REPORT

과 목 명 : 네트워크 응용 설계

담당교수 : 백 정 엽 교수님

제 출 일 : 2019. 04. 05

전 공 명 : 소프트웨어전공

학 번 : 20165974

이 름 : 최 정 민

중앙대학교

[A]. The Basic HTTP GET/response interaction

Recall that since the HTTP message was carried inside a TCP segment, which was carried inside an IP datagram, which was carried within an Ethernet frame, Wireshark displays the Frame, Ethernet, IP, and TCP packet information as well.

Please focus on the HTTP message only for now, but you may also need to look at other packet information, and you're more than welcome to look at other protocol fields for your own interest (and study).

By looking at the information in the HTTP GET and response messages, answer the following questions. When answering the following questions, you should print out the GET and response messages and indicate where in the message you've found the information that answers the following questions.

When you hand in your assignment, annotate the output so that it's clear where in the output you're getting the information for your answer (e.g., for our classes, we ask that students markup paper copies with a pen, or annotate electronic copies with text in a colored font).

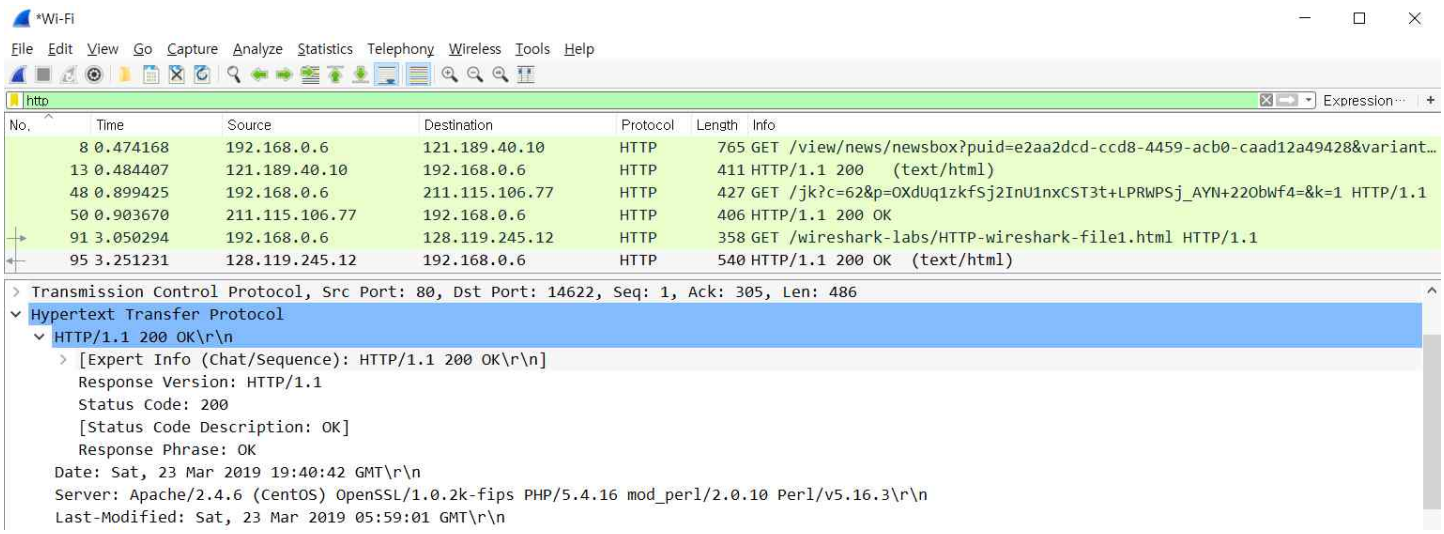
Answer the following questions:

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

Answer :

My browser is running http version 1.1

The server is running http version 1.1

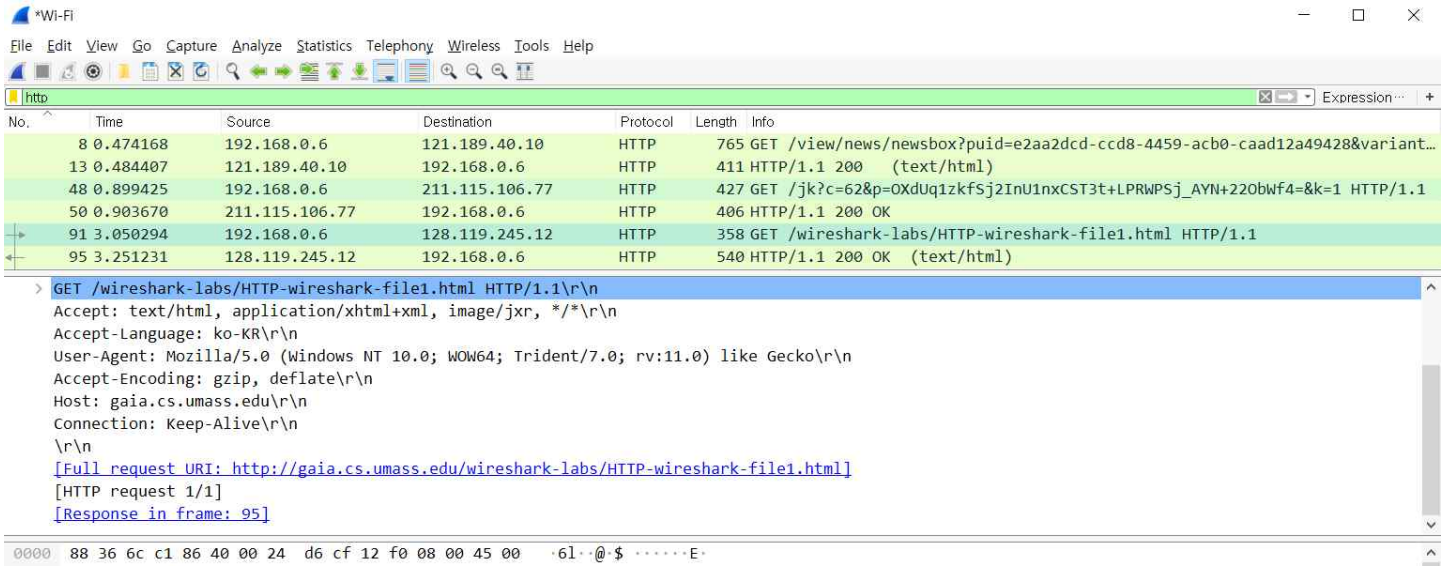


2. What is the IP address of your computer? Of the gaia.cs.umass.edu server?

Answer：

The IP address of my computter is 192.168.0.6

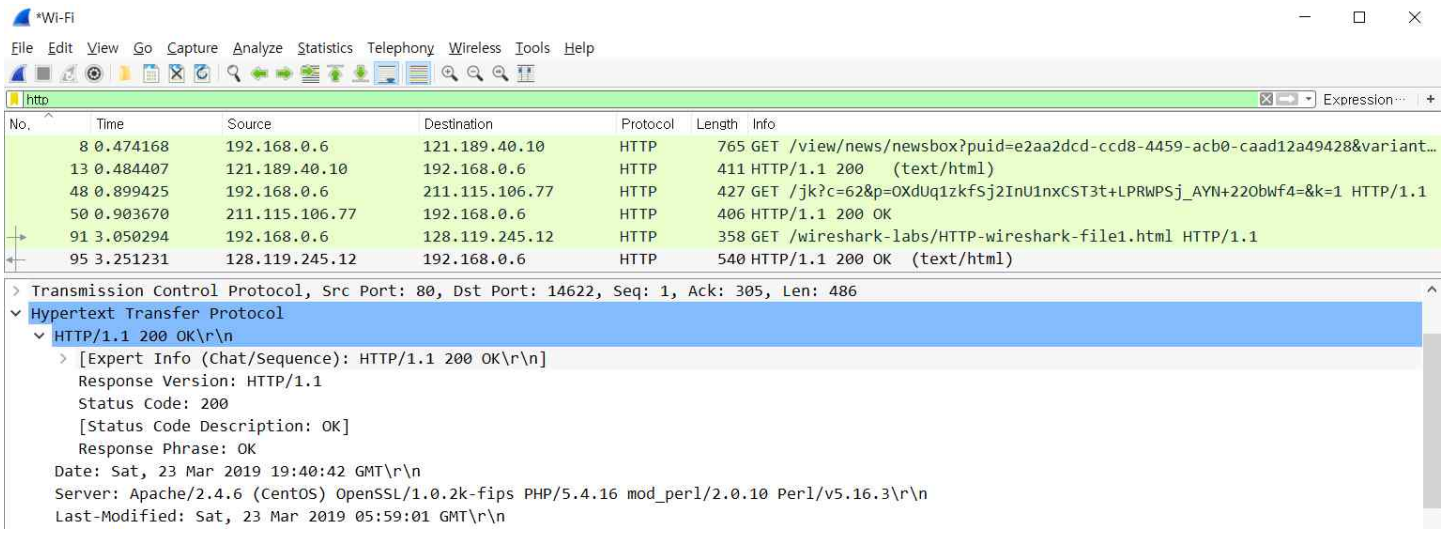
The IP address of the gaia.cs.umass.edu server is 128.119.245.12



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

Answer：

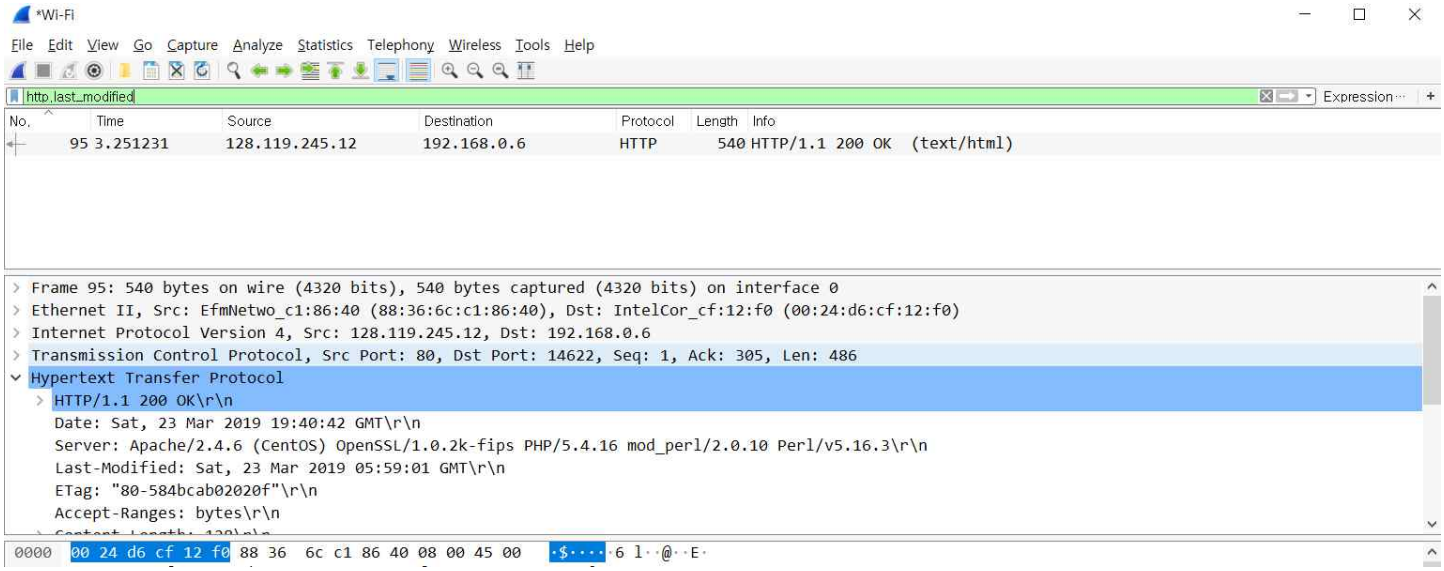
The status code returned was 200 OK



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

Answer：

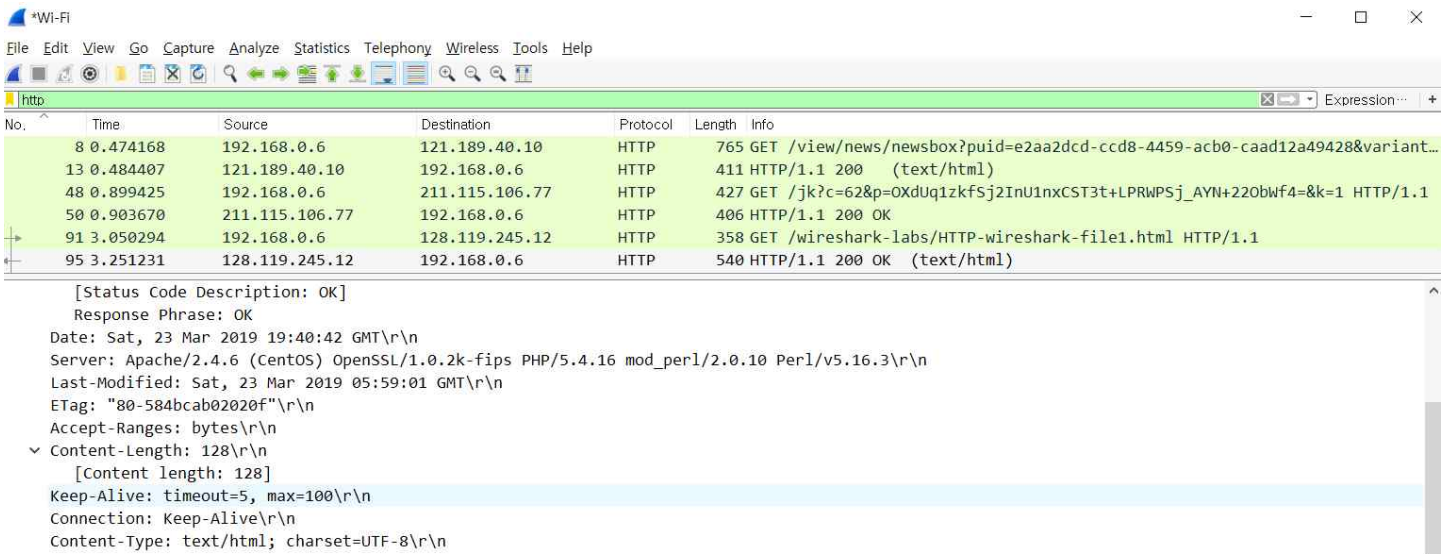
The file was last modified on Saturday, March 23, 2019 at 05:59:01 GMT



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

Answer：

128 bytes of content are being returned



[B]. The HTTP CONDITIONAL GET/response interaction → HTTP CONDITIONAL GET / 응답 상호 작용

Most web browsers perform object caching and thus perform a conditional GET when retrieving an HTTP object. Before performing the steps below, make sure your browser's cache is empty. (To do this under Internet Explorer, select Tools->Internet Options->Delete File; these actions will remove cached files from your browser's cache.) Now do the following:

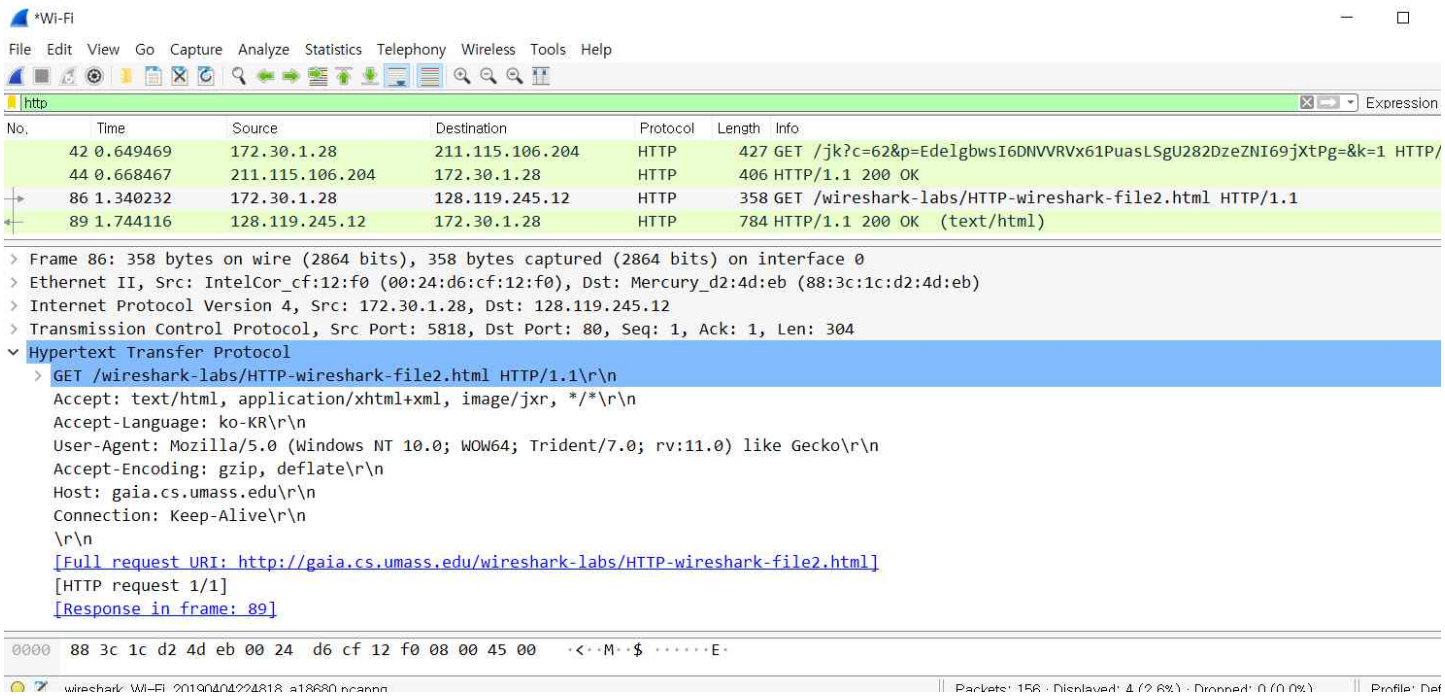
- Start up your web browser, and make sure your browser's cache is cleared, as discussed above.
- Start up the Wireshark packet sniffer
- Enter the following URL into your browser
  - http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file2.html
  - Your browser should display a very simple five-line HTML file.
- 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.

Answer the following questions:

6. 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?

Answer :

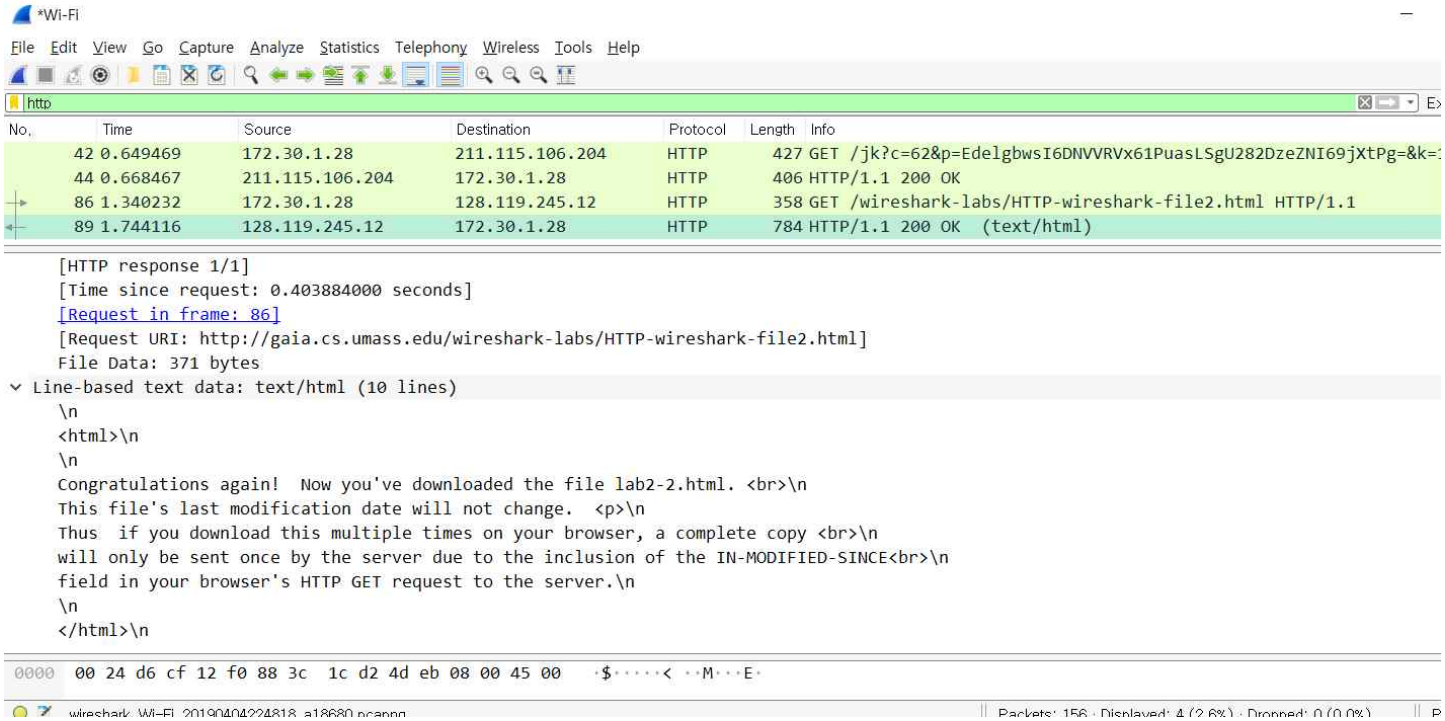
No there is no IF-MODIFIED-SINCE line in the GET message.



7. Inspect the contents of the server response. Did the server explicitly return the contents of the file?

Answer：

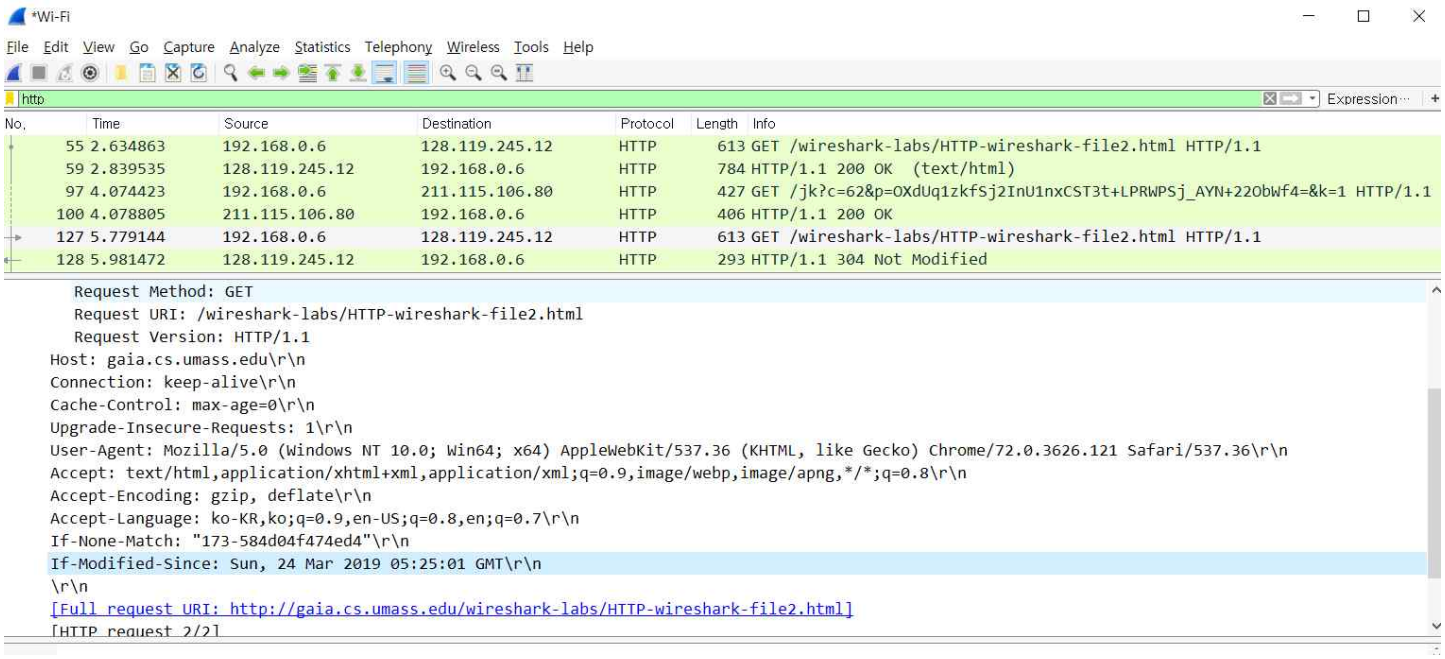
The server did explicitly return the contents of the file.



8. 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?

Answer：

Yes in the second HTTP message an IF-MODIFIED-SINCE line is included. The information that follows is the data and time (sun, 24 Mar 2019 05:25:01 GMT)

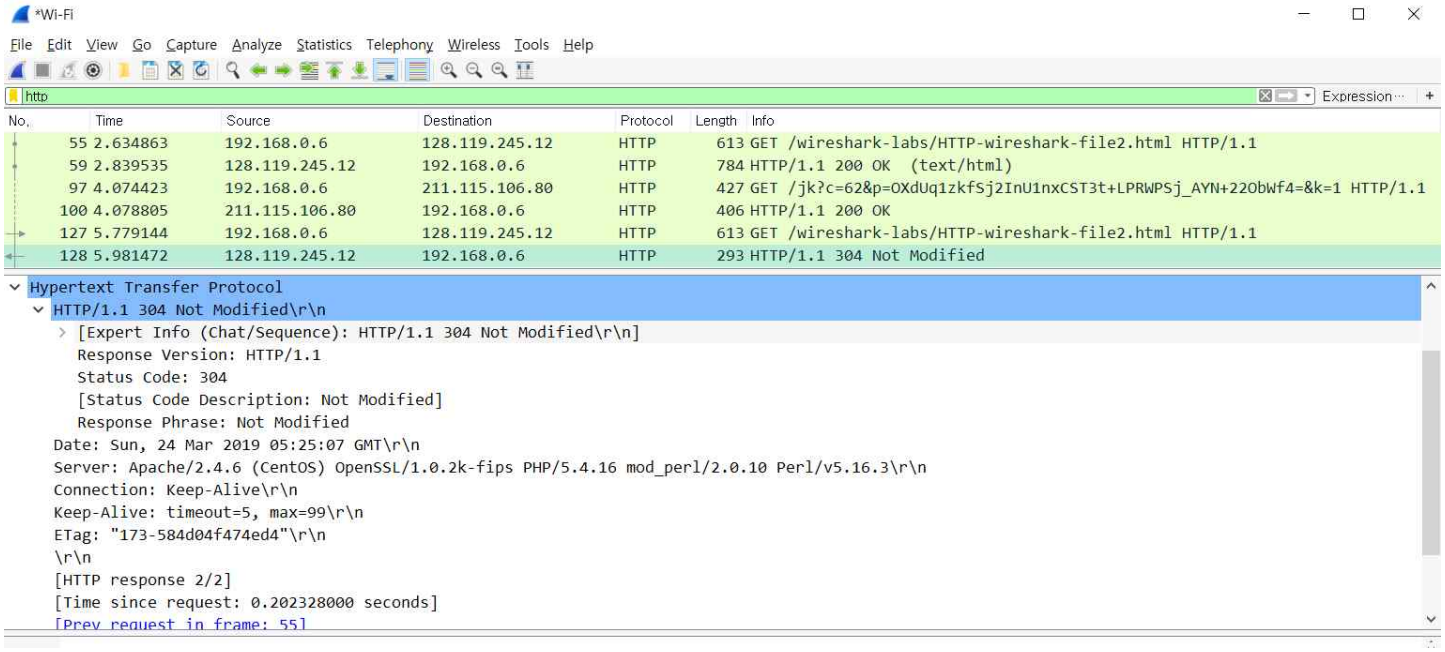


9. What is the HTTP status code and phrase returned from the server in response to this second HTTP GET? Did the server explicitly return the contents of the file? Explain.

Answer：

The HTTP status code is 304 and Response Phrase is Not Modified

The server did not return the contents of the file because the browser simply retrieved the contents from its cache



[C]. Retrieving Long Documents //긴 문서 검색

In our examples thus far, the documents retrieved have been simple and short HTML files. Let's next see what happens when we download a long HTML file. Do the following:

- Start up your web browser, and make sure your browser's cache is cleared
- Start up the Wireshark packet sniffer
- Enter the following URL into your browser
  - http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file3.html
- Stop Wireshark packet capture, and filter "http"

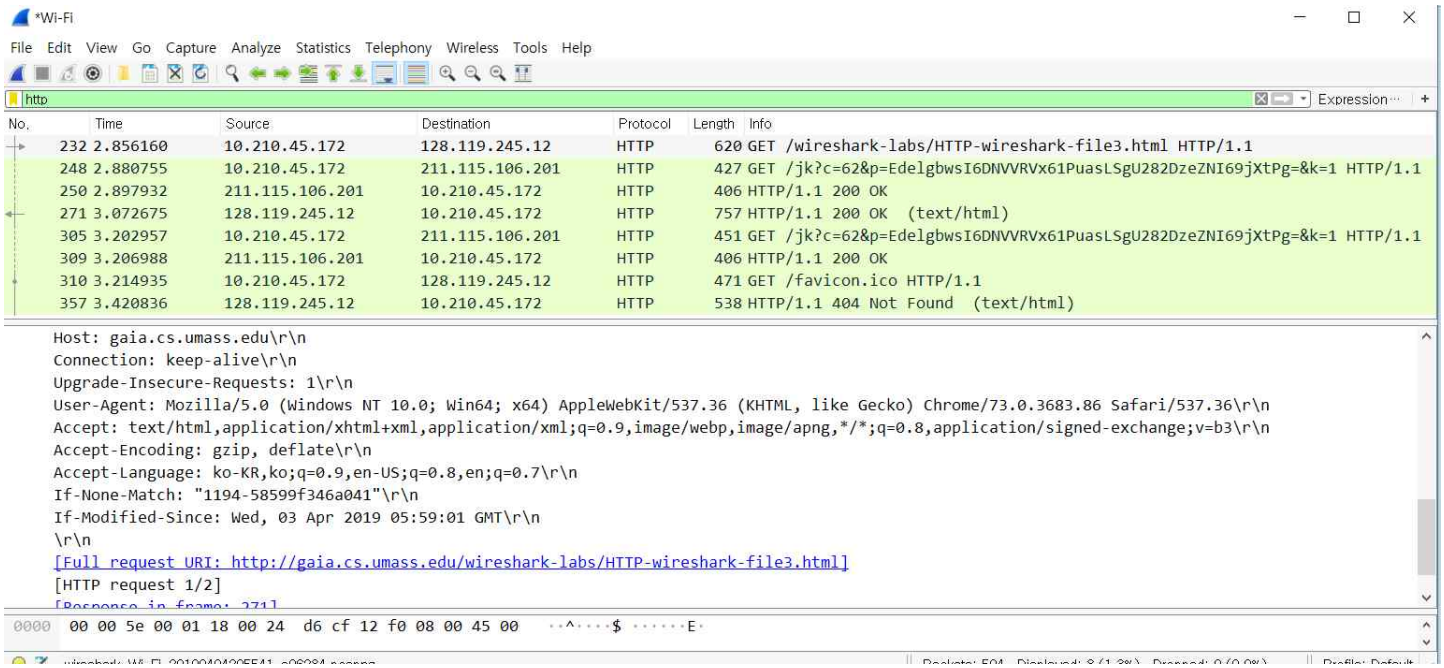
In the packet-listing window, you should see your HTTP GET message, followed by a multiple-packet TCP response to your HTTP GET request.

Answer the following questions:

10. 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'?

Answer :

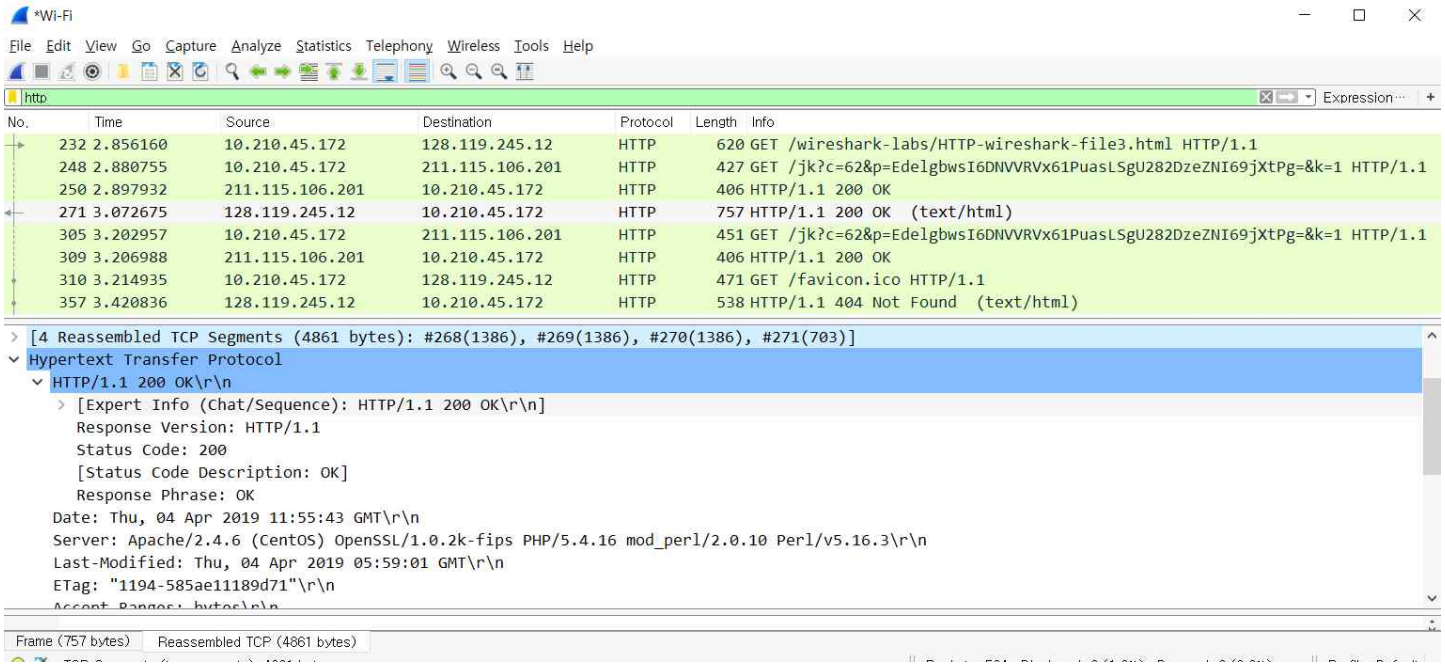
My browser sent 1 HTTP GET request to the server except favicon.ico.
The Packet that contained the GET message was packet number 232



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

Answer：

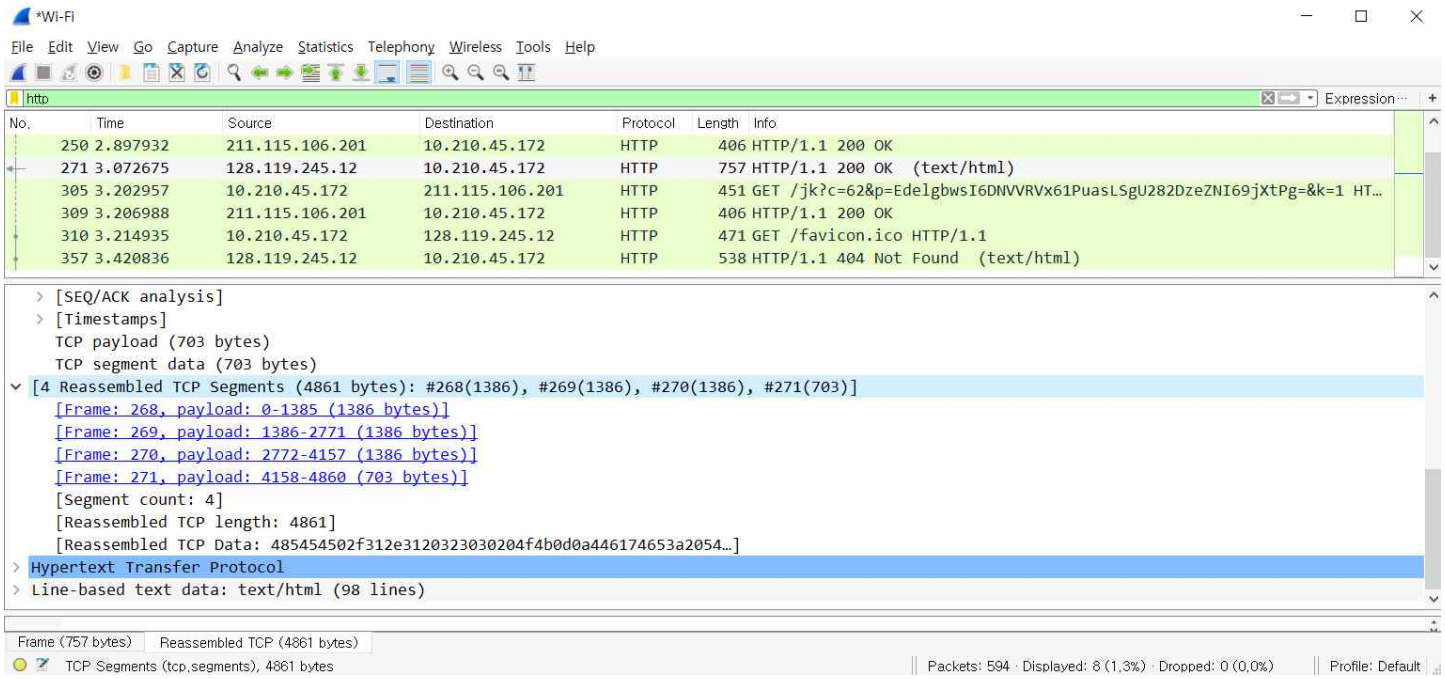
The status code was 200 and phrase was OK in the response



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

Answer：

The data was sent in 4 TCP segments to browser, then reassembled.



[D]. HTML Documents with Embedded Objects // 내장된 개체가 있는 HTML 문서

Now that we've seen how Wireshark displays the captured packet traffic for large HTML files, we can look at what happens when your browser downloads a file with embedded objects, i.e., a file that includes other objects (in the example below, image files) that are stored on another server(s). Do the following:

- Start up your web browser, and make sure your browser's cache is cleared.
- 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. Your browser will have to retrieve these logos from the indicated web sites, one from the gaia.cs.umass.edu web site, and another from caite.cs.umass.edu server.

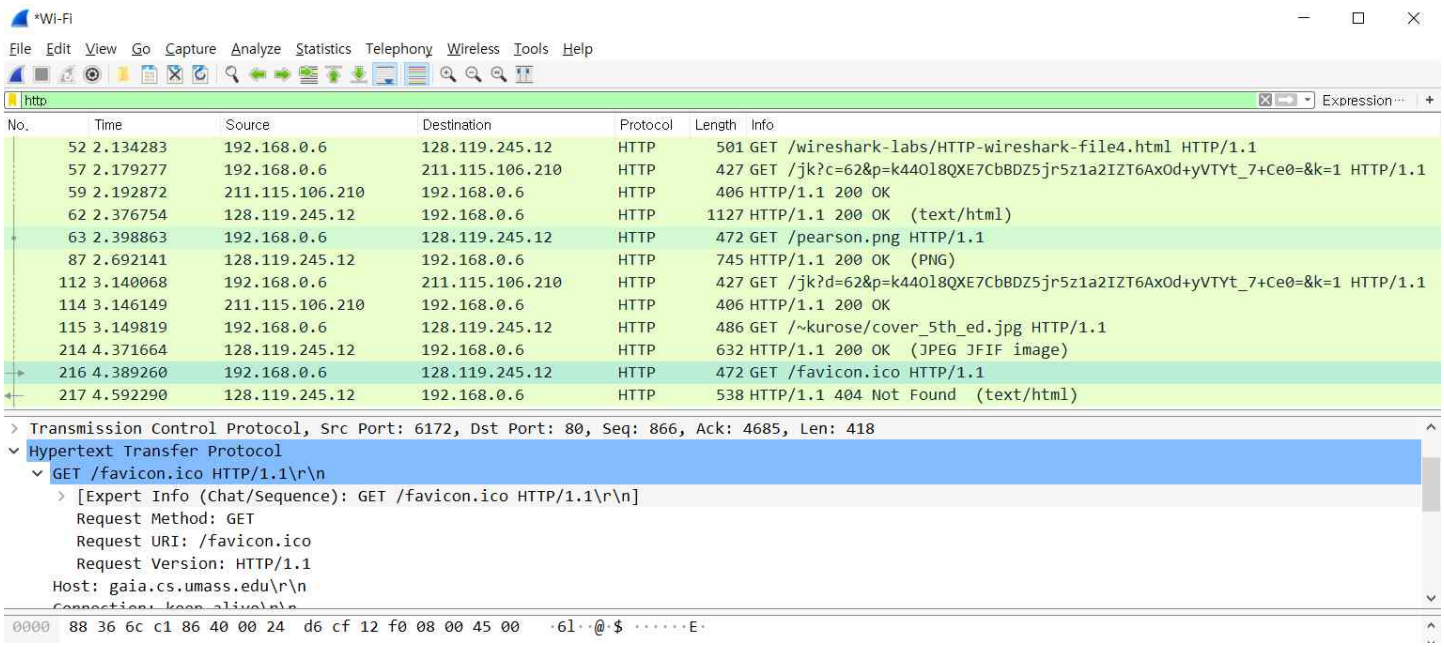
- Stop Wireshark packet capture, and filter "http".

Answer the following questions:

13. How many HTTP GET request messages did your browser send?

<u>Answer :</u>

<u>My browser sent 3 http GET message requests except favicon.ico.</u>

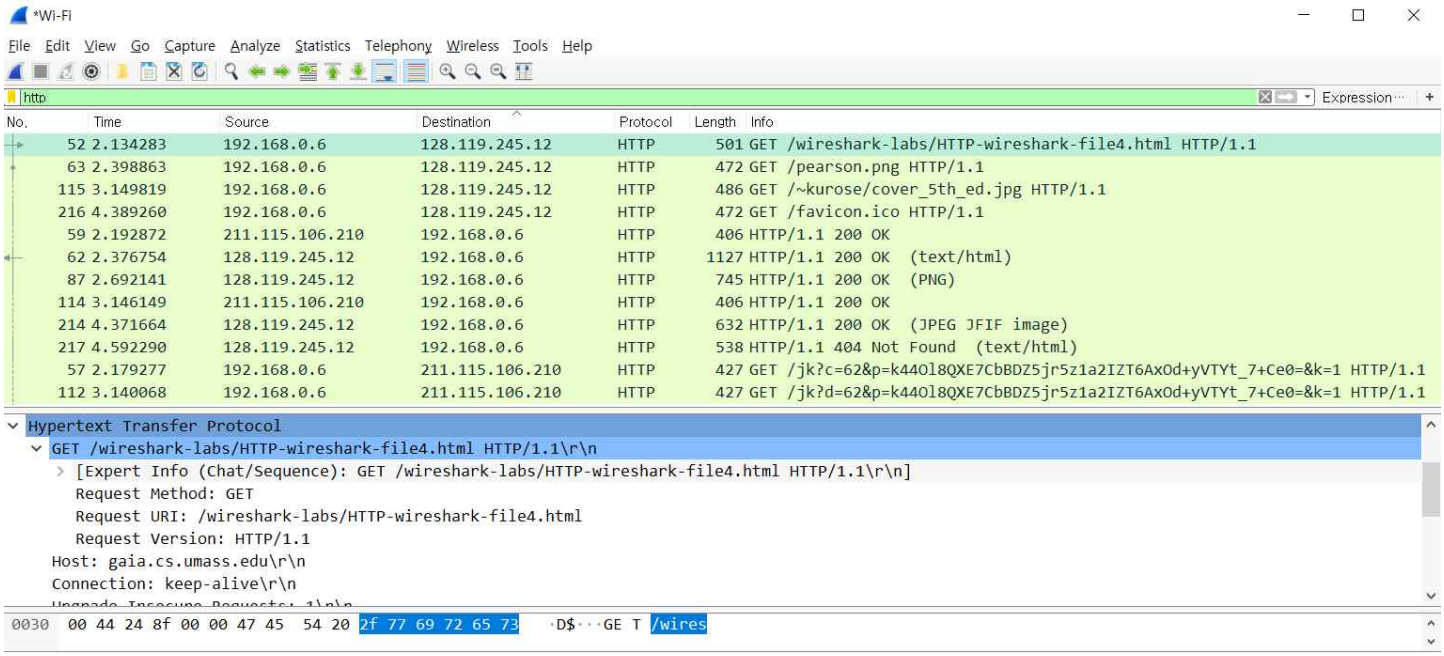


To which Internet addresses were these GET requests sent?

Answer：

Initial Page address : 128.119.245.12
Pearson.png : 128.119.245.12
cover_5th_ed.jpg : 128.119.245.12

(favicon.ico (128.119.245.12))



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

Answer：

In the captures below, the time in the PNG and JPEG JFIF images will differ by 1 second. Therefore, you can see that they were downloaded sequentially from both websites.