Lab 1: Packet analysis at application layer using Wireshark

SCSR1213 Network Communications

Universiti Teknologi Malaysia

Objective:

1. Understanding of network protocols by observing the sequence of messages exchanged between two protocol entities, delving down into the details of protocol operation, and causing protocols to perform certain actions and then observing these actions and their consequences.

2. To introduce student with Wireshark software tool for packet analyzer.

3. To analyze protocol used in application layer such as http and dns.

Reference material: Computer Networking: A Top-Down Approach, 7th ed., J.F. Kurose and K.W. Ross.

Name : TEH RU QIAN

Metric No : A23CS0191

Section : Section 02

Mark

PART A: Wireshark Getting Started

1.0 Introduction

The basic tool for observing the messages exchanged between executing protocol entities is called a **packet sniffer**. As the name suggests, a packet sniffer captures ("sniffs") messages being sent/received from/by your computer; it will also typically store and/or display the contents of the various protocol fields in these captured messages. A packet sniffer itself is passive. It observes messages being sent and received by applications and protocols running on your computer, but never sends packets itself. Similarly, received packets are never explicitly addressed to the packet sniffer. Instead, a packet sniffer receives a *copy* of packets that are sent/received from/by application and protocols executing on your machine.

Figure A.1 shows the structure of a packet sniffer. At the right of Figure 1 are the protocols (in this case, Internet protocols) and applications (such as a web browser or ftp client) that normally run on your computer. The packet sniffer, shown within the dashed rectangle in Figure A.1 is an addition to the usual software in your computer, and consists of two parts. The **packet capture library** receives a copy of every link-layer frame that is sent from or received by your computer. In Figure A.1, the assumed physical media is an Ethernet, and so all upper-layer protocols are eventually encapsulated within an Ethernet frame. Capturing all link-layer frames thus gives you all messages sent/received from/by all protocols and applications executing in your computer.

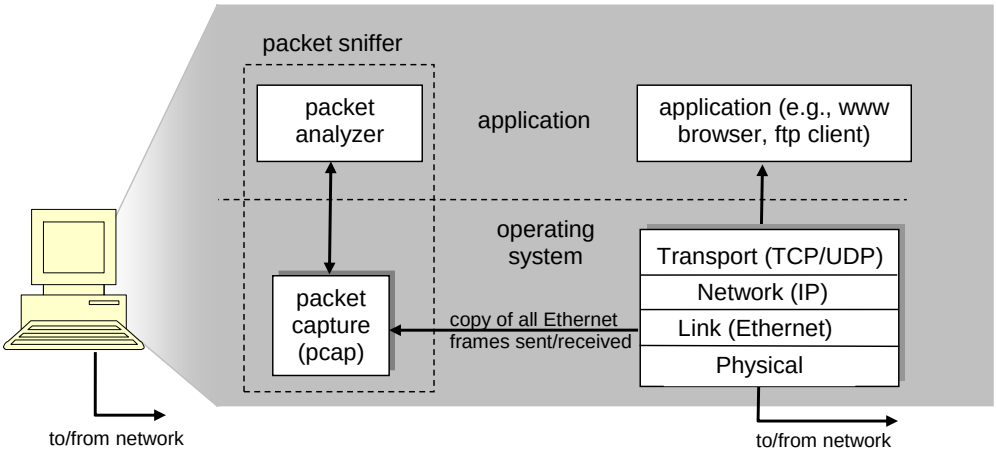


Figure A.1: Packet sniffer structure

The second component of a packet sniffer is the **packet analyzer**, which displays the contents of all fields within a protocol message. In order to do so, the packet analyzer must "understand" the structure of all messages exchanged by protocols. The packet analyzer understands the format of Ethernet frames, and so can identify the IP datagram within an Ethernet frame. It also understands the IP datagram format, so that it can extract the TCP segment within the IP datagram. Finally, it understands the TCP segment structure, so it can extract the HTTP message contained in the TCP segment. Finally, it understands the HTTP protocol and so, for example, knows that the first bytes of an HTTP message will contain the string "GET," "POST," or "HEAD".

2.0 Getting Wireshark Ready

- Download and install the Wireshark software
- Run Wireshark. Wireshark startup screen shown in Figure A.2.

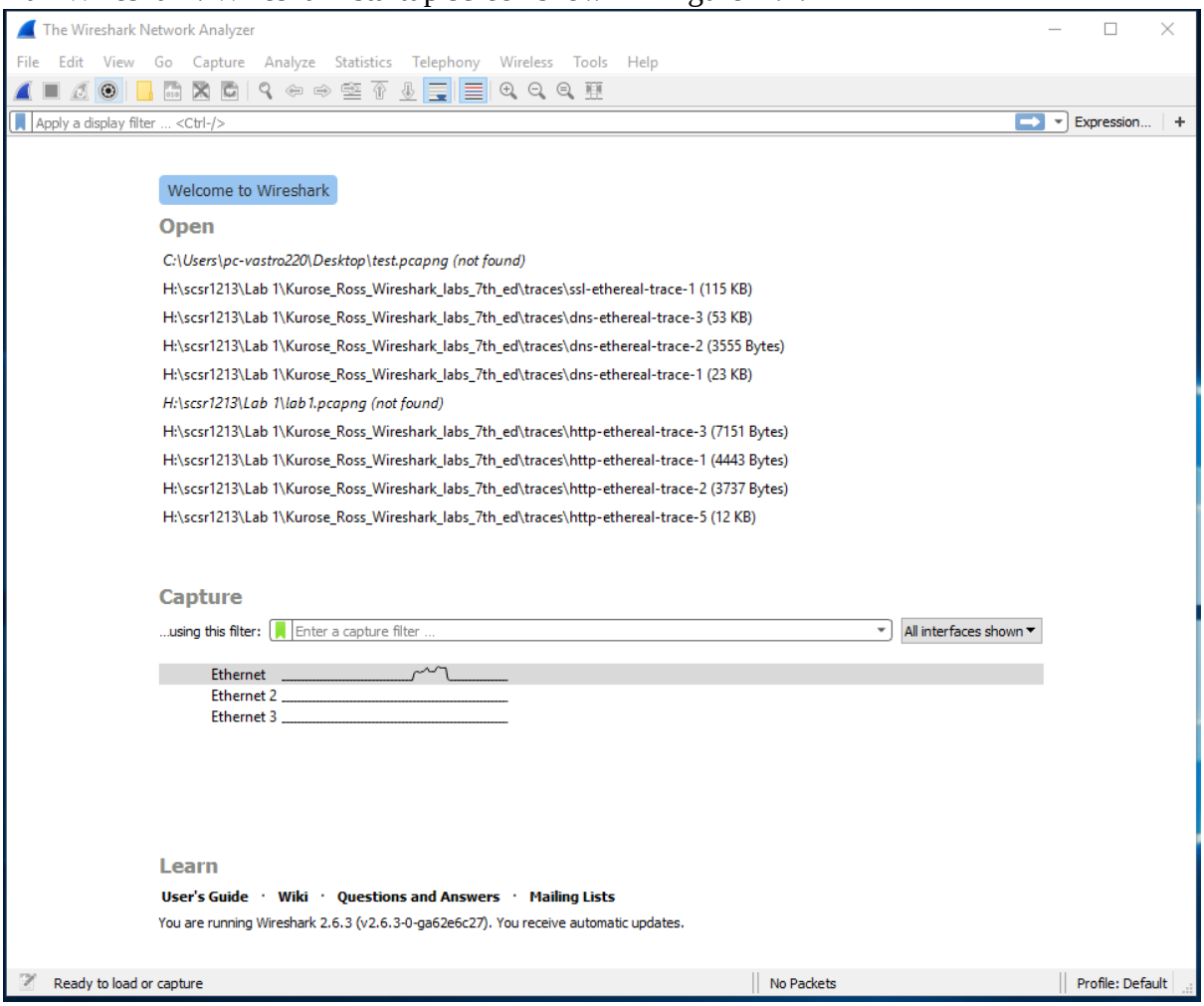


Figure A.2: Initial Wireshark startup screen

- The Wireshark interface has five major components as shown in Figure A.3.

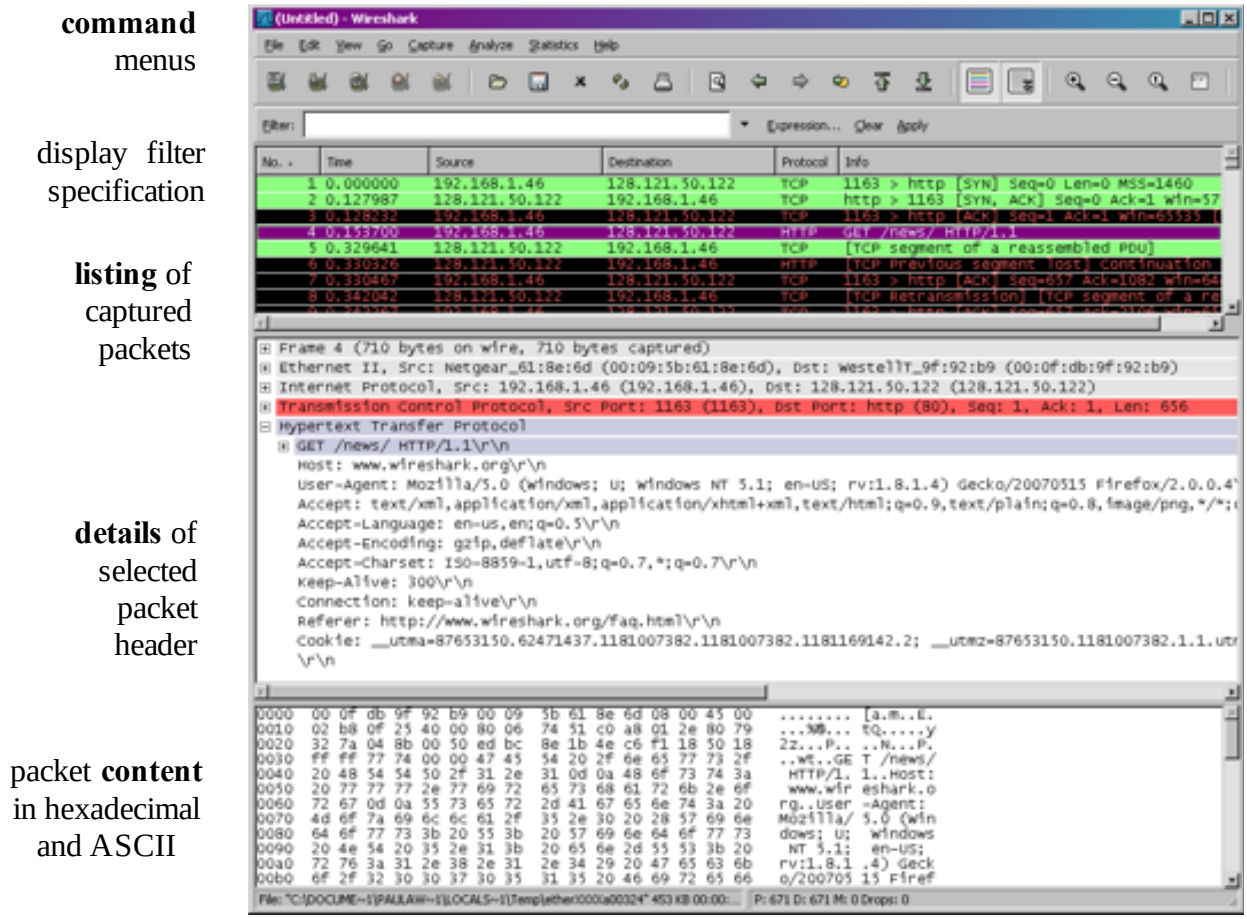


Figure A.3: Wireshark Graphical User Interface, during packet capture and

- The **command menus** are standard pulldown menus located at the top of the window.
- The **packet display filter field,** into which a protocol name or other information can be entered in order to filter the information displayed in the packet-listing window.
- The **packet-listing window** displays a one-line summary for each packet captured, including the packet number, the time at which the packet was captured, the packet's source and destination addresses, the protocol type, and protocol-specific information contained in the packet.
- The **packet-header details window** provides details about the packet selected (highlighted) in the packet-listing window. These details include information about the Ethernet frame and IP datagram that contains this packet. The amount of Ethernet and IP-layer detail displayed can be expanded or minimized by clicking on the plus minus boxes to the left of the Ethernet frame or IP datagram line in the packet details window. If the packet has been carried over TCP or UDP, TCP or UDP details will also be displayed, which can similarly be expanded or minimized. Finally, details about the highest-level protocol that sent or received this packet are also provided.
- The **packet-contents window** displays the entire contents of the captured frame, in both ASCII and hexadecimal format.

3.0 Test Run Wireshark

- Start up the Wireshark software.
- To begin packet capture, select the Capture pull down menu and pick Options menu. Select appropriate interfaces on your compute and click Start button to begin packet capture. Refer to Figure A.4

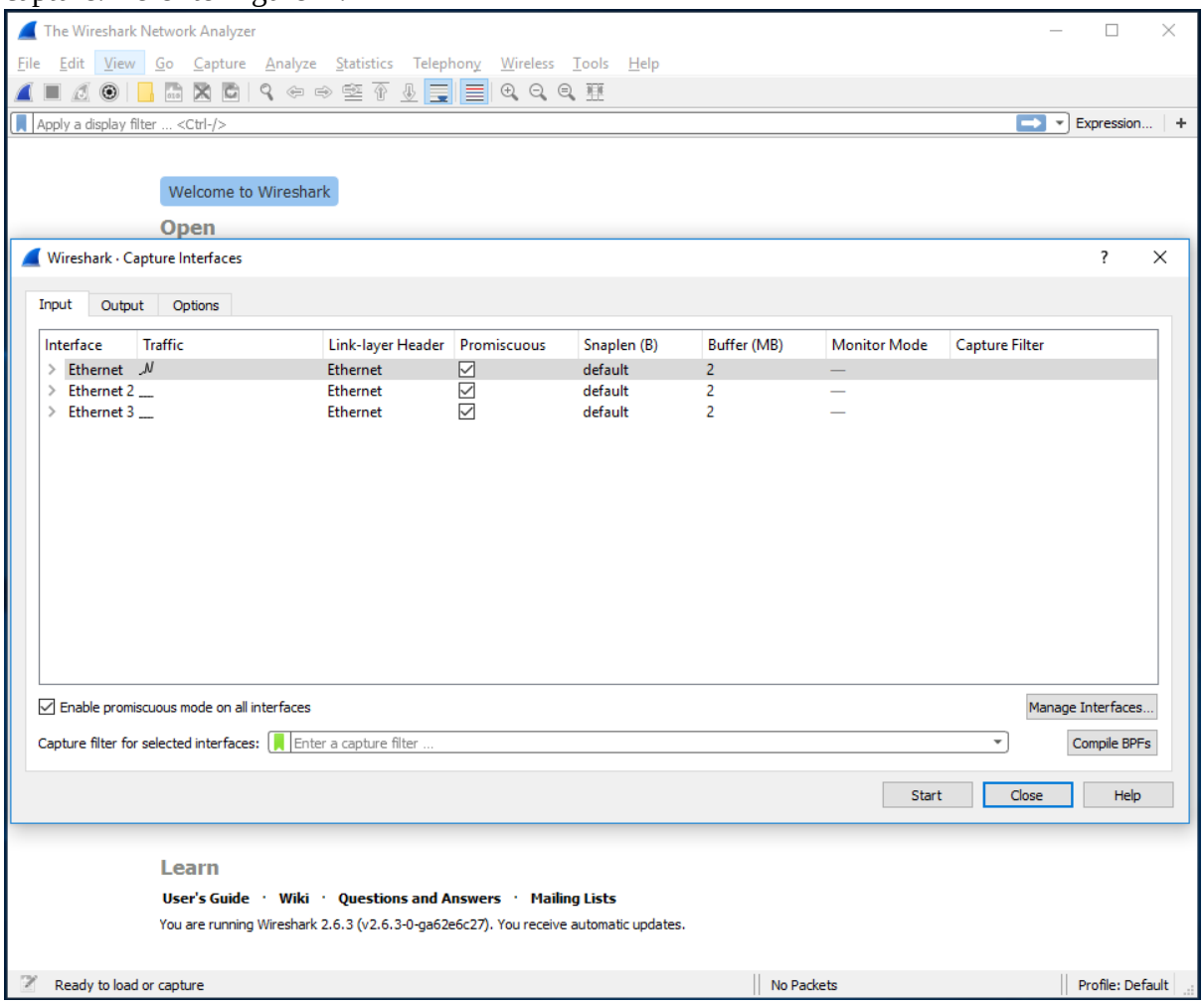


Figure A.4: Capture and Options Menu

- Once you begin packet capture, result will be shown as in Figure A.5.

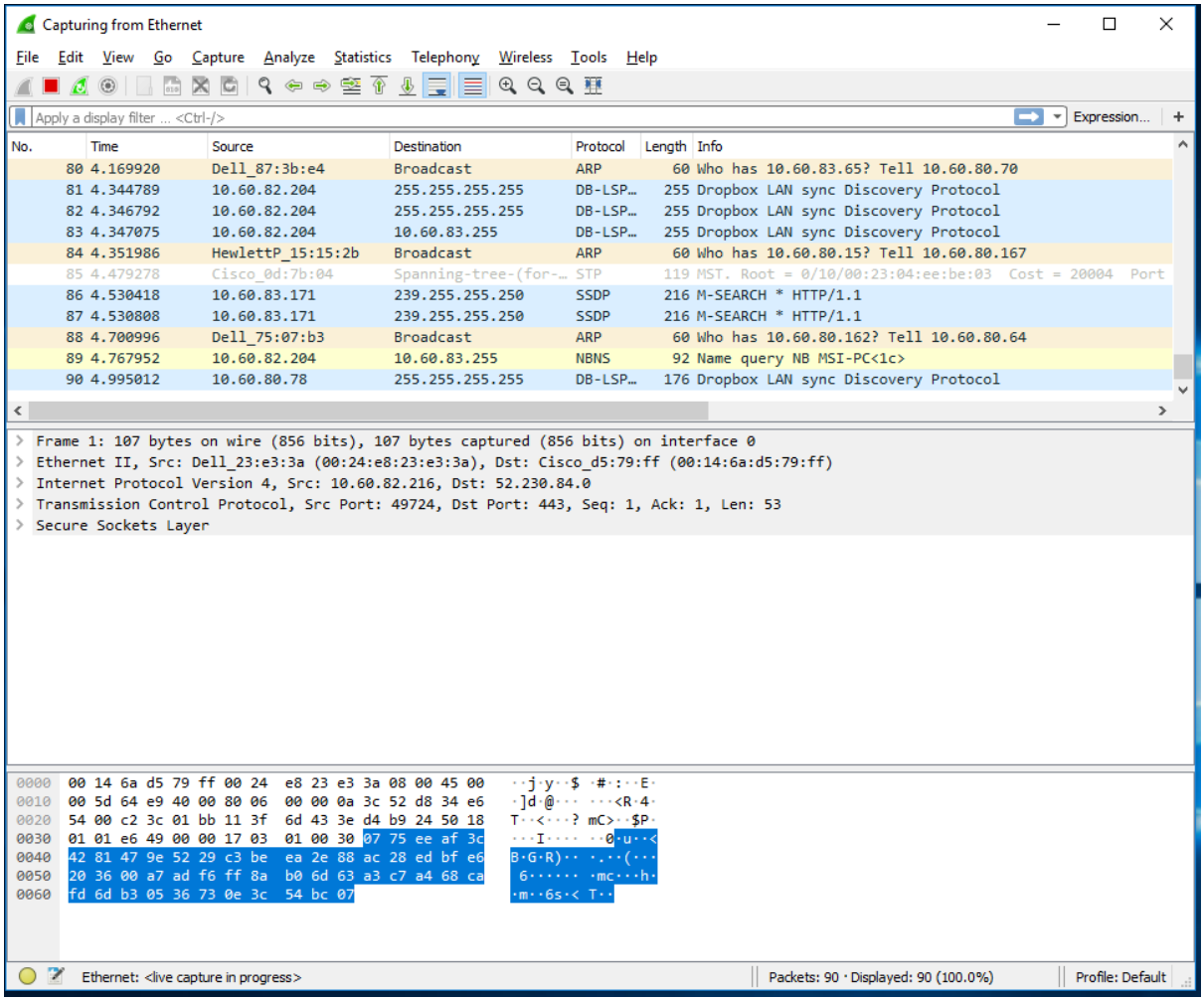


Figure A.5: Wireshark packet capture result

- By selecting Capture pulldown menu and selecting Stop, you can stop packet capture.

- Type “arp” in packet display filter field and press Enter key. This will cause only ARP message to be displayed in the packet-listing window as shown in Figure A.6.

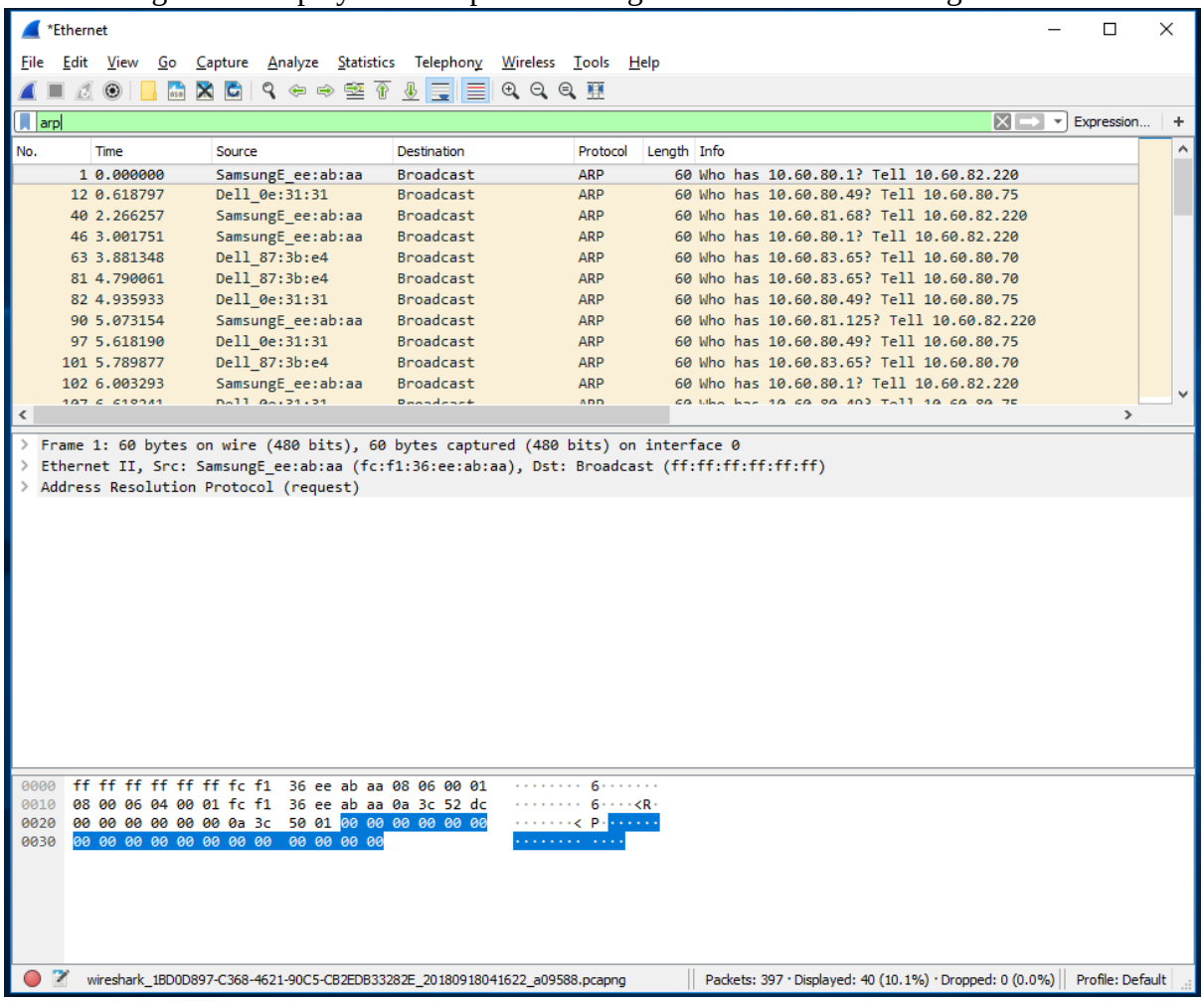


Figure A.6: ARP packet capture

- To save the trace result, use File pulldown menu and select Save function as shown in Figure A.7.

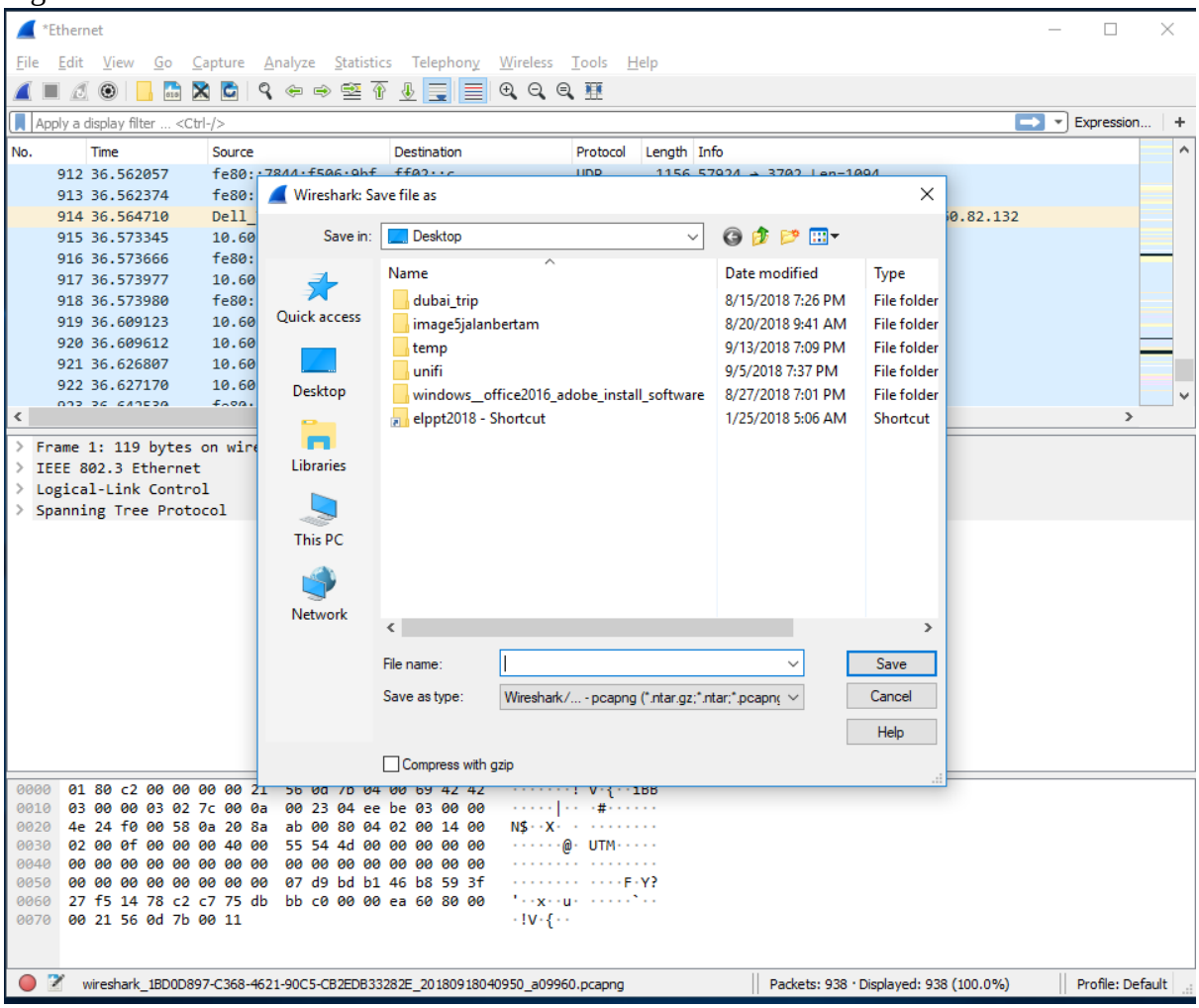


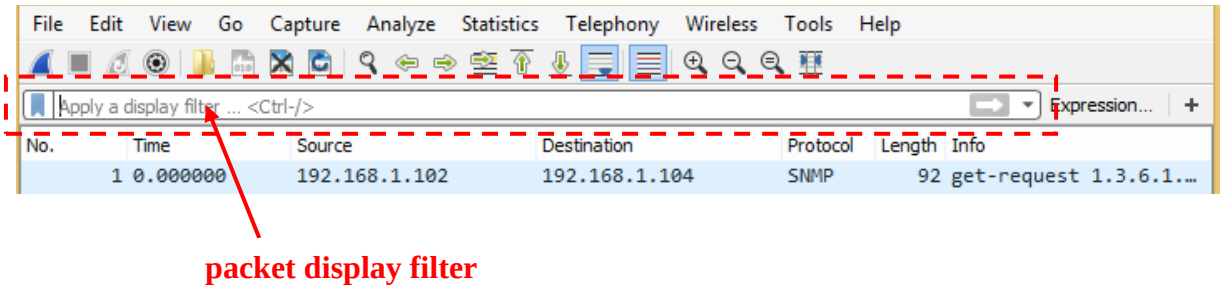
Figure A.7: Save Wireshark trace result

PART B: HTTP Trace

In this part, we'll explore several aspects of the HTTP protocol: the basic GET/response interaction, HTTP message formats and retrieving HTML files with embedded objects. Before beginning these labs, you might want to review Section 2.2 of the textbook.

B.1 The Basic HTTP GET/response interaction

- Open packet trace file **lab1-http-B01.pcapng**.
- Enter "**http**" (just the letters, not the quotation marks) in the **packet display filter field**, so that only captured HTTP messages will be displayed later in the packet-listing window. Refer to figure below:



- By looking at the information in the HTTP GET and response messages, answer the following questions:

1. What version of HTTP is the server running?
   Answer:
   HTTP is running the server of Version 1.1. Based on Figures B.1.1.1 and B.1.1.2, the version displayed in the captured HTTP messages shows that Version 1.1 is the same version being used by all HTTP traffic.

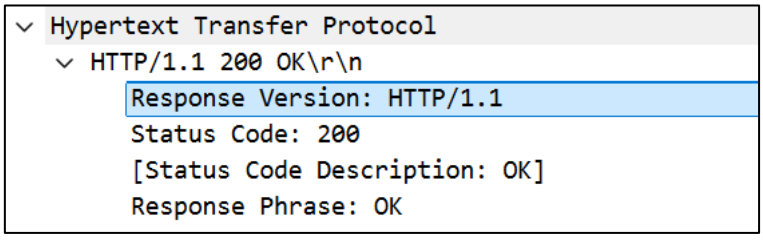


Figure B.1.1.1

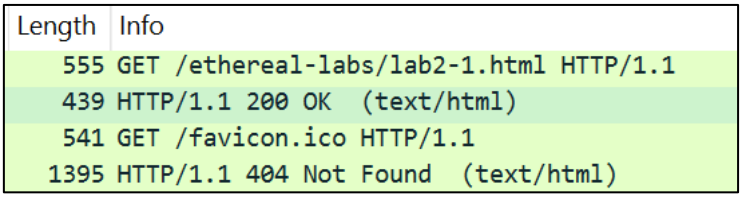


Figure B.1.1.2

2. What is the IP address of the client computer?
   Answer:
   The client computer server's IP address is 192.168.1.102. This is shown in Figure B.1.2.1 and appears as a destination address within the captured HTML messages.

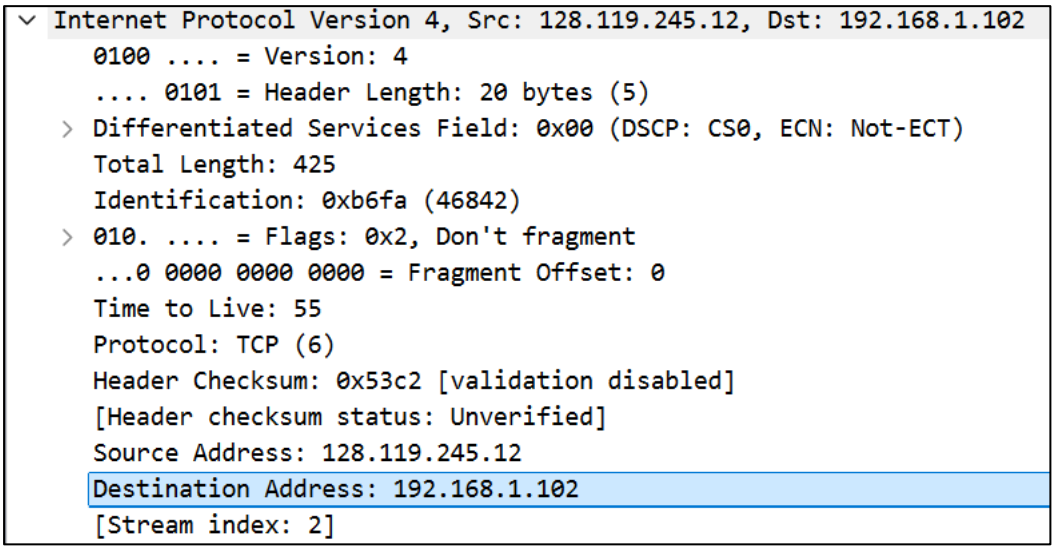


Figure B.1.2.1

3. What is the IP address of the gaia.cs.umass.edu server?
   Answer:
   The gaia.cs.umass.edu server's IP address is 128.119.245.12. This is shown in Figure B.1.3.1 and display as a source address within the captured HTML messages.

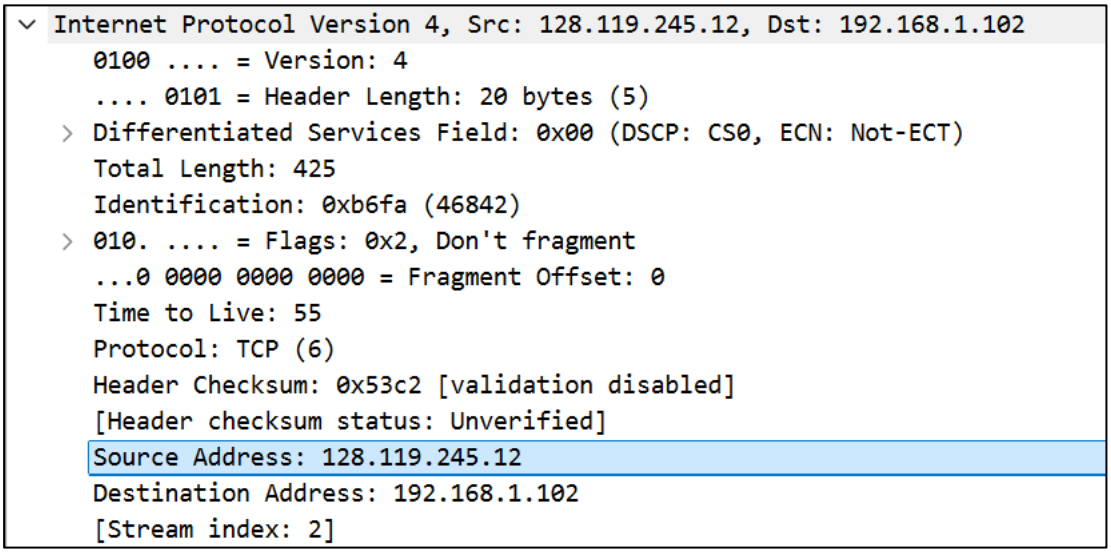


Figure B.1.3.1

4. How many bytes of content are being returned to client browser?
   Answer:

The bytes of content are being returned to client browser is 73 bytes that shown in Figure B.1.4.1.

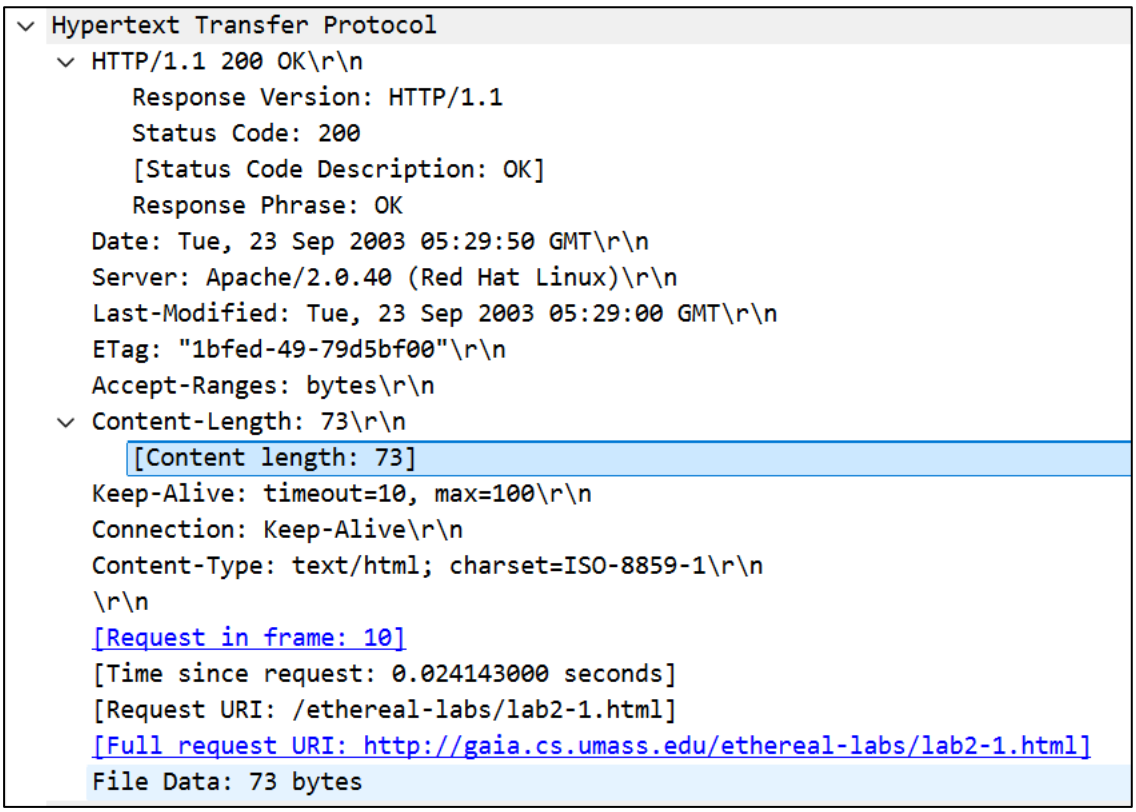


Figure B.1.4.1

5. What is the status code returned from the server to client browser?

   Answer:

   The status code returned from the server to client browser is 200 when request the URL/ethereal-labs/lab2-1.html and the response phrase is "OK" shown in Figure B.1.5.1. However, the status code returned from the server to client browser is 400 and the response phrase is "Not Found" when request URL /favicon.ico shown in Figure B.1.5.2. It is also supported by Figure B.1.5.3, which displays the overall status code and response phrase information for both response messages.

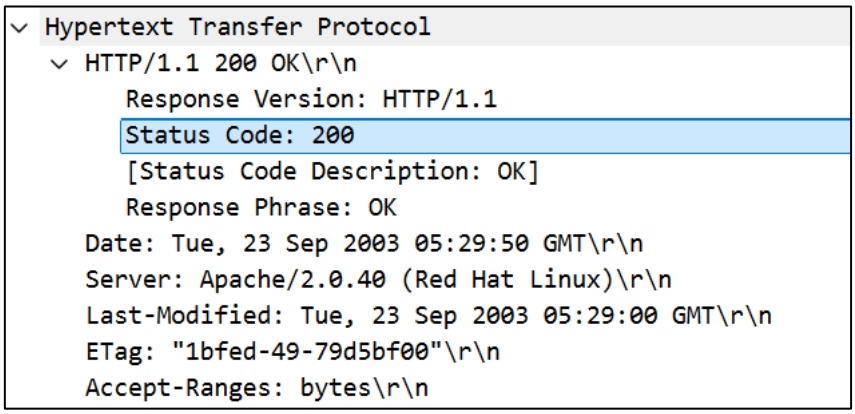


Figure B.1.5.1

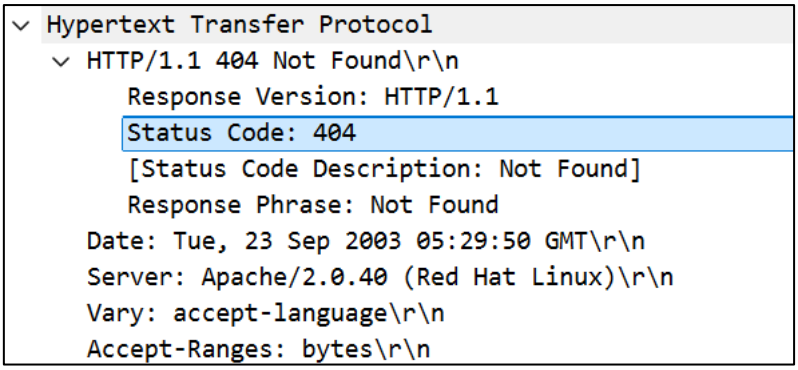


Figure B.1.5.2

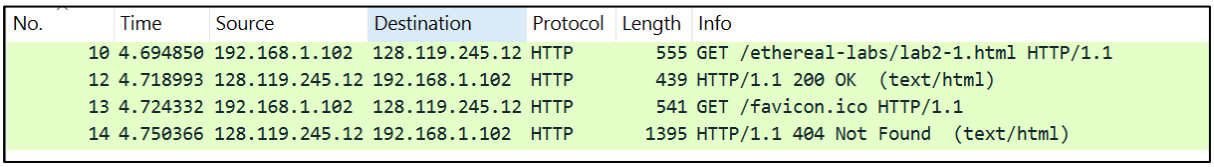


Figure B.1.5.3

B.2 The HTTP CONDITIONAL GET/response interaction

- Open packet trace file **lab1-http-B02.pcapng**.
- By looking at the information in the HTTP GET and response messages, answer the following questions:

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?
   Answer:
   No, there is no "IF-MODIFIED-SINCE" line in the HTTP GET after analyzing the contents of the browser's first HTTP GET request to the server shown in Figure B.2.1.1.

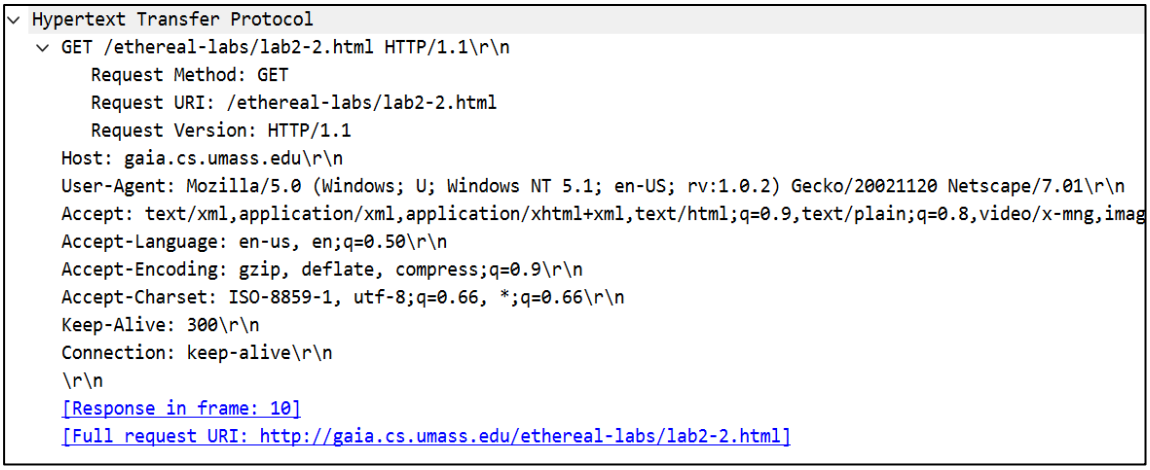


Figure B.2.1.1

2. Inspect the contents of the server response after the first GET request from client. Did the server explicitly return the contents of the file? How can you tell?
   Answer:
   Yes, the server explicitly returns the contents of the file. This is shown by the status code 200 and the response phrase "OK" based on Figure B.2.2.1. It is also supported by Figure B.2.2.2 that shows the contents of the file that return.

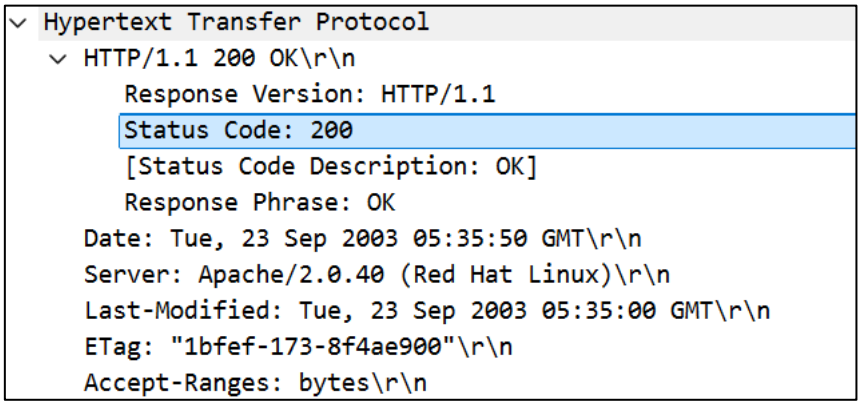


Figure B.2.2.1

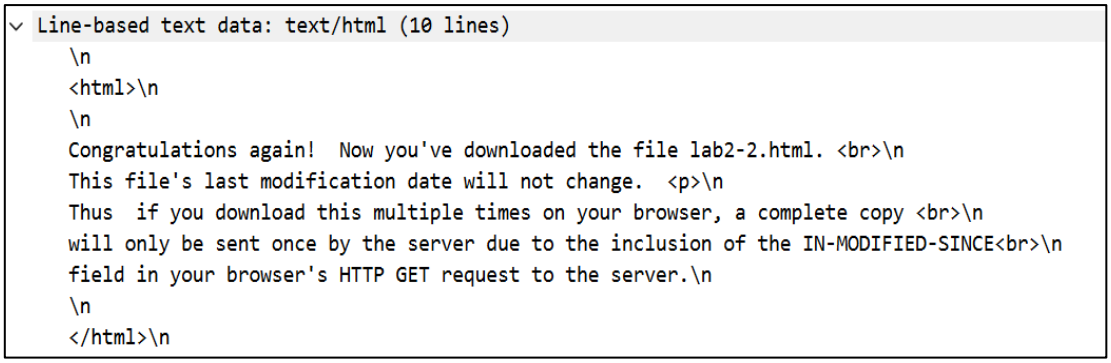


Figure B.2.2.2

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?

   Answer:

   Yes, it has contained the "IF-MODIFIED-SINCE:" line. The following information is "Tue, 23 Sep 2003 05:35:00 GMT\r\n" that shown in Figure B.2.3.1.

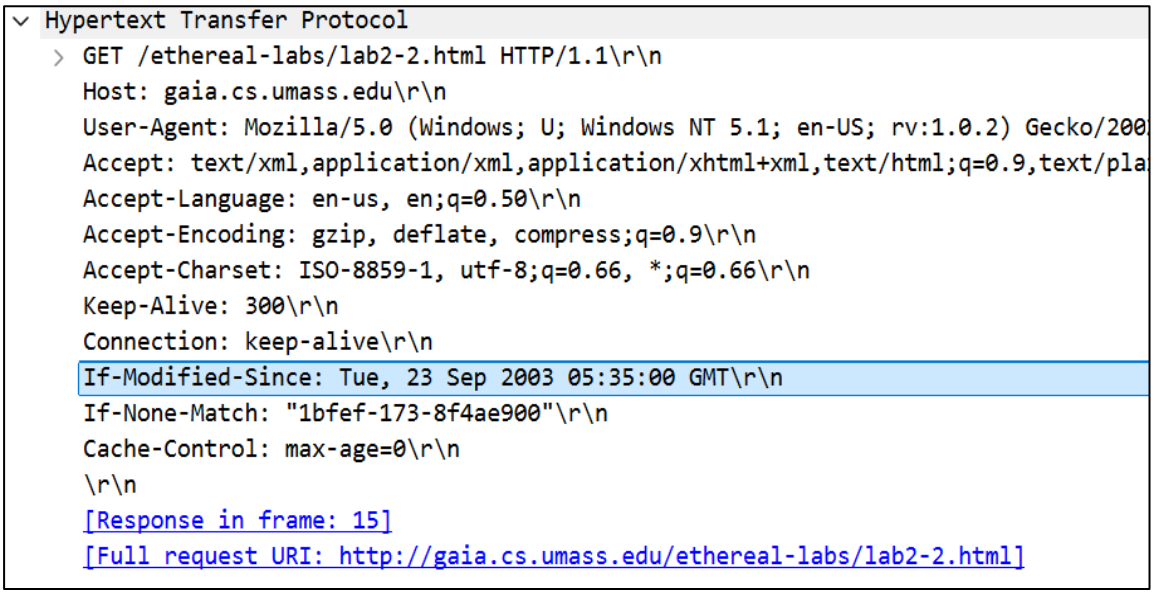


Figure B.2.3.1

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.

   Answer:

   The HTTP status code from the server in response to this second HTTP GET is 304 and the phrase returned is "Not Modified". The server did not return the contents of the file because the server detects the file does not change since last access and send the status code 304 to inform the browser to use the previous content in the cache and show it to the users that shown in Figure B.2.4.1.

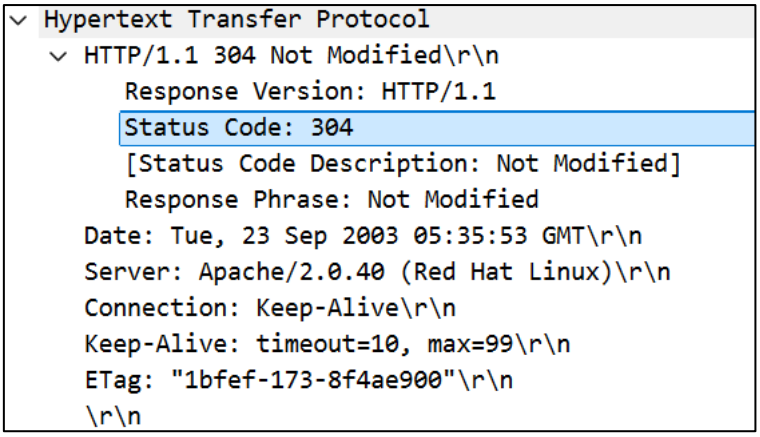
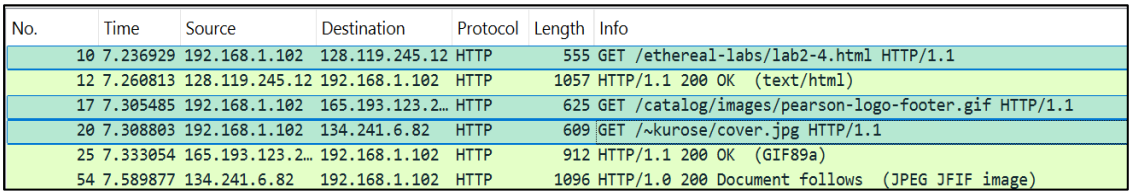

Hypertext Transfer Protocol
HTTP/1.1 304 Not Modified\r\n
Response Version: HTTP/1.1
Status Code: 304
[Status Code Description: Not Modified]
Response Phrase: Not Modified
Date: Tue, 23 Sep 2003 05:35:53 GMT\r\n
Server: Apache/2.0.40 (Red Hat Linux)\r\n
Connection: Keep-Alive\r\n
Keep-Alive: timeout=10, max=99\r\n
ETag: "1bfef-173-8f4ae900"\r\n
\r\n

Figure B.2.4.1

B.3 HTML Documents with Embedded Objects

- Open packet trace file **lab1-http-B03.pcapng**.
- By looking at the information in the HTTP GET and response messages, answer the following questions:

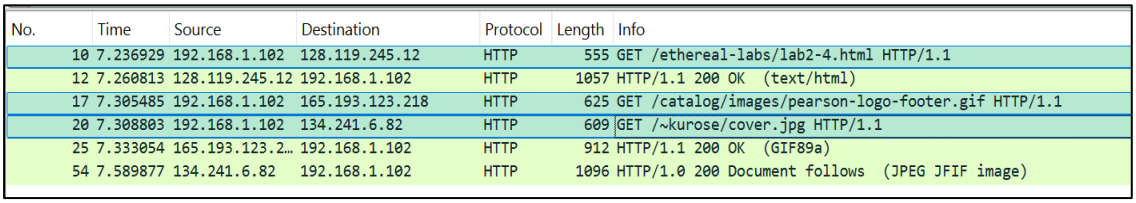
1. How many HTTP GET request messages did client browser send?
   Answer:
   There are 3 HTTP GET request messages sent by the client browser that are shown in Figure B.3.1.1.



No.	Time	Source	Destination	Protocol	Length	Info
10	7.236929	192.168.1.102	128.119.245.12	HTTP	555	GET /ethereal-labs/lab2-4.html HTTP/1.1
12	7.260813	128.119.245.12	192.168.1.102	HTTP	1057	HTTP/1.1 200 OK (text/html)
17	7.305485	192.168.1.102	165.193.123.2…	HTTP	625	GET /catalog/images/pearson-logo-footer.gif HTTP/1.1
20	7.308803	192.168.1.102	134.241.6.82	HTTP	609	GET /~kurose/cover.jpg HTTP/1.1
25	7.333054	165.193.123.2…	192.168.1.102	HTTP	912	HTTP/1.1 200 OK (GIF89a)
54	7.589877	134.241.6.82	192.168.1.102	HTTP	1096	HTTP/1.0 200 Document follows (JPEG JFIF image)

Figure B.3.1.1

2. To which Internet addresses were these GET requests sent?
   Answer:
   These GET requests sent to the internet addresses 128.119.245.12, 165.193.123.218 and 134.241.6.82 that shown in Figure B.3.2.1.



No.	Time	Source	Destination	Protocol	Length	Info
10	7.236929	192.168.1.102	128.119.245.12	HTTP	555	GET /ethereal-labs/lab2-4.html HTTP/1.1
12	7.260813	128.119.245.12	192.168.1.102	HTTP	1057	HTTP/1.1 200 OK (text/html)
17	7.305485	192.168.1.102	165.193.123.218	HTTP	625	GET /catalog/images/pearson-logo-footer.gif HTTP/1.1
20	7.308803	192.168.1.102	134.241.6.82	HTTP	609	GET /~kurose/cover.jpg HTTP/1.1
25	7.333054	165.193.123.2…	192.168.1.102	HTTP	912	HTTP/1.1 200 OK (GIF89a)
54	7.589877	134.241.6.82	192.168.1.102	HTTP	1096	HTTP/1.0 200 Document follows (JPEG JFIF image)

Figure B.3.2.1

3. any bytes of content are being returned to client browser for the **pearson-logo-footer.gif** image file?
   Answer:
   The bytes of content are being returned to client browser for the pearson-logo-footer.gif image file is 3357 bytes shown in Figure B.3.3.1.

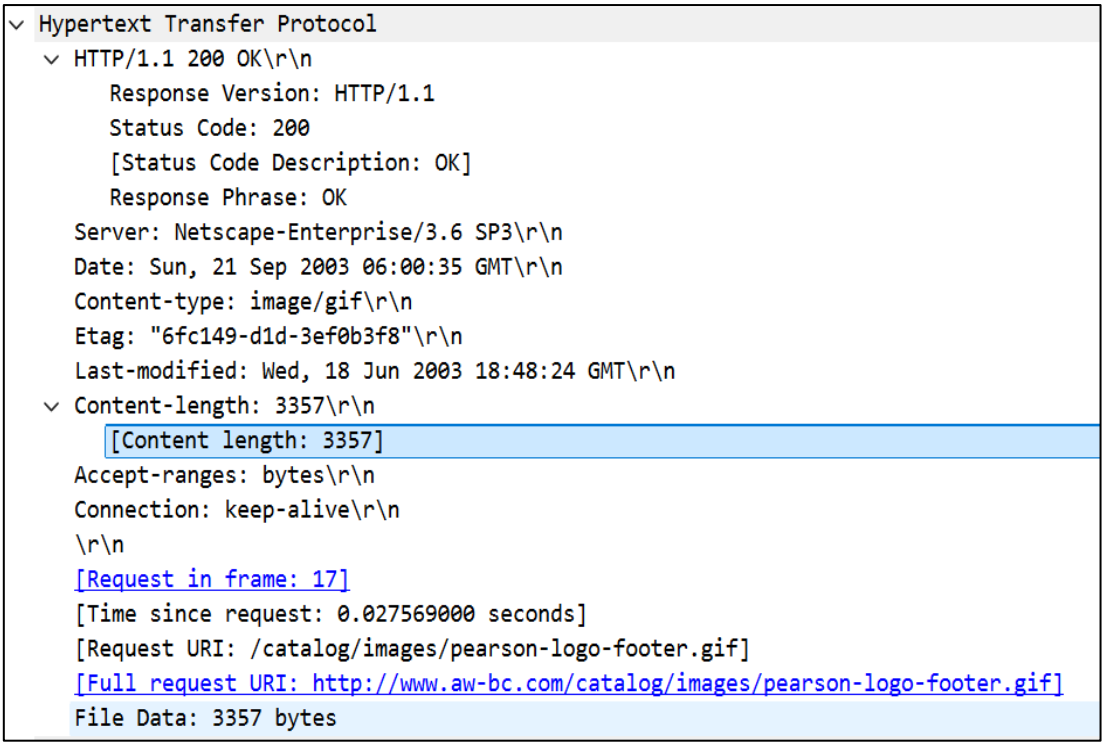


Figure B.3.3.1.

4. How many bytes of content are being returned to client browser for the **cover.jpg** image file?

   Answer:

   The bytes of content are being returned to client browser for the cover.jpg image file is 15642 bytes shown in Figure B.3.4.1.

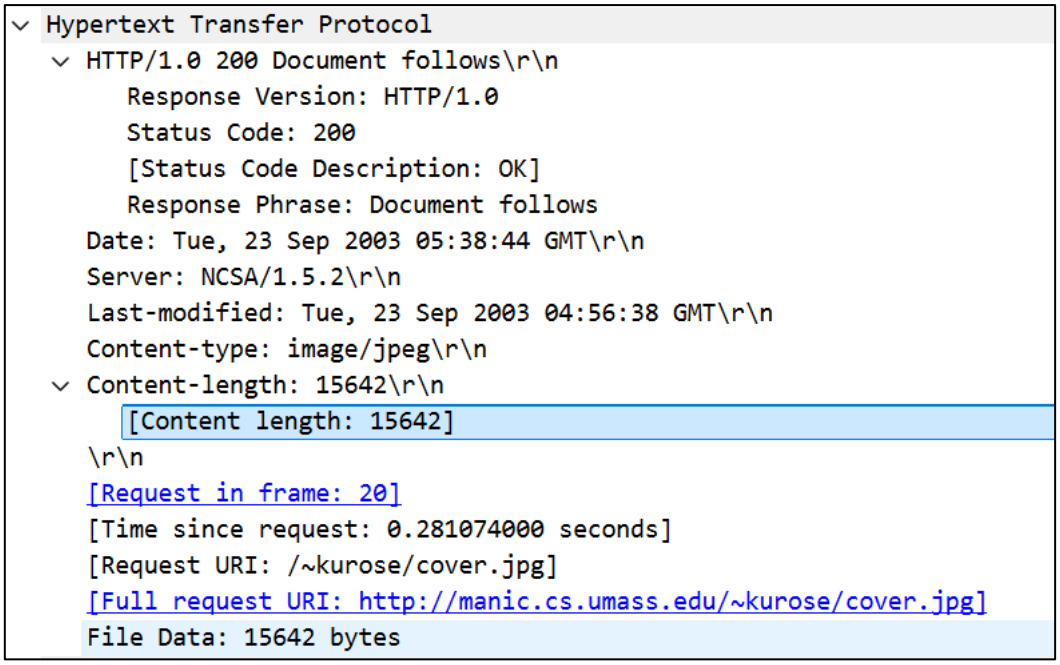


Figure B.3.4.1

PART C: DNS Trace

1.0 nslookup

nslookup tool allows the host running the tool to query any specified DNS server for a DNS record. The queried DNS server can be a root DNS server, a top-level-domain DNS server, an authoritative DNS server, or an intermediate DNS server. To accomplish this task, nslookup sends a DNS query to the specified DNS server, receives a DNS reply from that same DNS server, and displays the result.

- To run it in Windows, open the Command Prompt (cmd) and run nslookup on the command line as shown in Figure C.1 and Figure C.2

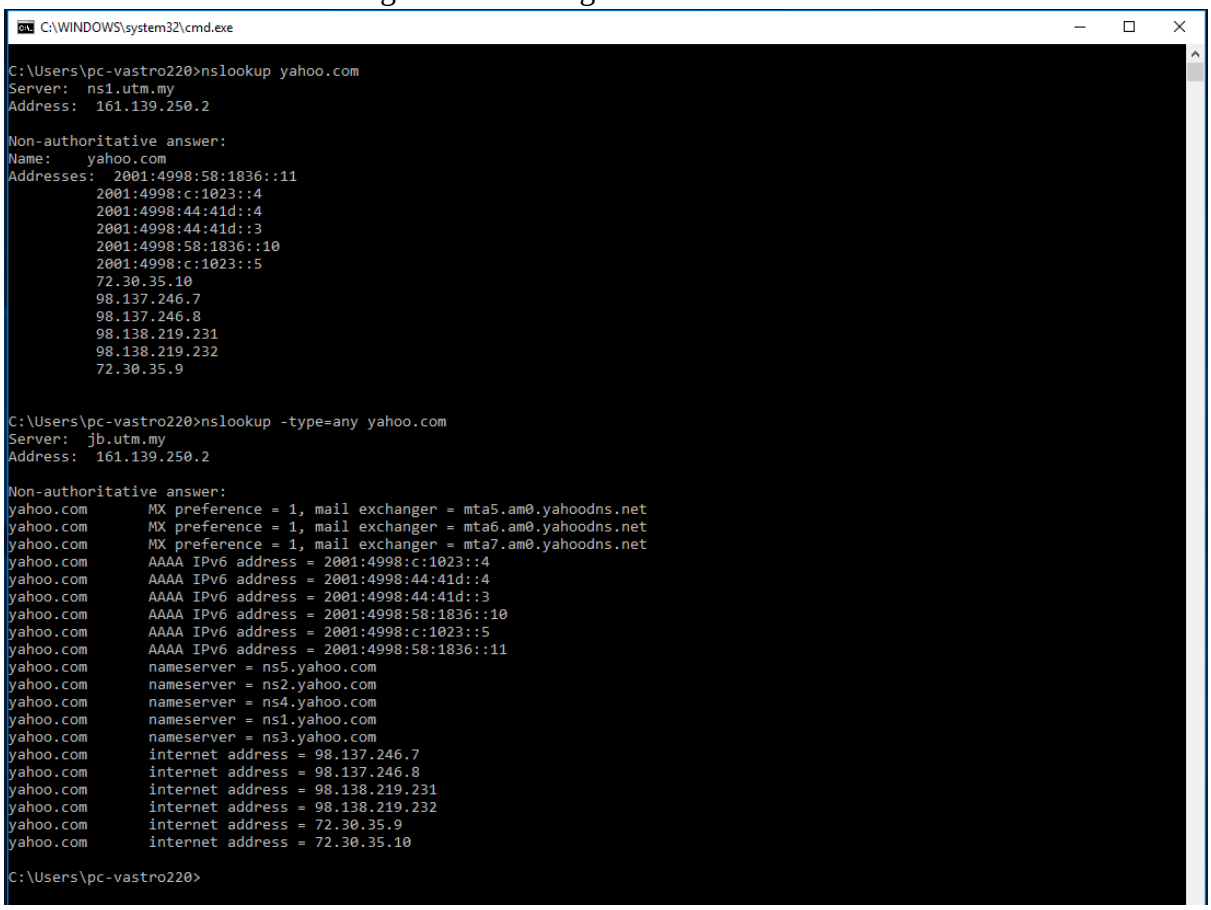


Figure C.1: nslookup result

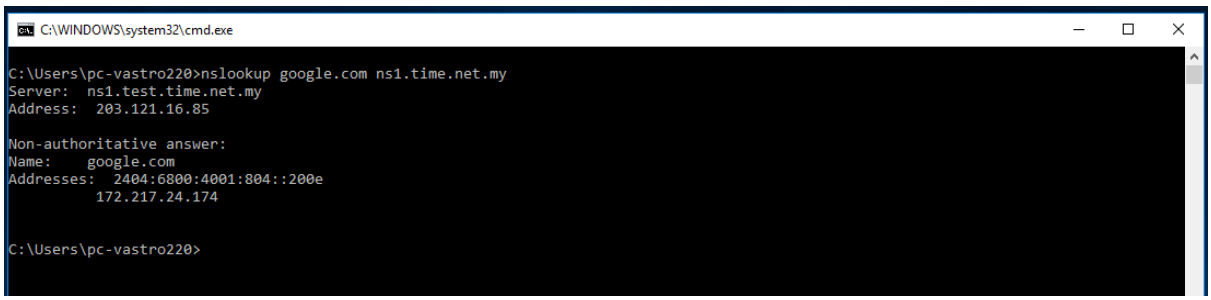


Figure C.2: nslookup result

1. Run nslookup to obtain the IP address of a www.microsoft.com server. What is the IP address of that server? Add screenshot to your answer.
   Answer:
   The IP address of the www.microsoft.com server is 23.0.222.32, shown in Figure C.1.1.

```
C:\Users\Teh>nslookup www.microsoft.com
Server:  ns3.utm.my
Address:  161.139.168.168

Non-authoritative answer:
Name:    e13678.dscb.akamaiedge.net
Addresses:  2600:1411:2000:39e::356e
          2600:1411:2000:38a::356e
          23.0.222.32
Aliases:  www.microsoft.com
          www.microsoft.com-c-3.edgekey.net
          www.microsoft.com-c-3.edgekey.net.globalredir.akadns.net
```

Figure C.1.1

2. Run nslookup to determine the non-authoritative DNS servers for domain microsoft.com. Add screenshot to your answer.
   Answer:
   The non-authoritative DNS servers for domain microsoft.com are 20.70.246.20, 20.236.44.162, 20.231.239.246, 20.76.201.171 and 20.112.250.133 shown in Figure C.1.2.

```
C:\Users\Teh>nslookup microsoft.com
Server:  ns3.utm.my
Address:  161.139.168.168

Non-authoritative answer:
Name:    microsoft.com
Addresses:  2603:1030:20e:3::23c
          2603:1020:201:10::10f
          2603:1010:3:3::5b
          2603:1030:b:3::152
          2603:1030:c02:8::14
          20.70.246.20
          20.236.44.162
          20.231.239.246
          20.76.201.171
          20.112.250.133
```

Figure C.1.2

2.0 ipconfig

ipconfig can be used to show your current TCP/IP information, including your address, DNS server addresses, adapter type and so on.

- Information about host, use the following command: ipconfig /all

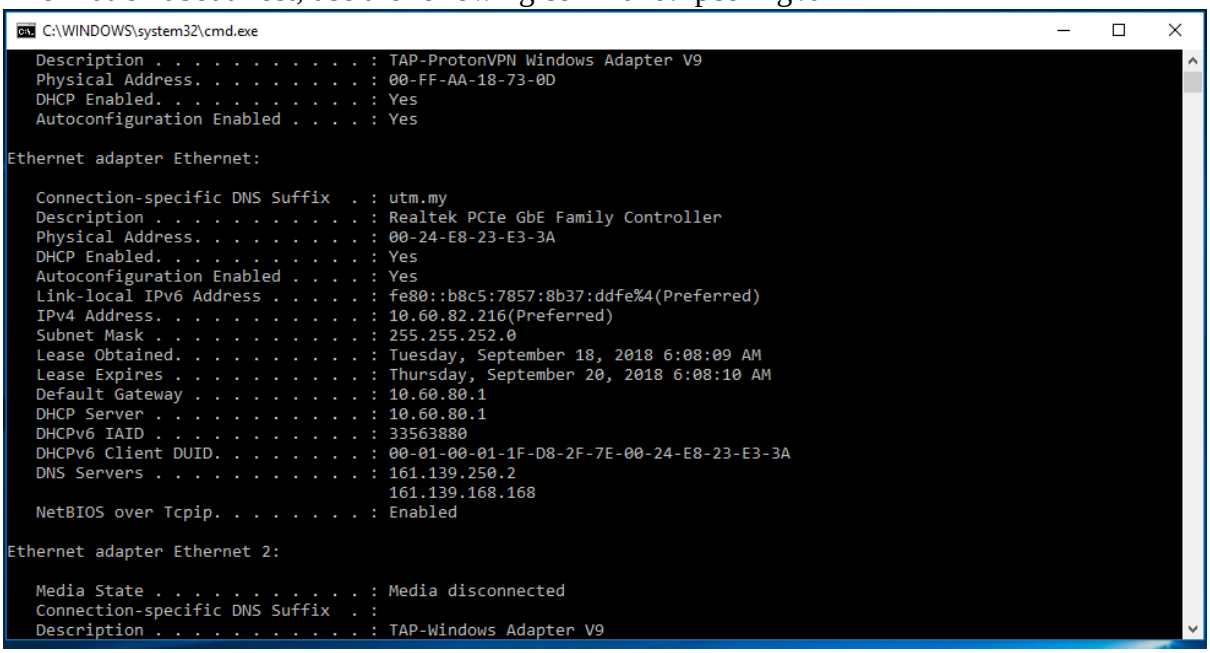


Figure C.3: ipconfig /all result

- ipconfig is also very useful for managing the DNS information stored in your host. Each entry shows the remaining Time to Live (TTL) in seconds.
  Command: ipconfig /displaydns

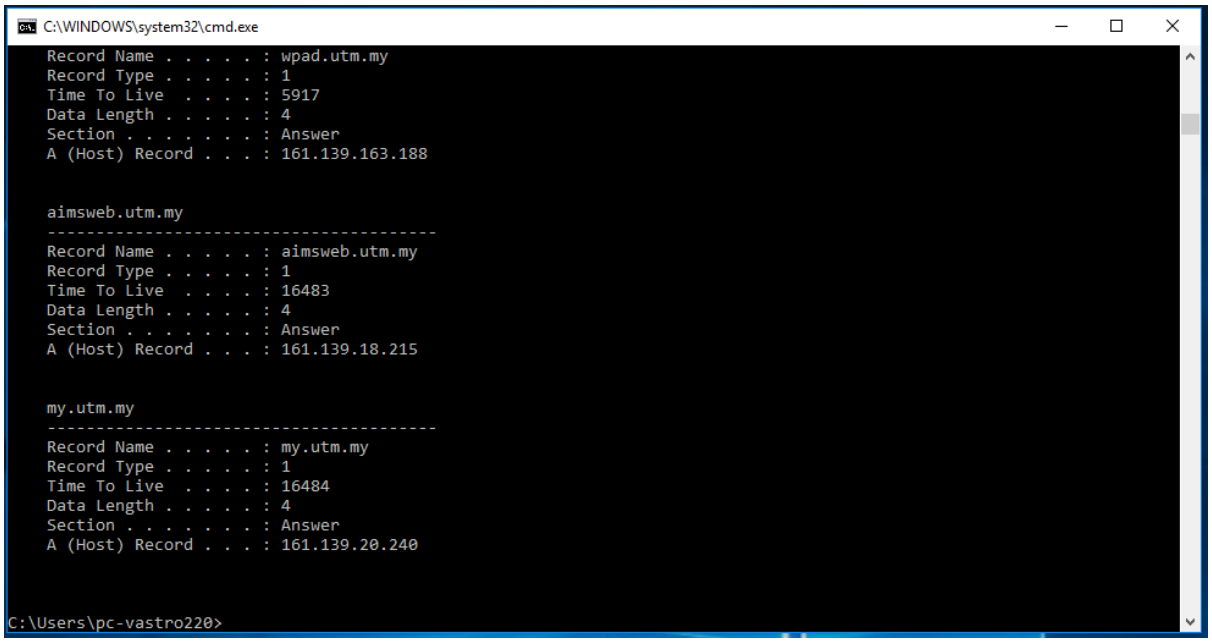


Figure C.4: ipconfig /displaydns result

- Flushing the DNS cache clears all entries and reloads the entries from the hosts file.

Command: ipconfig /flushdns

Figure C.5: ipconfig /flushdns result

3.0 Tracing DNS with Wireshark

- Open packet trace file dns-trace-1. Answer the following questions.

1. Locate the DNS query and response messages. Are then sent over UDP or TCP? Add screenshots in your answer.

   Answer:

   The DNS query and response messages that are located on the line number 8 and 9 are sent over User Datagram Protocol (UDP) as shown in Figure C.3.1.1. Besides, Figure C.3.1.2 and C.3.1.3 proving that DNS query and response messages on the line number 8 and 9 are sent over UDP.

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	Cisco_fc:f0:de	Spanning-tree…	STP	60	Conf. Root = 32768/0/00:01:96:45:05:9a Cost = 12 Port = 0x802d
2	0.148791	00000004.0001…	00000004.ffff…	IPX SAP	113	General Response
3	0.374081	Cisco_83:e4:54	Broadcast	ARP	60	Who has 128.238.38.248? Tell 128.238.38.2
4	1.981736	00000004.0001…	00000004.ffff…	IPX SAP	113	General Response
5	1.999786	Cisco_fc:f0:de	Spanning-tree…	STP	60	Conf. Root = 32768/0/00:01:96:45:05:9a Cost = 12 Port = 0x802d
6	2.031956	128.238.38.2	224.0.0.2	HSRP	62	Hello (state Active)
7	2.527474	Cisco_83:e4:54	Broadcast	ARP	60	Who has 128.238.38.38? Tell 128.238.38.2
8	3.075845	128.238.38.160	128.238.29.23	DNS	72	Standard query 0x006e A www.ietf.org
9	3.076689	128.238.29.23	128.238.38.160	DNS	104	Standard query response 0x006e A www.ietf.org A 132.151.6.75 A 65.246.255.51
10	3.078479	128.238.38.160	132.151.6.75	TCP	62	3369 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM
11	3.096413	132.151.6.75	128.238.38.160	TCP	62	80 → 3369 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1380 SACK_PERM
12	3.096463	128.238.38.160	132.151.6.75	TCP	54	3369 → 80 [ACK] Seq=1 Ack=1 Win=64860 Len=0

Figure C.3.1.1

```
User Datagram Protocol, Src Port: 3163, Dst Port: 53
    Source Port: 3163
    Destination Port: 53
    Length: 38
    Checksum: 0x8acb [unverified]
    [Checksum Status: Unverified]
    [Stream index: 1]
    [Stream Packet Number: 1]
  > [Timestamps]
    UDP payload (30 bytes)
```

Figure C.3.1.2

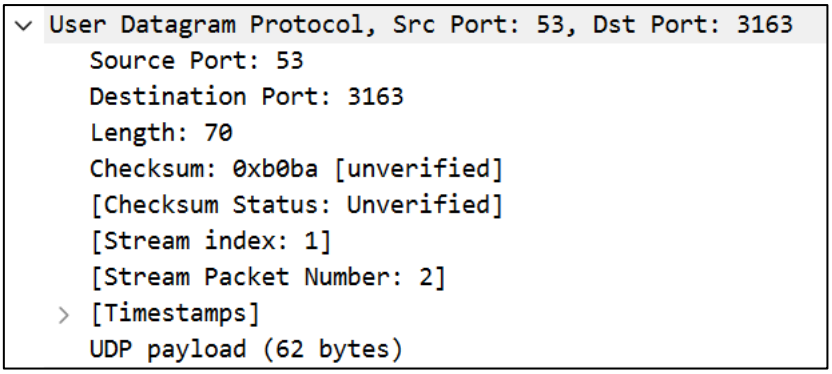


Figure C.3.1.3

2. What is the destination port for the DNS query message? What is the source port of DNS response message? Add screenshots in your answer.

   Answer:

   The destination port for the DNS query message is 53 that shown in Figure C.3.2.1. The source port of DNS response message is 53 that shown in Figure C.3.2.2.

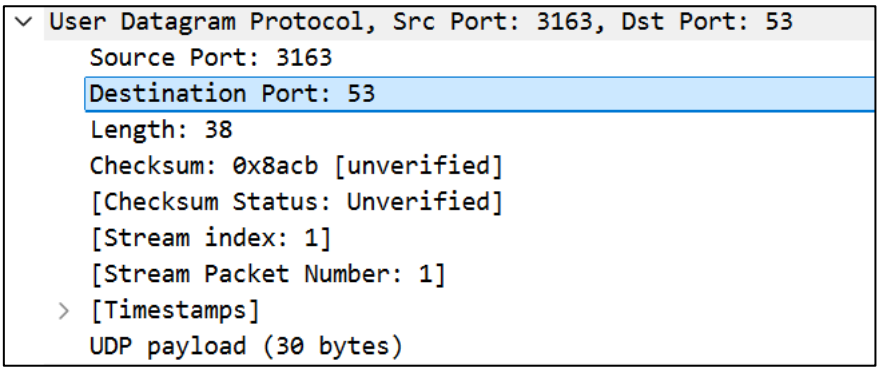


Figure C.3.2.1

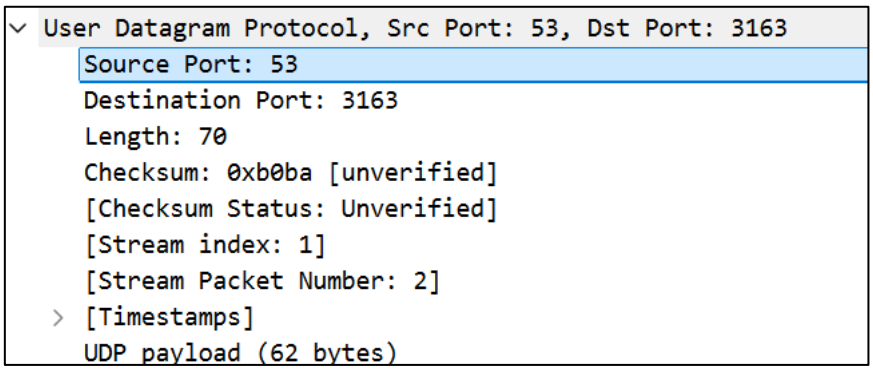


Figure C.3.2.2

3. To what IP address is the DNS query message sent? Add screenshots in your answer.
   Answer:
   The DNS query message is sent to IP address 128.238.29.23 that is shown in Figure C.3.3.1 and Figure C.3.3.2.

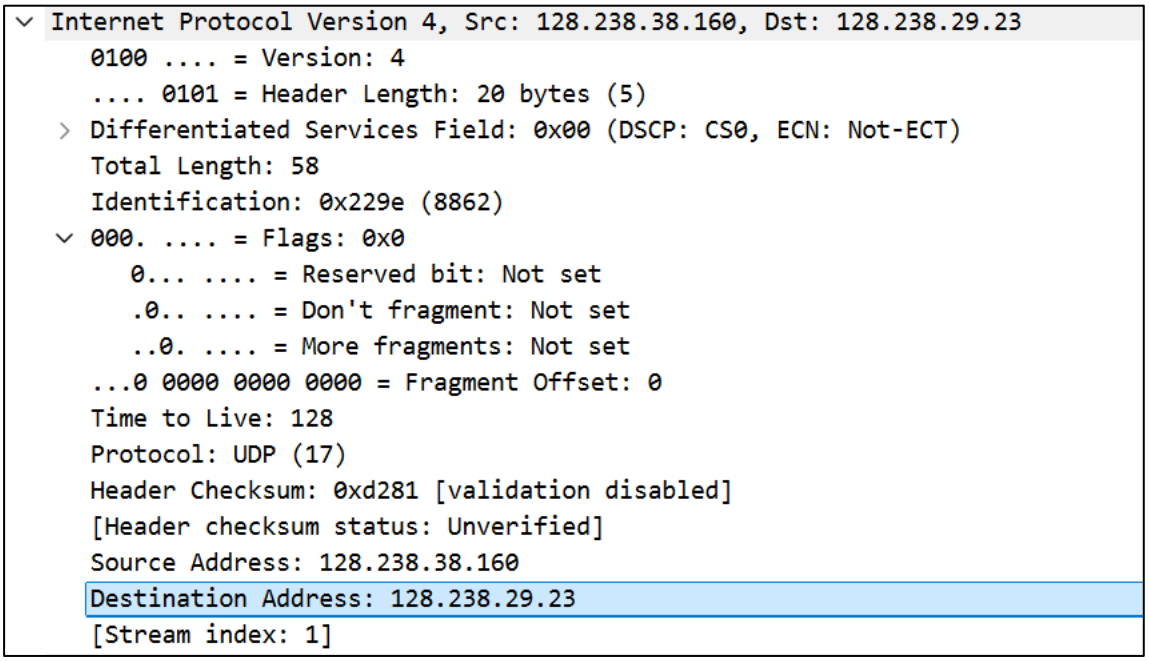


Figure C.3.3.1

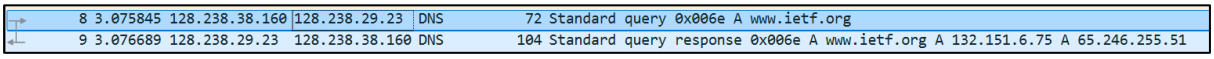


Figure C.3.3.2

4. Examine the DNS query message. What "Type" of DNS query is it? Does the query message contain any "answers"? Add screenshots in your answer.
   Answer:
   The "Type" of DNS query is Type A and query message does not contain any "answers" that shown in Figure C.3.4.1.

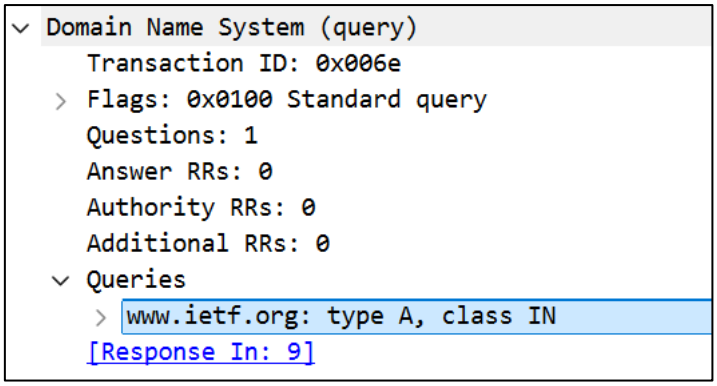


Figure C.3.4.1

5. Examine the DNS response message. How many “answers” are provided? What do each of these answers contain? Add screenshots in your answer.
   Answer:
   The DNS response message is provided 2 “answers”. Each of these answers contains the domain name, type of the address, the class, time to live and data length that is shown in Figure C.3.5.1.

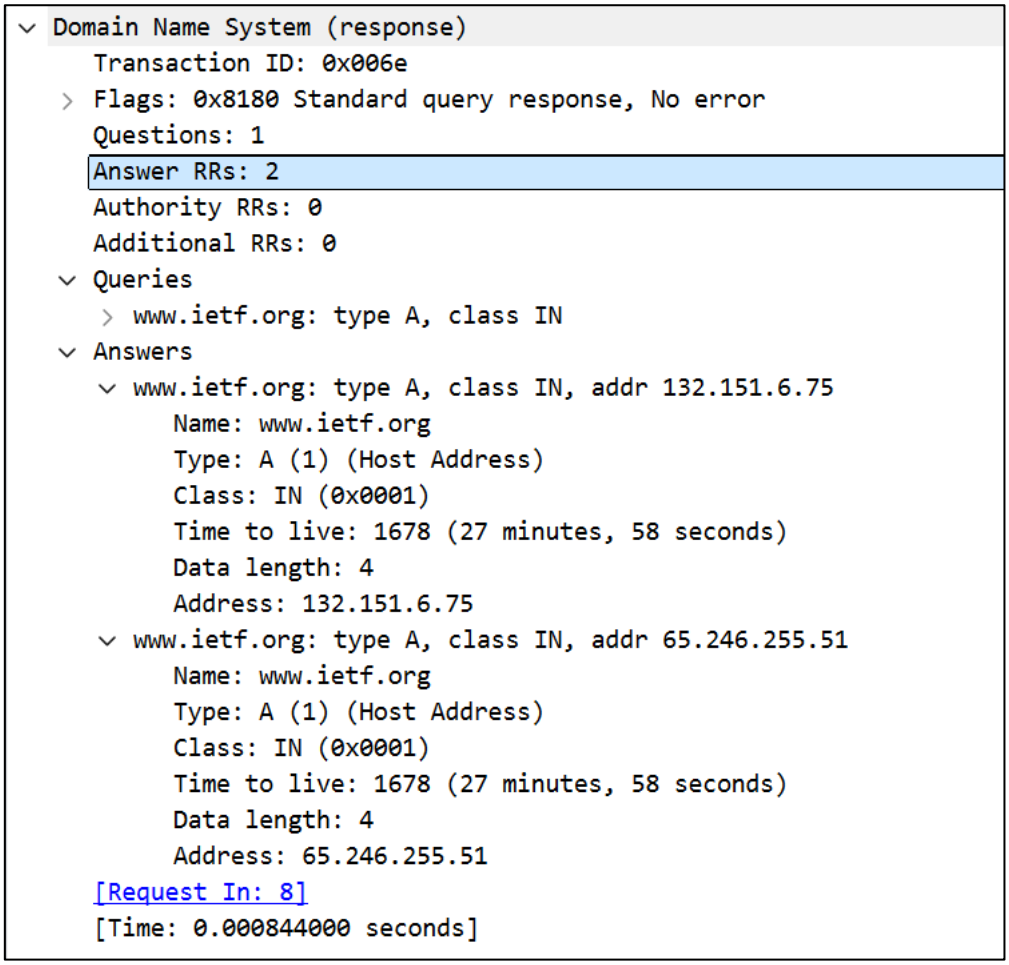


Figure C.3.5.1

6. 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? Add screenshots in your answer.
   Answer:
   The destination IP address of the SYN packet is corresponded to the IP addresses provided in the DNS response message which is 132.151.6.75 that shown in Figure C.3.6.1 and Figure C.3.6.2.

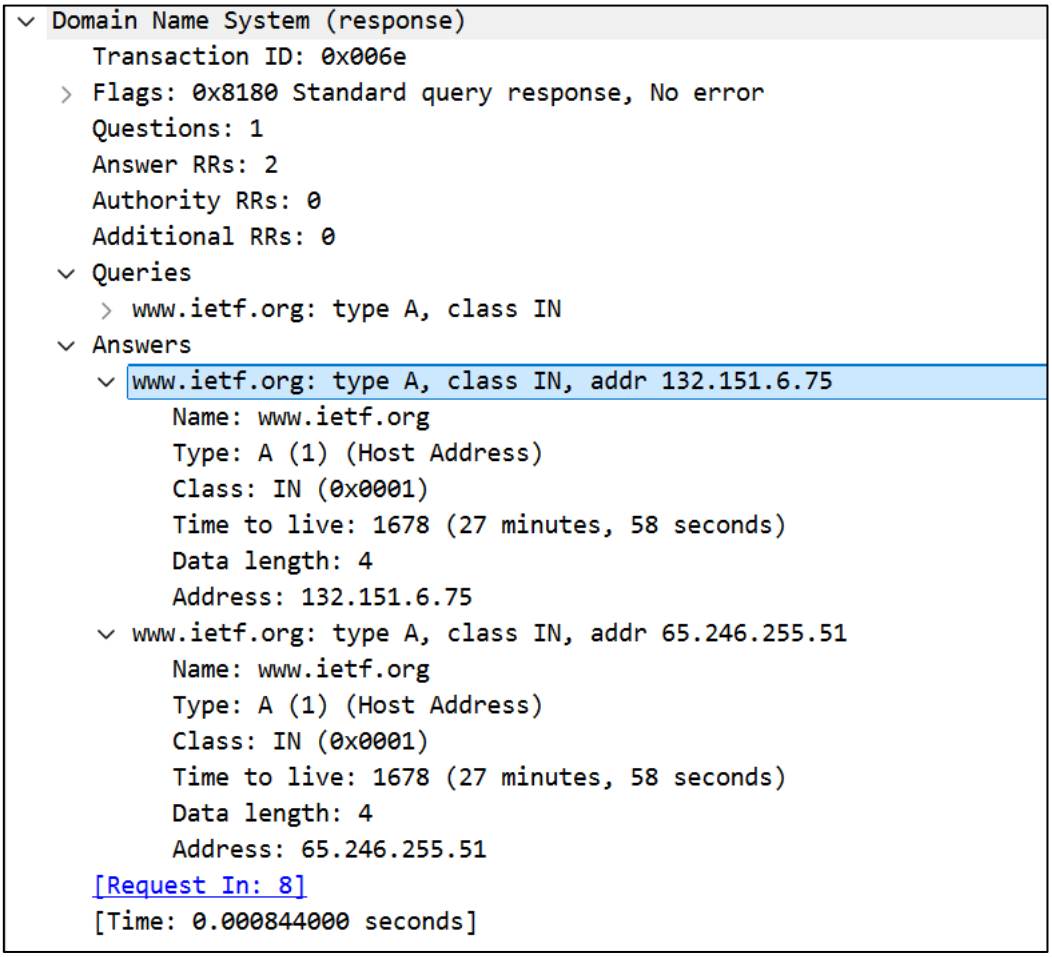


Figure C.3.6.1

```
Internet Protocol Version 4, Src: 128.238.38.160, Dst: 132.151.6.75
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 48
    Identification: 0x229f (8863)
  v 010. .... = Flags: 0x2, Don't fragment
      0... .... = Reserved bit: Not set
      .1.. .... = Don't fragment: Set
      ..0. .... = More fragments: Not set
    ...0 0000 0000 0000 = Fragment Offset: 0
    Time to Live: 128
    Protocol: TCP (6)
    Header Checksum: 0xa5b8 [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 128.238.38.160
    Destination Address: 132.151.6.75
    [Stream index: 2]
```

Figure C.3.6.2

7. This web page contains images. Before retrieving each image, does your host issue new DNS queries?
   Answer:
   No, the host does not issue new DNS queries before retrieving each image. In Figure C.3.7.1, there is no new DNS queries issue by host between line number 10 and number 27 before retrieving image in line number 28.

No.	Time	Source	Destination	Protocol	Length	Info
10	3.078479	128.238.38.160	132.151.6.75	TCP	62	3369 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM
11	3.096413	132.151.6.75	128.238.38.160	TCP	62	80 → 3369 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1380 SACK_PERM
12	3.096463	128.238.38.160	132.151.6.75	TCP	54	3369 → 80 [ACK] Seq=1 Ack=1 Win=64860 Len=0
13	3.096708	128.238.38.160	132.151.6.75	HTTP	429	GET / HTTP/1.1
14	3.111678	132.151.6.75	128.238.38.160	TCP	60	80 → 3369 [ACK] Seq=1 Ack=376 Win=6432 Len=0
15	3.120640	132.151.6.75	128.238.38.160	TCP	1434	80 → 3369 [ACK] Seq=1 Ack=376 Win=6432 Len=1380 [TCP PDU reassembled in 20]
16	3.128093	132.151.6.75	128.238.38.160	TCP	1434	80 → 3369 [ACK] Seq=1381 Ack=376 Win=6432 Len=1380 [TCP PDU reassembled in 20]
17	3.128148	128.238.38.160	132.151.6.75	TCP	54	3369 → 80 [ACK] Seq=376 Ack=2761 Win=64860 Len=0
18	3.148016	132.151.6.75	128.238.38.160	TCP	1434	80 → 3369 [ACK] Seq=2761 Ack=376 Win=6432 Len=1380 [TCP PDU reassembled in 20]
19	3.148069	128.238.38.160	132.151.6.75	TCP	54	3369 → 80 [ACK] Seq=376 Ack=4141 Win=64860 Len=0
20	3.153211	132.151.6.75	128.238.38.160	HTTP	1055	HTTP/1.1 200 OK (text/html)
21	3.153293	128.238.38.160	132.151.6.75	TCP	54	3369 → 80 [ACK] Seq=376 Ack=5143 Win=63859 Len=0
22	3.161867	128.238.38.160	132.151.6.75	TCP	54	3369 → 80 [FIN, ACK] Seq=376 Ack=5143 Win=63859 Len=0
23	3.174716	132.151.6.75	128.238.38.160	TCP	60	80 → 3369 [ACK] Seq=5143 Ack=377 Win=6432 Len=0
24	3.178159	128.238.38.160	132.151.6.75	TCP	62	3370 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM
25	3.179283	128.238.38.160	132.151.6.75	TCP	62	3371 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM
26	3.191649	132.151.6.75	128.238.38.160	TCP	62	80 → 3370 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1380 SACK_PERM
27	3.191726	128.238.38.160	132.151.6.75	TCP	54	3370 → 80 [ACK] Seq=1 Ack=1 Win=64860 Len=0
28	3.191998	128.238.38.160	132.151.6.75	HTTP	320	GET /images/ietflogo2e.gif HTTP/1.1

Figure C.3.7.1

- Open packet trace file dns-trace-2 for nslookup.
- We see from Wireshark that nslookup actually sent three DNS queries and received three DNS responses. For the purpose of this lab, ignore the first two sets of queries/responses, as they are specific to nslookup and are not normally generated by standard Internet applications. You should instead focus on the last query and response messages.
- Answer the following questions.

8. What is the destination port for the DNS query message? What is the source port of DNS response message? Add screenshots in your answer.

Answer:
The destination port for the DNS query message is 53 shown in Figure C.3.8.1 and the source port of DNS response message is 53 shown in Figure C.3.8.2.

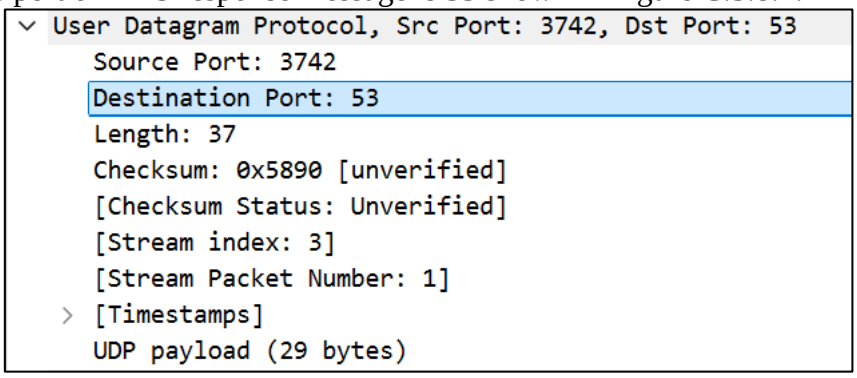


Figure C.3.8.1

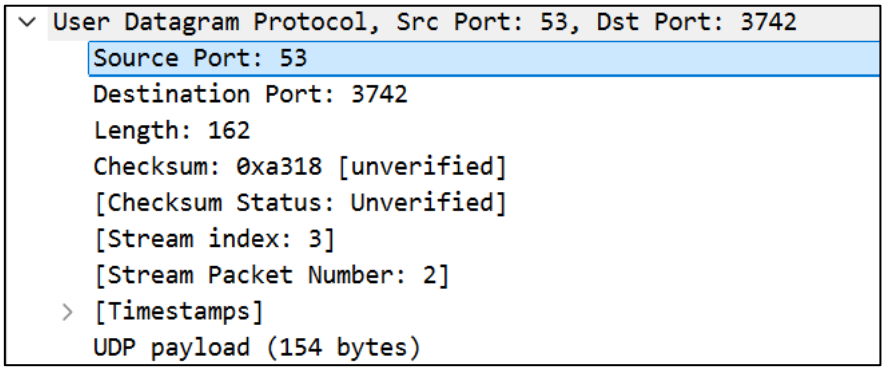


Figure C.3.8.2

9. To what IP address is the DNS query message sent? Is this the IP address of your default local DNS server? Add screenshots in your answer.
Answer:
The IP address of the DNS query message sent is 128.238.28.22 shown in Figure C.3.9.1. This IP address is different from the default local DNS server which is 192.168.43.1 shown in Figure C.3.9.2 and Figure C.3.9.3.

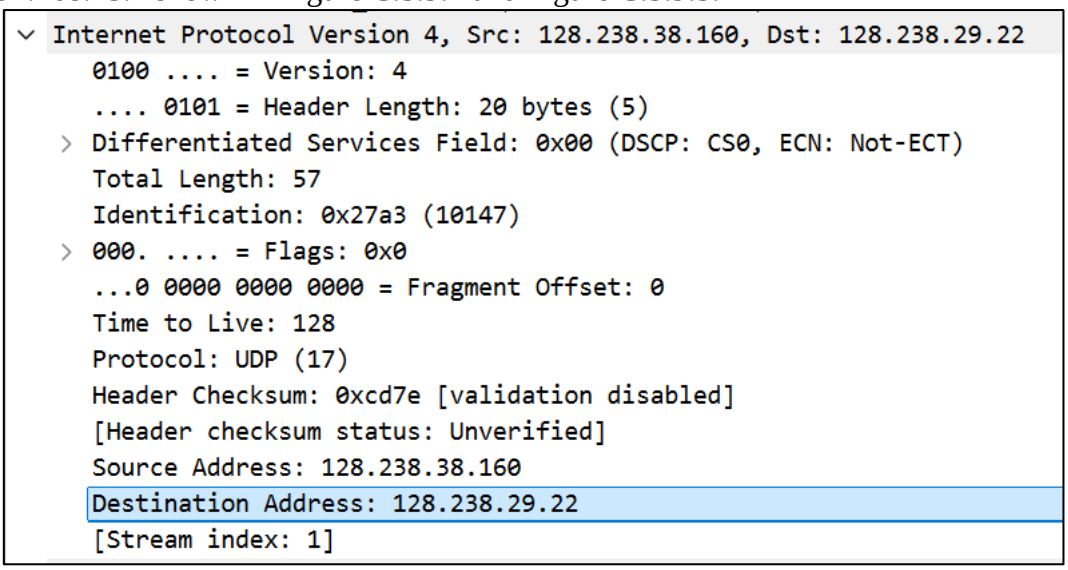


Figure C.3.9.1

```
Wireless LAN adapter Wi-Fi:

   Connection-specific DNS Suffix  . :
   Description . . . . . . . . . . . : MediaTek MT7921 Wi-Fi 6 802.11ax PCIe Adapter
   Physical Address. . . . . . . . . : 90-E8-68-71-0D-59
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes
   IPv6 Address. . . . . . . . . . . : 2001:d08:1282:8782:a1ac:c55a:acc4:44a6(Preferred)
   Temporary IPv6 Address. . . . . . : 2001:d08:1282:8782:d970:bf60:f087:d414(Preferred)
   Link-local IPv6 Address . . . . . : fe80::ee33:1917:e658:1808%12(Preferred)
   IPv4 Address. . . . . . . . . . . : 192.168.43.182(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Lease Obtained. . . . . . . . . . : Friday, January 24, 2025 8:02:47 PM
   Lease Expires . . . . . . . . . . : Friday, January 24, 2025 10:28:18 PM
   Default Gateway . . . . . . . . . : fe80::5ec3:7ff:feeb:70bb%12
                                       192.168.43.1
   DHCP Server . . . . . . . . . . . : 192.168.43.1
   DHCPv6 IAID . . . . . . . . . . . : 177268840
   DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-2A-16-E7-63-90-E8-68-71-0D-59
   DNS Servers . . . . . . . . . . . : 192.168.43.1
   NetBIOS over Tcpip. . . . . . . . : Enabled
```

Figure C.3.9.2

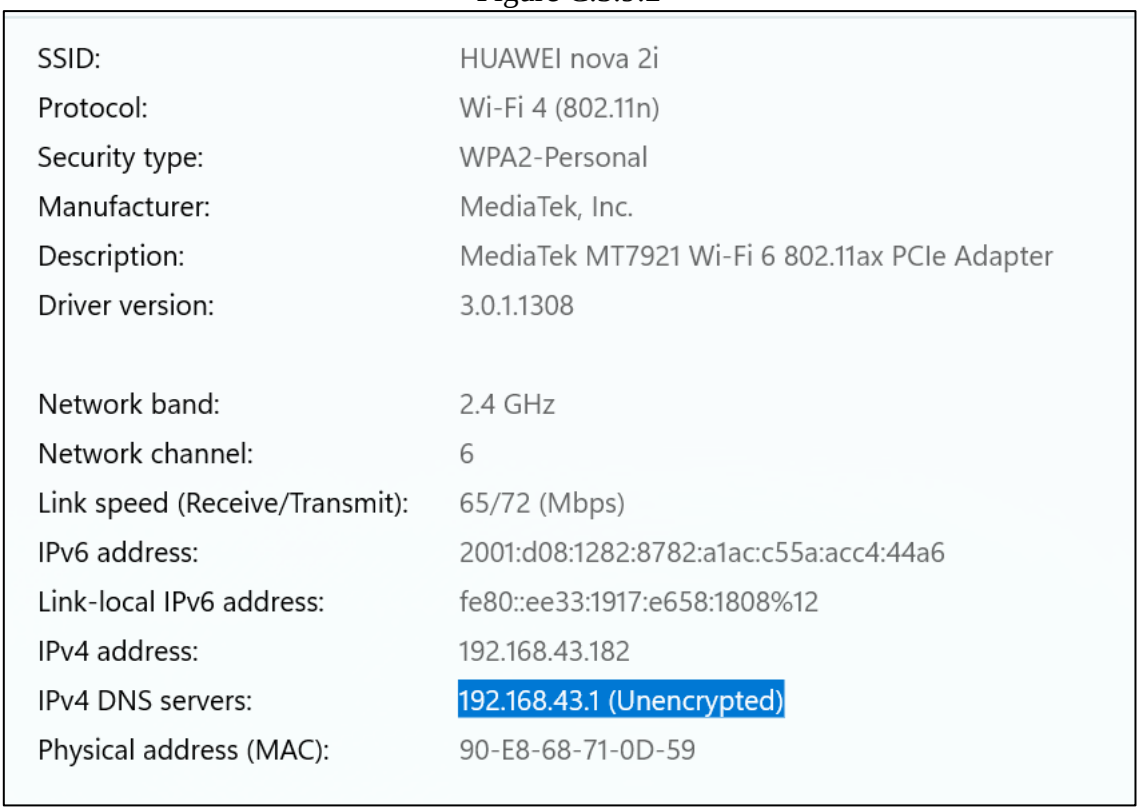


Figure C.3.9.3

10. Examine the DNS query message. What "Type" of DNS query is it? Does the query message contain any "answers"? Add screenshots in your answer.
    Answer:

    The "Type" of DNS query is Type A. The query does not contain any "answers". Figure C.3.10.1 shows the information.

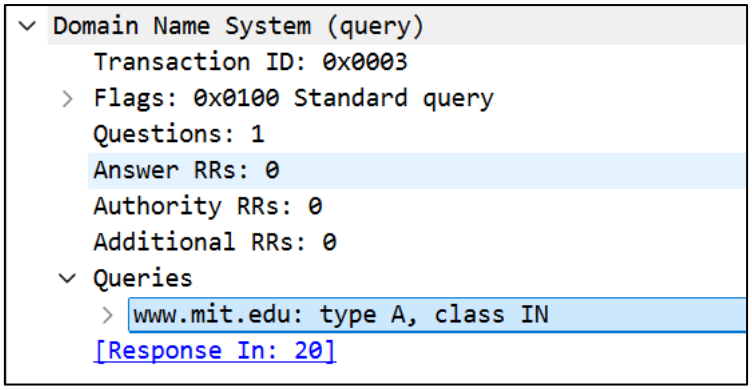


Figure C.3.10.1

11. Examine the DNS response message. How many "answers" are provided? What do each of these answers contain? Add screenshots in your answer.
Answer:
The DNS response message provided 1 "answer" and the answer contain the domain name, the type of address, the class, time to live, data length and IP address as shown in the Figure C.3.11.1.

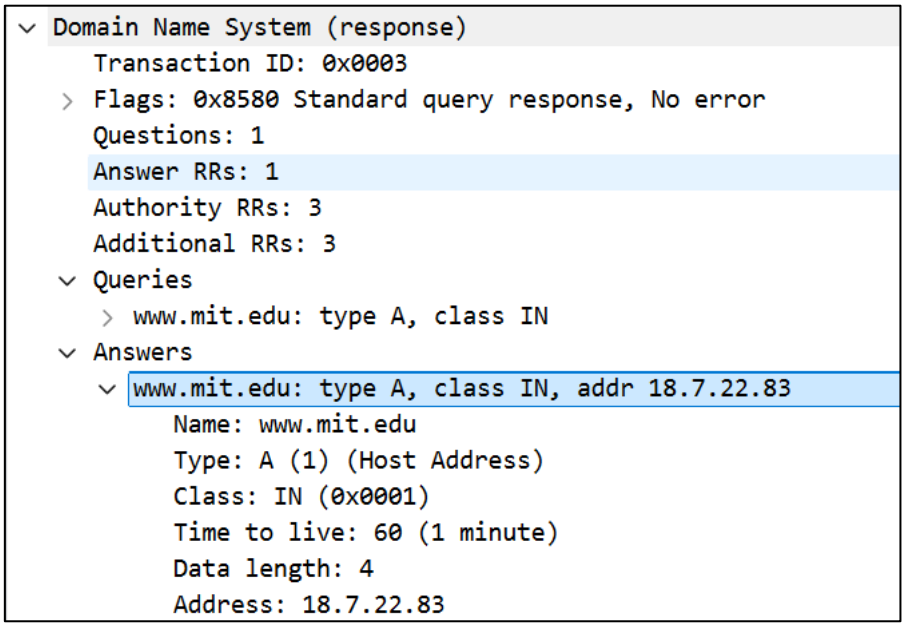


Figure C.3.11.1