LAB 1: Introduction to wireshark and analyzing HTTP.

Akshar Panchani ID- 202101522

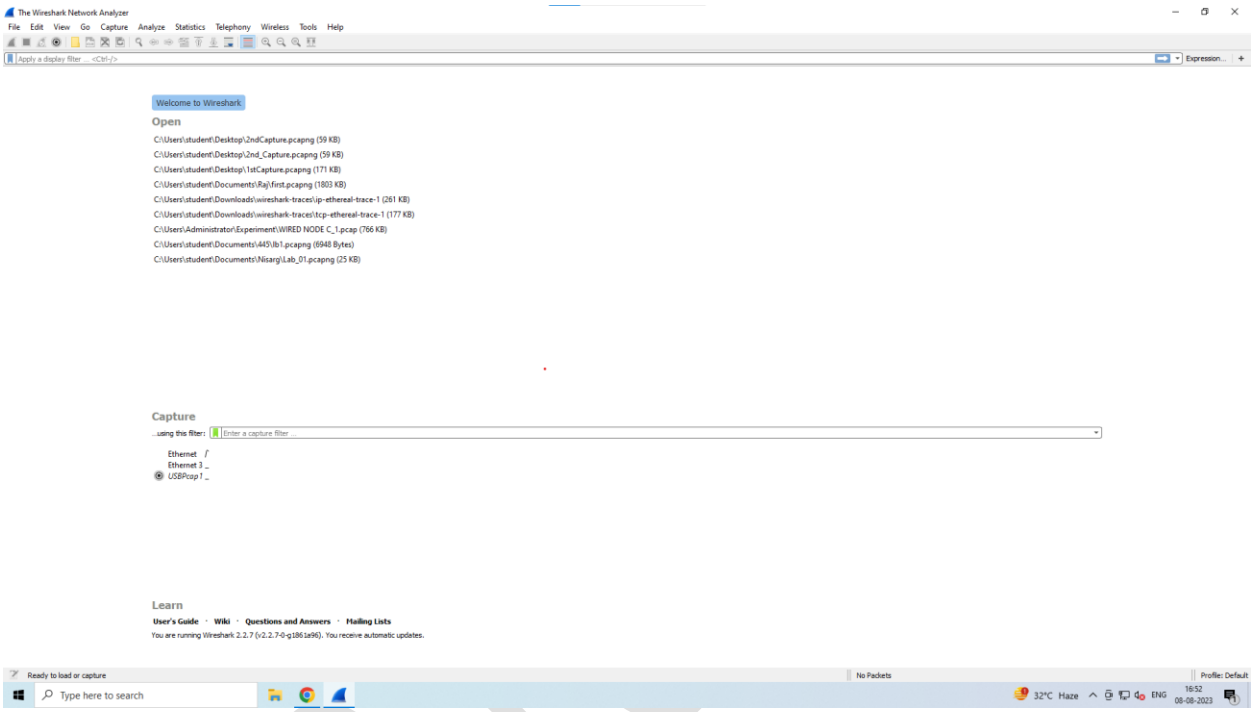
IT304 Computer Networks

8/8/23

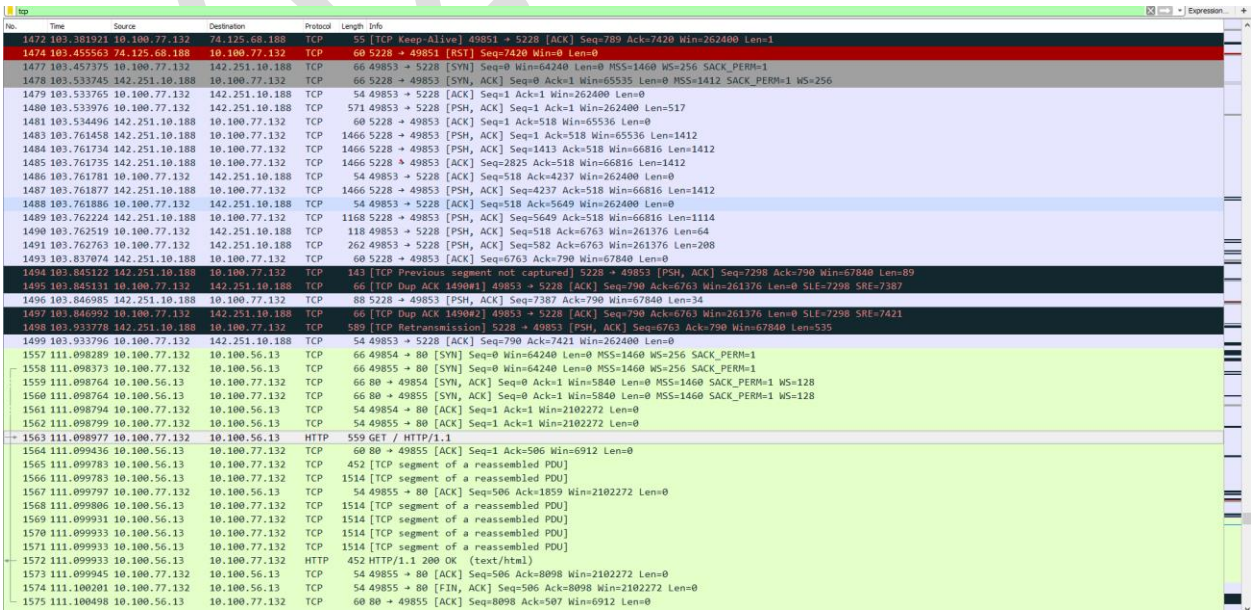
Exercise:

2.9:

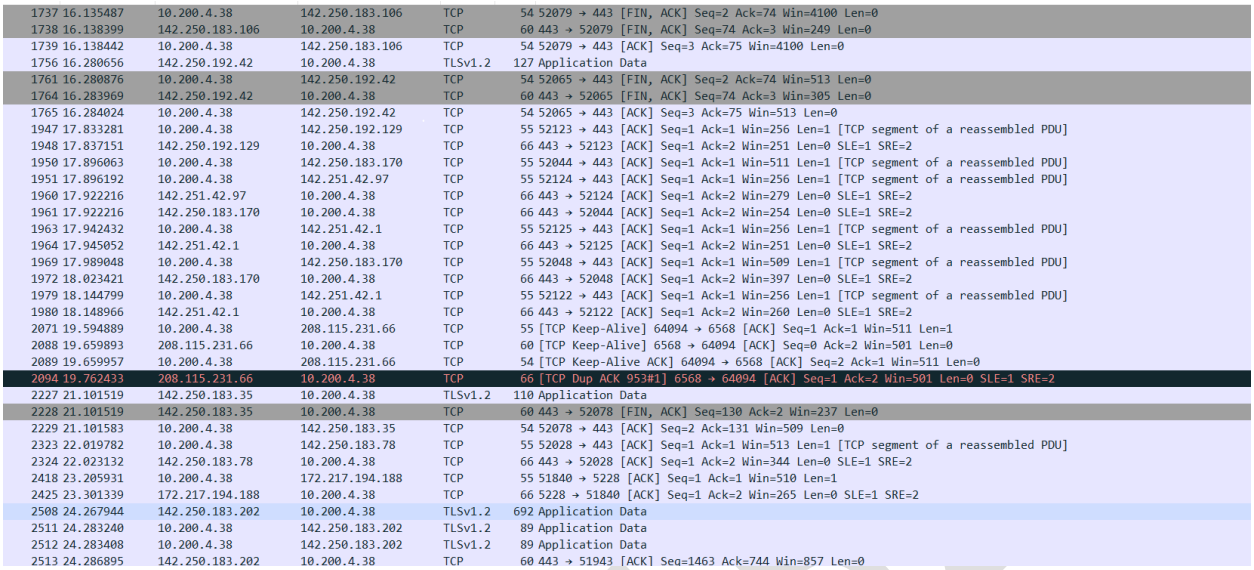
1. what are the interfaces shown on your default screen?



2. Begin packet capturing by selecting an interface that is being used to send and receive packets. AND 3. Filter out all the TCP packets and capture the screen.

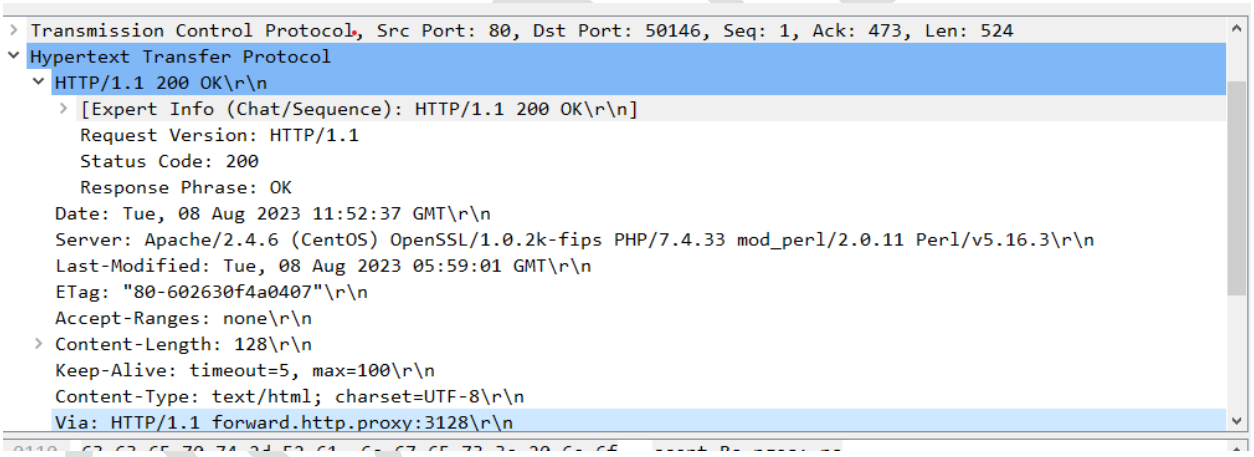


4. Filter out all the TCP data packets which has packet length more than 30 bytes.



1737	16.135487	10.200.4.38	142.250.183.106	TCP	54 52079 → 443 [FIN, ACK] Seq=2 Ack=74 Win=4100 Len=0
1738	16.138399	142.250.183.106	10.200.4.38	TCP	60 443 → 52079 [FIN, ACK] Seq=74 Ack=3 Win=249 Len=0
1739	16.138442	10.200.4.38	142.250.183.106	TCP	54 52079 → 443 [ACK] Seq=3 Ack=75 Win=4100 Len=0
1756	16.280656	142.250.192.42	10.200.4.38	TLSv1.2	127 Application Data
1761	16.280876	10.200.4.38	142.250.192.42	TCP	54 52065 → 443 [FIN, ACK] Seq=2 Ack=74 Win=513 Len=0
1764	16.283969	142.250.192.42	10.200.4.38	TCP	60 443 → 52065 [FIN, ACK] Seq=74 Ack=3 Win=305 Len=0
1765	16.284024	10.200.4.38	142.250.192.42	TCP	54 52065 → 443 [ACK] Seq=3 Ack=75 Win=513 Len=0
1947	17.833281	10.200.4.38	142.250.192.129	TCP	55 52123 → 443 [ACK] Seq=1 Ack=1 Win=256 Len=1 [TCP segment of a reassembled PDU]
1948	17.837151	142.250.192.129	10.200.4.38	TCP	66 443 → 52123 [ACK] Seq=1 Ack=2 Win=251 Len=0 SLE=1 SRE=2
1950	17.896063	10.200.4.38	142.250.183.170	TCP	55 52044 → 443 [ACK] Seq=1 Ack=1 Win=511 Len=1 [TCP segment of a reassembled PDU]
1951	17.896192	10.200.4.38	142.251.42.97	TCP	55 52124 → 443 [ACK] Seq=1 Ack=1 Win=256 Len=1 [TCP segment of a reassembled PDU]
1960	17.922216	142.251.42.97	10.200.4.38	TCP	66 443 → 52124 [ACK] Seq=1 Ack=2 Win=279 Len=0 SLE=1 SRE=2
1961	17.922216	142.250.183.170	10.200.4.38	TCP	66 443 → 52044 [ACK] Seq=1 Ack=2 Win=254 Len=0 SLE=1 SRE=2
1963	17.942432	10.200.4.38	142.251.42.1	TCP	55 52125 → 443 [ACK] Seq=1 Ack=1 Win=256 Len=1 [TCP segment of a reassembled PDU]
1964	17.945052	142.251.42.1	10.200.4.38	TCP	66 443 → 52125 [ACK] Seq=1 Ack=2 Win=251 Len=0 SLE=1 SRE=2
1969	17.989048	10.200.4.38	142.250.183.170	TCP	55 52048 → 443 [ACK] Seq=1 Ack=1 Win=509 Len=1 [TCP segment of a reassembled PDU]
1972	18.023421	142.250.183.170	10.200.4.38	TCP	66 443 → 52048 [ACK] Seq=1 Ack=2 Win=397 Len=0 SLE=1 SRE=2
1979	18.144799	10.200.4.38	142.251.42.1	TCP	55 52122 → 443 [ACK] Seq=1 Ack=1 Win=256 Len=1 [TCP segment of a reassembled PDU]
1980	18.148966	142.251.42.1	10.200.4.38	TCP	66 443 → 52122 [ACK] Seq=1 Ack=2 Win=260 Len=0 SLE=1 SRE=2
2071	19.594889	10.200.4.38	208.115.231.66	TCP	55 [TCP Keep-Alive] 64094 → 6568 [ACK] Seq=1 Ack=1 Win=511 Len=1
2088	19.659893	208.115.231.66	10.200.4.38	TCP	60 [TCP Keep-Alive] 6568 → 64094 [ACK] Seq=0 Ack=2 Win=501 Len=0
2089	19.659957	10.200.4.38	208.115.231.66	TCP	54 [TCP Keep-Alive ACK] 64094 → 6568 [ACK] Seq=2 Ack=1 Win=511 Len=0
2094	19.762433	208.115.231.66	10.200.4.38	TCP	66 [TCP Dup ACK 953#1] 6568 → 64094 [ACK] Seq=1 Ack=2 Win=501 Len=0 SLE=1 SRE=2
2227	21.101519	142.250.183.35	10.200.4.38	TLSv1.2	110 Application Data
2228	21.101519	142.250.183.35	10.200.4.38	TCP	60 443 → 52078 [FIN, ACK] Seq=130 Ack=2 Win=237 Len=0
2229	21.101583	10.200.4.38	142.250.183.35	TCP	54 52078 → 443 [ACK] Seq=2 Ack=131 Win=509 Len=0
2323	22.019782	10.200.4.38	142.250.183.78	TCP	55 52028 → 443 [ACK] Seq=1 Ack=1 Win=513 Len=1 [TCP segment of a reassembled PDU]
2324	22.023132	142.250.183.78	10.200.4.38	TCP	66 443 → 52028 [ACK] Seq=1 Ack=2 Win=344 Len=0 SLE=1 SRE=2
2418	23.205931	10.200.4.38	172.217.194.188	TCP	55 51840 → 5228 [ACK] Seq=1 Ack=1 Win=510 Len=1
2425	23.301339	172.217.194.188	10.200.4.38	TCP	66 5228 → 51840 [ACK] Seq=1 Ack=2 Win=265 Len=0 SLE=1 SRE=2
2508	24.267944	142.250.183.202	10.200.4.38	TLSv1.2	692 Application Data
2511	24.283240	10.200.4.38	142.250.183.202	TLSv1.2	89 Application Data
2512	24.283408	10.200.4.38	142.250.183.202	TLSv1.2	89 Application Data
2513	24.286895	142.250.183.202	10.200.4.38	TCP	60 443 → 51943 [ACK] Seq=1463 Ack=744 Win=857 Len=0

4.2:



```
> Transmission Control Protocol, Src Port: 80, Dst Port: 50146, Seq: 1, Ack: 473, Len: 524
v Hypertext Transfer Protocol
  v HTTP/1.1 200 OK\r\n
    > [Expert Info (Chat/Sequence): HTTP/1.1 200 OK\r\n]
      Request Version: HTTP/1.1
      Status Code: 200
      Response Phrase: OK
    Date: Tue, 08 Aug 2023 11:52:37 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
    Last-Modified: Tue, 08 Aug 2023 05:59:01 GMT\r\n
    ETag: "80-602630f4a0407"\r\n
    Accept-Ranges: none\r\n
  > Content-Length: 128\r\n
    Keep-Alive: timeout=5, max=100\r\n
    Content-Type: text/html; charset=UTF-8\r\n
    Via: HTTP/1.1 forward.http.proxy:3128\r\n
```

From the above screenshot we extract this answers.

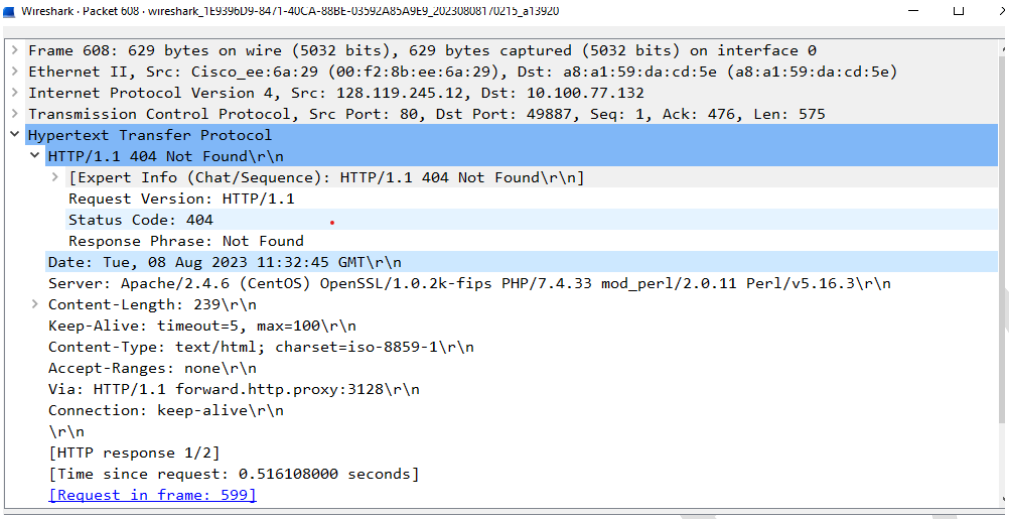
1. Is your browser running HTTP version 1.0 or 1.1? What version of HTTP is the server running?

HTTP Version 1.1 is the version for both server and browser.

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

Language shown is :- en-US, en q=0.9

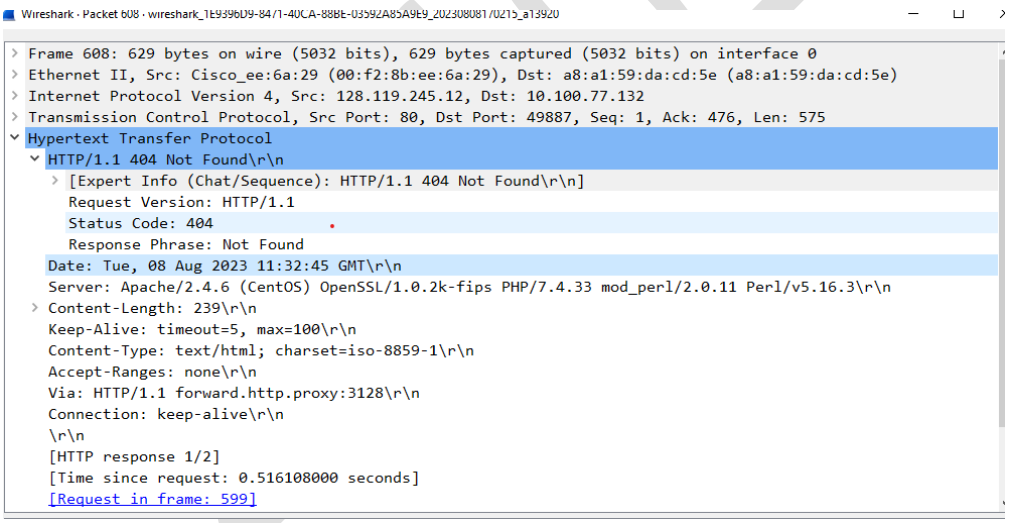
3. What is the IP address of your computer? and the gaia.cs.umass.edu server?



IP address of Computer (Src) 128.119.245.12 and the given server (dst)

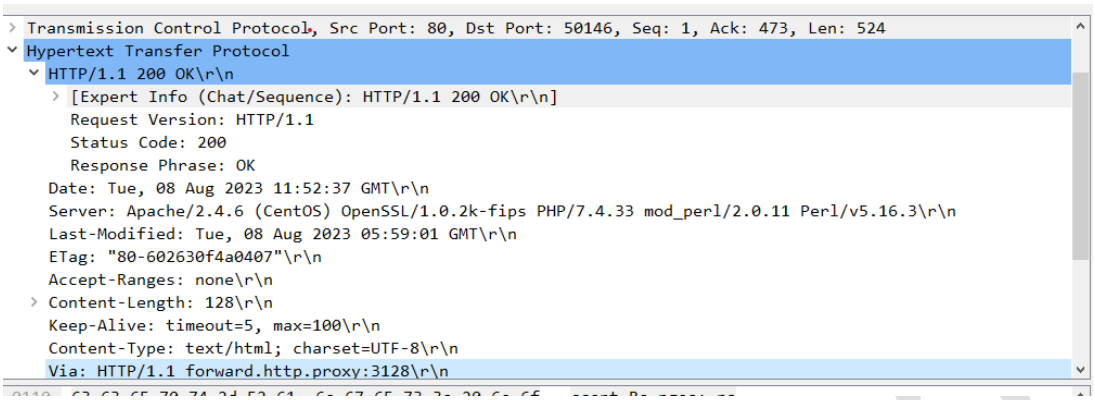
10.100.77.132

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



404 Not Found as it was reloaded twice.

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



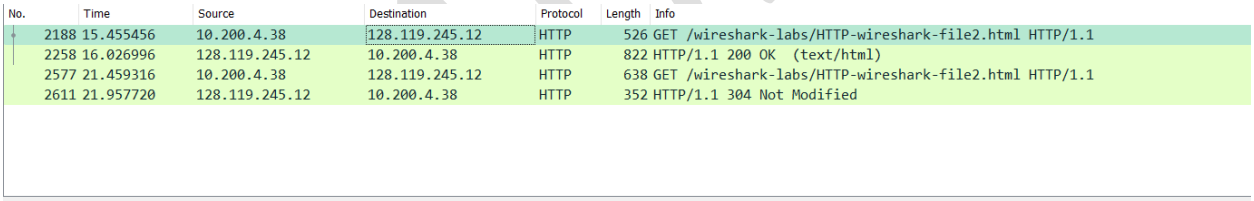
6. How many bytes of content are being returned to your browser?

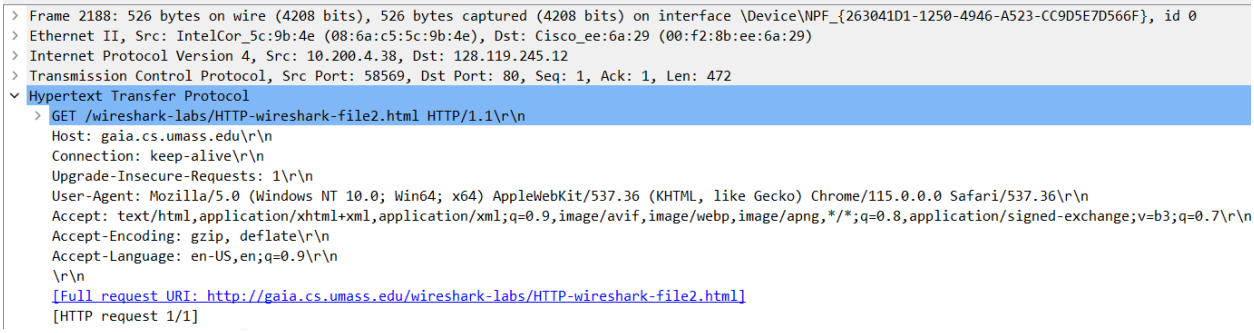
It is 128 Bytes.

5.2:

1. 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?

No it is not shown.





Yes as mentioned above.

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

Yes as it shows the content length with the line based content as shown

```
Hypertext Transfer Protocol
  HTTP/1.1 200 OK\r\n
  Date: Tue, 08 Aug 2023 13:28:40 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
  Last-Modified: Tue, 08 Aug 2023 05:59:01 GMT\r\n
  ETag: "173-602630f49fc37"\r\n
  Accept-Ranges: none\r\n
  Content-Length: 371\r\n
  Keep-Alive: timeout=5, max=100\r\n
  Content-Type: text/html; charset=UTF-8\r\n
  Via: HTTP/1.1 forward.http.proxy:3128\r\n
  Connection: keep-alive\r\n
  \r\n
  [HTTP response 1/1]
  [Time since request: 0.501677000 seconds]
  [Request in frame: 329]
  [Request URI: http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file2.html]
  File Data: 371 bytes
Line-based text data: text/html (10 lines)
  \n
  <html>\n
  \n
  Congratulations again!  Now you've downloaded the file lab2-2.html. <br>\n
  This file's last modification date will not change.  <p>\n
  Thus  if you download this multiple times on your browser, a complete copy <br>\n
  will only be sent once by the server due to the inclusion of the IN-MODIFIED-SINCE<br>\n
  field in your browser's HTTP GET request to the server.\n
  \n
  </html>\n
```

3. 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?

Yes it is shown as below.

```
448 4.825744    10.200.4.38     128.119.245.12  HTTP    638 GET /wireshark-labs/HTTP-wireshark-file2.html HTTP/1.1
516 5.329295    128.119.245.12  10.200.4.38     HTTP    352 HTTP/1.1 304 Not Modified

Frame 448: 638 bytes on wire (5104 bits), 638 bytes captured (5104 bits) on interface \Device\NPF_{263041D1-1250-4946-A523-CC9D5E7D566F}, id 0
Ethernet II, Src: IntelCor_5c:9b:4e (08:6a:c5:5c:9b:4e), Dst: Cisco_ee:6a:29 (00:f2:8b:ee:6a:29)
Internet Protocol Version 4, Src: 10.200.4.38, Dst: 128.119.245.12
Transmission Control Protocol, Src Port: 55617, Dst Port: 80, Seq: 1, Ack: 1, Len: 584
Hypertext Transfer Protocol
  GET /wireshark-labs/HTTP-wireshark-file2.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 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/115.0.0.0 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=0.7\r\n
  Accept-Encoding: gzip, deflate\r\n
  Accept-Language: en-US,en;q=0.9\r\n
  If-None-Match: "173-602630f49fc37"\r\n
  If-Modified-Since: Tue, 08 Aug 2023 05:59:01 GMT\r\n
```

4. 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.

It is 304 Not modified and this time the line based data is not shown as it is the same.

2577	21.459316	10.200.4.38	128.119.245.12	HTTP	638	GET /wireshark-labs/HTTP-wireshark-file2.html HTTP/1.1
2611	21.957720	128.119.245.12	10.200.4.38	HTTP	352	HTTP/1.1 304 Not Modified

```
> Frame 2611: 352 bytes on wire (2816 bits), 352 bytes captured (2816 bits) on interface \Device\NPF_{263041D1-1250-4946-A523-CC9D5E7D566F}, id 0
> Ethernet II, Src: Cisco_ee:6a:29 (00:f2:8b:ee:6a:29), Dst: IntelCor_5c:9b:4e (08:6a:c5:5c:9b:4e)
> Internet Protocol Version 4, Src: 128.119.245.12, Dst: 10.200.4.38
> Transmission Control Protocol, Src Port: 80, Dst Port: 58570, Seq: 1, Ack: 585, Len: 298
v Hypertext Transfer Protocol
  > HTTP/1.1 304 Not Modified\r\n
    Date: Tue, 08 Aug 2023 13:34:54 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
    Keep-Alive: timeout=5, max=100\r\n
    ETag: "173-602630f49fc37"\r\n
  > Content-Length: 0\r\n
    Via: HTTP/1.1 forward.http.proxy:3128\r\n
    Connection: keep-alive\r\n
    \r\n
    [HTTP response 1/1]
    [Time since request: 0.498404000 seconds]
    [Request in frame: 2577]
    [Request URI: http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file2.html]
  [Community ID: 1:/u6rQd17PpgC5l8AIv/GV903LXs=]
```

6.2:

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

Only 1 GET request was send by the browser.

No.	Time	Source	Destination	Protocol	Length	Info
402	3.636033	10.200.4.38	128.119.245.12	HTTP	526	GET /wireshark-labs/HTTP-wireshark-file3.html HTTP/1.1
561	4.150820	128.119.245.12	10.200.4.38	HTTP	573	HTTP/1.1 200 OK (text/html)
954	8.677286	10.200.4.38	128.119.245.12	HTTP	639	GET /wireshark-labs/HTTP-wireshark-file3.html HTTP/1.1
989	8.957826	128.119.245.12	10.200.4.38	HTTP	352	HTTP/1.1 304 Not Modified

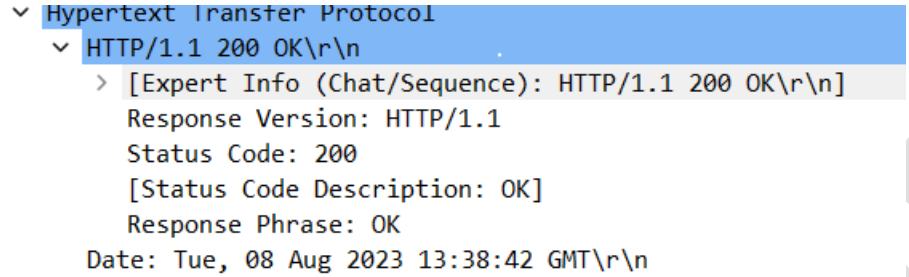
2. How many data-containing TCP segments were needed to carry the single HTTP response?

There were 4.

```
[4 Reassembled TCP Segments (4899 bytes): #558(1460), #559(1460), #560(1460), #561(519)]
   [Frame: 558, payload: 0-1459 (1460 bytes)]
   [Frame: 559, payload: 1460-2919 (1460 bytes)]
   [Frame: 560, payload: 2920-4379 (1460 bytes)]
   [Frame: 561, payload: 4380-4898 (519 bytes)]
   [Segment count: 4]
   [Reassembled TCP length: 4899]
   [Reassembled TCP Data: 485454502f312e3120323030204f4b0d0a446174653a205475652c203038204175672032…]
```

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

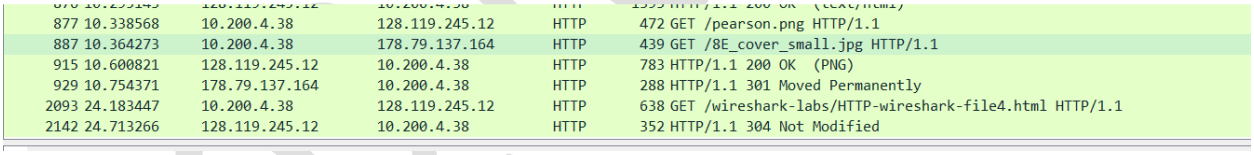
Status code is 200 and phrase is OK



7.2:

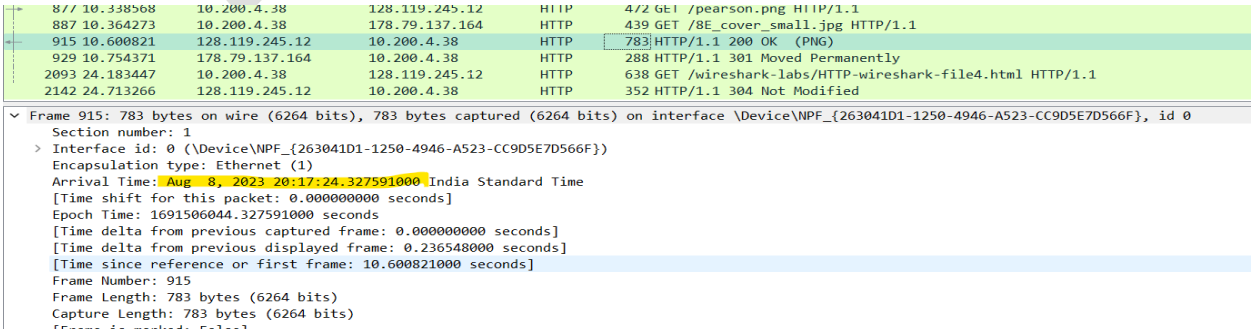
1. How many HTTP GET request messages were sent by your browser? To which Internet addresses were these GET requests sent?

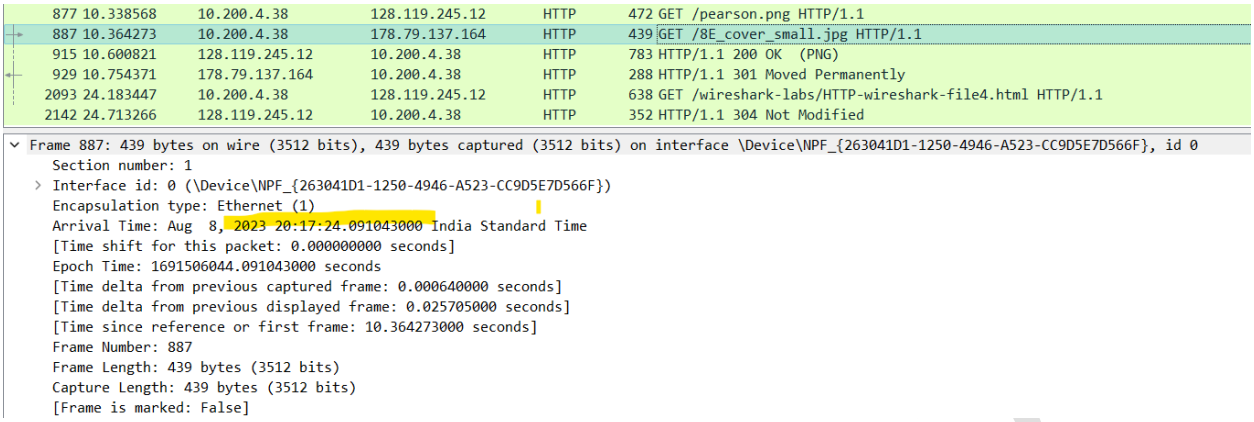
There were 3 GET request , 2 for the images (Pearson and BE cover small) and 1 webpage



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

It will be in series as both the timestamp are different from image so this is justified.

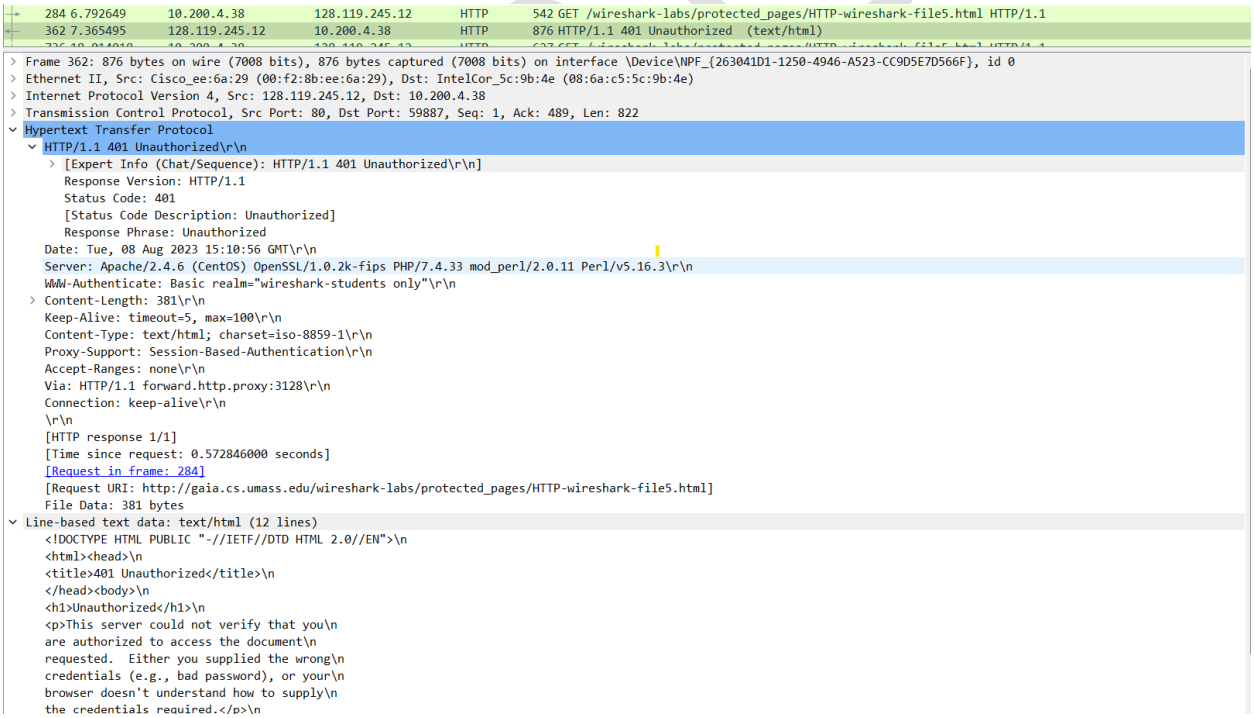




8.2:

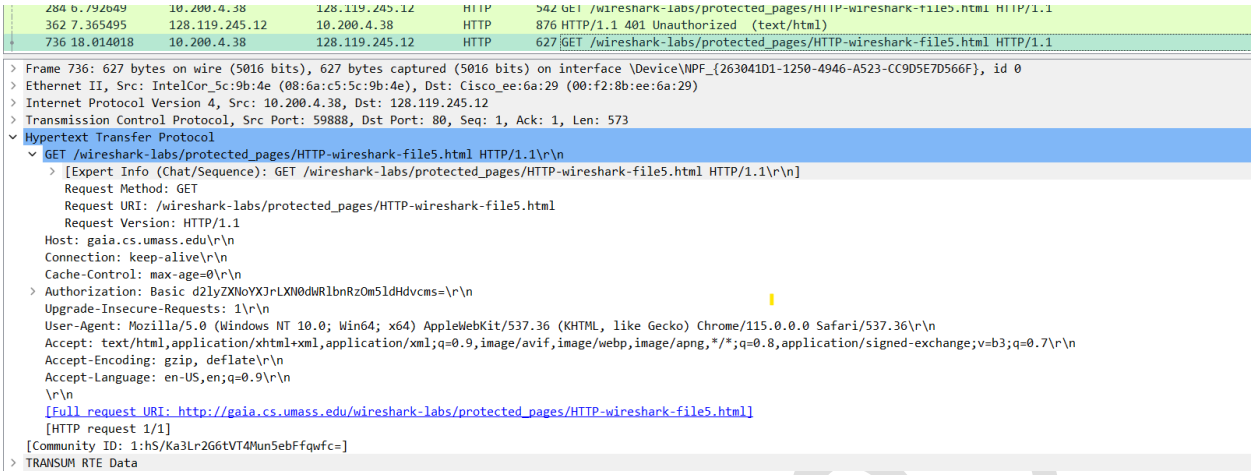
1. What is the servers response (status code and phrase) in response to the initial HTTP GET message from your browser?

It showed the unauthorized 401 error.



2. When your browsers sends the HTTP GET message for the second time, what new field is included in the HTTP GET message?

The new message of Authorization is included in the field and this time it accepts the request and leads to the webpage with 200 status code and OK phrase.



```
284 6.792649    10.200.4.38     128.119.245.12  HTTP    542 GET /wireshark-labs/protected_pages/HTTP-wireshark-file5.html HTTP/1.1
362 7.365495    128.119.245.12  10.200.4.38     HTTP    876 HTTP/1.1 401 Unauthorized  (text/html)
736 18.014018   10.200.4.38     128.119.245.12  HTTP    627 GET /wireshark-labs/protected_pages/HTTP-wireshark-file5.html HTTP/1.1

> Frame 736: 627 bytes on wire (5016 bits), 627 bytes captured (5016 bits) on interface \Device\NPF_{263041D1-1250-4946-A523-CC9D5E7D566F}, id 0
> Ethernet II, Src: IntelCor_5c:9b:4e (08:6a:c5:5c:9b:4e), Dst: Cisco_ee:6a:29 (00:f2:8b:ee:6a:29)
> Internet Protocol Version 4, Src: 10.200.4.38, Dst: 128.119.245.12
> Transmission Control Protocol, Src Port: 59888, Dst Port: 80, Seq: 1, Ack: 1, Len: 573
v Hypertext Transfer Protocol
  v GET /wireshark-labs/protected_pages/HTTP-wireshark-file5.html HTTP/1.1\r\n
    > [Expert Info (Chat/Sequence): GET /wireshark-labs/protected_pages/HTTP-wireshark-file5.html HTTP/1.1\r\n]
      Request Method: GET
      Request URI: /wireshark-labs/protected_pages/HTTP-wireshark-file5.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
    > Authorization: Basic d2lyZXNoYXJrLXN0dWRlbnRzOm5ldHdvcms=\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/115.0.0.0 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=0.7\r\n
    Accept-Encoding: gzip, deflate\r\n
    Accept-Language: en-US,en;q=0.9\r\n
    \r\n
    [Full request URI: http://gaia.cs.umass.edu/wireshark-labs/protected_pages/HTTP-wireshark-file5.html]
    [HTTP request 1/1]
  [Community ID: 1:hS/Ka3Lr2G6tVT4Mun5ebFfqwfc=]
> TRANSUM RTE Data
```