COMP4621 Wireshark Labs: HTTP and DNS

TA: Waheed Ghali

E-mail: wggghali@connect.ust.hk

Outline

- HTTP
  - the basic GET/response interaction
  - HTTP message formats
  - retrieving large HTML files

- DNS
  - Review of DNS
  - How to use nslookup to send DNS queries
  - How to use ipconfig (or ifconfig) to check IP addresses and DNS servers
  - Example of tracing DNS packets with Wireshark

HTTP: the basic GET/response interaction

1. Start up your web browser

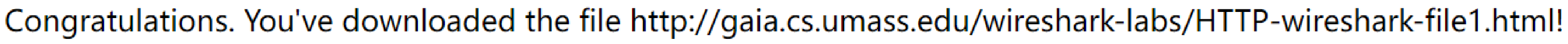
2. Start up the Wireshark packet sniffer and type ‘http’ in the display filter window

3. Select your interface and begin Wireshark packet capture

4. Enter the following to your web browser:

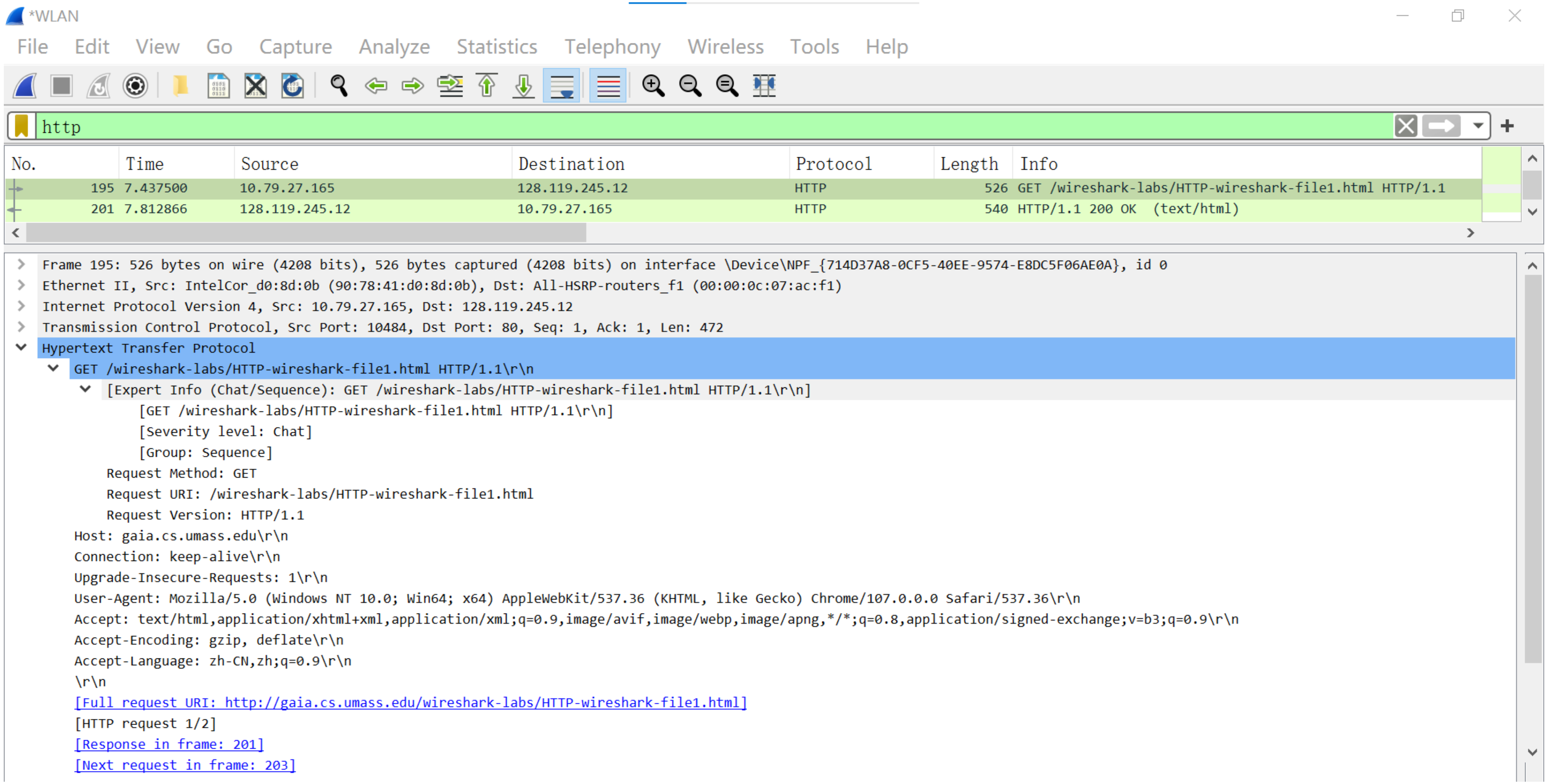
http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file1.html

5. After your browser has displayed the one-line HTML file, stop the Wireshark packet capture



Congratulations. You've downloaded the file http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file1.html!

HTTP: the basic GET/response interaction

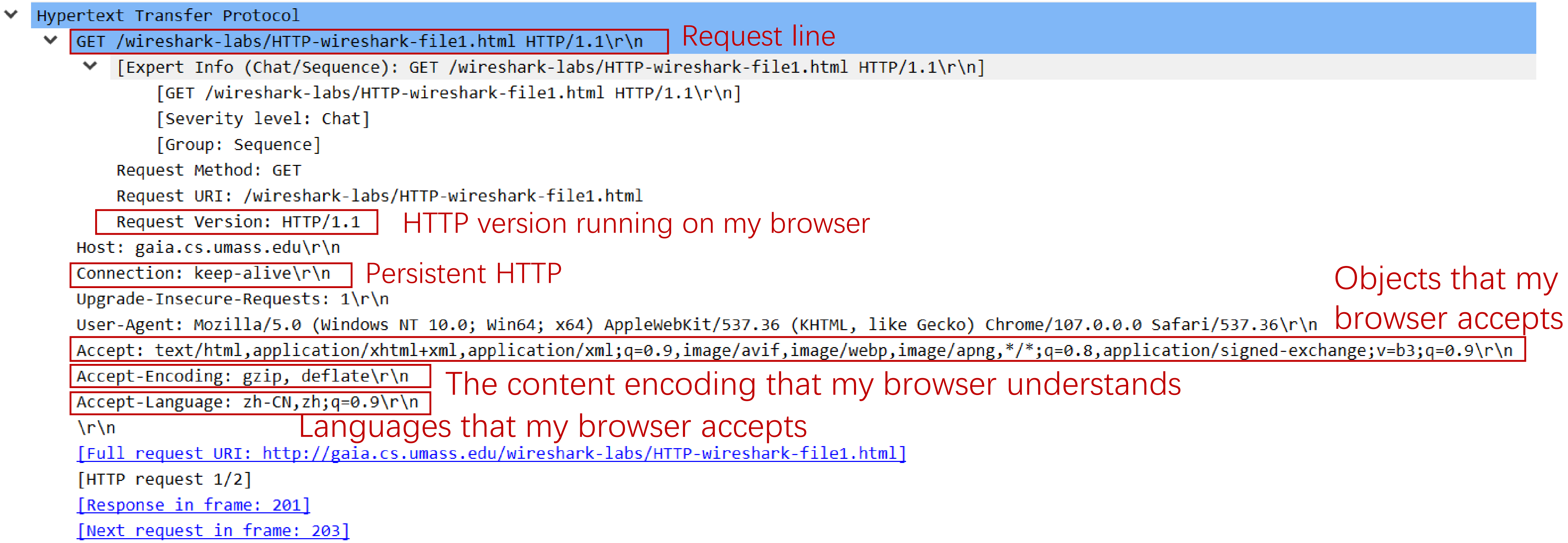


No.	Time	Source	Destination	Protocol	Length	Info
195	7.437500	10.79.27.165	128.119.245.12	HTTP	526	GET /wireshark-labs/HTTP-wireshark-file1.html HTTP/1.1
201	7.812866	128.119.245.12	10.79.27.165	HTTP	540	HTTP/1.1 200 OK (text/html)

```
> Frame 195: 526 bytes on wire (4208 bits), 526 bytes captured (4208 bits) on interface \Device\NPF_{714D37A8-0CF5-40EE-9574-E8DC5F06AE0A}, id 0
> Ethernet II, Src: IntelCor_d0:8d:0b (90:78:41:d0:8d:0b), Dst: All-HSRP-routers_f1 (00:00:0c:07:ac:f1)
> Internet Protocol Version 4, Src: 10.79.27.165, Dst: 128.119.245.12
> Transmission Control Protocol, Src Port: 10484, Dst Port: 80, Seq: 1, Ack: 1, Len: 472
v Hypertext Transfer Protocol
    v GET /wireshark-labs/HTTP-wireshark-file1.html HTTP/1.1\r\n
        v [Expert Info (Chat/Sequence): GET /wireshark-labs/HTTP-wireshark-file1.html HTTP/1.1\r\n]
              [GET /wireshark-labs/HTTP-wireshark-file1.html HTTP/1.1\r\n]
              [Severity level: Chat]
              [Group: Sequence]
          Request Method: GET
          Request URI: /wireshark-labs/HTTP-wireshark-file1.html
          Request Version: HTTP/1.1
      Host: gaia.cs.umass.edu\r\n
      Connection: keep-alive\r\n
      Upgrade-Insecure-Requests: 1\r\n
      User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/107.0.0.0 Safari/537.36\r\n
      Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9\r\n
      Accept-Encoding: gzip, deflate\r\n
      Accept-Language: zh-CN,zh;q=0.9\r\n
      \r\n
      [Full request URI: http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file1.html]
      [HTTP request 1/2]
      [Response in frame: 201]
      [Next request in frame: 203]
```

HTTP GET message

HTTP: HTTP message formats



```
⌄ Hypertext Transfer Protocol
  ⌄ GET /wireshark-labs/HTTP-wireshark-file1.html HTTP/1.1\r\n
    ⌄ [Expert Info (Chat/Sequence): GET /wireshark-labs/HTTP-wireshark-file1.html HTTP/1.1\r\n]
          [GET /wireshark-labs/HTTP-wireshark-file1.html HTTP/1.1\r\n]
          [Severity level: Chat]
          [Group: Sequence]
      Request Method: GET
      Request URI: /wireshark-labs/HTTP-wireshark-file1.html
      Request Version: HTTP/1.1
    Host: gaia.cs.umass.edu\r\n
    Connection: keep-alive\r\n
    Upgrade-Insecure-Requests: 1\r\n
    User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/107.0.0.0 Safari/537.36\r\n
    Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9\r\n
    Accept-Encoding: gzip, deflate\r\n
    Accept-Language: zh-CN,zh;q=0.9\r\n
    \r\n
    [Full request URI: http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file1.html]
    [HTTP request 1/2]
    [Response in frame: 201]
    [Next request in frame: 203]
```

Request line

HTTP version running on my browser

Persistent HTTP

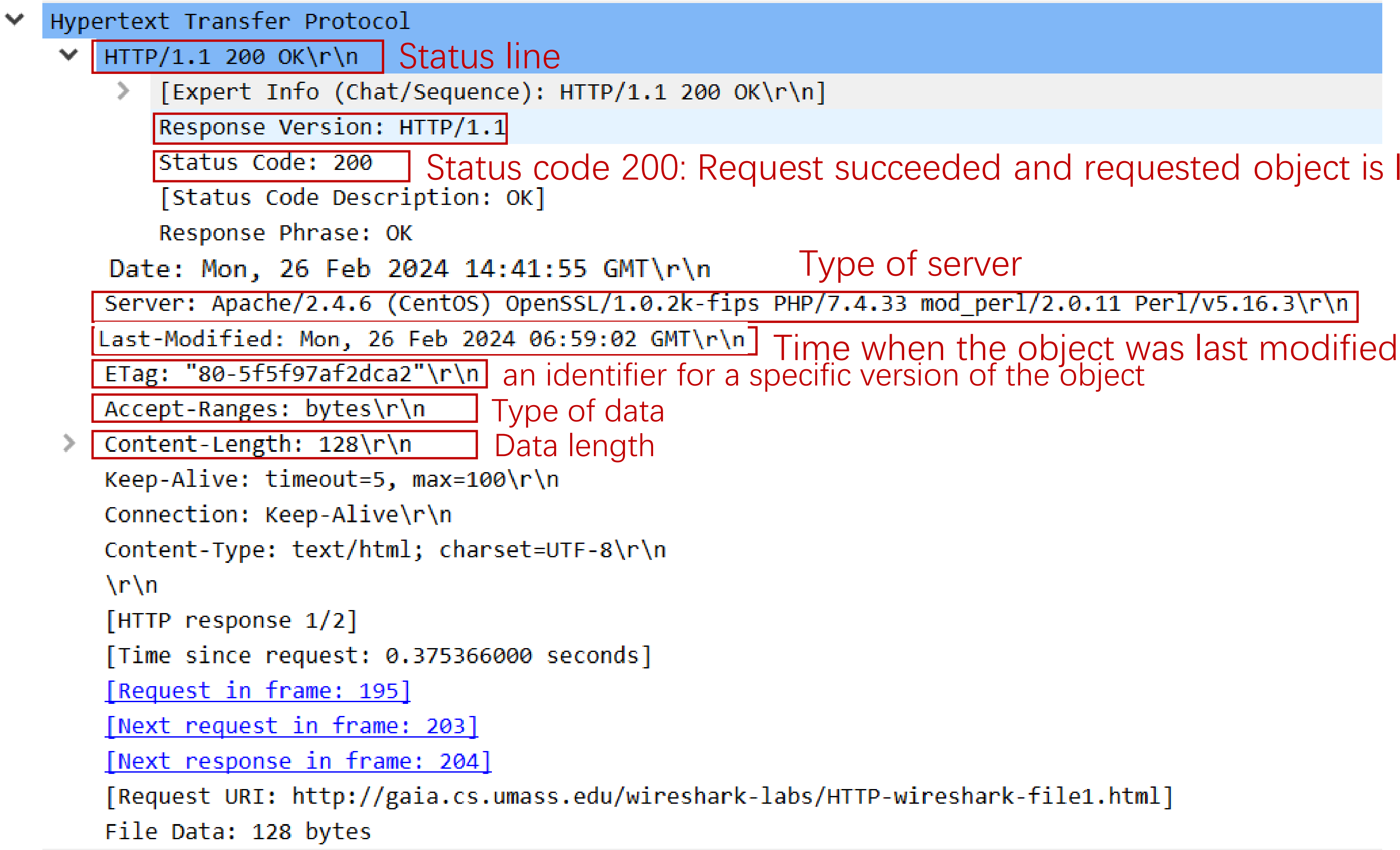
Objects that my browser accepts

The content encoding that my browser understands

Languages that my browser accepts

HTTP GET message

HTTP: HTTP message formats



```
Hypertext Transfer Protocol
  HTTP/1.1 200 OK\r\n
    [Expert Info (Chat/Sequence): HTTP/1.1 200 OK\r\n]
    Response Version: HTTP/1.1
    Status Code: 200
    [Status Code Description: OK]
    Response Phrase: OK
  Date: Mon, 26 Feb 2024 14:41:55 GMT\r\n
  Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips PHP/7.4.33 mod_perl/2.0.11 Perl/v5.16.3\r\n
  Last-Modified: Mon, 26 Feb 2024 06:59:02 GMT\r\n
  ETag: "80-5f5f97af2dca2"\r\n
  Accept-Ranges: bytes\r\n
  Content-Length: 128\r\n
  Keep-Alive: timeout=5, max=100\r\n
  Connection: Keep-Alive\r\n
  Content-Type: text/html; charset=UTF-8\r\n
  \r\n
  [HTTP response 1/2]
  [Time since request: 0.375366000 seconds]
  [Request in frame: 195]
  [Next request in frame: 203]
  [Next response in frame: 204]
  [Request URI: http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file1.html]
  File Data: 128 bytes
```

Status line

Status code 200: Request succeeded and requested object is later in this message

Type of server

Time when the object was last modified

an identifier for a specific version of the object

Type of data

Data length

HTTP response message

HTTP: retrieving Long Documents

1. Start up your web browser, and make sure your browser’s cache is cleared.

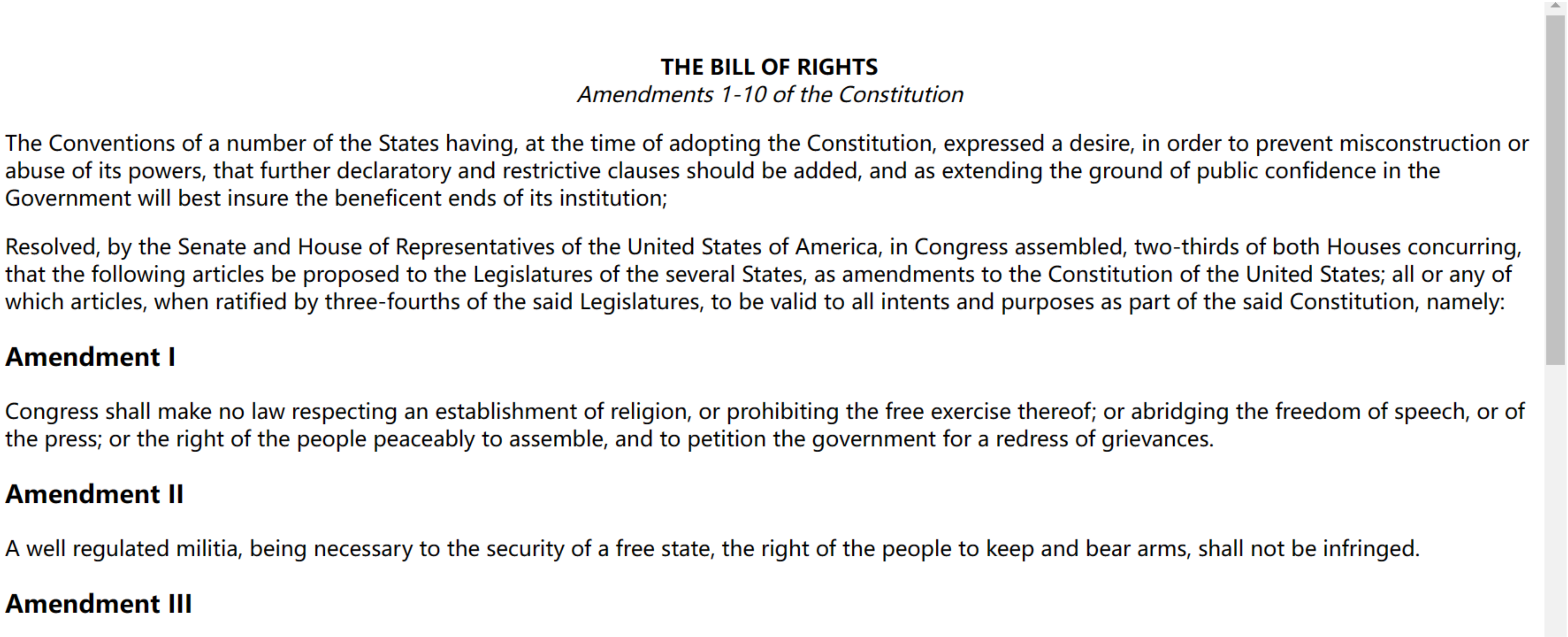
2. Start up the Wireshark packet sniffer

3. Enter the following URL into your browser: http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file3.html

Your browser should display the rather lengthy US Bill of Rights.

4. Stop Wireshark packet capture and enter “http” in the display-filter-specification window, so that only the captured HTTP messages are displayed.

HTTP: retrieving Long Documents



THE BILL OF RIGHTS
Amendments 1-10 of the Constitution

The Conventions of a number of the States having, at the time of adopting the Constitution, expressed a desire, in order to prevent misconstruction or abuse of its powers, that further declaratory and restrictive clauses should be added, and as extending the ground of public confidence in the Government will best insure the beneficent ends of its institution;

Resolved, by the Senate and House of Representatives of the United States of America, in Congress assembled, two-thirds of both Houses concurring, that the following articles be proposed to the Legislatures of the several States, as amendments to the Constitution of the United States; all or any of which articles, when ratified by three-fourths of the said Legislatures, to be valid to all intents and purposes as part of the said Constitution, namely:

Amendment I

Congress shall make no law respecting an establishment of religion, or prohibiting the free exercise thereof; or abridging the freedom of speech, or of the press; or the right of the people peaceably to assemble, and to petition the government for a redress of grievances.

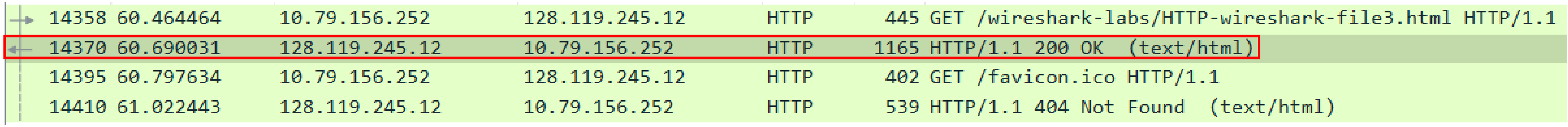
Amendment II

A well regulated militia, being necessary to the security of a free state, the right of the people to keep and bear arms, shall not be infringed.

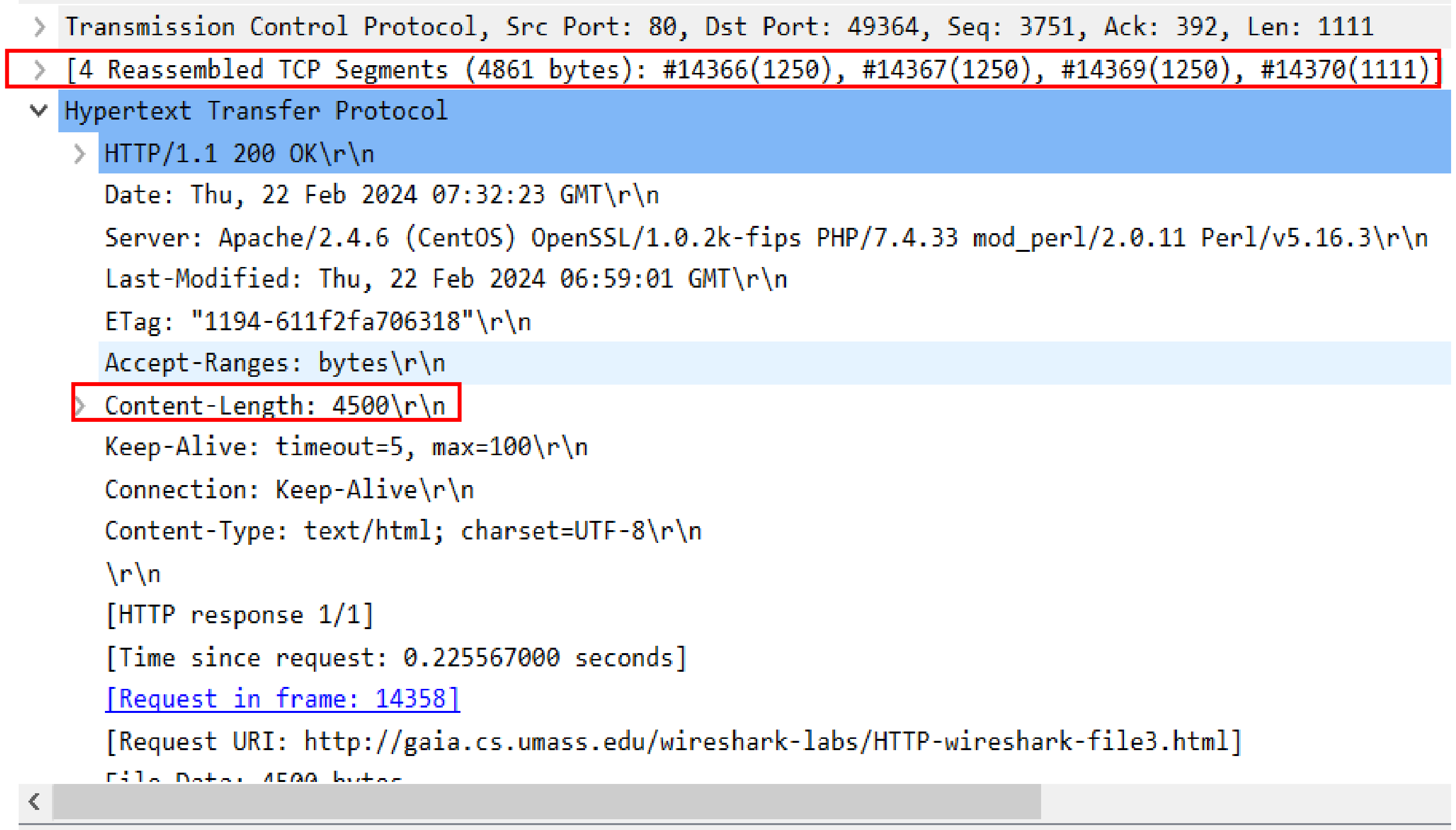
Amendment III

Web page

HTTP: retrieving Long Documents



```
14358 60.464464   10.79.156.252    128.119.245.12   HTTP    445 GET /wireshark-labs/HTTP-wireshark-file3.html HTTP/1.1
14370 60.690031   128.119.245.12   10.79.156.252    HTTP   1165 HTTP/1.1 200 OK  (text/html)
14395 60.797634   10.79.156.252    128.119.245.12   HTTP    402 GET /favicon.ico HTTP/1.1
14410 61.022443   128.119.245.12   10.79.156.252    HTTP    539 HTTP/1.1 404 Not Found  (text/html)
```



```
> Transmission Control Protocol, Src Port: 80, Dst Port: 49364, Seq: 3751, Ack: 392, Len: 1111
> [4 Reassembled TCP Segments (4861 bytes): #14366(1250), #14367(1250), #14369(1250), #14370(1111)]
v Hypertext Transfer Protocol
  > HTTP/1.1 200 OK\r\n
    Date: Thu, 22 Feb 2024 07:32:23 GMT\r\n
    Server: Apache/2.4.6 (CentOS) OpenSSL/1.0.2k-fips PHP/7.4.33 mod_perl/2.0.11 Perl/v5.16.3\r\n
    Last-Modified: Thu, 22 Feb 2024 06:59:01 GMT\r\n
    ETag: "1194-611f2fa706318"\r\n
    Accept-Ranges: bytes\r\n
  > Content-Length: 4500\r\n
    Keep-Alive: timeout=5, max=100\r\n
    Connection: Keep-Alive\r\n
    Content-Type: text/html; charset=UTF-8\r\n
    \r\n
    [HTTP response 1/1]
    [Time since request: 0.225567000 seconds]
    [Request in frame: 14358]
    [Request URI: http://gaia.cs.umass.edu/wireshark-labs/HTTP-wireshark-file3.html]
    File Data: 4500 bytes
```

Summary and Lab assignment: HTTP

- To conclude, we have studied
  - The basic GET/response interaction
  - The HTTP message formats
  - How to retrieve large HTML files
- Lab assignment:
  - follow the instructions in ‘Wireshark Lab: HTTP’ in section 2, 4 and 5 (The HTTP CONDITIONAL GET/response interaction, HTML Documents with Embedded Objects, HTTP Authentication) and answer the following questions: 9, 11, 16, 18, 19.
- Submit a **typed** response with cropped screen captures to the above questions through Canvas.

DNS: Domain Name System

- DNS
  - Distributed, hierarchical database of address/name translation (hostname to IP address translation, host aliasing, etc.)
  - Hosts, name servers communicate to resolve names
- DNS records
  - Type A: hostname, IP address
  - Type CNAME: alias name, canonical name
  - Type NS: domain, hostname of authoritative name server for this domain
- In this lab, we look at **the client side of DNS:** A client sends a query to its local DNS server, and receives a response containing the IP address
- We will learn how to analyze DNS queries using nslookup, ipconfig, and Wireshark

DNS : nslookup

- **nslookup** allows the host running it to query any specified DNS server (can be root, top-level domain, authoritative, etc.) for a DNS record
    - nslookup sends a DNS query to the specified DNS server, receives a DNS reply from that same DNS server, and displays the result
- General Syntax: **nslookup –option1 –option2 host-to-find dns-server**
    - nslookup can be run with zero(default: type=A), one, two or more options

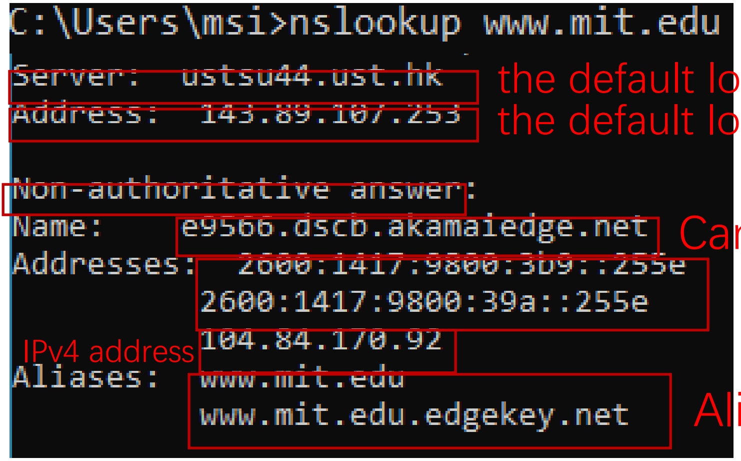
      eg. option -type=NS: query type-NS (name server record) from local DNS server
    - **host-to-find**: the host to find the DNS record for
    - **dns-server**(optional): the DNS server to send the query to

      default: the default local DNS server
- Run nslookup
    - In Linux/Unix: type the nslookup command on the command line
    - In Windows: open the Command Prompt and run nslookup on the command line

DNS : nslookup examples (in Windows)

- nslookup www.mit.edu

  Request IP address of ‘www.mit.edu’ from the default local DNS server



```
C:\Users\msi>nslookup www.mit.edu
Server:   ustsu44.ust.hk
Address:  143.89.107.253

Non-authoritative answer:
Name:    e9566.dscb.akamaiedge.net
Addresses:  2600:1417:9800:3b9::255e
          2600:1417:9800:39a::255e
          104.84.170.92
Aliases:  www.mit.edu
          www.mit.edu.edgekey.net
```

the DNS server that provides the answer

the default local DNS server name

the default local DNS server address

this answer came from the cache of some server rather than from an authoritative DNS server

Canonical name

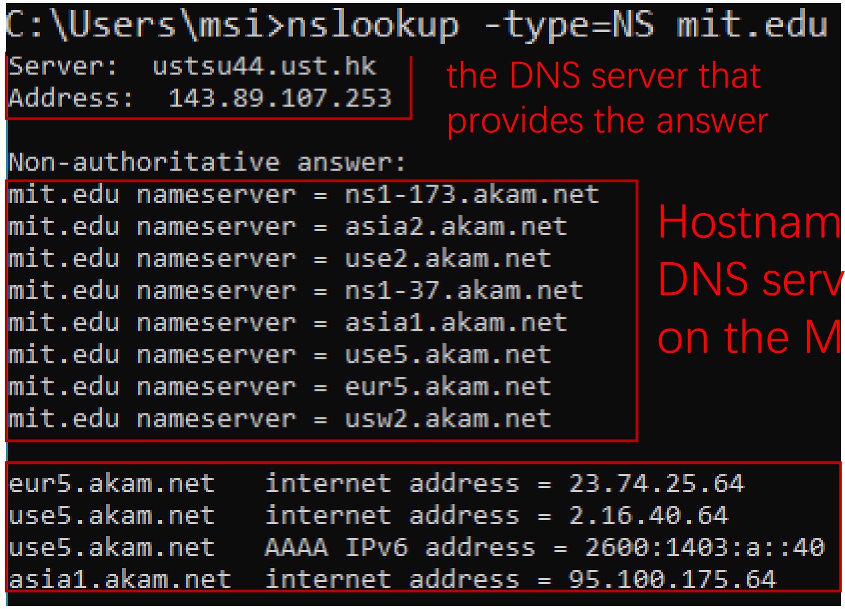
IPv6 addresses

IPv4 address

Alias names

- nslookup –type=NS mit.edu

  Request type-NS record (the hostnames of the authoritative DNS) for ‘mit.edu’ from the default local DNS server



```
C:\Users\msi>nslookup -type=NS mit.edu
Server:  ustsu44.ust.hk
Address:  143.89.107.253

Non-authoritative answer:
mit.edu nameserver = ns1-173.akam.net
mit.edu nameserver = asia2.akam.net
mit.edu nameserver = use2.akam.net
mit.edu nameserver = ns1-37.akam.net
mit.edu nameserver = asia1.akam.net
mit.edu nameserver = use5.akam.net
mit.edu nameserver = eur5.akam.net
mit.edu nameserver = usw2.akam.net

eur5.akam.net   internet address = 23.74.25.64
use5.akam.net   internet address = 2.16.40.64
use5.akam.net   AAAA IPv6 address = 2600:1403:a::40
asia1.akam.net  internet address = 95.100.175.64
```

the DNS server that provides the answer

Hostnames of authoritative DNS servers for the hosts on the MIT campus

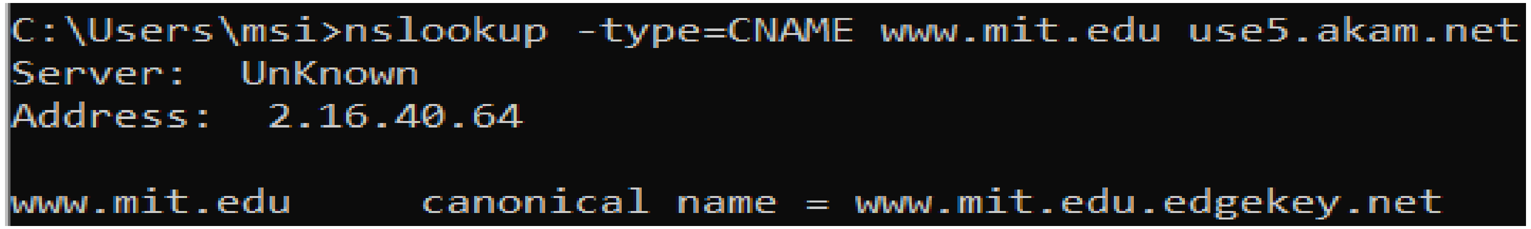
IP addresses of the authoritative DNS servers at MIT (return “for free”)

DNS : nslookup examples (in Windows)

- nslookup –type=CNAME www.mit.edu use5.akam.net

  Request type-CNAME record (canonical name) for ‘www.mit.edu’ from the DNS server use5.akam.net

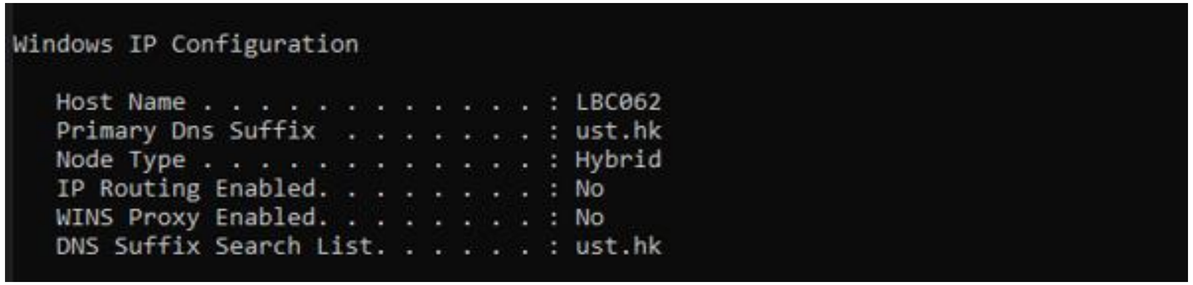
  (the query and reply transaction takes place directly between our querying host and use5.akam.net)



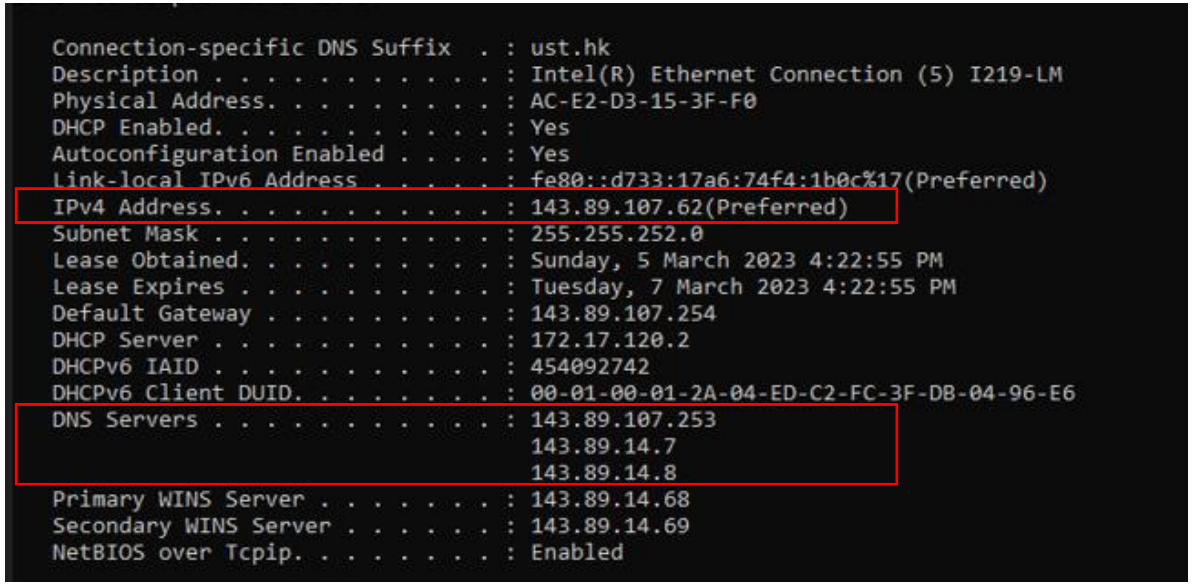
```
C:\Users\msi>nslookup -type=CNAME www.mit.edu use5.akam.net
Server:  UnKnown
Address:  2.16.40.64

www.mit.edu     canonical name = www.mit.edu.edgekey.net
```

DNS: ipconfig

- **ipconfig** (for windows) and **ifconfig** (for Linux/Mac OS) are used to show your current TCP/IP information, including your IP address, DNS address, adapter type, etc.
- **ipconfig /all**: show all the information about each network adapter (**ifconfig** for Linux or Mac OS)
  - IPv4 address
  - DNS Servers



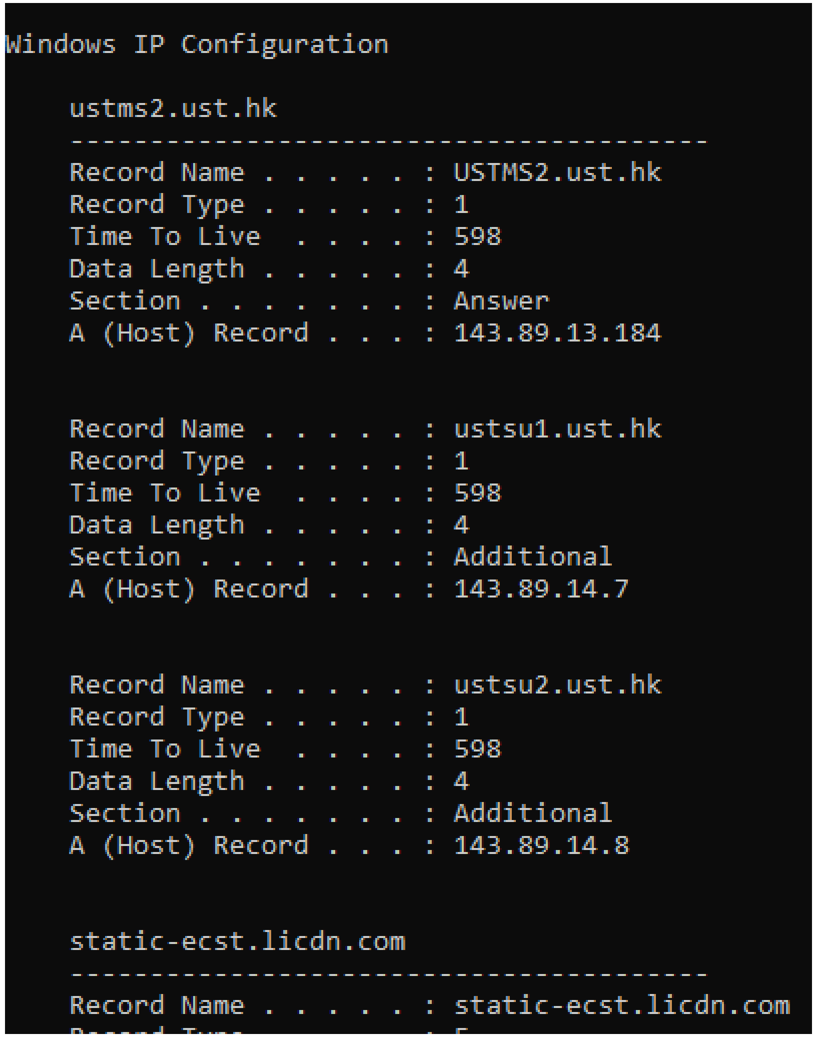
```
Windows IP Configuration

   Host Name . . . . . . . . . . . . : LBC062
   Primary Dns Suffix  . . . . . . . : ust.hk
   Node Type . . . . . . . . . . . . : Hybrid
   IP Routing Enabled. . . . . . . . : No
   WINS Proxy Enabled. . . . . . . . : No
   DNS Suffix Search List. . . . . . : ust.hk
```



```
   Connection-specific DNS Suffix  . : ust.hk
   Description . . . . . . . . . . . : Intel(R) Ethernet Connection (5) I219-LM
   Physical Address. . . . . . . . . : AC-E2-D3-15-3F-F0
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes
   Link-local IPv6 Address . . . . . : fe80::d733:17a6:74f4:1b0c%17(Preferred)
   IPv4 Address. . . . . . . . . . . : 143.89.107.62(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.252.0
   Lease Obtained. . . . . . . . . . : Sunday, 5 March 2023 4:22:55 PM
   Lease Expires . . . . . . . . . . : Tuesday, 7 March 2023 4:22:55 PM
   Default Gateway . . . . . . . . . : 143.89.107.254
   DHCP Server . . . . . . . . . . . : 172.17.120.2
   DHCPv6 IAID . . . . . . . . . . . : 454092742
   DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-2A-04-ED-C2-FC-3F-DB-04-96-E6
   DNS Servers . . . . . . . . . . . : 143.89.107.253
                                       143.89.14.7
                                       143.89.14.8
   Primary WINS Server . . . . . . . : 143.89.14.68
   Secondary WINS Server . . . . . . : 143.89.14.69
   NetBIOS over Tcpip. . . . . . . . : Enabled
```

DNS: ipconfig

- **ipconfig /displaydns**: show the cached DNS records (record name, value, TTL) on your host
- **ipconfig /flushdns**: clear the DNS cache
  - **sudo killall -HUP mDNSResponder** for Mac
- We will be using this in exercises to check our IP address & clear DNS cache



```
Windows IP Configuration

    ustms2.ust.hk
    ----------------------------------------
    Record Name . . . . . : USTMS2.ust.hk
    Record Type . . . . . : 1
    Time To Live  . . . . : 598
    Data Length . . . . . : 4
    Section . . . . . . . : Answer
    A (Host) Record . . . : 143.89.13.184


    Record Name . . . . . : ustsu1.ust.hk
    Record Type . . . . . : 1
    Time To Live  . . . . : 598
    Data Length . . . . . : 4
    Section . . . . . . . : Additional
    A (Host) Record . . . : 143.89.14.7


    Record Name . . . . . : ustsu2.ust.hk
    Record Type . . . . . : 1
    Time To Live  . . . . : 598
    Data Length . . . . . : 4
    Section . . . . . . . : Additional
    A (Host) Record . . . : 143.89.14.8


    static-ecst.licdn.com
    ----------------------------------------
    Record Name . . . . . : static-ecst.licdn.com
    Record Type . . . . . : 5
```

DNS: tracing DNS packets with Wireshark

- Capture DNS packets generated by ordinary Web-surfing activity.

1. Use ipconfig to empty the DNS cache in your host.

2. Open your browser and empty your browser cache.

   (With Internet Explorer, go to Tools menu and select Internet Options; then in the General tab select Delete Files.)

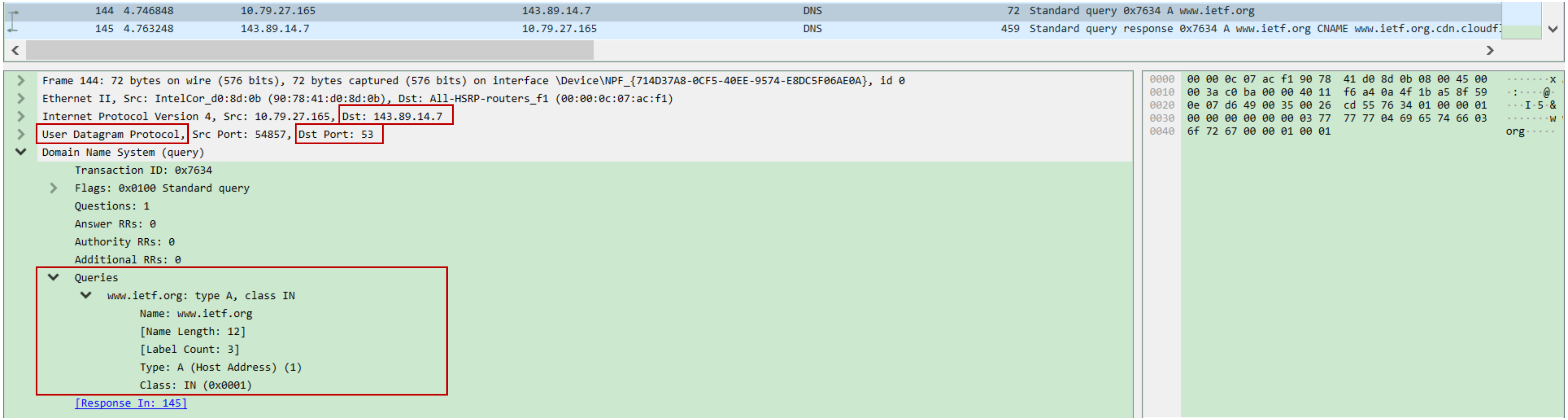
3. Open Wireshark and enter “ip.addr == your_IP_address” into the filter, where you obtain your_IP_address with ipconfig.

4. Start packet capture in Wireshark.

5. With your browser, visit the Web page: http://www.ietf.org

6. Stop packet capture.

DNS: tracing DNS packets with Wireshark



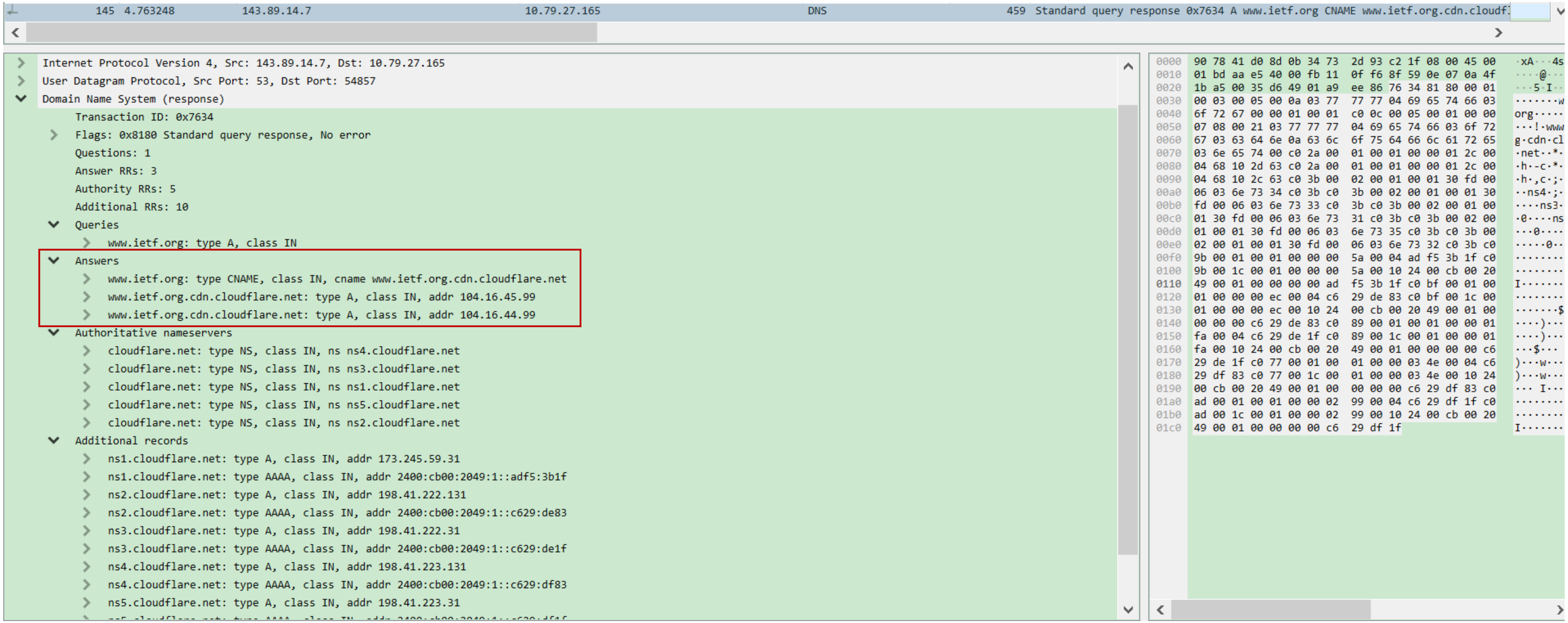
```
144  4.746848   10.79.27.165    143.89.14.7     DNS    72 Standard query 0x7634 A www.ietf.org
145  4.763248   143.89.14.7     10.79.27.165    DNS   459 Standard query response 0x7634 A www.ietf.org CNAME www.ietf.org.cdn.cloudfl
```

```
> Frame 144: 72 bytes on wire (576 bits), 72 bytes captured (576 bits) on interface \Device\NPF_{714D37A8-0CF5-40EE-9574-E8DC5F06AE0A}, id 0
> Ethernet II, Src: IntelCor_d0:8d:0b (90:78:41:d0:8d:0b), Dst: All-HSRP-routers_f1 (00:00:0c:07:ac:f1)
> Internet Protocol Version 4, Src: 10.79.27.165, Dst: 143.89.14.7
> User Datagram Protocol, Src Port: 54857, Dst Port: 53
v Domain Name System (query)
      Transaction ID: 0x7634
    > Flags: 0x0100 Standard query
      Questions: 1
      Answer RRs: 0
      Authority RRs: 0
      Additional RRs: 0
    v Queries
        v www.ietf.org: type A, class IN
              Name: www.ietf.org
              [Name Length: 12]
              [Label Count: 3]
              Type: A (Host Address) (1)
              Class: IN (0x0001)
      [Response In: 145]
```

```
0000  00 00 0c 07 ac f1 90 78  41 d0 8d 0b 08 00 45 00   ·······x
0010  00 3a c0 ba 00 00 40 11  f6 a4 0a 4f 1b a5 8f 59   ·:····@·
0020  0e 07 d6 49 00 35 00 26  cd 55 76 34 01 00 00 01   ···I·5·&
0030  00 00 00 00 00 00 03 77  77 77 04 69 65 74 66 03   ·······w
0040  6f 72 67 00 00 01 00 01                           org·····
```

DNS query message

DNS: tracing DNS packets with Wireshark



```
145  4.763248    143.89.14.7    10.79.27.165    DNS    459  Standard query response 0x7634 A www.ietf.org CNAME www.ietf.org.cdn.cloudf]
```

```
> Internet Protocol Version 4, Src: 143.89.14.7, Dst: 10.79.27.165
> User Datagram Protocol, Src Port: 53, Dst Port: 54857
v Domain Name System (response)
      Transaction ID: 0x7634
    > Flags: 0x8180 Standard query response, No error
      Questions: 1
      Answer RRs: 3
      Authority RRs: 5
      Additional RRs: 10
    v Queries
        > www.ietf.org: type A, class IN
    v Answers
        > www.ietf.org: type CNAME, class IN, cname www.ietf.org.cdn.cloudflare.net
        > www.ietf.org.cdn.cloudflare.net: type A, class IN, addr 104.16.45.99
        > www.ietf.org.cdn.cloudflare.net: type A, class IN, addr 104.16.44.99
    v Authoritative nameservers
        > cloudflare.net: type NS, class IN, ns ns4.cloudflare.net
        > cloudflare.net: type NS, class IN, ns ns3.cloudflare.net
        > cloudflare.net: type NS, class IN, ns ns1.cloudflare.net
        > cloudflare.net: type NS, class IN, ns ns5.cloudflare.net
        > cloudflare.net: type NS, class IN, ns ns2.cloudflare.net
    v Additional records
        > ns1.cloudflare.net: type A, class IN, addr 173.245.59.31
        > ns1.cloudflare.net: type AAAA, class IN, addr 2400:cb00:2049:1::adf5:3b1f
        > ns2.cloudflare.net: type A, class IN, addr 198.41.222.131
        > ns2.cloudflare.net: type AAAA, class IN, addr 2400:cb00:2049:1::c629:de83
        > ns3.cloudflare.net: type A, class IN, addr 198.41.222.31
        > ns3.cloudflare.net: type AAAA, class IN, addr 2400:cb00:2049:1::c629:de1f
        > ns4.cloudflare.net: type A, class IN, addr 198.41.223.131
        > ns4.cloudflare.net: type AAAA, class IN, addr 2400:cb00:2049:1::c629:df83
        > ns5.cloudflare.net: type A, class IN, addr 198.41.223.31
        > ns5.cloudflare.net: type AAAA, class IN, addr 2400:cb00:2049:1::c629:df1f
```

```
0000  90 78 41 d0 8d 0b 34 73  2d 93 c2 1f 08 00 45 00   ·xA···4s
0010  01 bd aa e5 40 00 fb 11  0f f6 8f 59 0e 07 0a 4f   ····@···
0020  1b a5 00 35 d6 49 01 a9  ee 86 76 34 81 80 00 01   ···5·I··
0030  00 03 00 05 00 0a 03 77  77 77 04 69 65 74 66 03   ·······w
0040  6f 72 67 00 00 01 00 01  c0 0c 00 05 00 01 00 00   org·····
0050  07 08 00 21 03 77 77 77  04 69 65 74 66 03 6f 72   ···!·www
0060  67 03 63 64 6e 0a 63 6c  6f 75 64 66 6c 61 72 65   g·cdn·cl
0070  03 6e 65 74 00 c0 2a 00  01 00 01 00 00 01 2c 00   ·net··*·
0080  04 68 10 2d 63 c0 2a 00  01 00 01 00 00 01 2c 00   ·h·-c·*·
0090  04 68 10 2c 63 c0 3b 00  02 00 01 00 01 30 fd 00   ·h·,c·;·
00a0  06 03 6e 73 34 c0 3b c0  3b 00 02 00 01 00 01 30   ··ns4·;·
00b0  fd 00 06 03 6e 73 33 c0  3b c0 3b 00 02 00 01 00   ····ns3·
00c0  01 30 fd 00 06 03 6e 73  31 c0 3b c0 3b 00 02 00   ·0····ns
00d0  01 00 01 30 fd 00 06 03  6e 73 35 c0 3b c0 3b 00   ···0····
00e0  02 00 01 00 01 30 fd 00  06 03 6e 73 32 c0 3b c0   ·····0··
00f0  9b 00 01 00 01 00 00 00  5a 00 04 ad f5 3b 1f c0   ········
0100  9b 00 1c 00 01 00 00 00  5a 00 10 24 00 cb 00 20   ········
0110  49 00 01 00 00 00 00 ad  f5 3b 1f c0 bf 00 01 00   I·······
0120  01 00 00 00 ec 00 04 c6  29 de 83 c0 bf 00 1c 00   ········
0130  01 00 00 00 ec 00 10 24  00 cb 00 20 49 00 01 00   ·······$
0140  00 00 00 c6 29 de 83 c0  89 00 01 00 01 00 00 01   ····)···
0150  fa 00 04 c6 29 de 1f c0  89 00 1c 00 01 00 00 01   ····)···
0160  fa 00 10 24 00 cb 00 20  49 00 01 00 00 00 00 c6   ···$···
0170  29 de 1f c0 77 00 01 00  01 00 00 03 4e 00 04 c6   )···w···
0180  29 df 83 c0 77 00 1c 00  01 00 00 03 4e 00 10 24   )···w···
0190  00 cb 00 20 49 00 01 00  00 00 00 c6 29 df 83 c0   ··· I···
01a0  ad 00 01 00 01 00 00 02  99 00 04 c6 29 df 1f c0   ········
01b0  ad 00 1c 00 01 00 00 02  99 00 10 24 00 cb 00 20   ········
01c0  49 00 01 00 00 00 00 c6  29 df 1f                  I·······
```

DNS response message

Summary and Lab assignment: DNS

- To conclude, we have seen
  - A review of DNS
  - How to use nslookup to send DNS queries
  - How to use ipconfig (or ifconfig) to check IP addresses and DNS servers
  - Example of tracing DNS packets with Wireshark

- Lab assignment:
  - For DNS, follow the instructions in ‘Wireshark Lab: DNS’ in section 3 (Tracing DNS with Wireshark) and answer the following questions: 16, 17, 18, 19.

- Submit a **typed** response with cropped screen captures to the above questions through Canvas.