not address as in UDP)
    request = connectionSocket.recv(1024).decode()

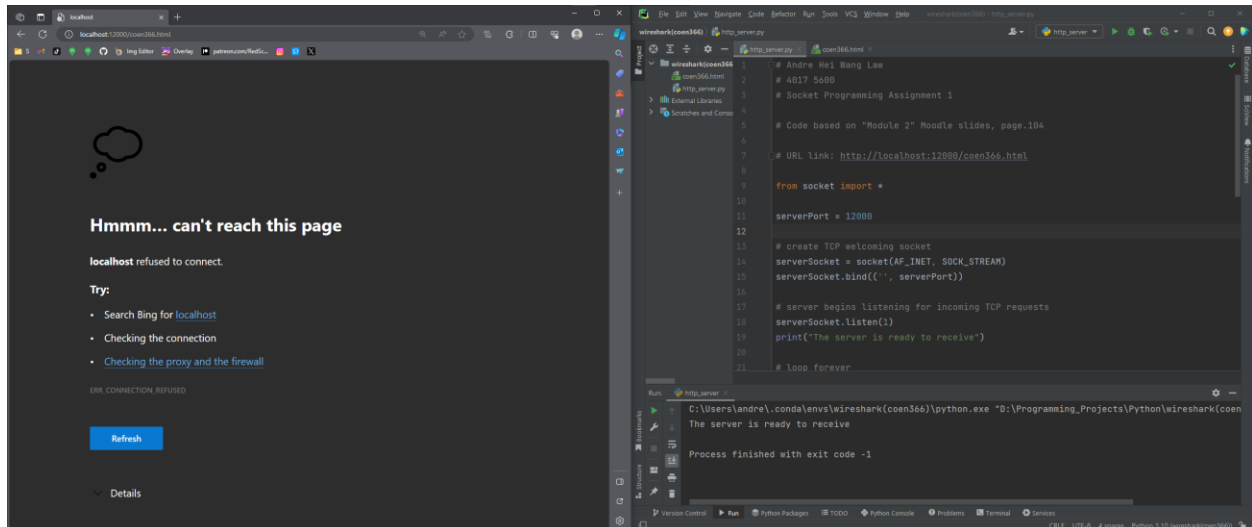
    # well-formed request (return error otherwise)
    try:
        with open('coen366.html', 'r') as htmlFile:
            htmlContent = htmlFile.read()
            response = "HTTP/1.1 200 OK\r\n\r\n" + htmlContent
    except FileNotFoundError: # error when file doesn't exist
        response = "HTTP/1.1 404 Not Found\r\n\r\nFile not found"
    except PermissionError: # error when permissions are set
        properly
        response = "HTTP/1.1 403 Forbidden\r\n\r\nPermission denied"

    # send HTTP response to the client (browser)
    connectionSocket.send(response.encode())

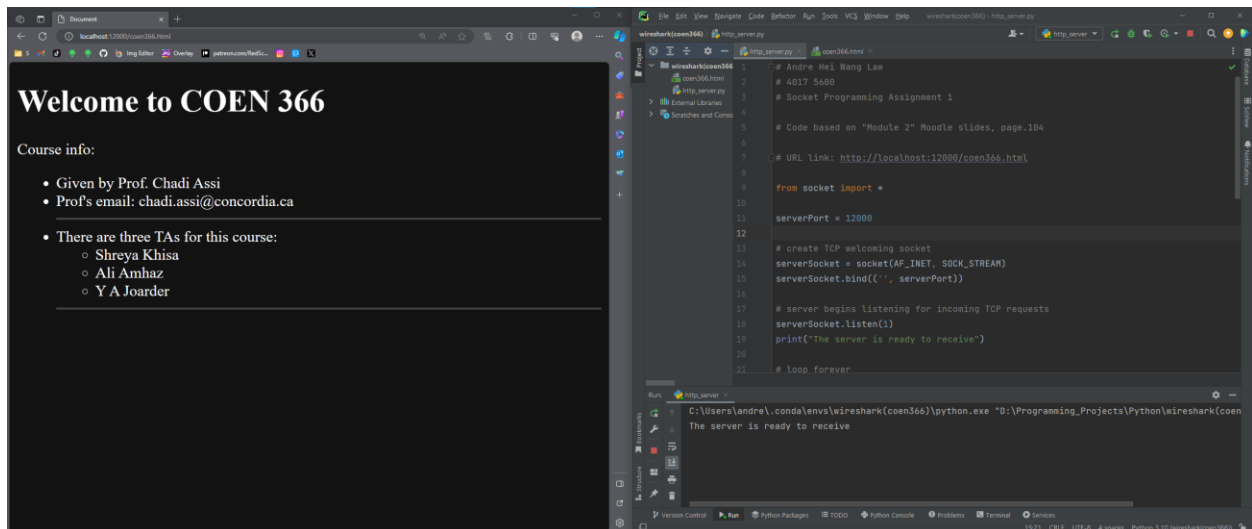
    # close the connection
    connectionSocket.close()
```

Before Running the Python Code:

- Url: <http://localhost:12000/coen366.html>

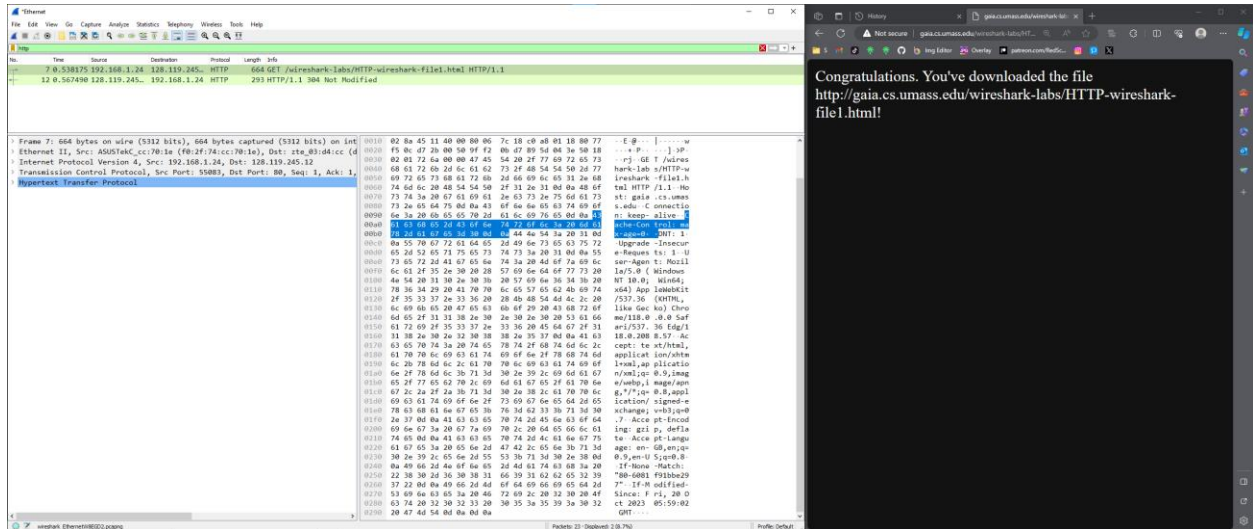


After Running the Python Code:



2. “Wireshark assignment 1.pdf” Assignment

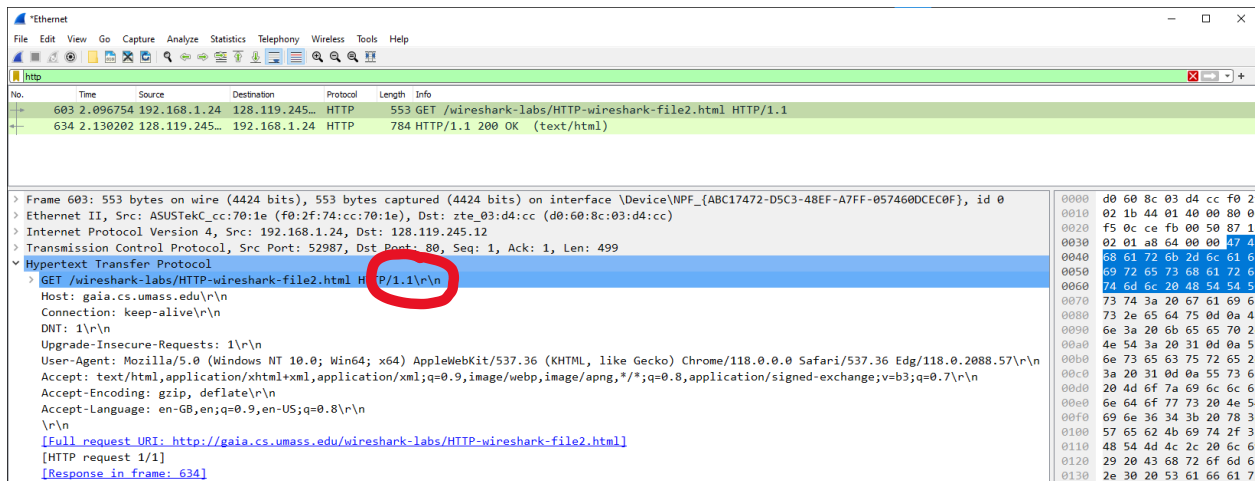
2.1 The Basic HTTP GET/response interaction



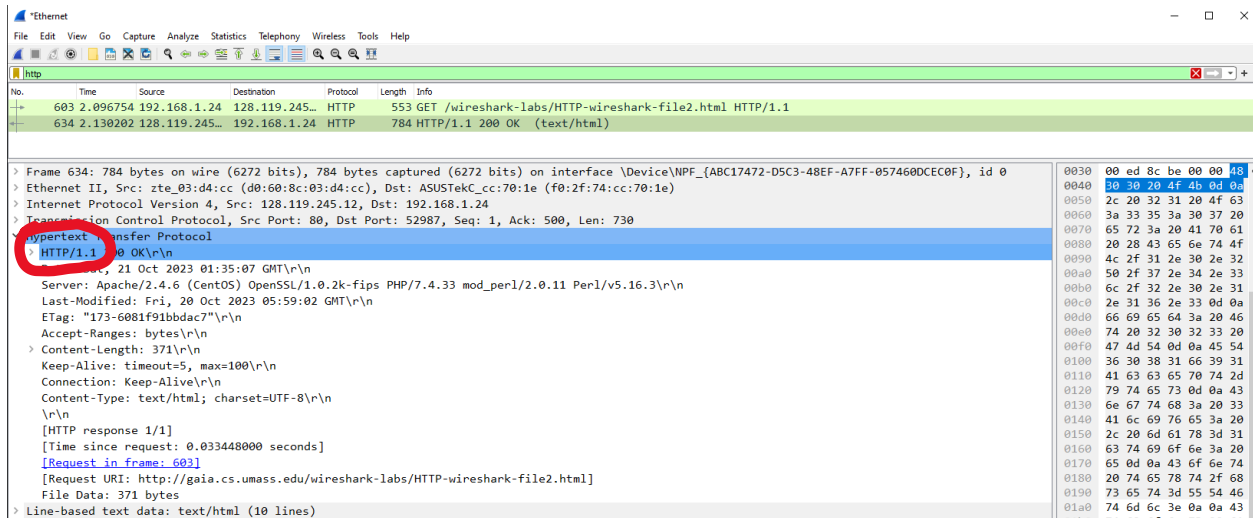
The above image is the baseline output after waiting a minute and opening gaia website.

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

From HTTP GET, Hypertext Transfer Protocol, my browser is running version 1.1.

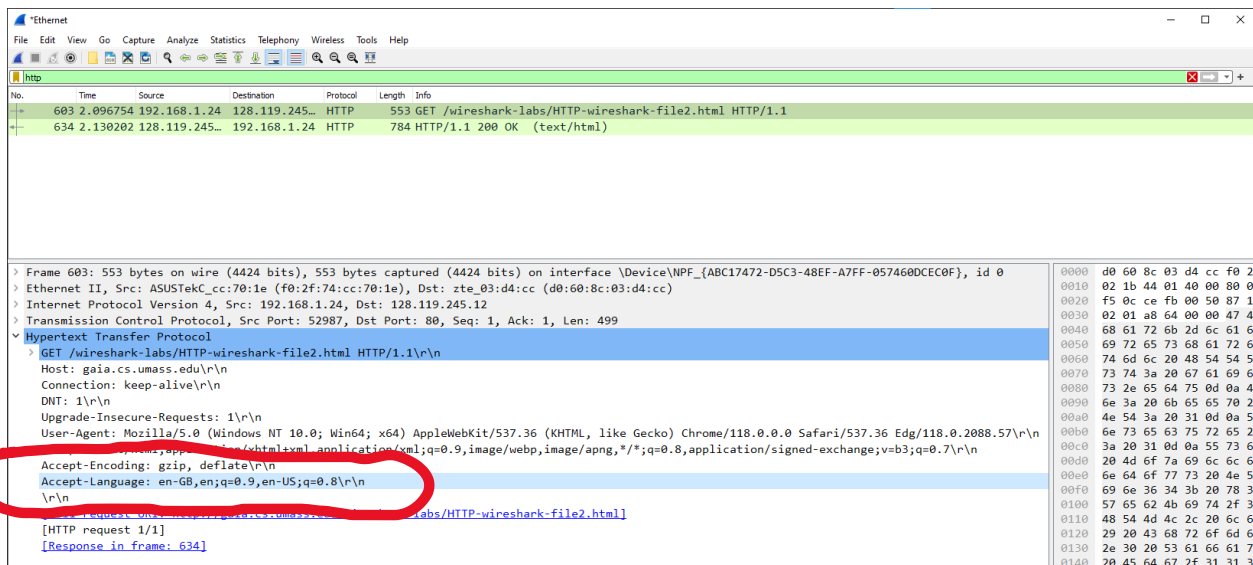


From HTTP response, Hypertext Transfer Protocol, the server is running version 1.1.



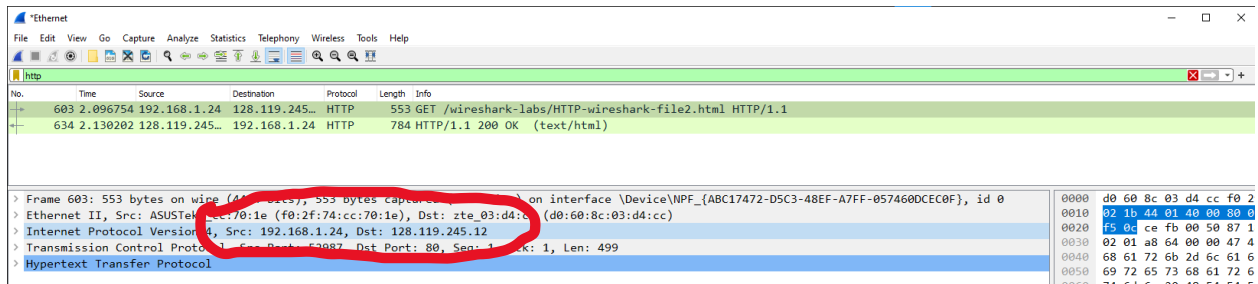
2. What languages (if any) does your browser indicate that it can accept to the server?

From HTTP GET, Hypertext Transfer Protocol, it accepts: en-GB,en;q=0.9,en-US;q=0.8\r\n



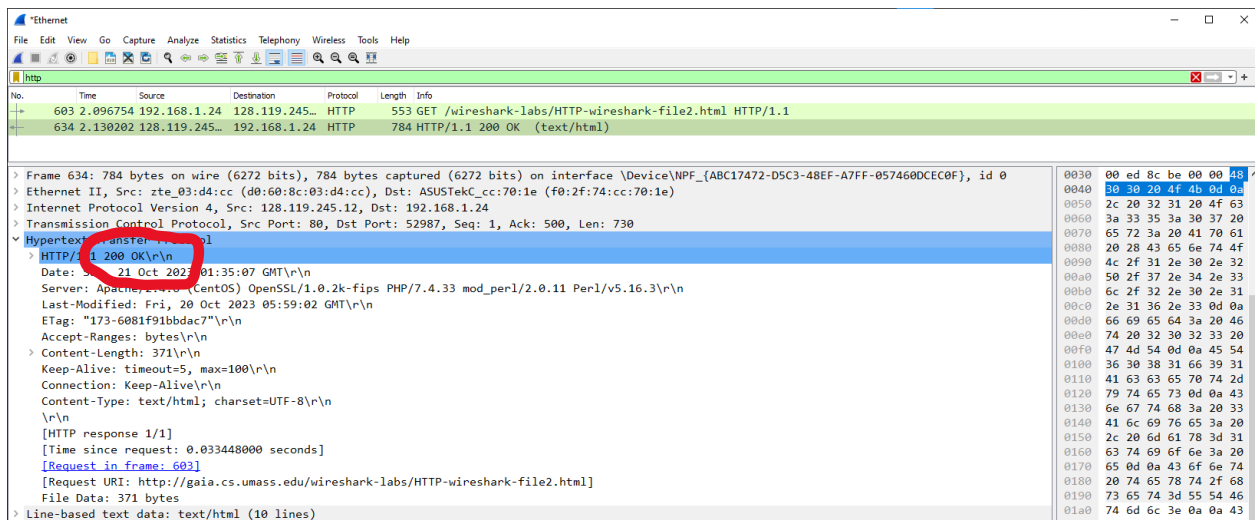
3. What is the IP address of your computer? Of the gaia.cs.umass.edu server?

From HTTP GET, my computer's IP is Src: 192.168.1.24 and gaia's is Dst: 128.119.245.12



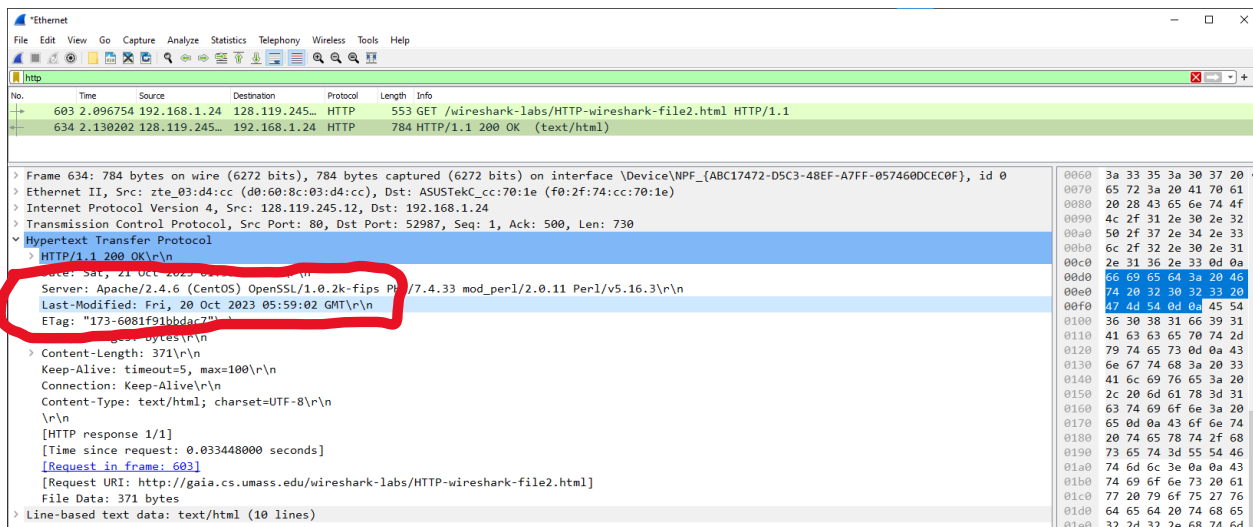
4. What is the status code returned from the server to your browser?

From HTTP response, Hypertext Transfer Protocol, the status code returned from the server to my browser is 200 OK.



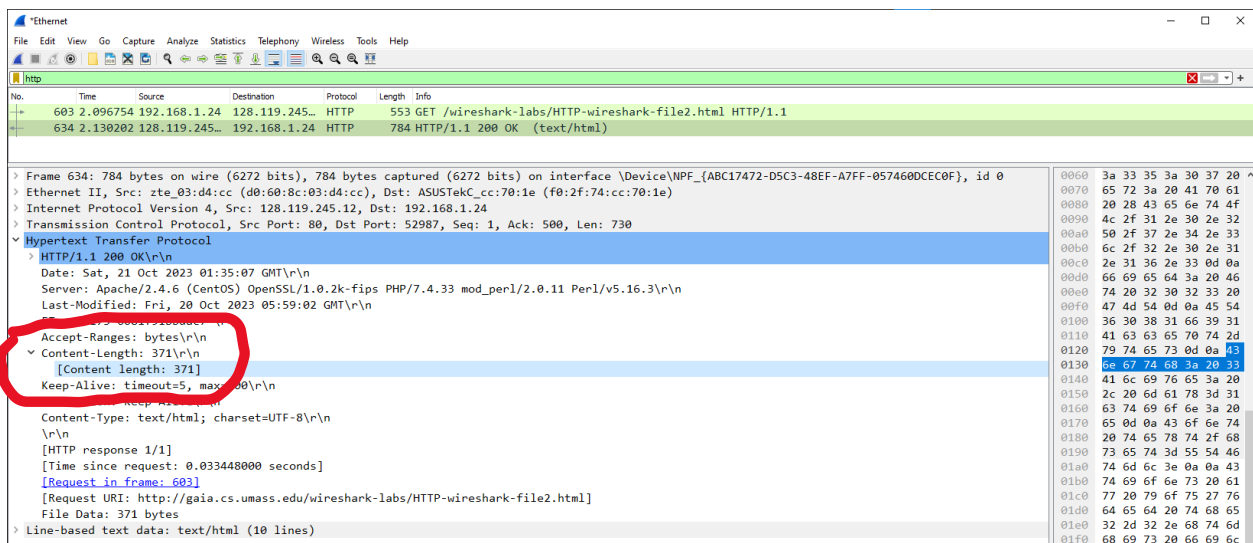
5. When was the HTML file that you are retrieving last modified at the server?

From HTTP response, Hypertext Transfer Protocol, the last modified time is Fri, 20 Oct 2023 05:59:02 GMT.



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

From HTTP response, Hypertext Transfer Protocol, the content length is 371 bytes.



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.

No, when inspecting the raw data in the packet content window, there are no additional headers within the data displayed in the packet-listing window.

> Frame 92: 553 bytes on wire (4424 bits), 553 bytes captured (4424 bits) on interface 0		0000	d0 60 8c 03 d4 cc f0 2f 74 cc 70 1e 08 00 45 00/ t p : ..E:
> Ethernet II, Src: ASUSTekC_cc:70:1e (f0:2f:74:cc:70:1e), Dst: zte_03:d4:cc (d0:60:8c:03:d4:cc)		0010	02 1b 44 a0 40 00 80 06 7c f8 c0 a8 01 18 80 77	..D:@... w
> Internet Protocol Version 4, Src: 192.168.1.24, Dst: 128.119.245.12		0020	f5 0c d3 53 00 50 81 07 ac 03 cc e1 d6 b1 50 18	...S-P... ..P:
> Transmission Control Protocol, Src Port: 54099, Dst Port: 80, Seq: 1, Ack: 1, Len: 499		0030	02 01 59 10 00 00 47 45 54 20 2f 77 69 72 65 73	..Y...GE T /wires
> Hypertext Transfer Protocol		0040	68 61 72 6b 2d 6c 61 62 73 2f 48 54 54 50 2d 77	hark-lab s/HTTP-w
		0050	69 72 65 73 68 61 72 6b 2d 66 69 6c 65 32 2e 68	inshark -file2.h
		0060	74 6d 6c 20 48 54 54 50 2f 31 2e 31 0d 0a 48 6f	tml HTTP /1.1..Ho
		0070	73 74 3a 20 67 61 69 61 2e 63 73 2e 75 6d 61 73	st: gaia .cs.umas
		0080	73 2e 65 64 75 0d 0a 43 6f 6e 6e 65 63 74 69 6f	s.edu..C connectio
		0090	6e 3a 20 6b 65 65 70 2d 61 6c 69 76 65 0d 0a 44	n: keep- alive..D
		00a0	4e 54 3a 20 31 0d 0a 55 70 67 72 61 64 65 2d 49	NT: 1..U pgrade-I
		00b0	6e 73 65 63 75 72 65 2d 52 65 71 75 65 73 74 73	nsecure- Requests
		00c0	3a 20 31 0d 0a 55 73 65 72 2d 41 67 65 6e 74 3a	: 1...Use r-Agent:
		00d0	20 4d 6f 7a 69 6c 6c 61 2f 35 2e 30 20 28 57 69	Mozilla /5.0 (Wi
		00e0	6e 64 6f 77 73 20 4e 54 20 31 30 2e 30 3b 20 57	ndows NT 10.0; W
		00f0	69 6e 36 34 3b 20 78 36 34 29 20 41 70 70 6c 65	in64; x6 4) Apple
		0100	57 65 62 4b 69 74 2f 35 33 37 2e 33 36 20 28 4b	WebKit/5 37.36 (K
		0110	48 54 4d 4c 2c 29 6c 69 6b 65 20 47 65 63 6b 6f	HTML, li ke Gecko
		0120	29 20 43 68 72 6f 6d 65 2f 31 31 38 2e 30 2e 30) Chrome /118.0.0
		0130	2e 30 20 53 61 66 61 72 69 2f 35 33 37 2e 33 36	.0 Safari 1/537.36
		0140	20 45 64 67 2f 31 31 38 2e 30 2e 32 30 38 38 2e	Edg/118 .0 2088.
		0150	35 37 0d 0a 41 63 63 65 70 74 3a 20 74 65 78 74	57..Accept: text
		0160	2f 68 74 6d 6c 2c 61 70 70 6c 69 63 61 74 69 6f	/html,ap plicatio
		0170	6e 2f 78 68 74 6d 6c 2b 78 6d 6c 2c 61 70 70 6c	n/xhtml+xml,appl
		0180	69 63 61 74 69 6f 6e 2f 78 6d 6c 3b 71 3d 30 2e	ication/ xml;q=0.
		0190	39 2c 69 6d 61 67 65 2f 77 65 62 70 2c 69 6d 61	9,image/ webp,ima
		01a0	67 65 2f 61 70 6e 67 2c 2a 2f 2a 3b 71 3d 30 2e	ge/apng, */*;q=0.
		01b0	38 2c 61 70 70 6c 69 63 61 74 69 6f 6e 2f 73 69	8,applic ation/si
		01c0	67 6e 65 64 2d 65 78 63 68 61 6e 67 65 3b 76 3d	gned-exc hange;v
		01d0	62 33 3b 71 3d 30 2e 37 0d 0a 41 63 63 65 70 74	b3;q=0.7 ..Accept
		01e0	2d 45 6e 63 6f 64 69 6e 67 3a 20 67 7a 69 70 2c	-Encoding: gzip,
		01f0	20 64 65 66 6c 61 74 65 0d 0a 41 63 63 65 70 74	deflate ..Accept
		0200	2d 4c 61 6e 67 75 61 67 65 3a 20 65 6e 2d 47 42	-Language: en-GB
		0210	2c 65 6e 3b 71 3d 30 2e 39 2c 65 6e 2d 55 53 3b	,en;q=0.9,en-US;
		0220	71 3d 30 2e 38 0d 0a 0d 0a	q=0.8...

2.2 The HTTP CONDITIONAL GET/response interaction

Wireshark

File Edit View Go Capture Analyze Statistics Display Windows Tools Help

Time

Source

Destination

Protocol

Length

Info

92	1.886905	192.168.1.24	128.119.245.12	HTTP	553	GET /wirespark-lab/wireless/wireless.html HTTP/1.1
100	1.831780	128.119.245.12	192.168.1.24	HTTP	784	HTTP/1.1 200 OK (text/html)
110	1.866667	192.168.1.24	128.119.245.12	HTTP	499	GET /favicon.ico HTTP/1.1
118	1.893100	128.119.245.12	192.168.1.24	HTTP	530	HTTP/1.1 404 Not Found [text/html]
178	2.406368	192.168.1.24	128.119.245.12	HTTP	605	GET /wirespark-lab/wireless/wireless.html HTTP/1.1
180	2.434294	128.119.245.12	192.168.1.24	HTTP	293	HTTP/1.1 304 Not Modified

> Frame 92: 553 bytes on wire (4424 bits), 553 bytes captured (4424 bits) on interface 0
> Ethernet II, Src: ASUSTekC_cc:70:1e (f0:2f:74:cc:70:1e), Dst: zte_03:d4:cc (d0:60:8c:03:d4:cc)
> Internet Protocol Version 4, Src: 192.168.1.24, Dst: 128.119.245.12
> Transmission Control Protocol, Src Port: 54099, Dst Port: 80, Seq: 1, Ack: 1, Len: 499
> Hypertext Transfer Protocol

0000

d0 60 8c 03 d4 cc f0 2f 74 cc 70 1e 08 00 45 00

...../ t p : ..E:

0010

02 1b 44 a0 40 00 80 06 7c f8 c0 a8 01 18 80 77

..D:@... |.....w

0020

f5 0c d3 53 00 50 81 07 ac 03 cc e1 d6 b1 50 18

...S-P... ..P:

0030

02 01 59 10 00 00 47 45 54 20 2f 77 69 72 65 73

..Y...GE T /wires

0040

68 61 72 6b 2d 6c 61 62 73 2f 48 54 54 50 2d 77

hark-lab s/HTTP-w

0050

69 72 65 73 68 61 72 6b 2d 66 69 6c 65 32 2e 68

inshark -file2.h

0060

74 6d 6c 20 48 54 54 50 2f 31 2e 31 0d 0a 48 6f

tml HTTP /1.1..Ho

0070

73 74 3a 20 67 61 69 61 2e 63 73 2e 75 6d 61 73

st: gaia .cs.umas

0080

73 2e 65 64 75 0d 0a 43 6f 6e 6e 65 63 74 69 6f

s.edu..C connectio

0090

6e 3a 20 6b 65 65 70 2d 61 6c 69 76 65 0d 0a 44

n: keep- alive..D

00a0

4e 54 3a 20 31 0d 0a 55 70 67 72 61 64 65 2d 49

NT: 1..U pgrade-I

00b0

6e 73 65 63 75 72 65 2d 52 65 71 75 65 73 74 73

nsecure- Requests

00c0

3a 20 31 0d 0a 55 73 65 72 2d 41 67 65 6e 74 3a

: 1...Use r-Agent:

00d0

20 4d 6f 7a 69 6c 6c 61 2f 35 2e 30 20 28 57 69

Mozilla /5.0 (Wi

00e0

6e 64 6f 77 73 20 4e 54 20 31 30 2e 30 3b 20 57

ndows NT 10.0; W

00f0

69 6e 36 34 3b 20 78 36 34 29 20 41 70 70 6c 65

in64; x6 4) Apple

0100

57 65 62 4b 69 74 2f 35 33 37 2e 33 36 20 28 4b

WebKit/5 37.36 (K

0110

48 54 4d 4c 2c 29 6c 69 6b 65 20 47 65 63 6b 6f

HTML, li ke Gecko

0120

29 20 43 68 72 6f 6d 65 2f 31 31 38 2e 30 2e 30

) Chrome /118.0.0

0130

2e 30 20 53 61 66 61 72 69 2f 35 33 37 2e 33 36

.0 Safari 1/537.36

0140

20 45 64 67 2f 31 31 38 2e 30 2e 32 30 38 38 2e

Edg/118 .0 2088.

0150

35 37 0d 0a 41 63 63 65 70 74 3a 20 74 65 78 74

57..Accept: text

0160

2f 68 74 6d 6c 2c 61 70 70 6c 69 63 61 74 69 6f

/html,ap plicatio

0170

6e 2f 78 68 74 6d 6c 2b 78 6d 6c 2c 61 70 70 6c

n/xhtml+xml,appl

0180

69 63 61 74 69 6f 6e 2f 78 6d 6c 3b 71 3d 30 2e

ication/ xml;q=0.

0190

39 2c 69 6d 61 67 65 2f 77 65 62 70 2c 69 6d 61

9,image/ webp,ima

01a0

67 65 2f 61 70 6e 67 2c 2a 2f 2a 3b 71 3d 30 2e

ge/apng, */*;q=0.

01b0

38 2c 61 70 70 6c 69 63 61 74 69 6f 6e 2f 73 69

8,applic ation/si

01c0

67 6e 65 64 2d 65 78 63 68 61 6e 67 65 3b 76 3d

gned-exc hange;v

01d0

62 33 3b 71 3d 30 2e 37 0d 0a 41 63 63 65 70 74

b3;q=0.7 ..Accept

01e0

2d 45 6e 63 6f 64 69 6e 67 3a 20 67 7a 69 70 2c

-Encoding: gzip,

01f0

20 64 65 66 6c 61 74 65 0d 0a 41 63 63 65 70 74

deflate ..Accept

0200

2d 4c 61 6e 67 75 61 67 65 3a 20 65 6e 2d 47 42

-Language: en-GB

0210

2c 65 6e 3b 71 3d 30 2e 39 2c 65 6e 2d 55 53 3b

,en;q=0.9,en-US;

0220

71 3d 30 2e 38 0d 0a 0d 0a

q=0.8...

gaia.com/wirespark-lab/wireless.html

Not secure | gaia.com/wirespark-lab/wireless.html

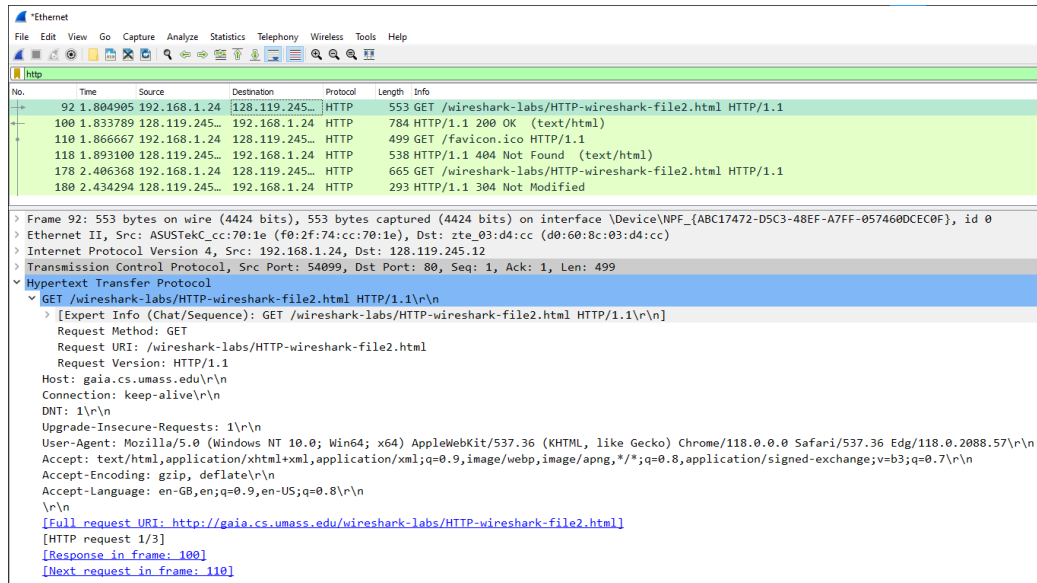
Congratulations again! Now you've downloaded the file lab2-2.html. This file's last modification date will not change.

Thus if you download this multiple times on your browser, a complete copy will only be sent once by the server due to the inclusion of the IN-MODIFIED-SINCE field in your browser's HTTP GET request to the server.

The above image is the baseline output opening gaia website and reloading the website.

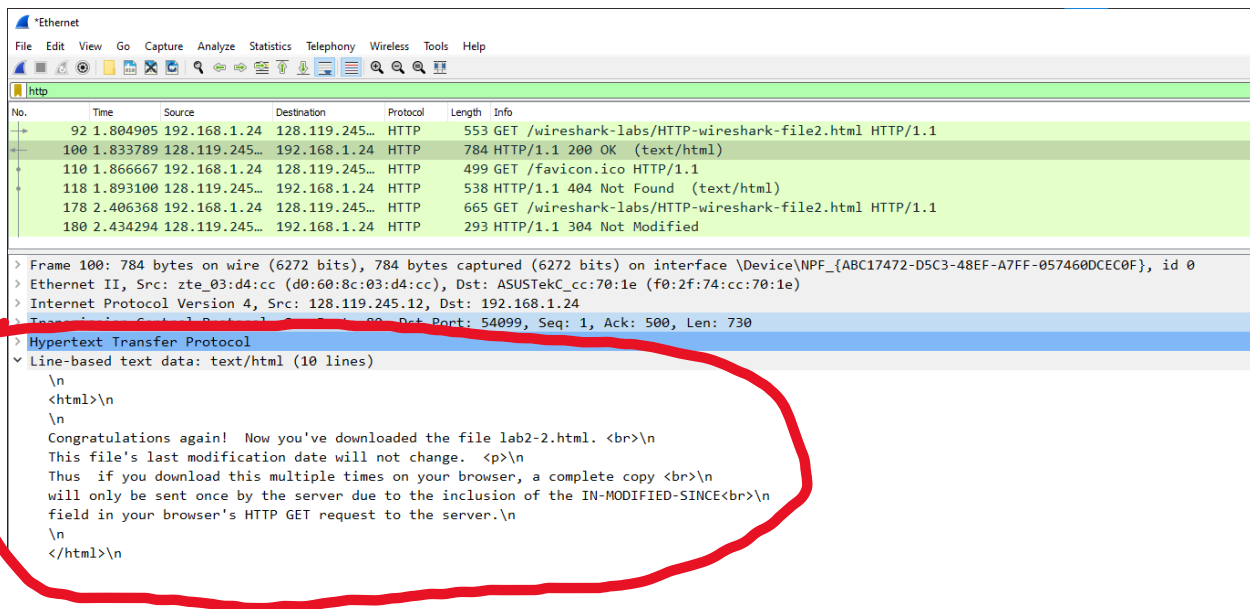
8. Inspect the contents of the first HTTP GET request from your browser to the server. Do you see an “IF-MODIFIED-SINCE” line in the HTTP GET?

From the first HTTP GET, Hypertext Transfer Protocol, it doesn't have any “IF-MODIFIED-SINCE” line.



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

From the first HTTP response, Line-based text data, yes, the server did explicitly return the contents of the file. It is an HTML code.



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?

From the second HTTP GET, Hypertext Transfer Protocol, it has the “IF-MODIFIED-SINCE” line. The information that follows this header is the date last modified, which is Fri, 20 Oct 2023 05:29:02 GMT\r\n.

The image shows a Wireshark packet capture of an HTTP transaction. The packet list on the left shows several packets, with packet 180 (GET /wireshark-labs/HTTP-wireshark-file2.html) selected. The packet details pane on the right shows the structure of the selected packet. The Hypertext Transfer Protocol section is expanded, showing the request line: GET /wireshark-labs/HTTP-wireshark-file2.html HTTP/1.1. Below this, various request headers are listed, including Host, Connection, Cache-Control, DNT, Upgrade-Insecure-Requests, User-Agent, Accept, Accept-Encoding, and Accept-Language. The 'If-Modified-Since' header is circled in red, showing its value: 'Fri, 20 Oct 2023 05:59:02 GMT'. The packet bytes pane at the bottom shows the raw data of the request, including the request line and headers.

```
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
http
No. Time Source Destination Protocol Length Info
92 1.804905 192.168.1.24 128.119.245... HTTP 553 GET /wireshark-labs/HTTP-wireshark-file2.html HTTP/1.1
100 1.833789 128.119.245... 192.168.1.24 HTTP 784 HTTP/1.1 200 OK (text/html)
110 1.866667 192.168.1.24 128.119.245... HTTP 499 GET /favicon.ico HTTP/1.1
118 1.893100 128.119.245... 192.168.1.24 HTTP 538 HTTP/1.1 404 Not Found (text/html)
178 2.406368 192.168.1.24 128.119.245... HTTP 665 GET /wireshark-labs/HTTP-wireshark-file2.html HTTP/1.1
180 2.434294 128.119.245... 192.168.1.24 HTTP 293 HTTP/1.1 304 Not Modified

> Frame 178: 665 bytes on wire (5320 bits), 665 bytes captured (5320 bits) on interface \Device\NPF_{ABC17472-D5C3-48EF-A7FF-057460DCEC0F}, id 0
> Ethernet II, Src: ASUSTekC_cc:70:1e (f0:2f:74:cc:70:1e), Dst: zte_03:d4:cc (d0:60:8c:03:d4:cc)
> Internet Protocol Version 4, Src: 192.168.1.24, Dst: 128.119.245.12
> Transmission Control Protocol, Src Port: 54099, Dst Port: 80, Seq: 945, Ack: 1215, Len: 611
Hypertext Transfer Protocol
  GET /wireshark-labs/HTTP-wireshark-file2.html HTTP/1.1\r\n
  > [Expert Info (Chat/Sequence): GET /wireshark-labs/HTTP-wireshark-file2.html HTTP/1.1\r\n]
    Request Method: GET
    Request URI: /wireshark-labs/HTTP-wireshark-file2.html
    Request Version: HTTP/1.1
    Host: gaia.cs.umass.edu\r\n
    Connection: keep-alive\r\n
    Cache-Control: max-age=0\r\n
    DNT: 1\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/118.0.0.0 Safari/537.36 Edg/118.0.2088.57\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=0.7\r\n
    Accept-Encoding: gzip, deflate\r\n
    Accept-Language: en-GB,en;q=0.9,en-US;q=0.8\r\n
    If-Modified-Since: Fri, 20 Oct 2023 05:59:02 GMT\r\n
  [Full] request URI: http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file2.html
  [HTTP request 3/3]
  [Prev request in frame: 110]
  [Response in frame: 180]
```

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.

From the second HTTP response, Hypertext Transfer Protocol, the status is “304 Not Modified”.

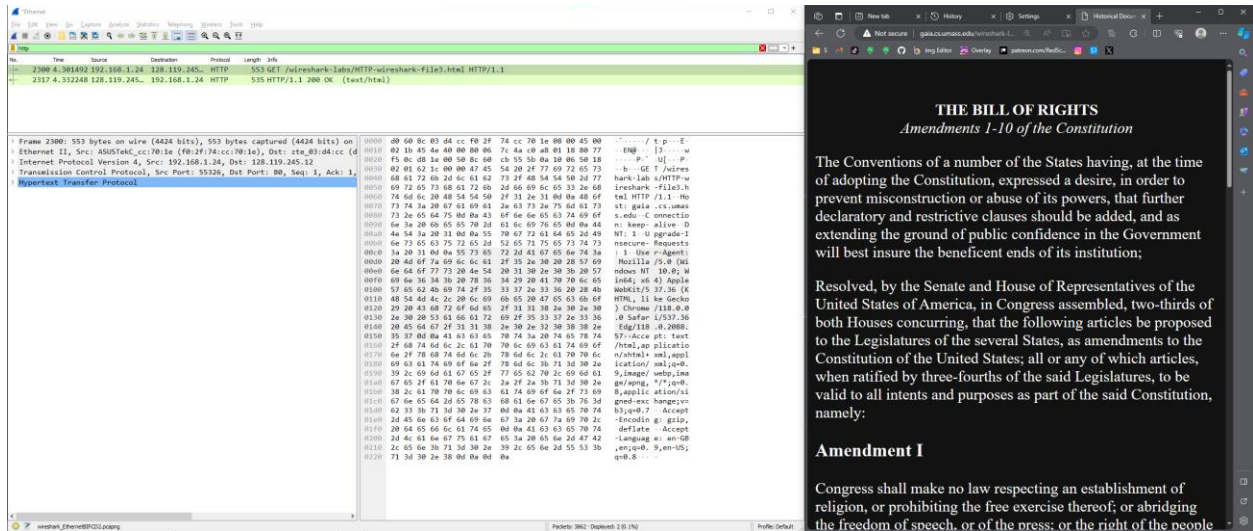
With no modification, no content is explicitly returned from the server (no Line-Based text data).

The image shows a Wireshark packet capture of an HTTP transaction. The packet list on the left shows several packets, with packet 180 (HTTP/1.1 304 Not Modified) selected. The packet details pane on the right shows the structure of the selected packet. The Hypertext Transfer Protocol section is expanded, showing the response line: HTTP/1.1 304 Not Modified. Below this, various response headers are listed, including Response Version, Status Code, Status Code Description, Response Phrase, Date, Server, Connection, Keep-Alive, and ETag. The 'Status Code' and 'Status Code Description' are circled in red, showing their values: '304' and 'Not Modified' respectively. The packet bytes pane at the bottom shows the raw data of the response, including the response line and headers.

```
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
http
No. Time Source Destination Protocol Length Info
92 1.804905 192.168.1.24 128.119.245... HTTP 553 GET /wireshark-labs/HTTP-wireshark-file2.html HTTP/1.1
100 1.833789 128.119.245... 192.168.1.24 HTTP 784 HTTP/1.1 200 OK (text/html)
110 1.866667 192.168.1.24 128.119.245... HTTP 499 GET /favicon.ico HTTP/1.1
118 1.893100 128.119.245... 192.168.1.24 HTTP 538 HTTP/1.1 404 Not Found (text/html)
178 2.406368 192.168.1.24 128.119.245... HTTP 665 GET /wireshark-labs/HTTP-wireshark-file2.html HTTP/1.1
180 2.434294 128.119.245... 192.168.1.24 HTTP 293 HTTP/1.1 304 Not Modified

> Frame 180: 293 bytes on wire (2344 bits), 293 bytes captured (2344 bits) on interface \Device\NPF_{ABC17472-D5C3-48EF-A7FF-057460DCEC0F}, id 0
> Ethernet II, Src: zte_03:d4:cc (d0:60:8c:03:d4:cc), Dst: ASUSTekC_cc:70:1e (f0:2f:74:cc:70:1e)
> Internet Protocol Version 4, Src: 128.119.245.12, Dst: 192.168.1.24
> Transmission Control Protocol, Src Port: 80, Dst Port: 54099, Seq: 1215, Ack: 1556, Len: 239
Hypertext Transfer Protocol
  HTTP/1.1 304 Not Modified\r\n
  > [Expert Info (Chat/Sequence): HTTP/1.1 304 Not Modified\r\n]
    Response Version: HTTP/1.1
    Status Code: 304
    [Status Code Description: Not Modified]
    Response Phrase: Not Modified
    Date: Sat, 21 Oct 2023 02:08:34 GMT\r\n
    Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips PHP/7.4.33 mod_perl/2.0.11 Perl/v5.16.3\r\n
    Connection: Keep-Alive\r\n
    Keep-Alive: timeout=5, max=98\r\n
    ETag: "173-6081f91bbdac7"\r\n
  [HTTP response 3/3]
  [Time since request: 0.027926000 seconds]
  [Prev request in frame: 110]
  [Prev response in frame: 118]
  [Request in frame: 178]
  [Request URI: http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file2.html]
```

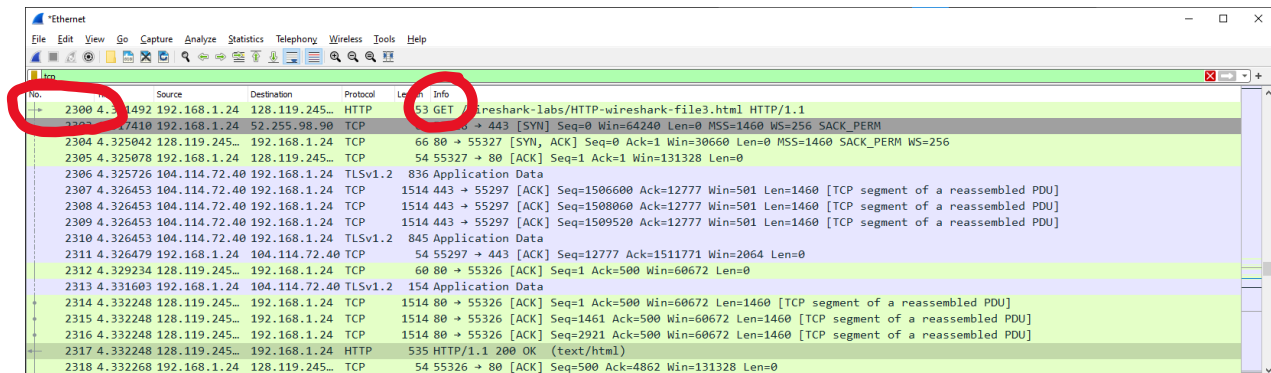
2.3 Retrieving Long Documents



The above image is the baseline output after running gaia website with a long HTML file.

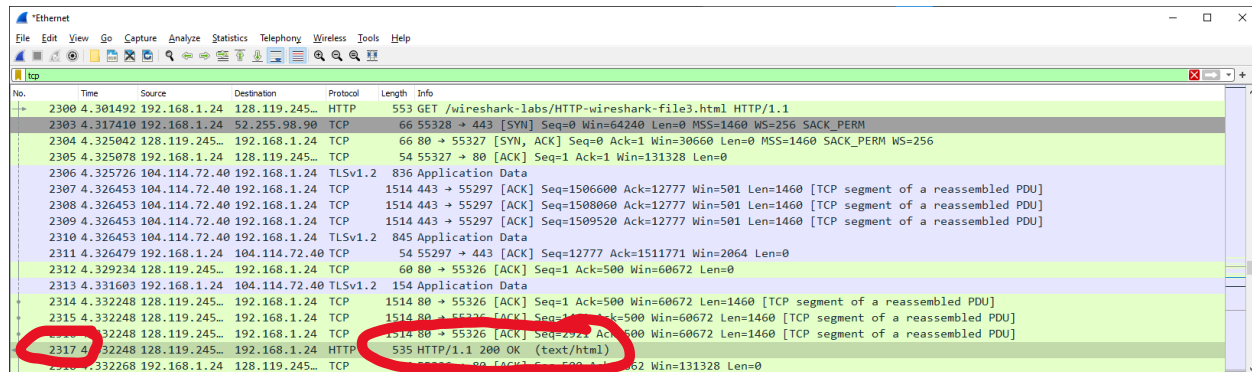
12. How many HTTP GET request messages did your browser send? Which packet number in the trace contains the GET message for the Bill of Rights?

There are only one HTTP GET request. The packet number is No.2300.



13. Which packet number in the trace contains the status code and phrase associated with the response to the HTTP GET request?

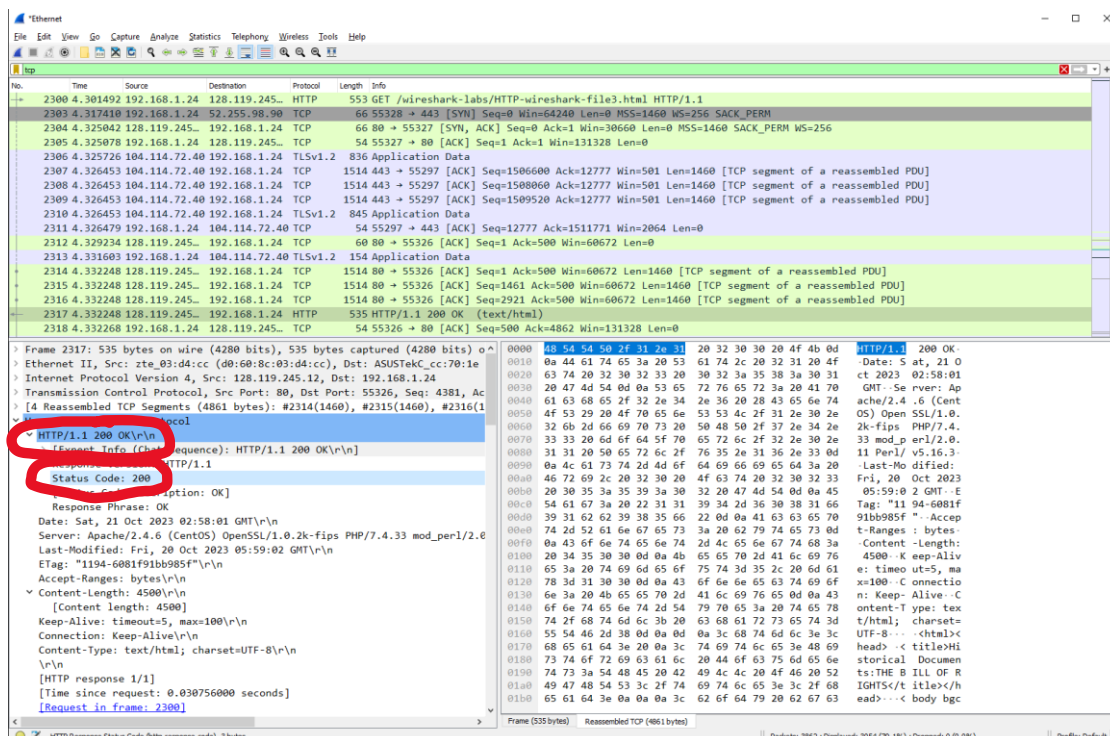
The packet number that contains the status code and phrase is No.2317.



No.	Time	Source	Destination	Protocol	Length	Info
2300	4.301492	192.168.1.24	128.119.245...	HTTP	553	GET /wireshark-labs/HTTP-wireshark-file3.html HTTP/1.1
2303	4.317410	192.168.1.24	52.255.98.90	TCP	66	55328 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
2304	4.325042	128.119.245...	192.168.1.24	TCP	66	80 → 55327 [SYN, ACK] Seq=0 Ack=1 Win=30660 Len=0 MSS=1460 SACK_PERM WS=256
2305	4.325078	192.168.1.24	128.119.245...	TCP	54	55327 → 80 [ACK] Seq=1 Ack=1 Win=131328 Len=0
2306	4.325726	104.114.72.40	192.168.1.24	TLSv1.2	836	Application Data
2307	4.326453	104.114.72.40	192.168.1.24	TCP	1514	443 → 55297 [ACK] Seq=1506600 Ack=12777 Win=501 Len=1460 [TCP segment of a reassembled PDU]
2308	4.326453	104.114.72.40	192.168.1.24	TCP	1514	443 → 55297 [ACK] Seq=1508060 Ack=12777 Win=501 Len=1460 [TCP segment of a reassembled PDU]
2309	4.326453	104.114.72.40	192.168.1.24	TCP	1514	443 → 55297 [ACK] Seq=1509520 Ack=12777 Win=501 Len=1460 [TCP segment of a reassembled PDU]
2310	4.326453	104.114.72.40	192.168.1.24	TLSv1.2	845	Application Data
2311	4.326479	192.168.1.24	104.114.72.40	TCP	54	55297 → 443 [ACK] Seq=12777 Ack=1511771 Win=2064 Len=0
2312	4.329234	128.119.245...	192.168.1.24	TCP	60	80 → 55326 [ACK] Seq=1 Ack=500 Win=60672 Len=0
2313	4.331603	192.168.1.24	104.114.72.40	TLSv1.2	154	Application Data
2314	4.332248	128.119.245...	192.168.1.24	TCP	1514	80 → 55326 [ACK] Seq=1 Ack=500 Win=60672 Len=1460 [TCP segment of a reassembled PDU]
2315	4.332248	128.119.245...	192.168.1.24	TCP	1514	80 → 55326 [ACK] Seq=1 Ack=500 Win=60672 Len=1460 [TCP segment of a reassembled PDU]
2316	4.332248	128.119.245...	192.168.1.24	TCP	1514	80 → 55326 [ACK] Seq=1 Ack=500 Win=60672 Len=1460 [TCP segment of a reassembled PDU]
2317	4.332248	128.119.245...	192.168.1.24	HTTP	535	HTTP/1.1 200 OK (text/html)
2318	4.332268	192.168.1.24	128.119.245...	TCP	54	55326 → 80 [ACK] Seq=500 Ack=4862 Win=131328 Len=0

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

From HTTP response, Hypertext Transfer Protocol, the status code and phrase is 200 OK.

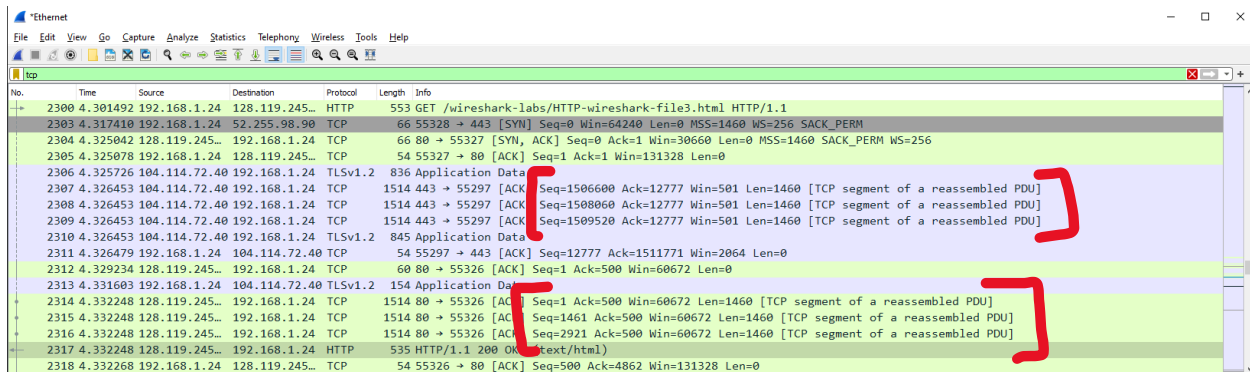


No.	Time	Source	Destination	Protocol	Length	Info
2300	4.301492	192.168.1.24	128.119.245...	HTTP	553	GET /wireshark-labs/HTTP-wireshark-file3.html HTTP/1.1
2303	4.317410	192.168.1.24	52.255.98.90	TCP	66	55328 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
2304	4.325042	128.119.245...	192.168.1.24	TCP	66	80 → 55327 [SYN, ACK] Seq=0 Ack=1 Win=30660 Len=0 MSS=1460 SACK_PERM WS=256
2305	4.325078	192.168.1.24	128.119.245...	TCP	54	55327 → 80 [ACK] Seq=1 Ack=1 Win=131328 Len=0
2306	4.325726	104.114.72.40	192.168.1.24	TLSv1.2	836	Application Data
2307	4.326453	104.114.72.40	192.168.1.24	TCP	1514	443 → 55297 [ACK] Seq=1506600 Ack=12777 Win=501 Len=1460 [TCP segment of a reassembled PDU]
2308	4.326453	104.114.72.40	192.168.1.24	TCP	1514	443 → 55297 [ACK] Seq=1508060 Ack=12777 Win=501 Len=1460 [TCP segment of a reassembled PDU]
2309	4.326453	104.114.72.40	192.168.1.24	TCP	1514	443 → 55297 [ACK] Seq=1509520 Ack=12777 Win=501 Len=1460 [TCP segment of a reassembled PDU]
2310	4.326453	104.114.72.40	192.168.1.24	TLSv1.2	845	Application Data
2311	4.326479	192.168.1.24	104.114.72.40	TCP	54	55297 → 443 [ACK] Seq=12777 Ack=1511771 Win=2064 Len=0
2312	4.329234	128.119.245...	192.168.1.24	TCP	60	80 → 55326 [ACK] Seq=1 Ack=500 Win=60672 Len=0
2313	4.331603	192.168.1.24	104.114.72.40	TLSv1.2	154	Application Data
2314	4.332248	128.119.245...	192.168.1.24	TCP	1514	80 → 55326 [ACK] Seq=1 Ack=500 Win=60672 Len=1460 [TCP segment of a reassembled PDU]
2315	4.332248	128.119.245...	192.168.1.24	TCP	1514	80 → 55326 [ACK] Seq=1 Ack=500 Win=60672 Len=1460 [TCP segment of a reassembled PDU]
2316	4.332248	128.119.245...	192.168.1.24	TCP	1514	80 → 55326 [ACK] Seq=1 Ack=500 Win=60672 Len=1460 [TCP segment of a reassembled PDU]
2317	4.332248	128.119.245...	192.168.1.24	HTTP	535	HTTP/1.1 200 OK (text/html)
2318	4.332268	192.168.1.24	128.119.245...	TCP	54	55326 → 80 [ACK] Seq=500 Ack=4862 Win=131328 Len=0

Frame 2317: 535 bytes on wire (4280 bits), 535 bytes captured (4280 bits) on Ethernet II, Src: zte_03:d4:cc (08:00:0c:03:d4:cc), Dst: ASUSTek.cc:70:1e
Internet Protocol Version 4, Src: 128.119.245.12, Dst: 192.168.1.24
Transmission Control Protocol, Src Port: 80, Dst Port: 55326, Seq: 4381, Ac
4 Reassembled TCP Segments (4861 bytes): #2314(1460), #2315(1460), #2316(1
HTTP Hypertext Transfer Protocol
HTTP/1.1 200 OK
Response Info (Content-Length: 4500) sequence: HTTP/1.1 200 OK
Status Code: 200
Response Phrase: OK
Option: OK
Date: Sat, 21 Oct 2023 02:58:01 GMT
Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips PHP/7.4.33 mod_perl/2.0
Last-Modified: Fri, 20 Oct 2023 05:59:02 GMT
ETag: "1194-6081f91bb985f"
Accept-Ranges: bytes
Content-Length: 4500
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html; charset=UTF-8
[HTTP response 1/1]
[Time since request: 0.030756000 seconds]
[Request in frame: 2300]

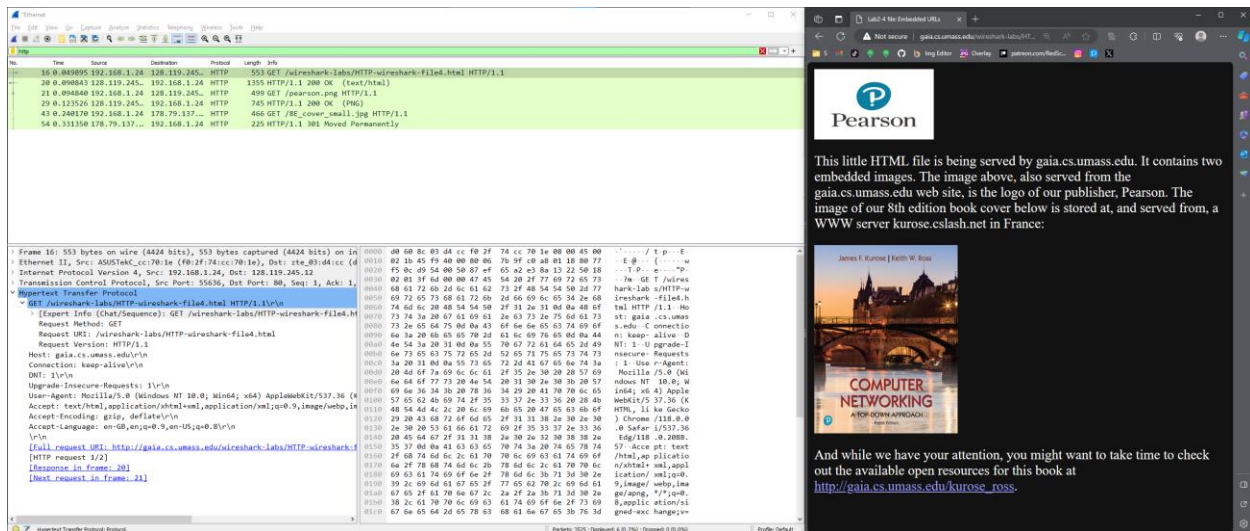
15. How many data-containing TCP segments were needed to carry the single HTTP response and the text of the Bill of Rights?

When filtering for TCP and looking for “TCP segment of a reassembled PDU”, we observe 6 of them. Their number are No.2307, No.2308, No.2309, No.2314, No.2315, No.2316.



No.	Time	Source	Destination	Protocol	Length	Info
2300	4.301492	192.168.1.24	128.119.245...	HTTP	553	GET /wireshark-labs/HTTP-wireshark-file3.html HTTP/1.1
2303	4.317410	192.168.1.24	52.255.98.90	TCP	66	55328 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM
2304	4.325042	128.119.245...	192.168.1.24	TCP	66	80 → 55327 [SYN, ACK] Seq=0 Ack=1 Win=30660 Len=0 MSS=1460 SACK_PERM WS=256
2305	4.325078	192.168.1.24	128.119.245...	TCP	54	55327 → 80 [ACK] Seq=1 Ack=1 Win=131328 Len=0
2306	4.325726	104.114.72.40	192.168.1.24	TLSv1.2	836	Application Data
2307	4.326453	104.114.72.40	192.168.1.24	TCP	1514	443 → 55297 [ACK] Seq=1506600 Ack=12777 Win=501 Len=1460 [TCP segment of a reassembled PDU]
2308	4.326453	104.114.72.40	192.168.1.24	TCP	1514	443 → 55297 [ACK] Seq=1508060 Ack=12777 Win=501 Len=1460 [TCP segment of a reassembled PDU]
2309	4.326453	104.114.72.40	192.168.1.24	TCP	1514	443 → 55297 [ACK] Seq=1509520 Ack=12777 Win=501 Len=1460 [TCP segment of a reassembled PDU]
2310	4.326453	104.114.72.40	192.168.1.24	TLSv1.2	845	Application Data
2311	4.326479	192.168.1.24	104.114.72.40	TCP	54	55297 → 443 [ACK] Seq=12777 Ack=1511771 Win=2064 Len=0
2312	4.329234	128.119.245...	192.168.1.24	TCP	60	80 → 55326 [ACK] Seq=1 Ack=500 Win=60672 Len=0
2313	4.331603	192.168.1.24	104.114.72.40	TLSv1.2	154	Application Data
2314	4.332248	128.119.245...	192.168.1.24	TCP	1514	80 → 55326 [ACK] Seq=1 Ack=500 Win=60672 Len=1460 [TCP segment of a reassembled PDU]
2315	4.332248	128.119.245...	192.168.1.24	TCP	1514	80 → 55326 [ACK] Seq=1461 Ack=500 Win=60672 Len=1460 [TCP segment of a reassembled PDU]
2316	4.332248	128.119.245...	192.168.1.24	TCP	1514	80 → 55326 [ACK] Seq=2921 Ack=500 Win=60672 Len=1460 [TCP segment of a reassembled PDU]
2317	4.332248	128.119.245...	192.168.1.24	HTTP	535	HTTP/1.1 200 OK (text/html)
2318	4.332268	192.168.1.24	128.119.245...	TCP	54	55326 → 80 [ACK] Seq=500 Ack=4862 Win=131328 Len=0

2.4 HTML Documents with Embedded Objects

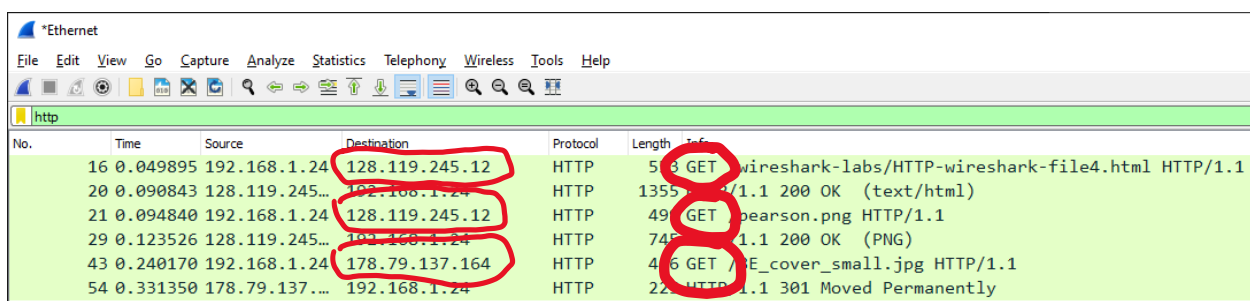


The screenshot displays a Wireshark packet capture of an HTTP GET request to a Pearson website. The packet list shows the request and the response. The packet details pane shows the request structure, including the URL, host, and user agent. The packet bytes pane shows the raw data of the request. On the right, a browser window shows the Pearson website with a book cover for 'Computer Networking' by James F. Kurose and Keith W. Ross.

The above image is the baseline output of opening gaia website with embedded object.

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

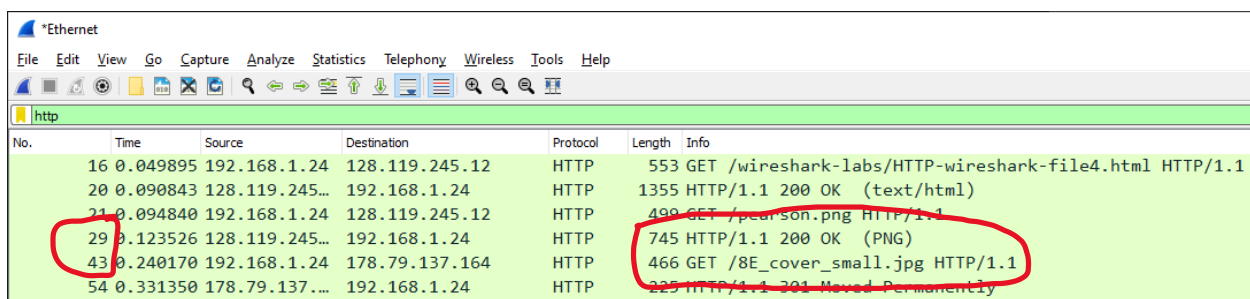
There were three HTTP GET requests. The first one is a GET request from my browser to gaia web server (destination 128.119.245.12). The second is to get “pearson.png” from gaia’s webstie logo publisher, Pearson (destination 128.119.245.12). The third GET request is to get “8E_cover_small.jpg” from Kurose.cslash.net server (destination 178.79.137.164).



No.	Time	Source	Destination	Protocol	Length	Info
16	0.049895	192.168.1.24	128.119.245.12	HTTP	553	GET /wireshark-labs/HTTP-wireshark-file4.html HTTP/1.1
20	0.090843	128.119.245...	192.168.1.24	HTTP	1355	HTTP/1.1 200 OK (text/html)
21	0.094840	192.168.1.24	128.119.245.12	HTTP	490	GET /pearson.png HTTP/1.1
29	0.123526	128.119.245...	192.168.1.24	HTTP	745	HTTP/1.1 200 OK (PNG)
43	0.240170	192.168.1.24	178.79.137.164	HTTP	466	GET /8E_cover_small.jpg HTTP/1.1
54	0.331350	178.79.137...	192.168.1.24	HTTP	225	HTTP/1.1 301 Moved Permanently

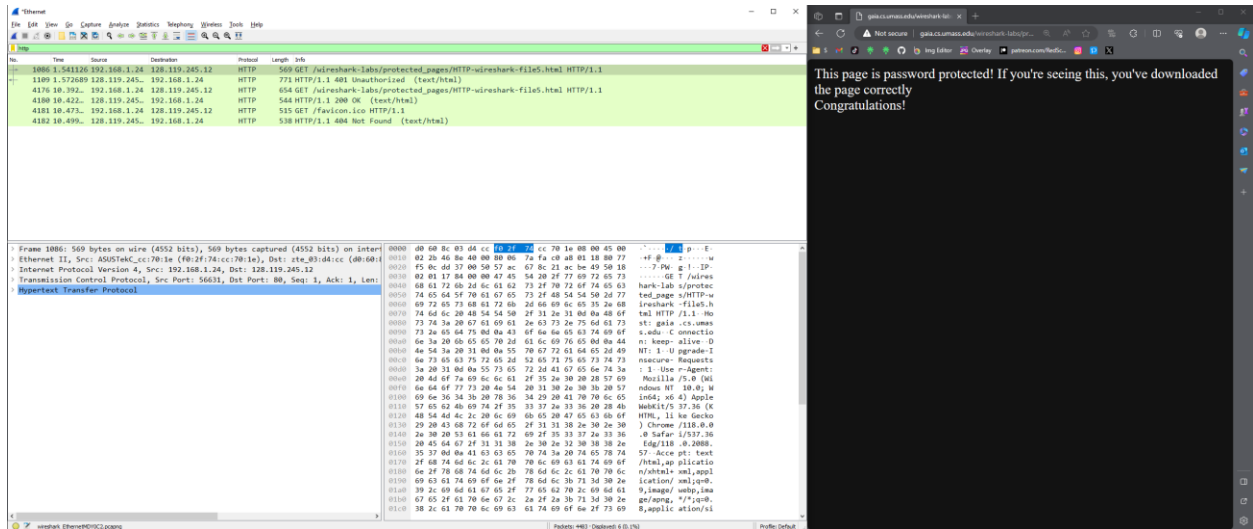
17. Can you tell whether your browser downloaded the two images serially, or whether they were downloaded from the two web sites in parallel? Explain.

The images were downloaded serially, because the first image finished downloading before the third HTTP GET request. Mainly, observe packet 29 “200 OK (PNG)” comes before packet 43 which is the third HTTP GET request.



No.	Time	Source	Destination	Protocol	Length	Info
16	0.049895	192.168.1.24	128.119.245.12	HTTP	553	GET /wireshark-labs/HTTP-wireshark-file4.html HTTP/1.1
20	0.090843	128.119.245...	192.168.1.24	HTTP	1355	HTTP/1.1 200 OK (text/html)
21	0.094840	192.168.1.24	128.119.245.12	HTTP	490	GET /pearson.png HTTP/1.1
29	0.123526	128.119.245...	192.168.1.24	HTTP	745	HTTP/1.1 200 OK (PNG)
43	0.240170	192.168.1.24	178.79.137.164	HTTP	466	GET /8E_cover_small.jpg HTTP/1.1
54	0.331350	178.79.137...	192.168.1.24	HTTP	225	HTTP/1.1 301 Moved Permanently

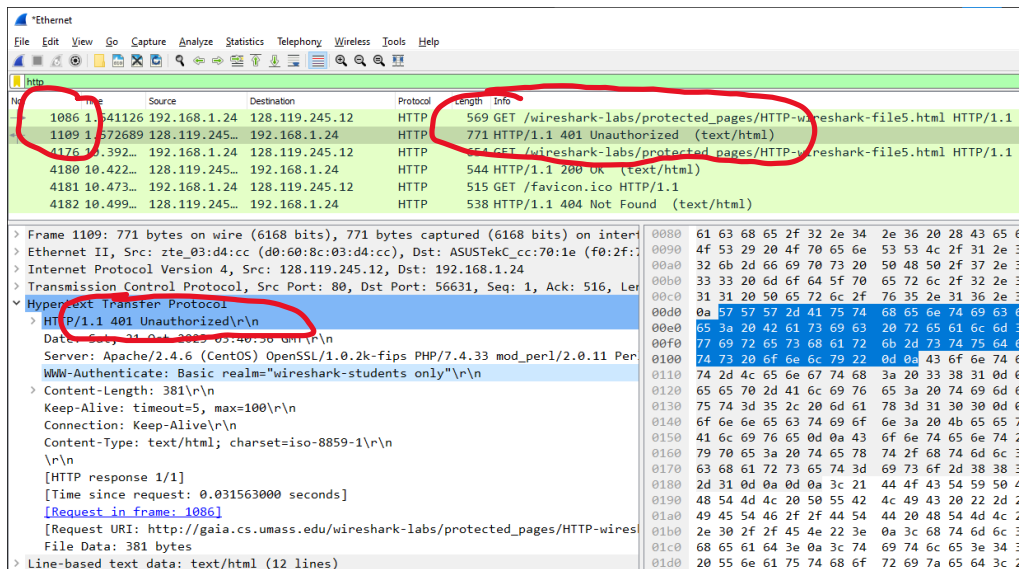
2.5 HTTP Authentication



The above image is the baseline output of opening gaia website with login authentication.

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

Packet 1109 is the response to the HTTP GET request packet 1086. It says "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?

When sending the second HTTP GET requests (packet 4176), “Authorization: Basic” is included with a string of Base64 format that represents the username and password.

The image shows a Wireshark packet capture of an HTTP GET request. The packet list on the left shows several packets, with packet 4176 selected. The packet details pane on the right shows the structure of the selected packet, including the Hypertext Transfer Protocol section. The 'Authorization: Basic' field is highlighted with a red circle, showing the Base64 encoded credentials 'wireshark-students:network'.

No.	Time	Source	Destination	Protocol	Length	Info
1086	1.541126	192.168.1.24	128.119.245.12	HTTP	569	GET /wireshark-labs/protected_pages/HTTP-wireshark-file5.html HTTP/1.1
1109	1.572689	128.119.245...	192.168.1.24	HTTP	771	HTTP/1.1 401 Unauthorized (text/html)
4176	10.392...	192.168.1.24	128.119.245.12	HTTP	654	GET /wireshark-labs/protected_pages/HTTP-wireshark-file5.html HTTP/1.1
4180	10.422...	128.119.245...	192.168.1.24	HTTP	544	HTTP/1.1 200 OK (text/html)
4181	10.473...	192.168.1.24	128.119.245.12	HTTP	515	GET /favicon.ico HTTP/1.1
4182	10.499...	128.119.245...	192.168.1.24	HTTP	538	HTTP/1.1 404 Not Found (text/html)

Frame 4176: 654 bytes on wire (5232 bits), 654 bytes captured (5232 bits) on interface 0
> Ethernet II, Src: ASUSTekC_cc:70:1e (f0:2f:74:cc:70:1e), Dst: zte_03:d4:cc (d0:60:88:00:00:00)
> Internet Protocol Version 4, Src: 192.168.1.24, Dst: 128.119.245.12
> Transmission Control Protocol, Src Port: 56632, Dst Port: 80, Seq: 1, Ack: 1, Len: 654
 Hypertext Transfer Protocol
 > GET /wireshark-labs/protected_pages/HTTP-wireshark-file5.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 Authorization: Basic d2lyZXNoYXJrLXN0dWR1bnRzM5ldHdvcms=\r\n Credentials: wireshark-students:network\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/80.0.4012.101 Safari/537.36\r\n Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/svg+xml\r\n Accept-Encoding: gzip, deflate\r\n Accept-Language: en-GB,en;q=0.9,en-US;q=0.8\r\n [Full request URI: http://gaia.cs.umass.edu/wireshark-labs/protected_pages/HTTP-wireshark-file5.html]\r\n [HTTP request 1/2]\r\n [Response in frame: 4180]\r\n [Next request in frame: 4181]

3. Concepts Learned from this Lab:

Socket programming lab allowed me to understand more about TCP connection and how to create a simple web server. The Wireshark lab provided hands-on experience with HTTP protocol aspects, including GET/response interactions, conditional GET requests, handling long documents, and the use of embedded objects. It also illustrated potential security issues with basic authentication and the need for more secure web access methods.