

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 : MUHAMMAD NUR SOLIHIN BIN MALIK RADZUAN
Metric No : A21EC0089
Section : 03



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.

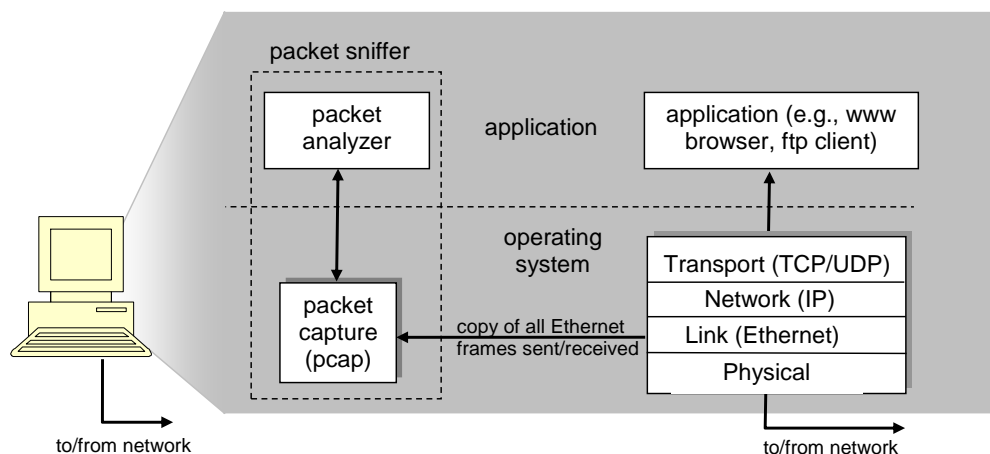


Figure A.1: Packet sniffer structure

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.

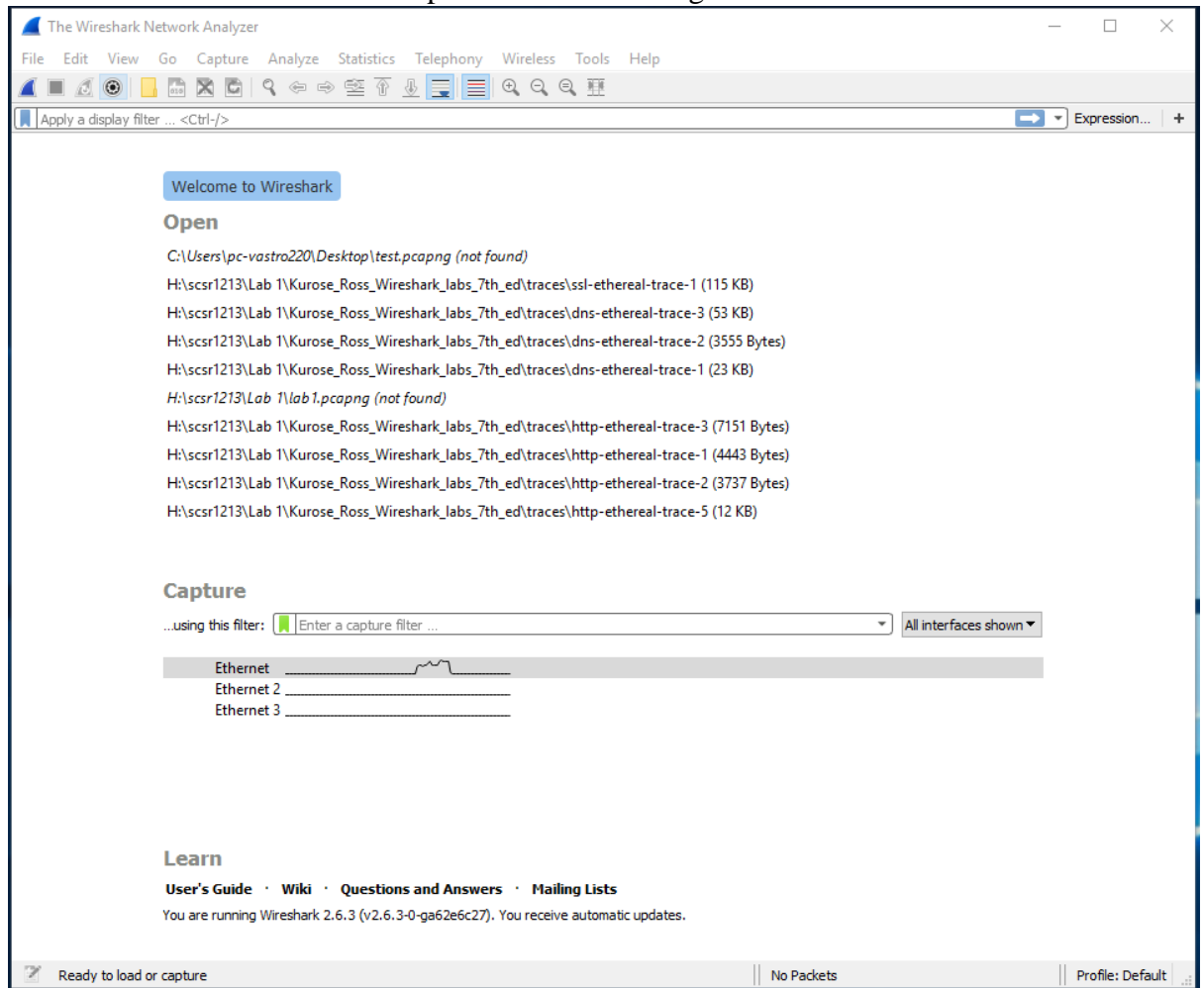


Figure A.2: Initial Wireshark startup screen

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

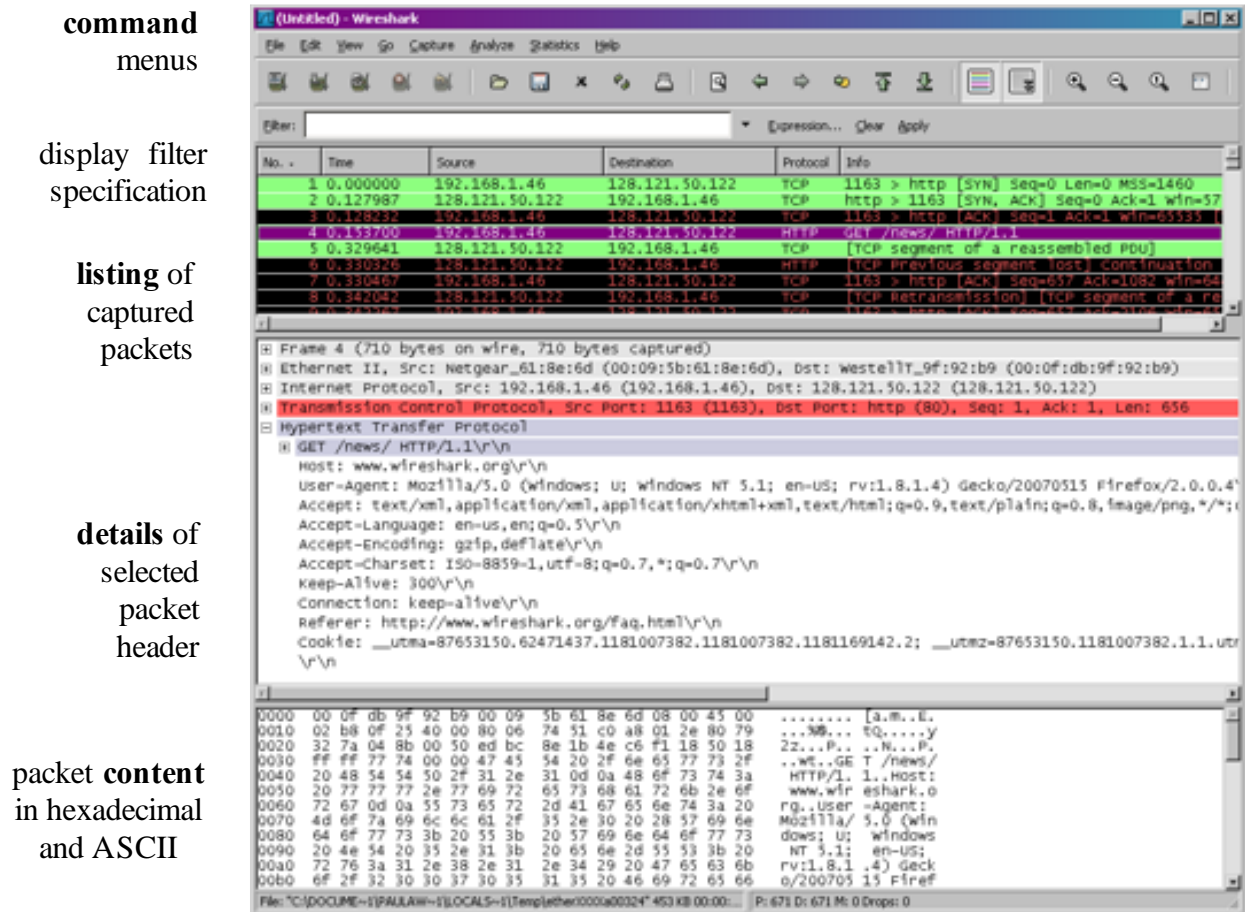


Figure A.3: Wireshark Graphical User Interface, during packet capture and

- 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

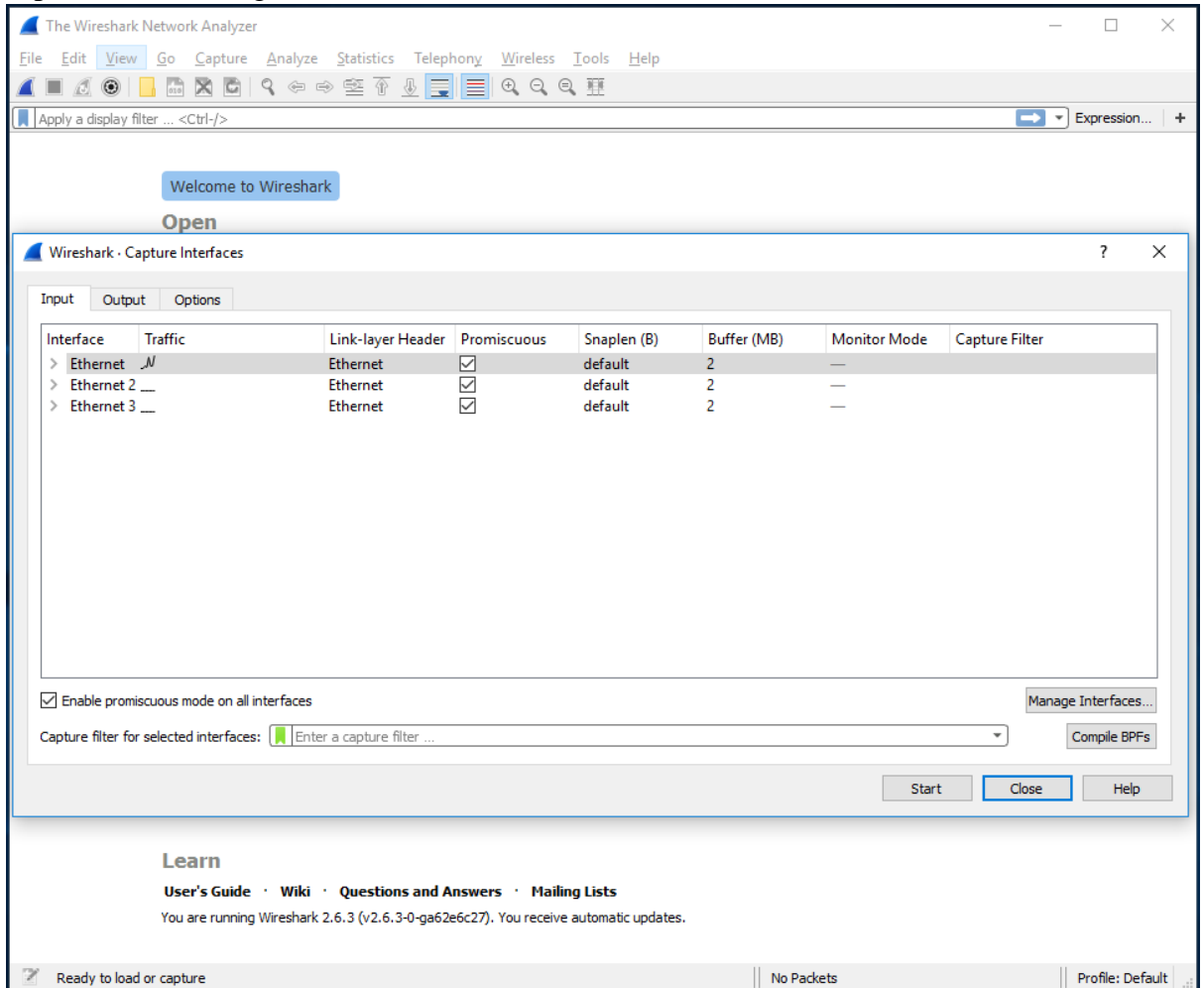


Figure A.4: Capture and Options Menu

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

The image shows the Wireshark network protocol analyzer interface. The top menu bar includes File, Edit, View, Go, Capture, Analyze, Statistics, Telephony, Wireless, Tools, and Help. Below the menu is a toolbar with various icons for file operations, capture control, and analysis. A display filter bar shows 'Apply a display filter ... <Ctrl-/>'. The main packet list pane displays a table of captured packets with columns for No., Time, Source, Destination, Protocol, Length, and Info. The bottom pane shows the detailed view of the selected packet (No. 107), including Ethernet II, Internet Protocol Version 4, Transmission Control Protocol, and Secure Sockets Layer headers. The bottom status bar indicates 'Ethernet: <live capture in progress>', 'Packets: 90 · Displayed: 90 (100.0%)', and 'Profile: Default'.

No.	Time	Source	Destination	Protocol	Length	Info
80	4.169920	Dell_87:3b:e4	Broadcast	ARP	60	Who has 10.60.83.65? Tell 10.60.80.70
81	4.344789	10.60.82.204	255.255.255.255	DB-LSP...	255	Dropbox LAN sync Discovery Protocol
82	4.346792	10.60.82.204	255.255.255.255	DB-LSP...	255	Dropbox LAN sync Discovery Protocol
83	4.347075	10.60.82.204	10.60.83.255	DB-LSP...	255	Dropbox LAN sync Discovery Protocol
84	4.351986	HewlettP_15:15:2b	Broadcast	ARP	60	Who has 10.60.80.15? Tell 10.60.80.167
85	4.479278	Cisco_0d:7b:04	Spanning-tree-(for-...	STP	119	MST. Root = 0/10/00:23:04:ee:be:03 Cost = 20004 Port
86	4.530418	10.60.83.171	239.255.255.250	SSDP	216	M-SEARCH * HTTP/1.1
87	4.530808	10.60.83.171	239.255.255.250	SSDP	216	M-SEARCH * HTTP/1.1
88	4.700996	Dell_75:07:b3	Broadcast	ARP	60	Who has 10.60.80.162? Tell 10.60.80.64
89	4.767952	10.60.82.204	10.60.83.255	NBNS	92	Name query NB MSI-PC<1c>
90	4.995012	10.60.80.78	255.255.255.255	DB-LSP...	176	Dropbox LAN sync Discovery Protocol

> Frame 1: 107 bytes on wire (856 bits), 107 bytes captured (856 bits) on interface 0
 > Ethernet II, Src: Dell_23:e3:3a (00:24:e8:23:e3:3a), Dst: Cisco_d5:79:ff (00:14:6a:d5:79:ff)
 > Internet Protocol Version 4, Src: 10.60.82.216, Dst: 52.230.84.0
 > Transmission Control Protocol, Src Port: 49724, Dst Port: 443, Seq: 1, Ack: 1, Len: 53
 > Secure Sockets Layer

0000 00 14 6a d5 79 ff 00 24 e8 23 e3 3a 08 00 45 00 ..j.y..\$ #:..E.
 0010 00 5d 64 e9 40 00 80 06 00 00 0a 3c 52 d8 34 e6 ..]d.@...<R.4.
 0020 54 00 c2 3c 01 bb 11 3f 6d 43 3e d4 b9 24 50 18 T...<...? mC>..\$P.
 0030 01 01 e6 49 00 00 17 03 01 00 30 07 75 ee af 3c ...I.....0-u...<
 0040 42 81 47 9e 52 29 c3 be ea 2e 88 ac 28 ed bf e6 B.G.R)...((...
 0050 20 36 00 a7 ad f6 ff 8a b0 6d 63 a3 c7 a4 68 ca 6.....mc...h.
 0060 fd 6d b3 05 36 73 0e 3c 54 bc 07 ..m...6s.<T..

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.

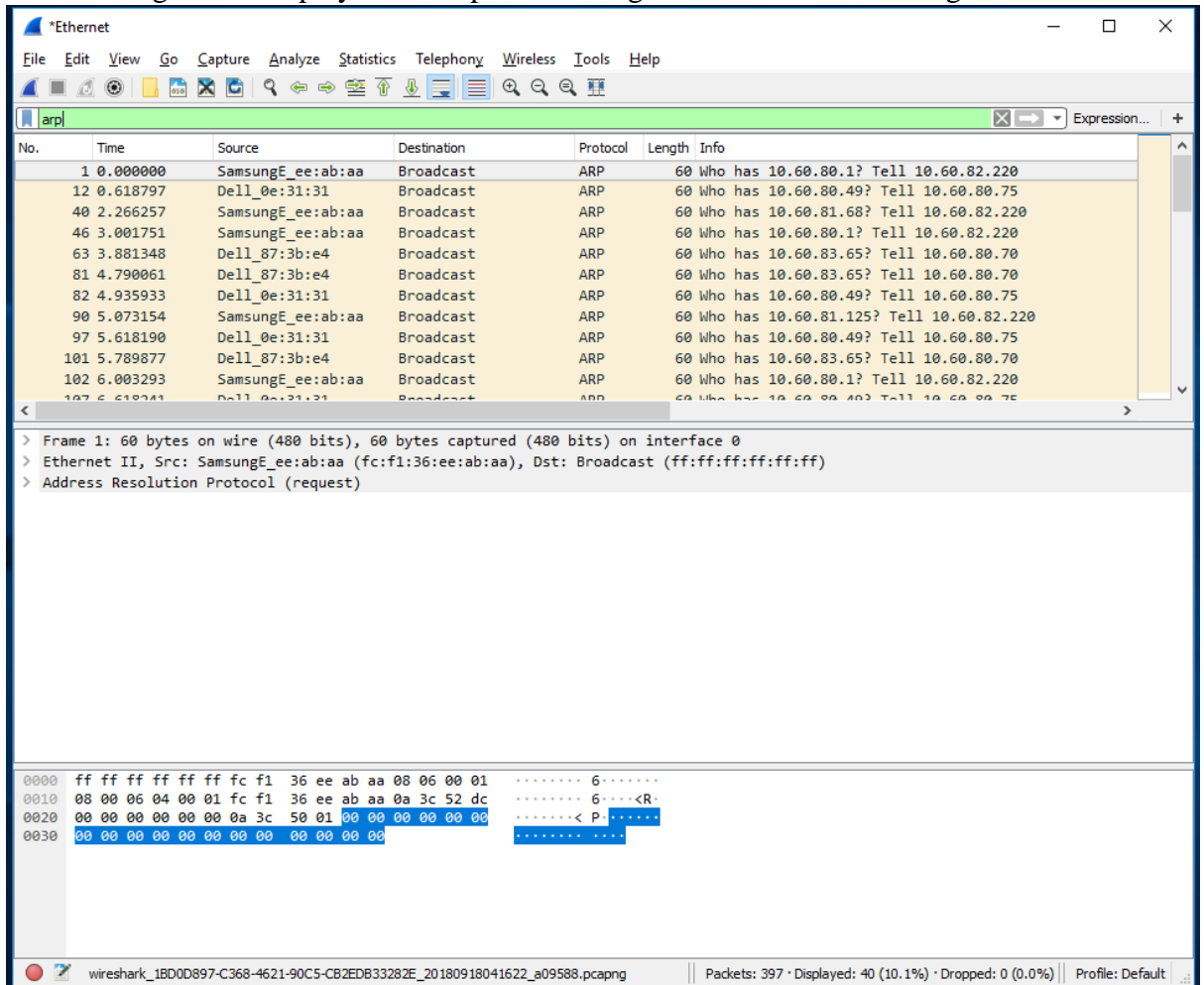


Figure A.6: ARP packet capture

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

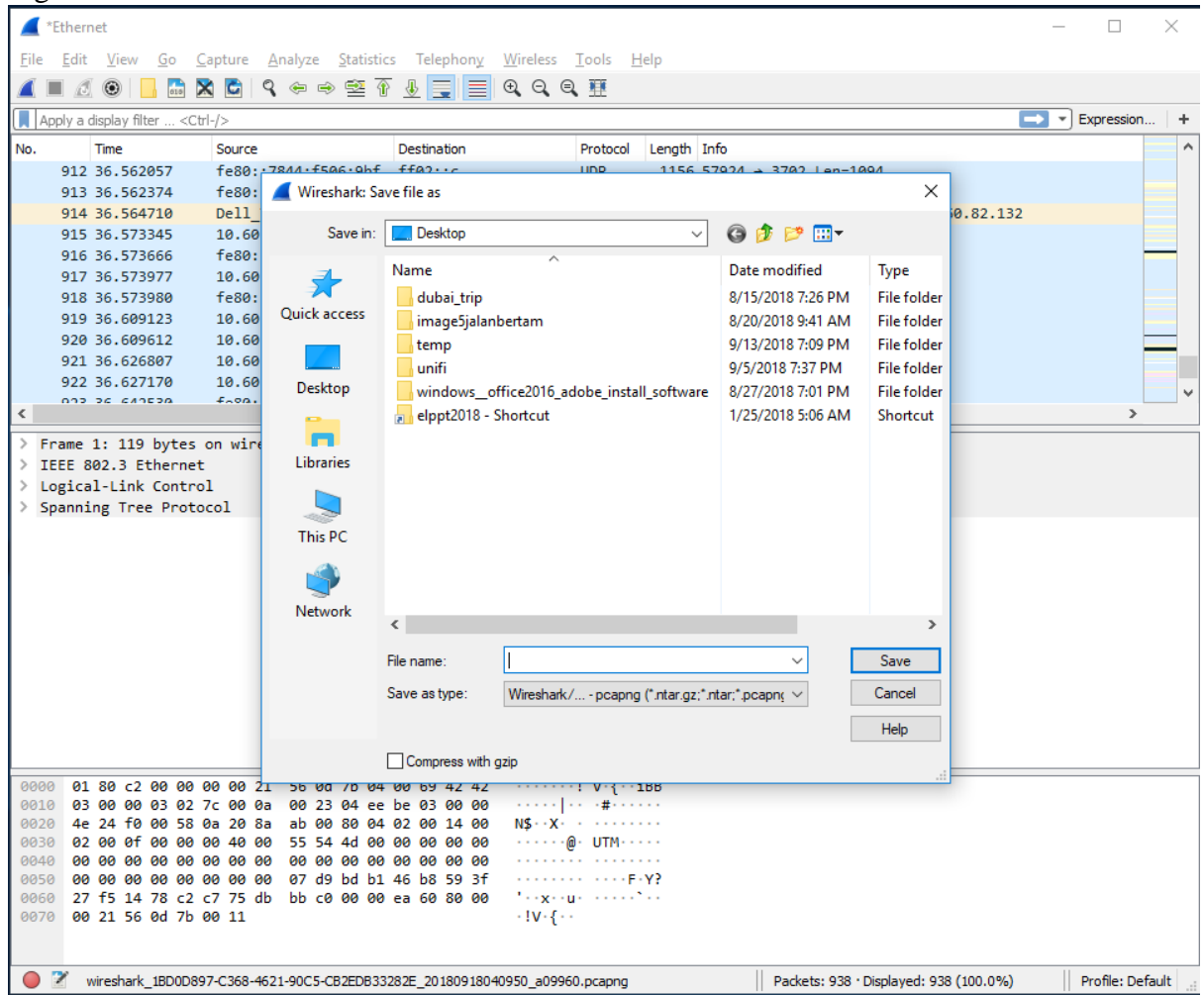


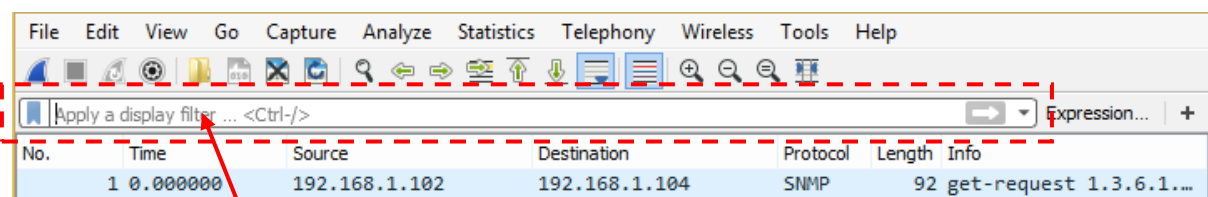
Figure A.7: Save Wireshark trace result

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:



packet display filter

- 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?

```
Request Method: GET
Request URI: /ethereal-labs/lab2-1.html
Request Version: HTTP/1.1
Host: gaia.cs.umass.edu\r\n
```

Request Version: HTTP/1.1

Hence the version of the HTTP is 1.1

2. What is the IP address of the client computer?

```
Internet Protocol Version 4, Src: 192.168.1.102, Dst: 128.119.245.12
```

IP address of the client computer: 192.168.1.102

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

Internet Protocol Version 4, Src: 192.168.1.102, Dst: 128.119.245.12

IP address of the gaia.cs.umass.edu.server : 128.119.245.12

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

```
> Content-Length: 73\r\n
  Keep-Alive: timeout=10, max=100\r\n
  Connection: Keep-Alive\r\n
  Content-Type: text/html; charset=ISO-8859-1\r\n
  \r\n
  [HTTP response 1/2]
  [Time since request: 0.024143000 seconds]
  [Request in frame: 10]
  [Next request in frame: 13]
  [Next response in frame: 14]
  [Request URI: http://gaia.cs.umass.edu/ethereal-la
  File Data: 73 bytes
```

File Data: 73 bytes

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

```
Response Version: HTTP/1.1
Status Code: 200
[Status Code Description: OK]
```

Status Code is 200

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?

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

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?

The server explicitly return the content of the files as the Wireshark provide the Line-based text data that have the content of the file.

```
> Hypertext Transfer Protocol
  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 there us IF-MODIFIED-SINCE line the second HTTP GET. The information that follows are the date which is Tue, 23 Sep 2003 and time is 05:35:00 GMT.

```
> Hypertext Transfer Protocol
  GET /ethereal-labs/lab2-2.html HTTP/1.1\r\n
  Host: gaia.cs.umass.edu\r\n
  User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US;
  Accept: text/xml,application/xml,application/xhtml+xml,text
  Accept-Language: en-us, en;q=0.50\r\n
  Accept-Encoding: gzip, deflate, compress;q=0.9\r\n
  Accept-Charset: ISO-8859-1, utf-8;q=0.66, *;q=0.66\r\n
  Keep-Alive: 300\r\n
  Connection: keep-alive\r\n
  If-Modified-Since: Tue, 23 Sep 2003 05:35:00 GMT\r\n
  If-None-Match: "1bfef-173-8f4ae900"\r\n
  Cache-Control: max-age=0\r\n
  \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.

```

  ▾ Hypertext Transfer Protocol
    ▾ HTTP/1.1 304 Not Modified\r\n
      ▾ [Expert Info (Chat/Sequence): HTTP/1.1 304 Not Modified\r\n]
        [HTTP/1.1 304 Not Modified\r\n]
        [Severity level: Chat]
        [Group: Sequence]
        Response Version: HTTP/1.1
        Status Code: 304
        [Status Code Description: Not Modified]
        Response Phrase: Not Modified

```

The status code is 304 and the phrase the returned from the server is Not Modified to the second HTTP GET. It means that the file is not modified since the specific date and time that was initially requested. Hence, if the files is modified since the date and time that had been requested initially, it would simply return the content of the files, but in this case, it doesn't.

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?

Protocol	Length	Info
HTTP	555	GET /ethereal-labs/lab2-4.html HTTP/1.1
HTTP	1057	HTTP/1.1 200 OK (text/html)
HTTP	625	GET /catalog/images/pearson-logo-footer.gif HTTP/1.1
HTTP	609	GET /~kurose/cover.jpg HTTP/1.1
HTTP	912	HTTP/1.1 200 OK (GIF89a)
HTTP	1096	HTTP/1.0 200 Document follows (JPEG JFIF image)

There are 3 HTTP GET request message that the client browser sent which are requesting the `ethereal-labs/lab2-4.html`, requesting for the `pearson-logo-folder` and requesting the `~kurose/cover.jpg`

2. To which Internet addresses were these GET requests sent?

128.119.245.12	HTTP	555 GET /ethereal-labs/lab2-4.html HTTP/1.1
192.168.1.102	HTTP	1057 HTTP/1.1 200 OK (text/html)
165.193.123.218	HTTP	625 GET /catalog/images/pearson-logo-footer.gif HTTP/1.1
134.241.6.82	HTTP	609 GET /~kurose/cover.jpg HTTP/1.1
192.168.1.102	HTTP	912 HTTP/1.1 200 OK (GIF89a)
192.168.1.102	HTTP	1096 HTTP/1.0 200 Document follows (JPEG JFIF image)

Internet Address for *GET /ethereal-labs/lab2-4.html HTTP/1.1*: 128.119.245.12

Internet Address for *GET /catalog/images/pearson-logo-footer.gif*: 165.193.123.218

Internet Address for *GET /~kurose/cover.jpg*: 165.193.123.218 : 134.241.6.82

3. any bytes of content are being returned to client browser for the **pearson-logo-footer.gif** image file?

```
> Content-length: 3357\r\n
Accept-ranges: bytes\r\n
Connection: keep-alive\r\n
\r\n
[HTTP response 1/1]
[Time since request: 0.027569000 seconds]
[Request in frame: 17]
[Request URI: http://www.aw-bc.com/catalog/images/pearson-logo-
File Data: 3357 bytes
```

Bytes of content to client browser for the **pearson-logo-footer.gif** image file is **3357 bytes**

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

```
✓ Hypertext Transfer Protocol
  > HTTP/1.0 200 Document follows\r\n
    Date: Tue, 23 Sep 2003 05:38:44 GMT\r\n
    Server: NCSA/1.5.2\r\n
    Last-modified: Tue, 23 Sep 2003 04:56:38 GMT\r\n
    Content-type: image/jpeg\r\n
  > Content-length: 15642\r\n
    \r\n
    [HTTP response 1/1]
    [Time since request: 0.281074000 seconds]
    [Request in frame: 20]
    [Request URI: http://manic.cs.umass.edu/~kurose/cover.jpg]
    File Data: 15642 bytes
```

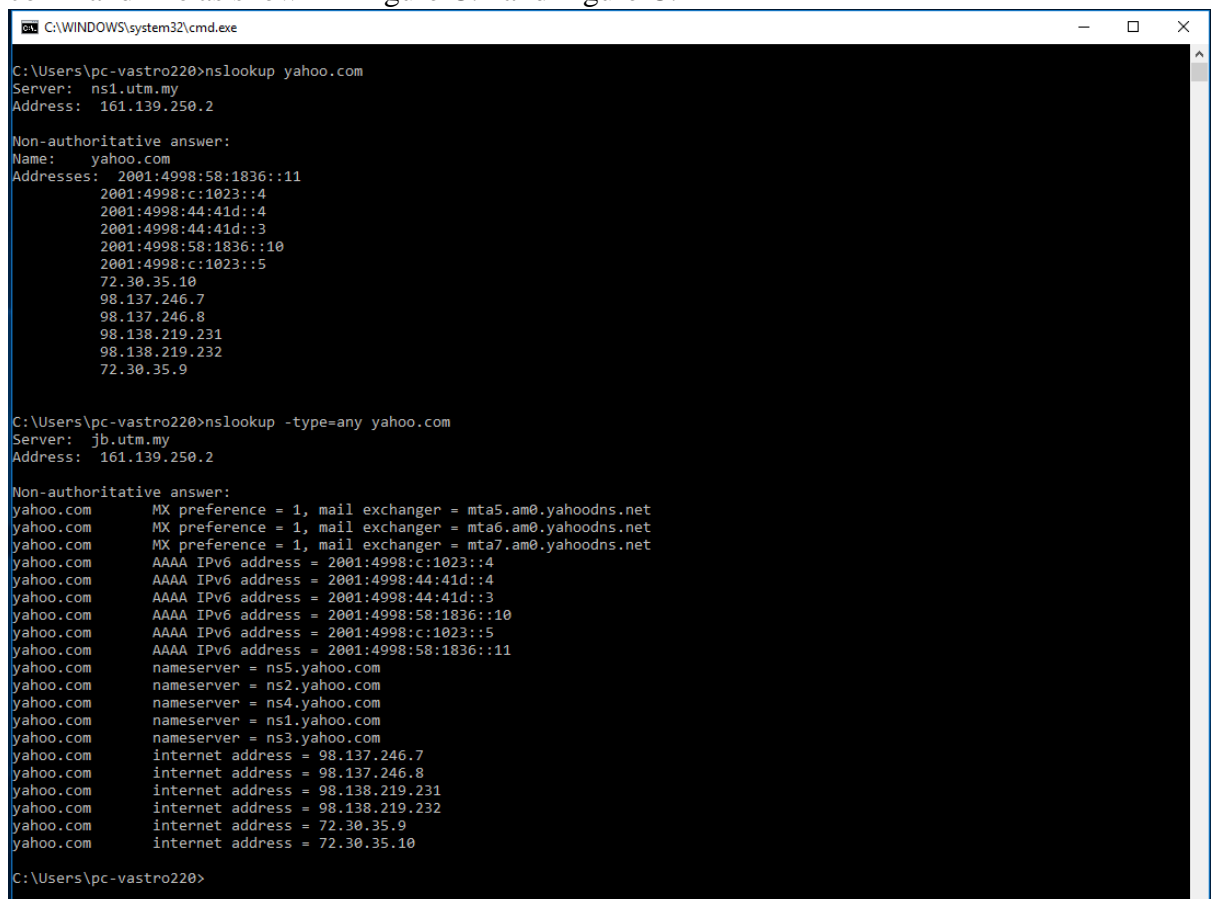
Bytes of content that are being returned to client browser for the **cover.jpg image file** is **15642 bytes**

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



```
C:\WINDOWS\system32\cmd.exe

C:\Users\pc-vastro220>nslookup yahoo.com
Server: ns1.utm.my
Address: 161.139.250.2

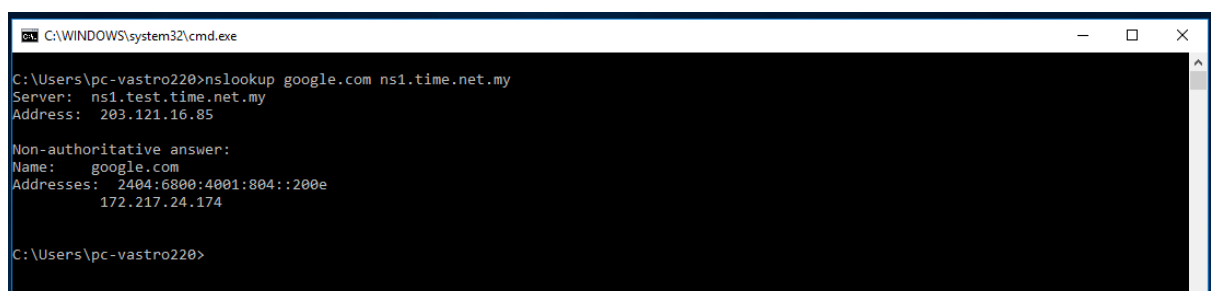
Non-authoritative answer:
Name: yahoo.com
Addresses: 2001:4998:58:1836::11
           2001:4998:c:1023::4
           2001:4998:44:41d::4
           2001:4998:44:41d::3
           2001:4998:58:1836::10
           2001:4998:c:1023::5
           72.30.35.10
           98.137.246.7
           98.137.246.8
           98.138.219.231
           98.138.219.232
           72.30.35.9

C:\Users\pc-vastro220>nslookup -type=any yahoo.com
Server: jlb.utm.my
Address: 161.139.250.2

Non-authoritative answer:
yahoo.com      MX preference = 1, mail exchanger = mta5.am0.yahoodns.net
yahoo.com      MX preference = 1, mail exchanger = mta6.am0.yahoodns.net
yahoo.com      MX preference = 1, mail exchanger = mta7.am0.yahoodns.net
yahoo.com      AAAA IPv6 address = 2001:4998:c:1023::4
yahoo.com      AAAA IPv6 address = 2001:4998:44:41d::4
yahoo.com      AAAA IPv6 address = 2001:4998:44:41d::3
yahoo.com      AAAA IPv6 address = 2001:4998:58:1836::10
yahoo.com      AAAA IPv6 address = 2001:4998:c:1023::5
yahoo.com      AAAA IPv6 address = 2001:4998:58:1836::11
yahoo.com      nameserver = ns5.yahoo.com
yahoo.com      nameserver = ns2.yahoo.com
yahoo.com      nameserver = ns4.yahoo.com
yahoo.com      nameserver = ns1.yahoo.com
yahoo.com      nameserver = ns3.yahoo.com
yahoo.com      internet address = 98.137.246.7
yahoo.com      internet address = 98.137.246.8
yahoo.com      internet address = 98.138.219.231
yahoo.com      internet address = 98.138.219.232
yahoo.com      internet address = 72.30.35.9
yahoo.com      internet address = 72.30.35.10

C:\Users\pc-vastro220>
```

Figure C.1: nslookup result



```
C:\WINDOWS\system32\cmd.exe

C:\Users\pc-vastro220>nslookup google.com ns1.time.net.my
Server: ns1.test.time.net.my
Address: 203.121.16.85

Non-authoritative answer:
Name: google.com
Addresses: 2404:6800:4001:804::200e
           172.217.24.174

C:\Users\pc-vastro220>
```

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.

IP address is 211.25.122.89

```
C:\Users\solih>nslookup www.microsoft.com
Server: ns1.utm.my
Address: 161.139.250.2

Non-authoritative answer:
Name: e13678.dscb.akamaiedge.net
Addresses: 2001:f40:a:c85::356e
           2001:f40:a:c84::356e
           2001:f40:a:c80::356e
           2001:f40:a:c81::356e
           2001:f40:a:c82::356e
           211.25.122.89
Aliases: www.microsoft.com
          www.microsoft.com-c-3.edgekey.net
          www.microsoft.com-c-3.edgekey.net.globalredir.akadns.net
```

2. Run nslookup to determine the non-authoritative DNS servers for domain microsoft.com. Add screenshot to your answer.

```
C:\Users\solih>nslookup microsoft.com
Server: ns1.utm.my
Address: 161.139.250.2

Non-authoritative answer:
Name: microsoft.com
Addresses: 20.81.111.85
           20.84.181.62
           20.53.203.50
           20.112.52.29
           20.103.85.33

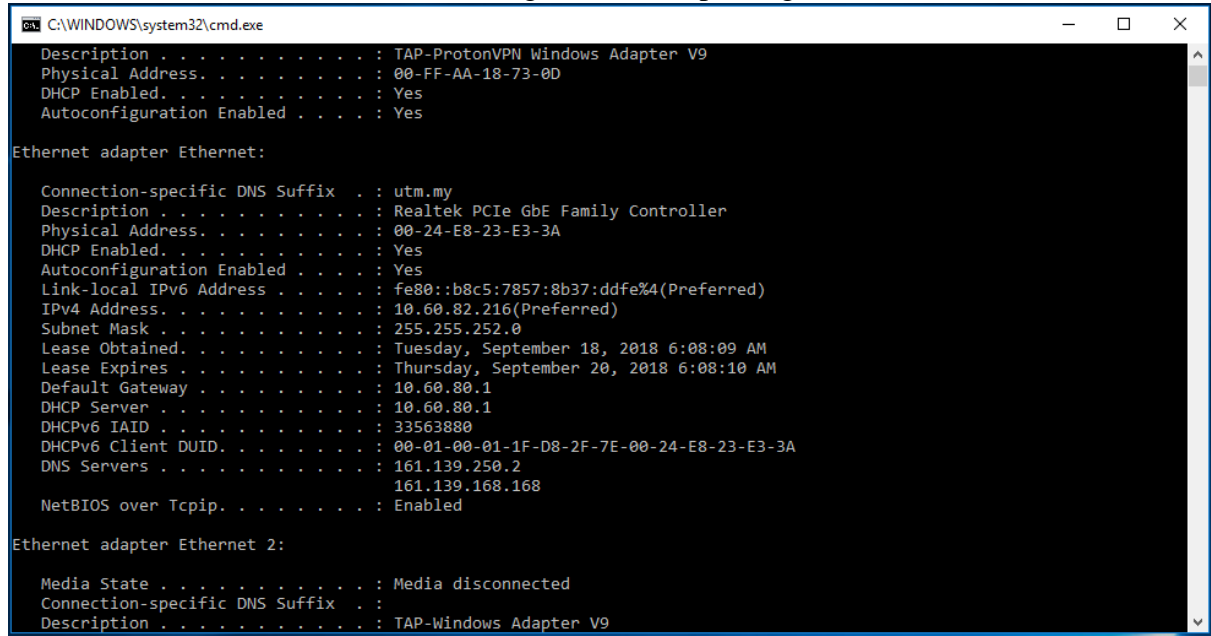
C:\Users\solih>nslookup -type=any microsoft.com
Server: ns1.utm.my
Address: 161.139.250.2

Non-authoritative answer:
microsoft.com MX preference = 10, mail exchanger = microsoft-com.mail.protection.outlook.com
microsoft.com internet address = 20.112.52.29
microsoft.com internet address = 20.103.85.33
microsoft.com internet address = 20.53.203.50
microsoft.com internet address = 20.84.181.62
microsoft.com internet address = 20.81.111.85
microsoft.com nameserver = ns2-39.azure-dns.net
microsoft.com nameserver = ns4-39.azure-dns.info
microsoft.com nameserver = ns3-39.azure-dns.org
microsoft.com nameserver = ns1-39.azure-dns.com
```

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



```
C:\WINDOWS\system32\cmd.exe
Description . . . . . : TAP-ProtonVPN Windows Adapter V9
Physical Address. . . . . : 00-FF-AA-18-73-0D
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . : Yes

Ethernet adapter Ethernet:

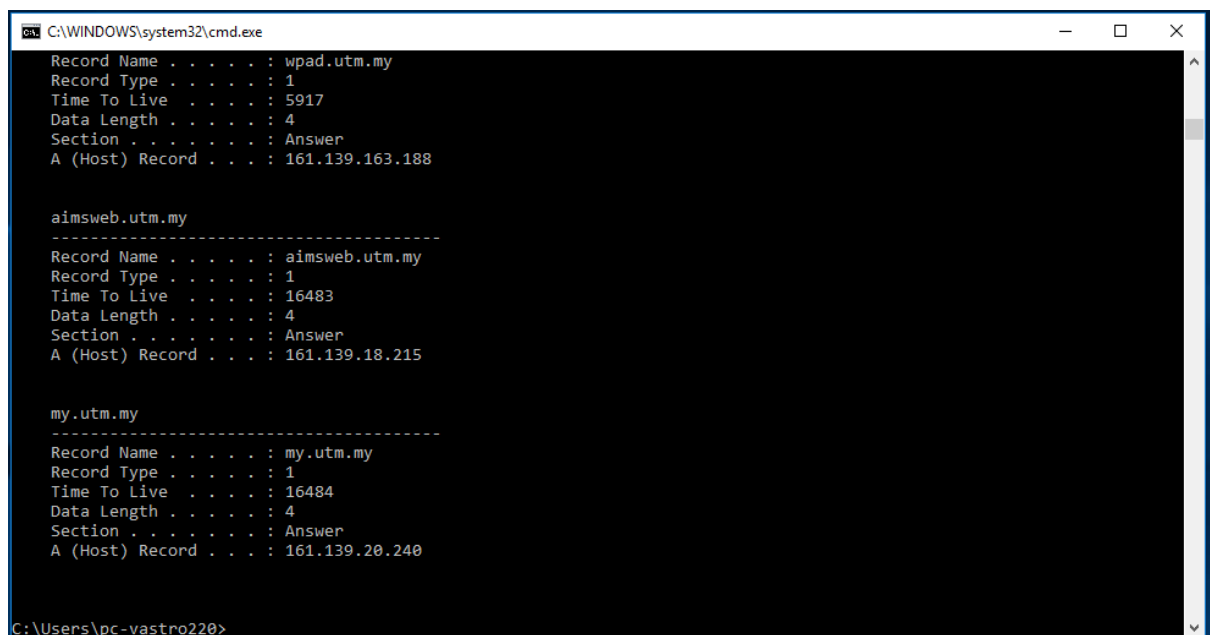
    Connection-specific DNS Suffix  . : utm.my
    Description . . . . . : Realtek PCIe GbE Family Controller
    Physical Address. . . . . : 00-24-E8-23-E3-3A
    DHCP Enabled. . . . . : Yes
    Autoconfiguration Enabled . . . . : Yes
    Link-local IPv6 Address . . . . . : fe80::b8c5:7857:8b37:ddfe%4(Preferred)
    IPv4 Address. . . . . : 10.60.82.216(Preferred)
    Subnet Mask . . . . . : 255.255.252.0
    Lease Obtained. . . . . : Tuesday, September 18, 2018 6:08:09 AM
    Lease Expires . . . . . : Thursday, September 20, 2018 6:08:10 AM
    Default Gateway . . . . . : 10.60.80.1
    DHCP Server . . . . . : 10.60.80.1
    DHCPv6 IAID . . . . . : 33563880
    DHCPv6 Client DUID. . . . . : 00-01-00-01-1F-D8-2F-7E-00-24-E8-23-E3-3A
    DNS Servers . . . . . : 161.139.250.2
                           161.139.168.168
    NetBIOS over Tcpip. . . . . : Enabled

Ethernet adapter Ethernet 2:

    Media State . . . . . : Media disconnected
    Connection-specific DNS Suffix  . :
    Description . . . . . : TAP-Windows Adapter V9
```

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



```
C:\WINDOWS\system32\cmd.exe
Record Name . . . . . : wpad.utm.my
Record Type . . . . . : 1
Time To Live . . . . . : 5917
Data Length . . . . . : 4
Section . . . . . : Answer
A (Host) Record . . . : 161.139.163.188

-----
aimsweb.utm.my
Record Name . . . . . : aimsweb.utm.my
Record Type . . . . . : 1
Time To Live . . . . . : 16483
Data Length . . . . . : 4
Section . . . . . : Answer
A (Host) Record . . . : 161.139.18.215

-----
my.utm.my
Record Name . . . . . : my.utm.my
Record Type . . . . . : 1
Time To Live . . . . . : 16484
Data Length . . . . . : 4
Section . . . . . : Answer
A (Host) Record . . . : 161.139.20.240

C:\Users\pc-vastro220>
```

Figure C.4: ipconfig /displaydns result

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

Command: ipconfig /flushdns



```
C:\WINDOWS\system32\cmd.exe
C:\Users\pc-vastro220>ipconfig /flushdns

Windows IP Configuration

Successfully flushed the DNS Resolver Cache.

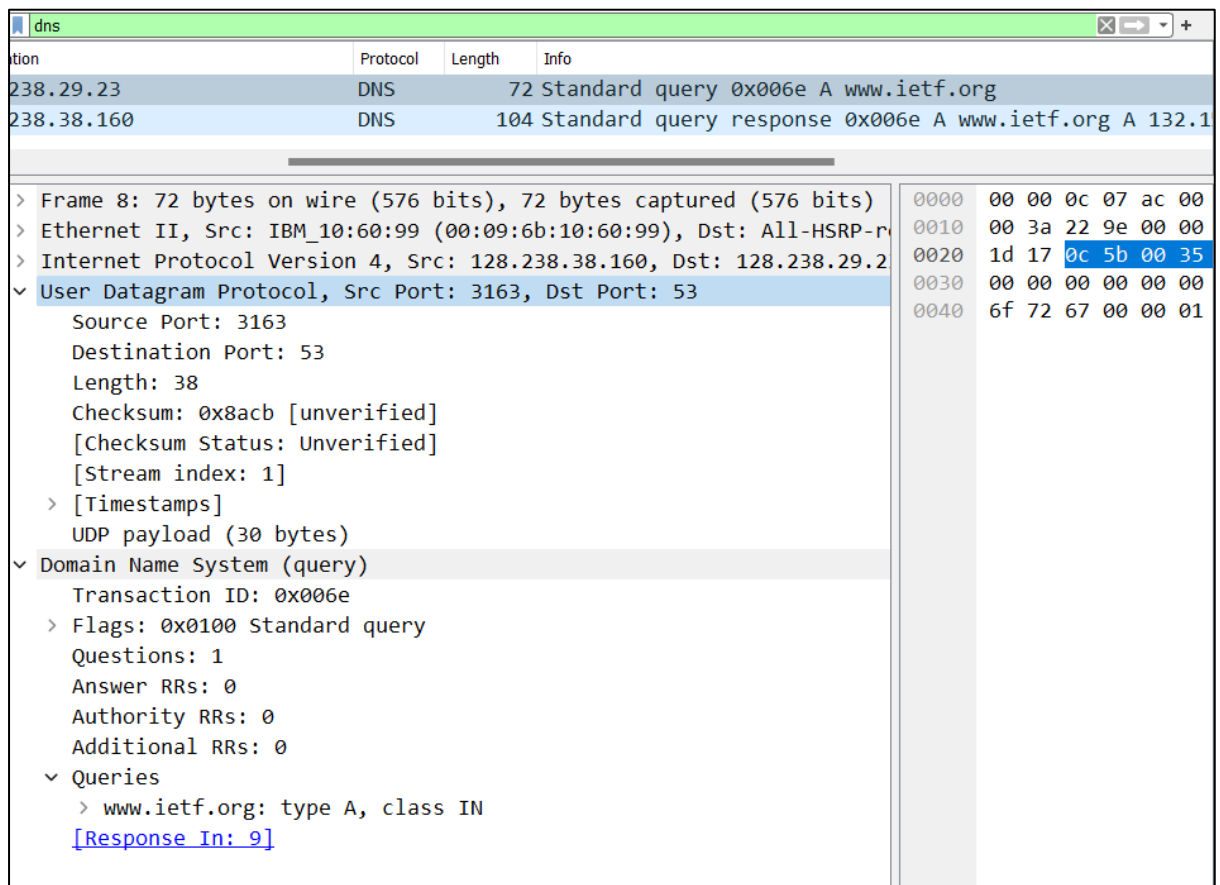
C:\Users\pc-vastro220>
```

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.

They are sent over UDP.



DNS Query

Source port of DNS Response message: 53

238.29.23	DNS	72 Standard query 0x006e A www.ietf.org
238.38.160	DNS	104 Standard query response 0x006e A www.ietf.org A 132.1

> Frame 9: 104 bytes on wire (832 bits), 104 bytes captured (832 bits)

> Ethernet II, Src: Cisco_83:e4:54 (00:b0:8e:83:e4:54), Dst: IBM_10:6

> Internet Protocol Version 4, Src: 128.238.29.23, Dst: 128.238.38.16

▼ User Datagram Protocol, Src Port: 53, Dst Port: 3163

Source Port: 53

Destination Port: 3163

Length: 70

Checksum: 0xb0ba [unverified]

[Checksum Status: Unverified]

[Stream index: 1]

> [Timestamps]

UDP payload (62 bytes)

0000	00 09 6b 10 60 99
0010	00 5a d5 95 00 00
0020	26 a0 00 35 0c 5b
0030	00 02 00 00 00 00
0040	6f 72 67 00 00 01
0050	06 8e 00 04 84 97
0060	06 8e 00 04 41 f6

3. To what IP address is the DNS query message sent? Add screenshots in your answer.

IP address that the DNS query message sent 128.238.29.23 as the Destination Address for the DNS query message is 128.238.29.23

```

  ▾ Internet Protocol Version 4, Src: 128.238.38.160, Dst: 128.238.29.23
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
    > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
      Total Length: 58
      Identification: 0x229e (8862)
    > 000. .... = Flags: 0x0
      ...0 0000 0000 0000 = Fragment Offset: 0
      Time to Live: 128
      Protocol: UDP (17)
      Header Checksum: 0xd281 [validation disabled]
      [Header checksum status: Unverified]
      Source Address: 128.238.38.160
      Destination Address: 128.238.29.23

```

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.

The DNS query message is a Standard query type. Hence, it doesn't contain any answers.

```

Domain Name System (query)
  Transaction ID: 0x006e
  > Flags: 0x0100 Standard query
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 0
  > Queries
    > www.ietf.org: type A, class IN
      [Response In: 9]
```

5. Examine the DNS response message. How many “answers” are provided? What do each of these answers contain? Add screenshots in your answer.

There are a total of 2 answer that are provided in the DNS response message which provides the name of the host, type, class, data length and the address.

```

Answers
  > www.ietf.org: type A, class IN, addr 132.151.6.75
    Name: www.ietf.org
    Type: A (Host Address) (1)
    Class: IN (0x0001)
    Time to live: 1678 (27 minutes, 58 seconds)
    Data length: 4
    Address: 132.151.6.75
  > www.ietf.org: type A, class IN, addr 65.246.255.51
    Name: www.ietf.org
    Type: A (Host Address) (1)
    Class: IN (0x0001)
    Time to live: 1678 (27 minutes, 58 seconds)
    Data length: 4
    Address: 65.246.255.51
  [Request In: 8]
  [Time: 0.000844000 seconds]
```

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.

128.238.29.23	DNS	72 Standard query 0x006e A www.ietf.org
128.238.38.160	DNS	104 Standard query response 0x006e A www.ietf.org

<p>> Frame 9: 104 bytes on wire (832 bits), 104 bytes captured (832 bits) on interface 0</p> <p>> Ethernet II, Src: Cisco_83:e4:54 (00:b0:8e:83:e4:54), Dst: IBM_10:00:00:00:00:00</p> <p>> Internet Protocol Version 4, Src: 128.238.29.23, Dst: 128.238.38.160</p> <p> 0100 = Version: 4</p> <p> 0101 = Header Length: 20 bytes (5)</p> <p> > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)</p> <p> Total Length: 90</p> <p> Identification: 0xd595 (54677)</p> <p> > 000. = Flags: 0x0</p> <p> ...0 0000 0000 0000 = Fragment Offset: 0</p> <p> Time to Live: 126</p> <p> Protocol: UDP (17)</p> <p> Header Checksum: 0x216a [validation disabled]</p> <p> [Header checksum status: Unverified]</p> <p> Source Address: 128.238.29.23</p> <p> Destination Address: 128.238.38.160</p>	<p>0000 00 09 6b 10 60 99</p> <p>0010 00 5a d5 95 00 00</p> <p>0020 26 a0 00 35 0c 5b</p> <p>0030 00 02 00 00 00 00</p> <p>0040 6f 72 67 00 00 01</p> <p>0050 06 8e 00 04 84 97</p> <p>0060 06 8e 00 04 41 f6</p>
---	---

DNS

132.151.6.75	TCP	62 3371 → 80 [FIN, ACK] Seq=201 Ack=355
132.151.6.75	TCP	62 3372 → 80 [SYN] Seq=0 Win=64240 Len=0
132.151.6.75	TCP	62 3373 → 80 [SYN] Seq=0 Win=64240 Len=0
128.238.38.160	TCP	60 80 → 3371 [ACK] Seq=355 Ack=262 Win=64
128.238.38.160	TCP	62 80 → 3372 [SYN, ACK] Seq=0 Ack=1 Win=!

> Frame 47: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) > Ethernet II, Src: Cisco_83:e4:54 (00:b0:8e:83:e4:54), Dst: IBM_10:6 > Internet Protocol Version 4, Src: 132.151.6.75, Dst: 128.238.38.160 0100 = Version: 4 0101 = Header Length: 20 bytes (5) > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT) Total Length: 40 Identification: 0x6ac3 (27331) > 010. = Flags: 0x2, Don't fragment ...0 0000 0000 0000 = Fragment Offset: 0 Time to Live: 53 Protocol: TCP (6) Header Checksum: 0xa89c [validation disabled] [Header checksum status: Unverified] Source Address: 132.151.6.75 Destination Address: 128.238.38.160 > Transmission Control Protocol, Src Port: 80, Dst Port: 3371, Seq: 3	0000 00 09 6b 10 60 0010 00 28 6a c3 40 0020 26 a0 00 50 0d 0030 19 20 a9 06 00
---	--

TCP SYN

Yes, it is the same. As it seems based on the picture provided, the destination address for of the SYN packet is 128.238.38.160 which the same with the IP address that provided in the DNS response message which is 128.238.38.160

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

No.

- 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.

Destination for DNS Query message: 53

```

User Datagram Protocol, Src Port: 3742, Dst Port: 53
  Source Port: 3742
  Destination Port: 53
  Length: 37
  Checksum: 0x5890 [unverified]
  [Checksum Status: Unverified]
  [Stream index: 3]
  > [Timestamps]
  UDP payload (29 bytes)
  > Domain Name System (query)

```

Source port of DNS response message: 53

```

> Frame 20: 196 bytes on wire (1568 bits), 196 bytes captured (1568
> Ethernet II, Src: Cisco_83:e4:54 (00:b0:8e:83:e4:54), Dst: IBM_10
> Internet Protocol Version 4, Src: 128.238.29.22, Dst: 128.238.38.
User Datagram Protocol, Src Port: 53, Dst Port: 3742
  Source Port: 53
  Destination Port: 3742
  Length: 162
  Checksum: 0xa318 [unverified]
  [Checksum Status: Unverified]
  [Stream index: 3]
  > [Timestamps]
  UDP payload (154 bytes)
  > Domain Name System (response)
    Transaction ID: 0x0003
    > Flags: 0x8580 Standard query response, No error
    Questions: 1

```

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.

IP address that the DNS query message sent 128.238.29.23 as the Destination Address for the DNS query message is 128.238.29.23 and it is not same with my default local DNS server which is 192.168.1.1

```
0100 .... = Version: 4
.... 0101 = Header Length: 20 bytes (5)
> Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
Total Length: 57
Identification: 0x27a3 (10147)
> 000. .... = Flags: 0x0
...0 0000 0000 0000 = Fragment Offset: 0
Time to Live: 128
Protocol: UDP (17)
Header Checksum: 0xcd7e [validation disabled]
[Header checksum status: Unverified]
Source Address: 128.238.38.160
Destination Address: 128.238.29.22
> User Datagram Protocol, Src Port: 3742, Dst Port: 53
> Domain Name System (query)
Transaction ID: 0x0003
```

ipconfig/all result:

```
Wireless LAN adapter Wi-Fi:

Connection-specific DNS Suffix . : realtek
Description . . . . . : MediaTek Wi-Fi 6 MT7921 Wireless LAN Card
Physical Address. . . . . : EC-2E-98-CB-D9-2D
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . : Yes
IPv4 Address. . . . . : 192.168.1.7(Preferred)
Subnet Mask . . . . . : 255.255.255.0
Lease Obtained. . . . . : Saturday, 19 November, 2022 11:05:20 PM
Lease Expires . . . . . : Sunday, 20 November, 2022 3:05:20 AM
Default Gateway . . . . . : 192.168.1.1
DHCP Server . . . . . : 192.168.1.1
DNS Servers . . . . . : 192.168.1.1
NetBIOS over Tcpip. . . . . : Enabled
```

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.

DNS query message is a type A and doesn't have any answers

```
✓ Domain Name System (query)
  Transaction ID: 0x0003
  > Flags: 0x0100 Standard query
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 0
  ✓ Queries
    > www.mit.edu: type A, class IN
      [Response In: 20]
```

11. Examine the DNS response message. How many “answers” are provided? What do each of these answers contain? Add screenshots in your answer.

There is only 1 answer. It contains the name of the host, the address of the host, type of the host, class, time to live and the data length.

```
✓ Domain Name System (response)
  Transaction ID: 0x0003
  > Flags: 0x8580 Standard query response, No error
    Questions: 1
    Answer RRs: 1
    Authority RRs: 3
    Additional RRs: 3
  > Queries
  ✓ Answers
    > www.mit.edu: type A, class IN, addr 18.7.22.83
      Name: www.mit.edu
      Type: A (Host Address) (1)
      Class: IN (0x0001)
      Time to live: 60 (1 minute)
      Data length: 4
      Address: 18.7.22.83
    > Authoritative nameservers
    > Additional records
      [Request In: 19]
      [Time: 0.016757000 seconds]
```