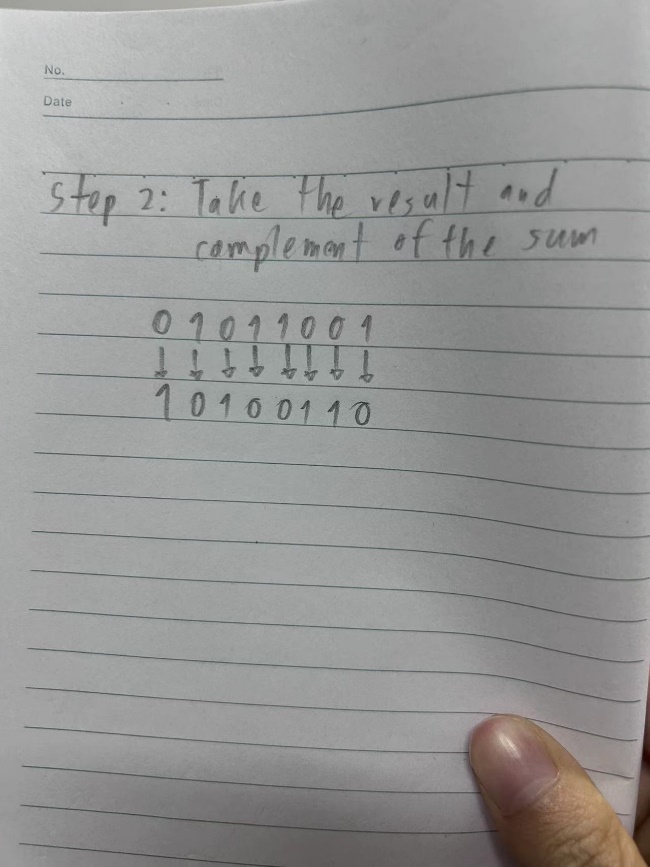
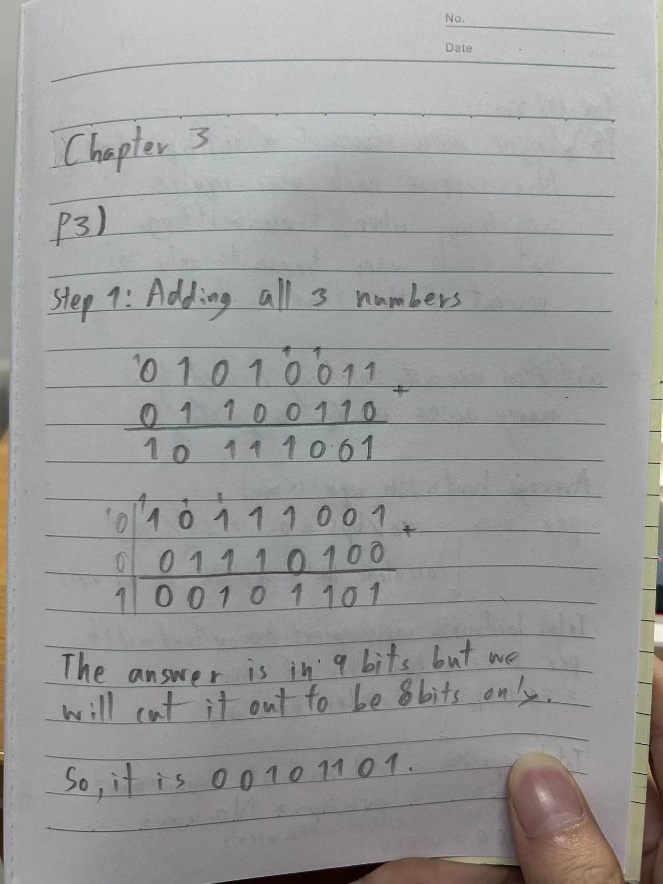
Assignment

**Homework 7**

P3. UDP and TCP use 1s complement for their checksums. Suppose you have the following three 8-bit bytes: 01010011, 01100110, 01110100. What is the 1s complement of the sum of these 8-bit bytes? (Note that although UDP and TCP use 16-bit words in computing the checksum, for this problem you are being asked to consider 8-bit sums.) Show all work.

**Answer:**

****

Why is it that UDP takes the 1s complement of the sum; that is, why not just use the sum?

**Answer:** UDP uses the one's complement of the sum to provide better error detection, helping to ensure the data is transmitted accurately.

With the 1s complement scheme, how does the receiver detect errors?

**Answer:** The receiver calculates its own checksum using the same method as the sender. It adds up all the parts of the data and takes the one's complement of the sum. Then it compares this calculated checksum with the one it received.

If the calculated checksum matches the received checksum, it means the data is likely to have no error. But if the calculated checksum is different from the received checksum, it means errors may have occurred during transmission.

So, by comparing the calculated checksum with the received checksum, the receiver can tell if there are errors in the data.

Is it possible that a 1-bit error will go undetected? How about a 2-bit error?

**Answer:** For 1 bit error, it will always be detected by the checksum, but for 2 bits errors sometimes it could not be detected because there are certain combinations of errors that can cancel each other out.

P4.

a. Suppose you have the following 2 bytes: 01011100 and 01100101. What is the 1s complement of the sum of these 2 bytes?

**Answer:** Adding = 11000001**,** Complement = 00111110

b. Suppose you have the following 2 bytes: 11011010 and 01100101. What is the 1s complement of the sum of these 2 bytes?

**Answer:** Adding = 100111111, Complement = 011000000

c. For the bytes in part (a), give an example where one bit is flipped in each of the 2 bytes and yet the 1s complement doesn’t change.

**Answer:** First byte = 01010100, Second byte = 01101101

**Homework 6**

P26. Suppose Bob joins a BitTorrent torrent, but he does not want to upload any data to any other peers (so called free-riding).

a. Bob claims that he can receive a complete copy of the file that is shared by the swarm. Is Bob’s claim possible? Why or why not?

**Answer**: It is possible because the system of BitTorrent decided to make the users being P2P in order to download and share data together. Nevertheless, in Bob’s case he wants to download the file only which he might be able to download, but the other peers will see that he does not upload anything, so they will not support him much and it is going to make Bob downloading file slower.

b. Bob further claims that he can further make his “free-riding” more efficient by using a collection of multiple computers (with distinct IP addresses) in the computer lab in his department. How can he do that?

**Answer**: In my own perspective, I think that Bob does is according to these following steps.

1. Install BitTorrent to computers: Bob would install BitTorrent into his main computer and into other computers.

2. Start download on each computer: Let say Bob downloaded the same file into every of his computers and with BitTorrent these computers will receive and share data between themselves, Bob’s main computer, and the server.

3. Combine downloaded parts: Once each computer has downloaded parts of the file, Bob can combine these parts manually or use synchronization tools to gather all the downloaded fragments into a complete file on one of the computers.

These are the steps that I think Bob does in order to make him available to download file without sharing back to the society, however, it violates the purpose of BitTorrent that would like very users share and receive the information together as if they are working in group project.

**Homework 5**

P12. Write a simple TCP program for a server that accepts lines of input from a client and prints the lines onto the server’s standard output. (You can do this by modifying the TCPServer.py program in the text.) Compile and execute your program. On any other machine that contains a Web browser, set the proxy server in the browser to the host that is running your server program; also, configure the port number appropriately. Your browser should now send its GET request messages to your server, and your server should display the messages on its standard output. Use this platform to determine whether your browser generates conditional GET messages for objects that are locally cached.

P15. What is the difference between MAIL FROM: in SMTP and From: in the mail message itself?

*MAIL FROM in SMTP:*

It is a part of the SMTP protocol used for sending emails. It is used by the sending mail server to the receiving mail server to start the process of delivering an email message. Moreover, the purpose of it is for routing the receiving email server to determine where non-delivery notifications should be sent in case of delivery failure.

*From: in the mail itself:*

The “From:” field is a header field in the email message itself. It is used to specify the email address of the sender that is displayed to the recipients. Furthermore, it identifies the sender of the message to the recipients.

In summary, while both "MAIL FROM" in SMTP and "From:" in the email message itself involve specifying the sender's email address, "MAIL FROM" is used in the email delivery process between mail servers, whereas "From:" is a header field in the email message that identifies the sender to the recipients.

P20. Suppose you can access the caches in the local DNS servers of your department. Can you propose a way to roughly determine the Web servers (outside your department) that are most popular among the users in your department? Explain.

**1. Access DNS Cache Records:**

Gain access to the DNS cache records in your department's local DNS servers.

**2. Extract Hostnames:**

Extract the hostnames or domain names from the DNS cache records.

**3. Count Frequency:**

Count the frequency of each hostname or domain name.

**4. Rank Websites:**

Rank the websites based on their frequency of access.

**5. Analyze Trends:**

Monitor trends over time to understand changes in website popularity.

**6. Consider Hosting Providers:**

Identify the hosting providers associated with popular websites.

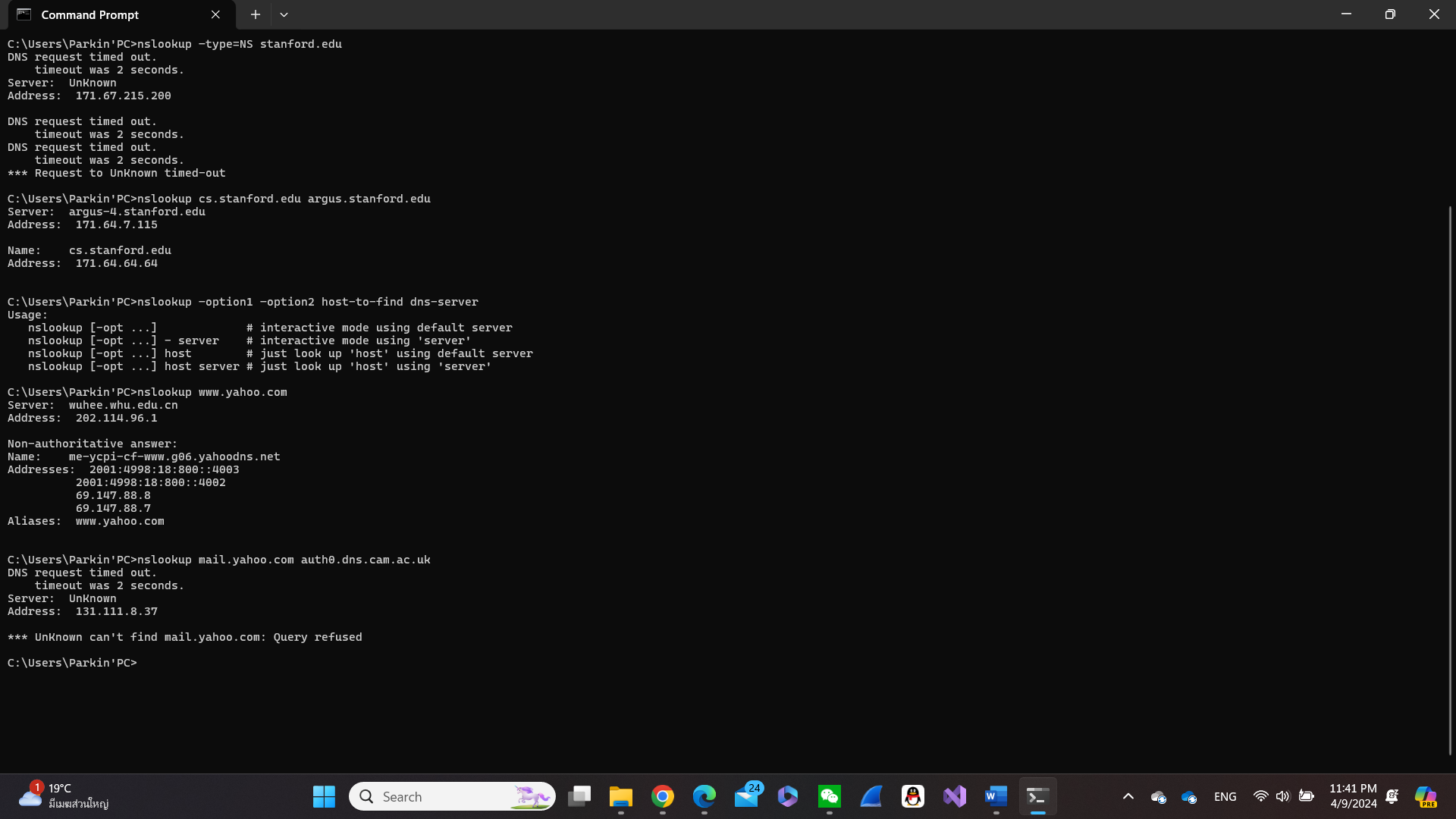
**7. Aggregate Data:**

Aggregate data over time for accuracy.

**8. Ensure Privacy:**

Respect user privacy and comply with relevant regulations

**Lab 4 Wireshark DNS1**

****

1. Run *nslookup* to obtain the IP address of a Web server in China. What is the IP address of that server?

Answer: Addresses: 2001:250:4001:2::1223

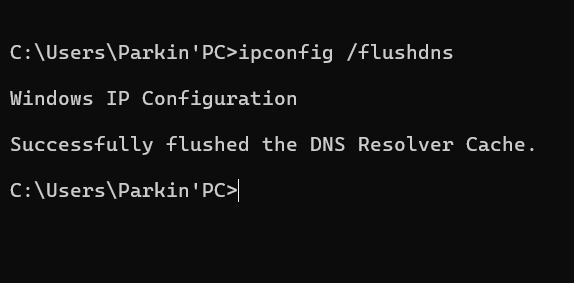
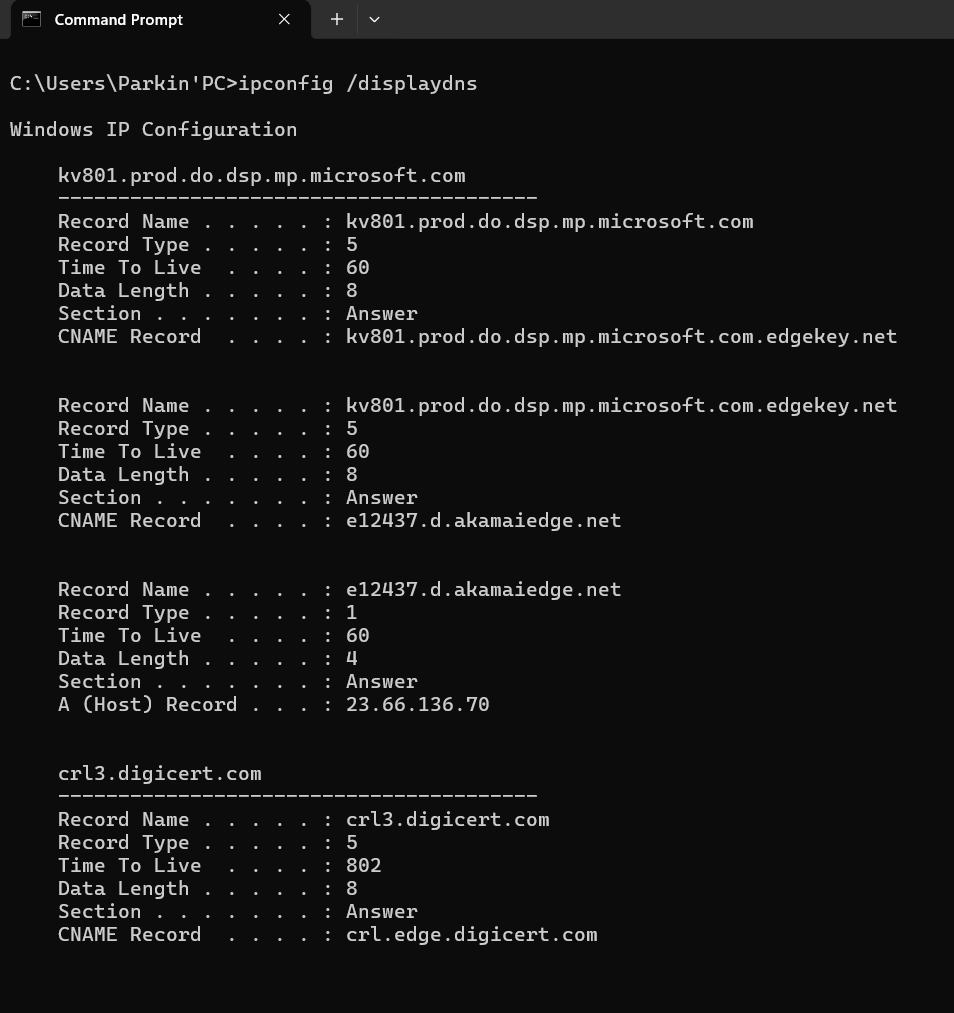
202.114.64.221

1. Run *nslookup* to determine the authoritative DNS servers for a university in Europe.

Answer: Address: 171.67.215.200

1. Run *nslookup* so that one of the DNS servers obtained in Question 2 is queried for the mail servers for Yahoo! mail. What is its IP address?

Answer: Address: 131.111.8.37



1. Locate the DNS query and response messages. Are they sent over UDP or TCP?

Answer: Yes, they do. “User Datagram Protocol, Src Port: 63037, Dst Port: 53”

1. What is the destination port for the DNS query message? What is the source port of DNS response message?

Answer: Destination of the port is Dst Port: 53

1. To what IP address is the DNS query message sent? Use ipconfig to determine the IP address of your local DNS server. Are these two IP addresses the same?

Answer: DNS query message sent Source: 10.131.191.53 and Destination: 202.114.96. Furthermore, they sent the same information.

Wireless LAN adapter Wi-Fi:

Connection-specific DNS Suffix . :

Description . . . . . . . . . . . : Intel(R) Wireless-AC 9560 160MHz

Physical Address. . . . . . . . . : D8-3B-BF-5B-1A-9E

DHCP Enabled. . . . . . . . . . . : Yes

Autoconfiguration Enabled . . . . : Yes

Link-local IPv6 Address . . . . . : fe80::b8e7:feb7:f398:efd5%17(Preferred)

IPv4 Address. . . . . . . . . . . : 10.131.191.53(Preferred)

Subnet Mask . . . . . . . . . . . : 255.255.128.0

Lease Obtained. . . . . . . . . . : Tuesday, April 9, 2024 9:41:02 PM

Lease Expires . . . . . . . . . . : Wednesday, April 10, 2024 1:37:25 AM

Default Gateway . . . . . . . . . : 10.131.255.254

DHCP Server . . . . . . . . . . . : 10.131.255.254

DHCPv6 IAID . . . . . . . . . . . : 282606527

DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-26-0C-2A-74-F8-75-A4-E2-D9-B8

DNS Servers . . . . . . . . . . . : 202.114.96.1

202.114.64.1

NetBIOS over Tcpip. . . . . . . . : Enabled

1. Examine the DNS query message. What “Type” of DNS query is it? Does the query message contain any “answers”?

Answer: It is query type A and it did not contain any answers.

1. Examine the DNS response message. How many “answers” are provided? What do each of these answers contain?

Answer: In my Wireshark, it appeared no answers.

1. Consider the subsequent TCP SYN packet sent by your host. Does the destination IP address of the SYN packet correspond to any of the IP addresses provided in the DNS response message?

Answer: It is TCP destination: 199.16.156.11

1. This web page contains images. Before retrieving each image, does your host issue new DNS queries?

Answer: No, it did not contain any images.

1. What is the destination port for the DNS query message? What is the source port of DNS response message?

Answer: Port53

1. To what IP address is the DNS query message sent? Is this the IP address of your default local DNS server?

Answer: Destination: 202.114.96.1 and it is the IP address of my local DNS server.

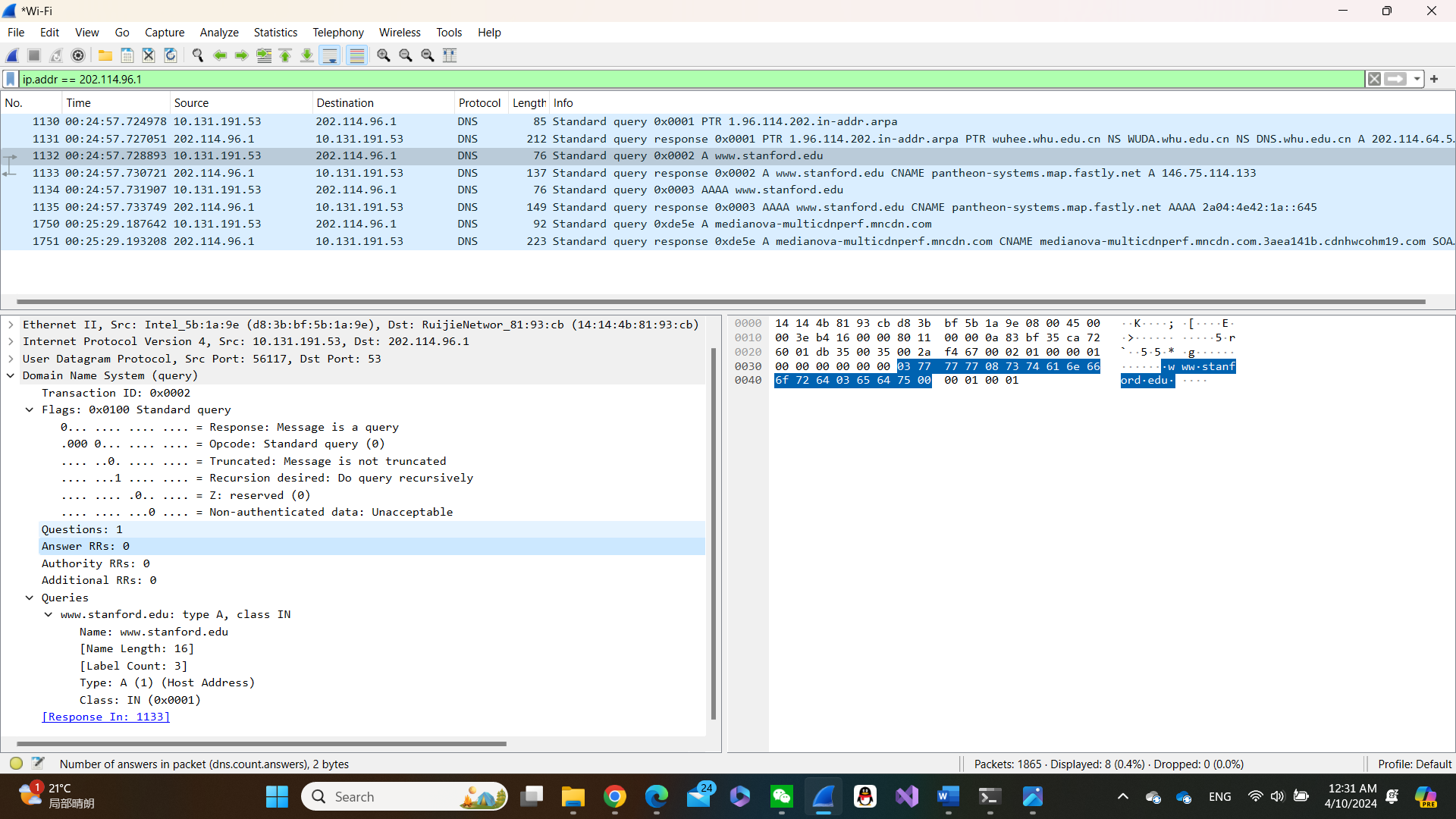
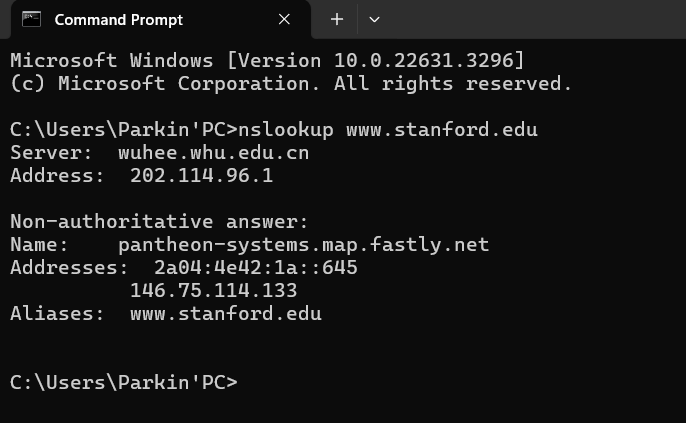
1. Examine the DNS query message. What “Type” of DNS query is it? Does the query message contain any “answers”?

Answer: They said the query are types A and AAQ but did not contain any answers.

1. Examine the DNS response message. How many “answers” are provided? What do each of these answers contain?

Answer: In my Wireshark it did not show the answers in the queries section.

1. Provide a screenshot.



16. To what IP address is the DNS query message sent? Is this the IP address of your default local DNS server?

Answer: Destination: 202.114.96.1 and it is the IP address of my local DNS server

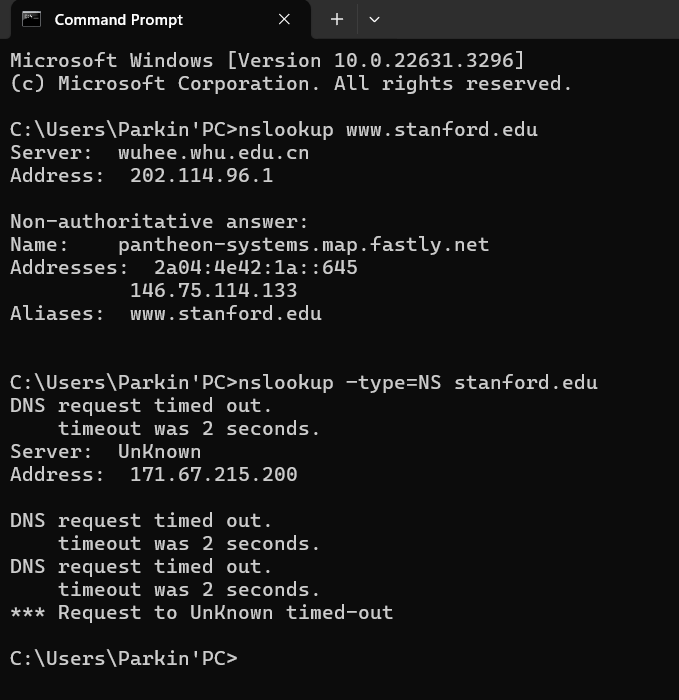
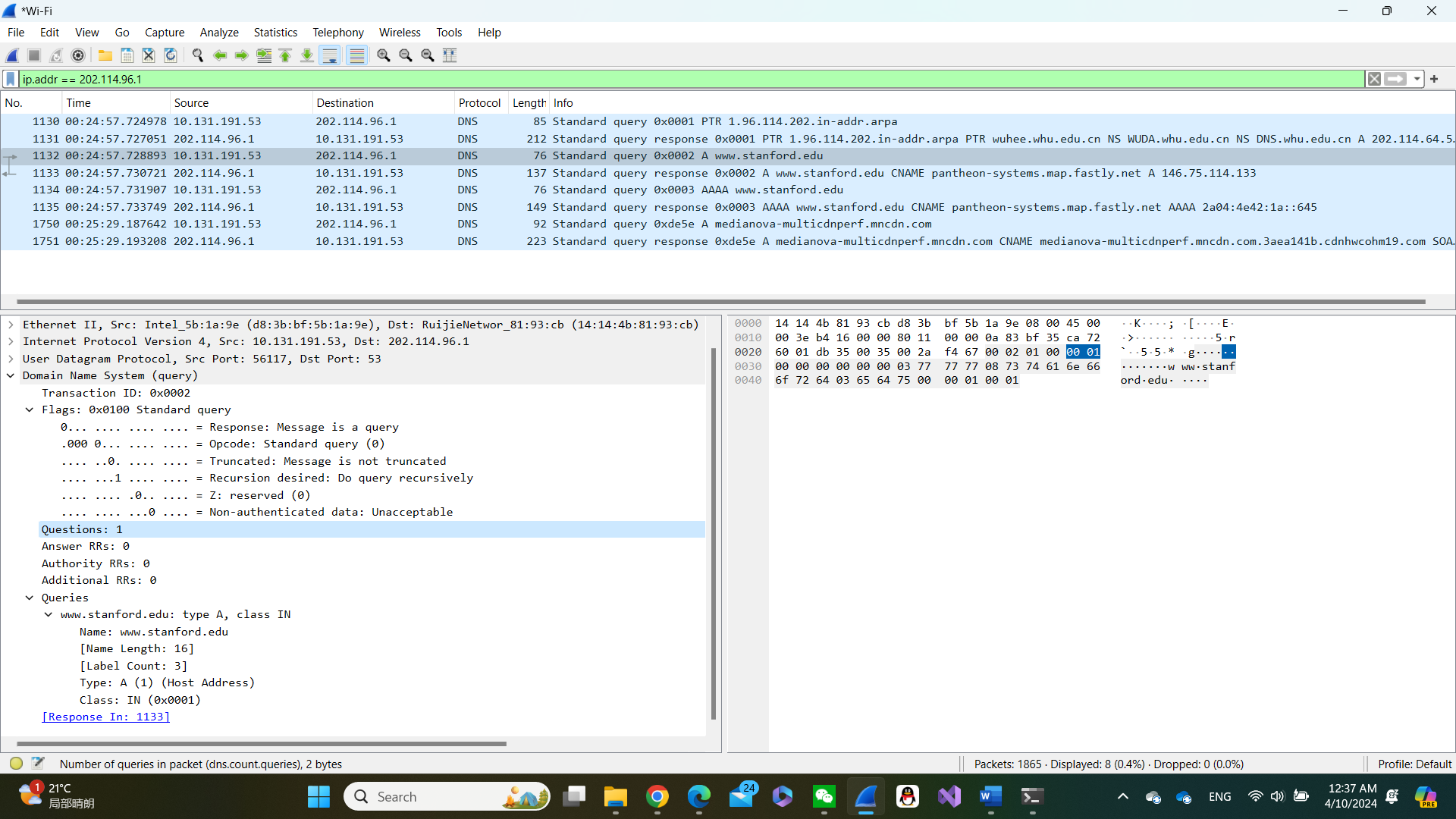
17. Examine the DNS query message. What “Type” of DNS query is it? Does the query message contain any “answers”?

Answer: Query type A and it does not contain any message.

18. Examine the DNS response message. What Stanford nameservers does the response message provide? Does this response message also provide the IP addresses of the Stanford name servers?

Answer: In my Wireshark, it did not show any message.

19. Provide a screenshot

****

20. To what IP address is the DNS query message sent? Is this the IP address of your default local DNS server? If not, what does the IP address correspond to?

Answer: First it is sent to get to default DNS query message to get address of [www.stanford.edu](http://www.stanford.edu) then it was sent to [www.stanford.edu](http://www.stanford.edu) .

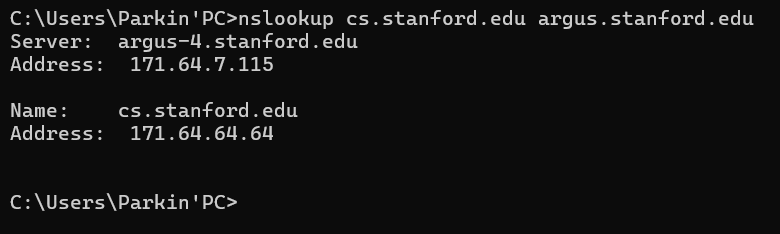
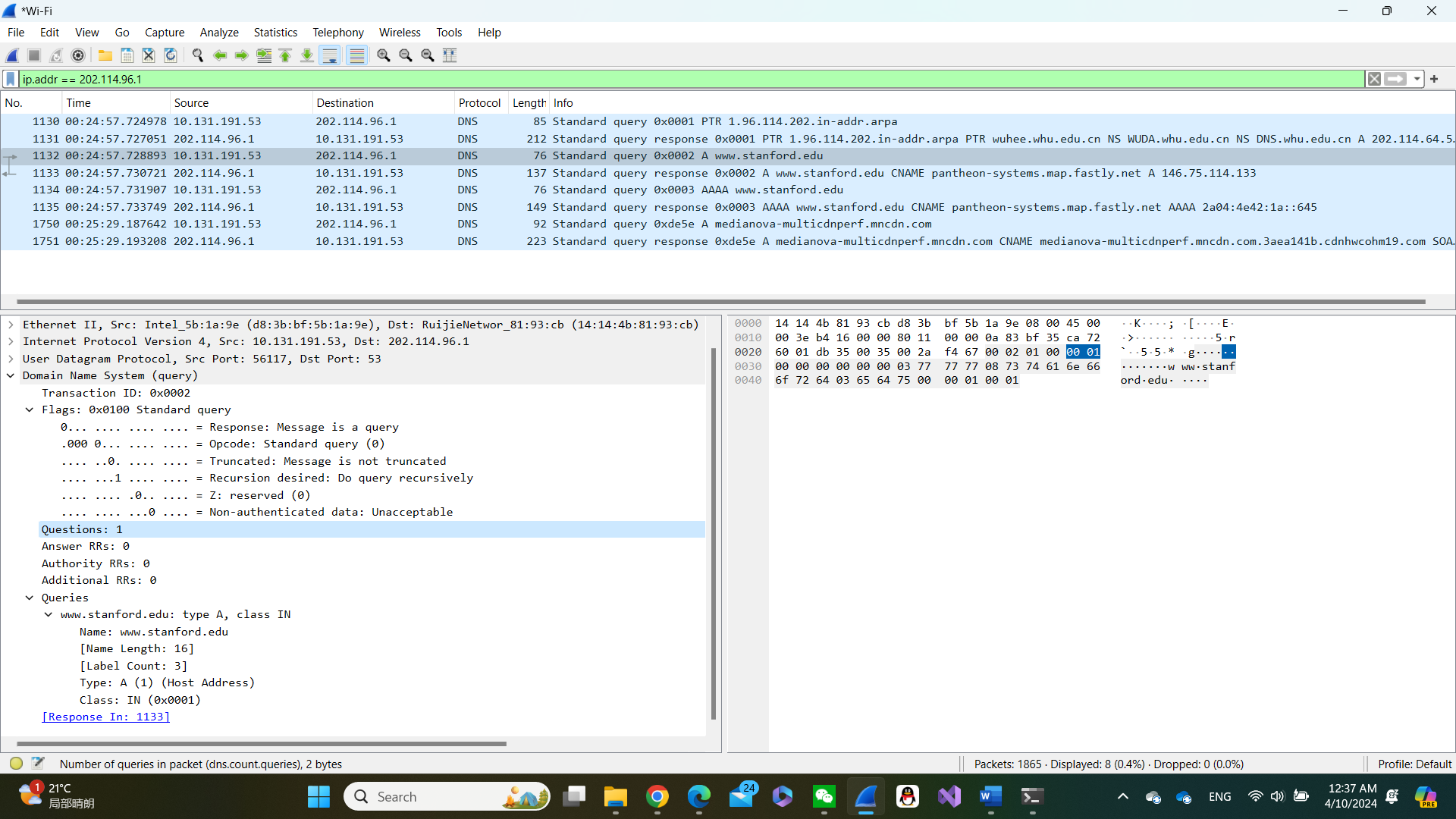
21. Examine the DNS query message. What “Type” of DNS query is it? Does the query message contain any “answers”?

Answer: The types of queries are A and AAA, moreover, it does not contain any message

22. Examine the DNS response message. How many “answers” are provided? What does each of these answers contain?

Answer: In my Wireshark, it does not show any answers.

23. Provide a screenshot.

****

**Homework 4**

**P4**. Consider the following string of ASCII characters that were captured by Wireshark when the browser sent an HTTP GET message (i.e., this is the actual content of an HTTP GET message). The characters are carriage return and line-feed characters (that is, the italized character string in the text below represents the single carriage-return character that was contained at that point in the HTTP header). Answer the following questions, indicating where in the HTTP GET message below you find the answer.

GET /cs453/index.html HTTP/1.1Host: gai a.cs.umass.eduUser-Agent: Mozilla/5.0 ( Windows;U; Windows NT 5.1; en-US; rv:1.7.2) Gec ko/20040804 Netscape/7.2 (ax) Accept:ex t/xml, application/xml, application/xhtml+xml, text /html;q=0.9, text/plain;q=0.8,image/png,\*/\*;q=0.5 M02\_KURO1557\_08\_SE\_C02.indd 168 12/02/20 4:08 PM PROBLEMS 169 Accept-Language: en-us,en;q=0.5AcceptEncoding: zip,deflateAccept-Charset: ISO -8859-1,utf-8;q=0.7,\*;q=0.7Keep-Alive: 300 Connection:keep-alive

a. What is the URL of the document requested by the browser?

- The URL of the document requested by the browser is “cs453/index.html”.

b. What version of HTTP is the browser running?

- The version of HTTP that the browser is running is HTTP/1.1.

c. Does the browser request a non-persistent or a persistent connection?

- The browser requests a persistent connection because “Connection: keep-alive”.

d. What is the IP address of the host on which the browser is running?

- No, the provided string doesn't contain the IP address

e. What type of browser initiates this message? Why is the browser type needed in an HTTP request message?

- The browser initiating this HTTP request message is Mozilla/5.0, which is common use with Firefox browser. There are several reasons why some browsers are using HTTP request message according to the following:

1) Content Negotiation: The server can use the information about the browser type to determine the appropriate content to send back.

2) Feature Detection: Servers may need to detect the capabilities of the browser to provide appropriate content or functionality. Different browsers support different features or standards, so this information can help tailor the response.

3) Logging and Analytics: Knowing the browser type helps in logging and analytics on the server side. Website owners can analyze the distribution of browsers accessing their site, which can inform decisions about website design and optimization.

4) Compatibility: It helps in ensuring compatibility with various browsers. By knowing the browser type, developers can ensure that the website or web application functions correctly across different browsers and versions.

**Lab 2 HTTP**

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

* HTTP 1.1

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

- The system has not told.

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

- My computer IP: 10.131.165.58

- Website IP: 128.119.245.12

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

- Status Code: 404

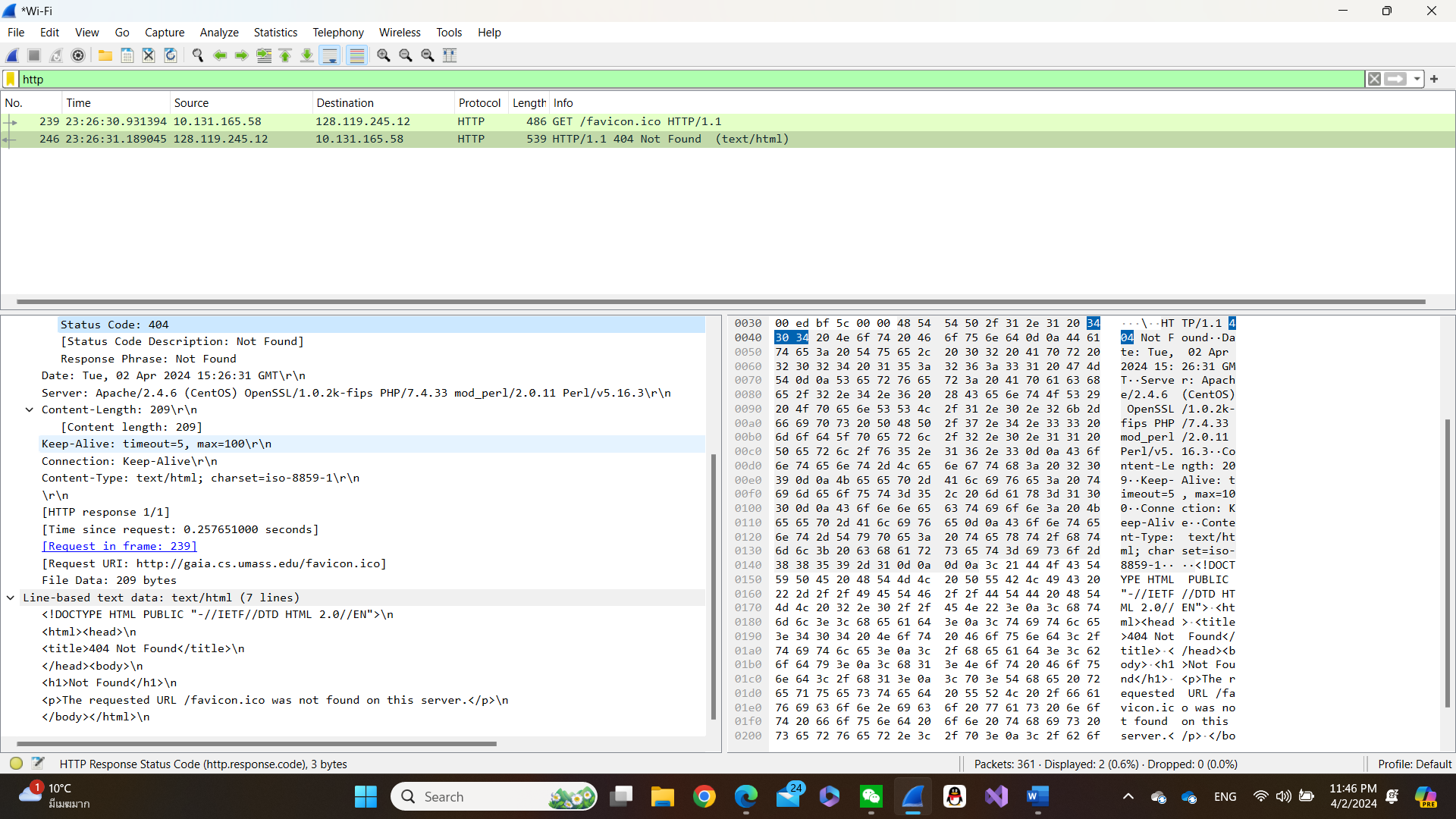
5. When was the HTML file that you are retrieving last modified at the server?

- Date: Tue, 02 Apr 2024 15:26:31 GMT\r\n

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

- File Data: 209 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, that's all

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?

- In the HTTP GET line I do not see “IF-MODIFIED-SINCE”.

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

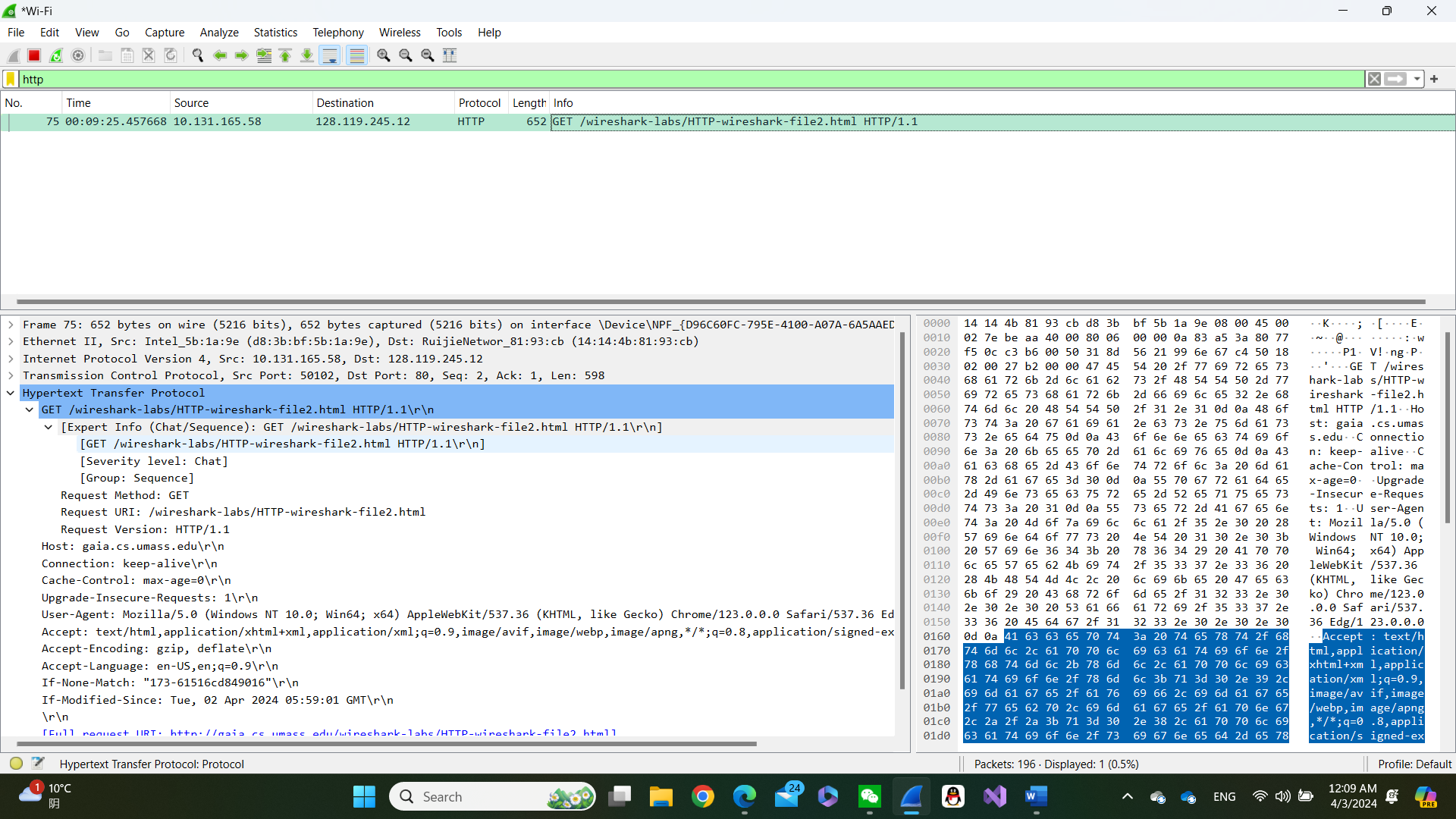
- The contents of the file in the response packet were discovered.

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?

- If-Modified-Since: Tue, 02 Apr 2024 05:59:01 GMT\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.

- Status Code: 304 that is, it has not changed, and the text has not been retransmitted, but the existing text in the cache is directly called. This offloads servers and increases efficiency.



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

- I saw 1 GET methods and the trace number is 62.

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

- Packet number 638

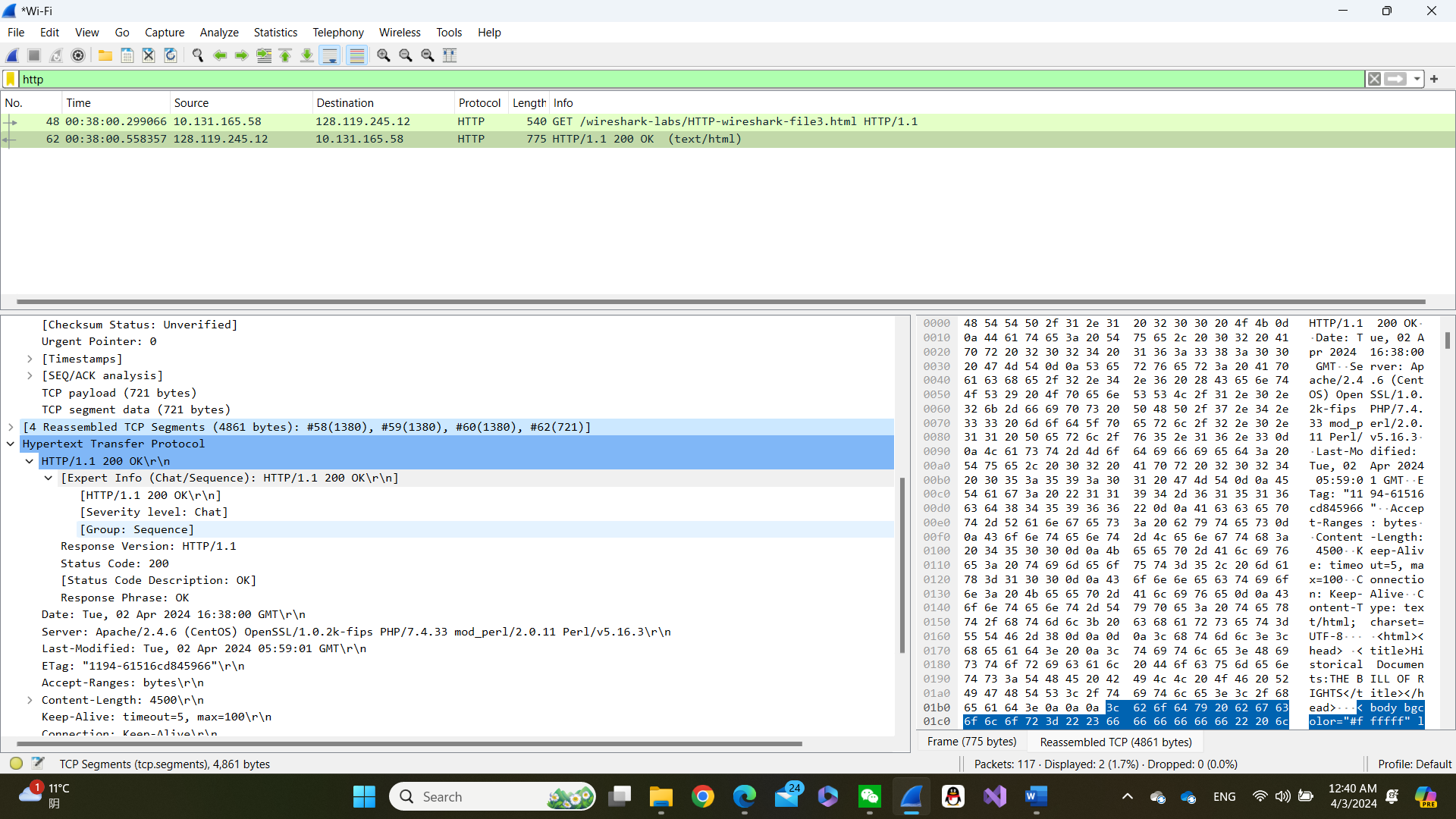
14. What is the status code and phrase in the response?

- Status Code: 200

- [Status Code Description: Not Modified]

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

- [4 Reassembled TCP Segments (4861 bytes): #58(1380), #59(1380), #60(1380), #62(721)]

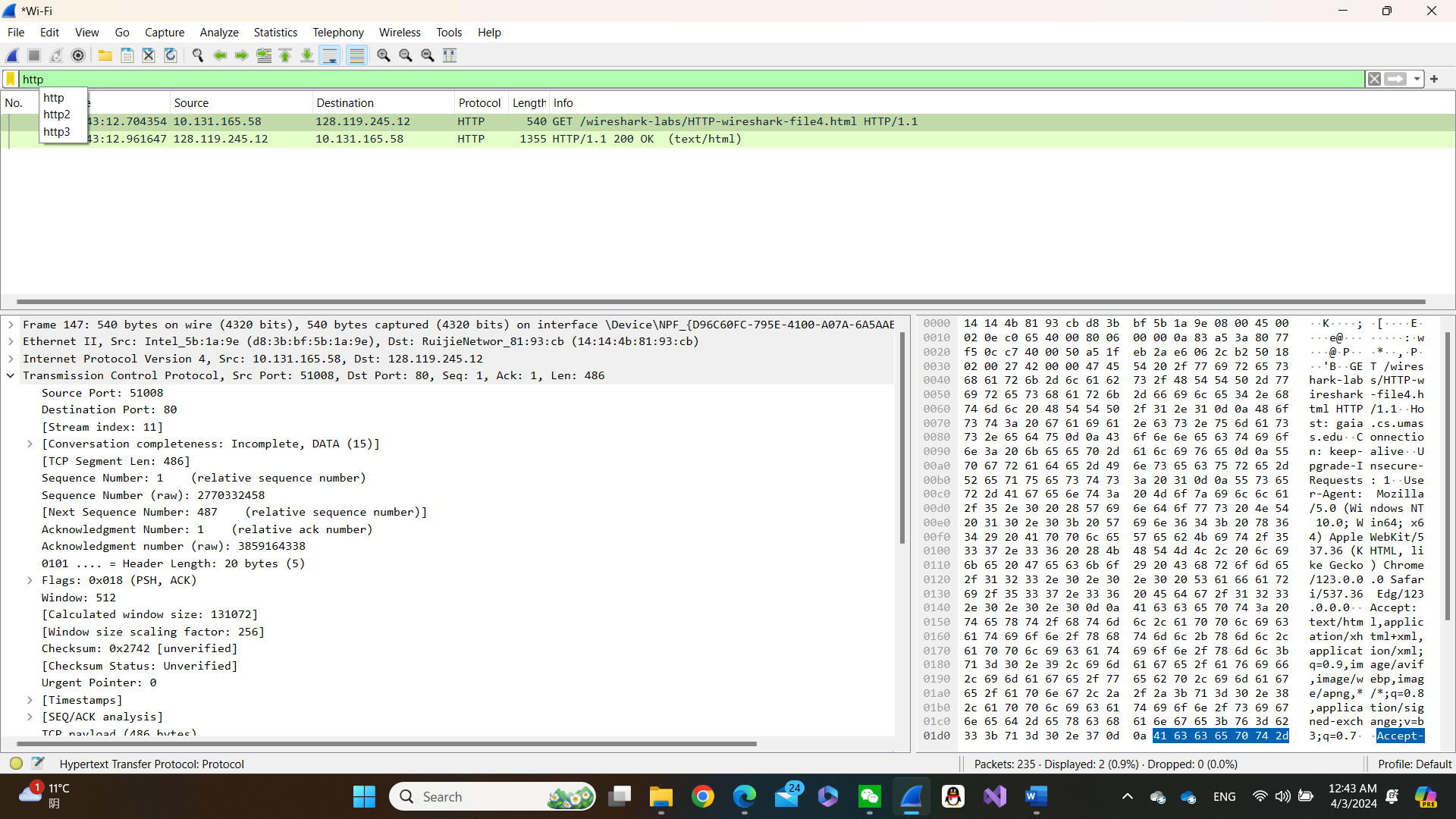


16. How many HTTP GET request messages did your browser send? To which Internet addresses were these GET requests sent?

- 1 GET requests

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.

- I'm not sure if this concurrency is really sent at the same time, but GET packets are sent at different times, so I think they should be sent sequential not parallel.

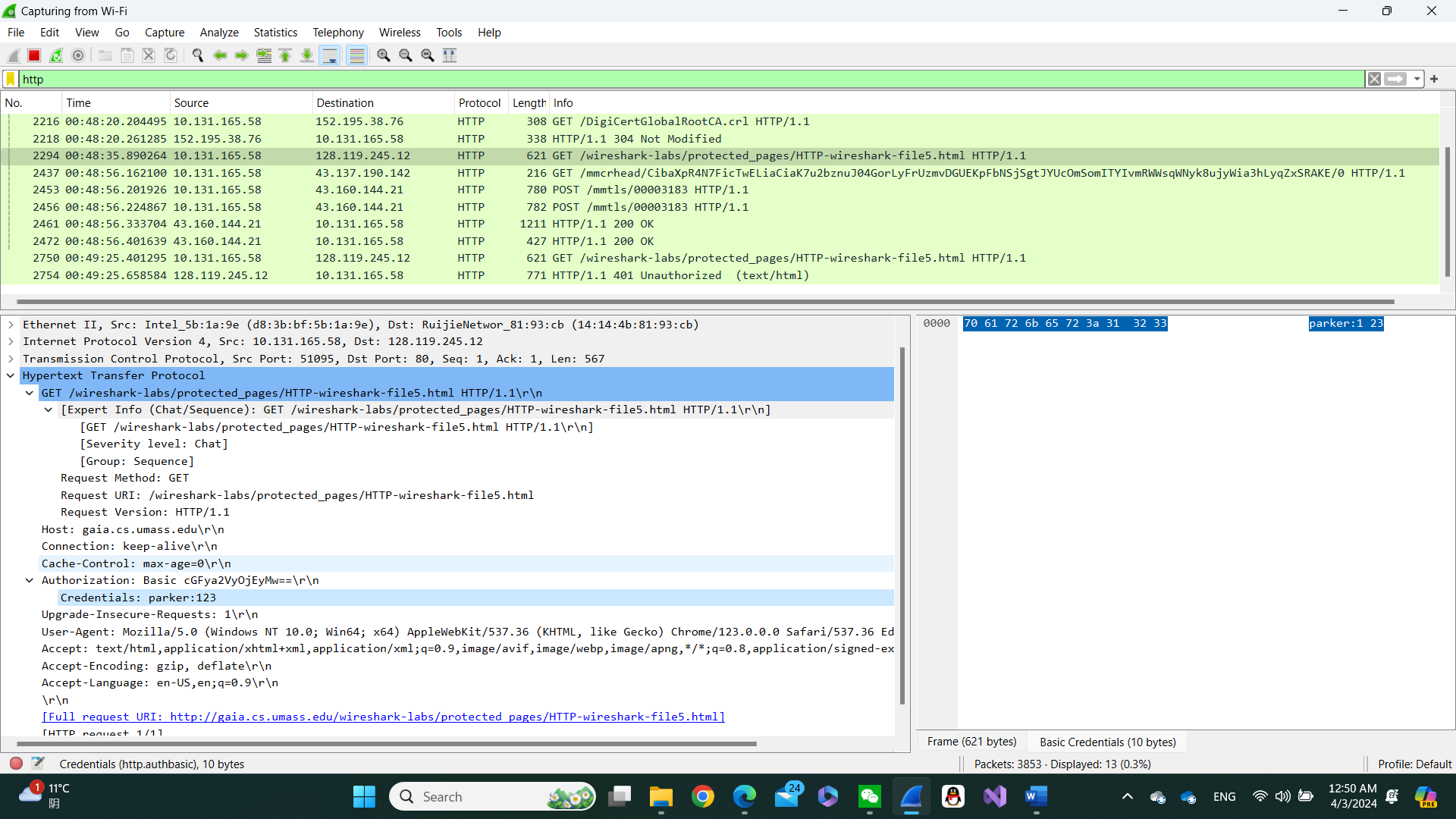


18. What is the server’s response (status code and phrase) in response to the initial HTTP GET message from your browser?

- [HTTP/1.1 401 Unauthorized\r\n]

- [Severity level: Chat]

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

- Authorization: Basic cGFya2VyOjEyMw==\r\n

- Credentials: parker:123

**Homework 3**

1. List 3 different protocols that appear in the protocol column in the unfiltered packet-listing window in step 7 above.

* UDP, TCP, ARP

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 packet-listing 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*.)

- GET message: 21:33:44.066485

- HTTP OK: 21:33:44.070267

Time to send = 0.01 seconds

3. What is the Internet address of the cs.whu.edu.cn? What is the Internet address of your computer?

* Address of cs.whu.ed.cn = 202.114.64.221
* Address of my computer = 10.131.166.249

4. Print the two HTTP messages (GET and OK) referred to in question 2 above. To do so, select *Print* from the Wireshark *File* command menu, and select the “*Selected Packet Only”* and *“Print as displayed”* radial buttons, and then click OK.

