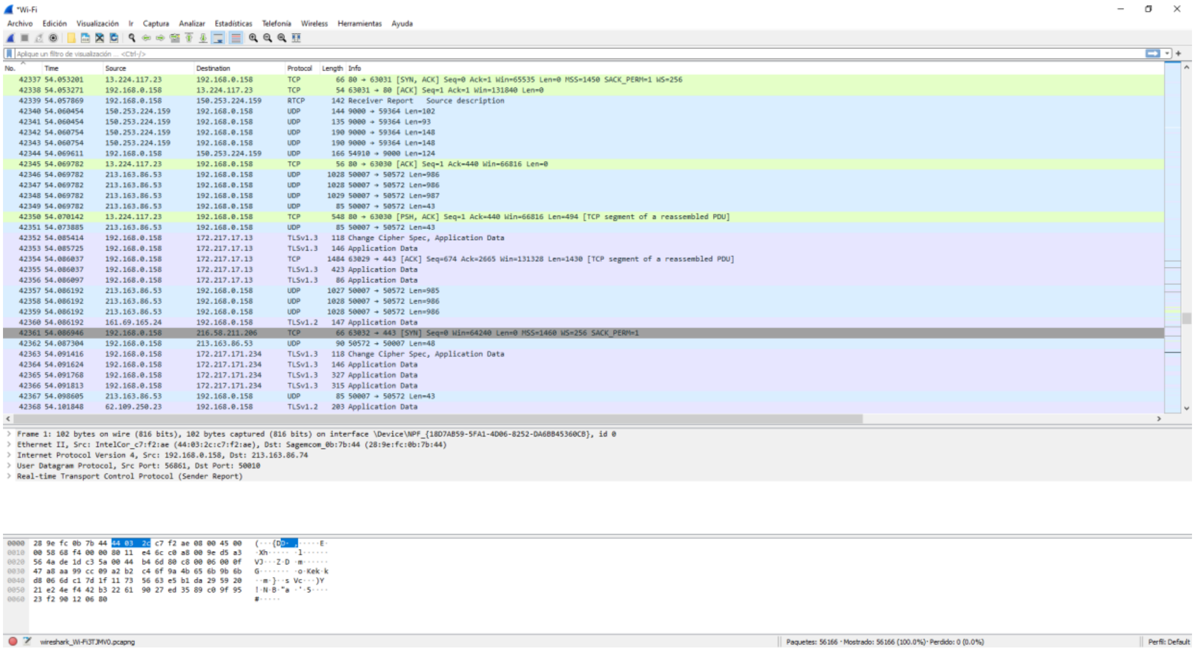
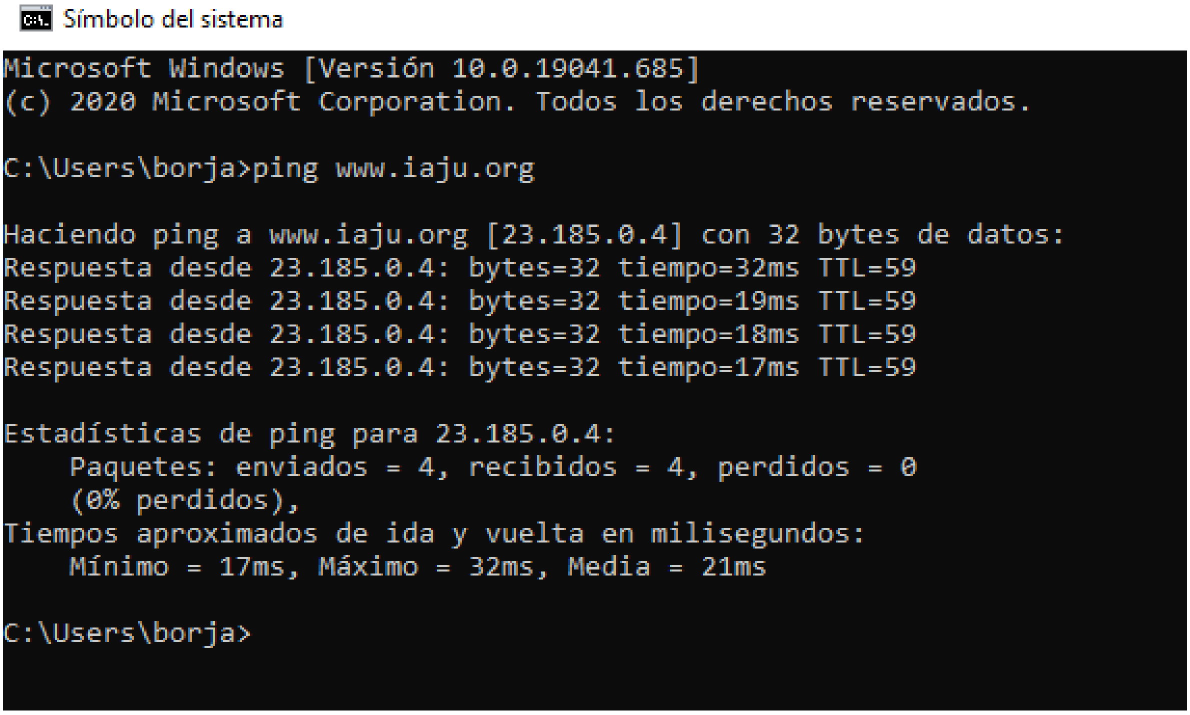
**LAB 2: Name:** Laura Ferrer Haba.

For this first step we will make a screenshot of Wireshark and it will be our first packet capture. First, we will have to open Wireshark, go to our active interface and we will make a screenshot of the *ping* [www.iaju.org](http://www.iaju.org). For doing the ping we have to go to shell of Windows (cmd).

## Question 1:

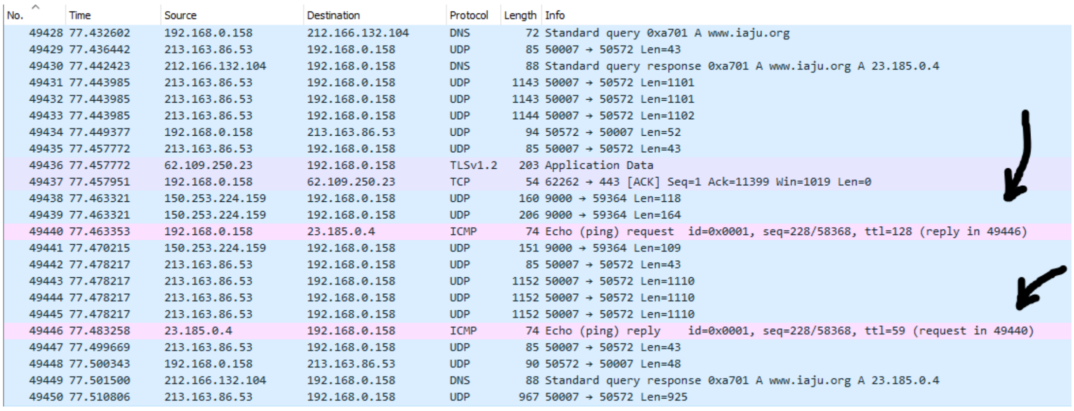

*Wi-Fi
Archivo Edición Visualización Ir Captura Analizar Estadísticas Telefonía Wireless Herramientas Ayuda
Aplique un filtro de visualización ... <Ctrl-/>

```
No.    Time      Source           Destination      Protocol Length Info
 42337 54.053201 13.224.117.23    192.168.0.158    TCP        66 80 → 63031 [SYN, ACK] Seq=0 Ack=1 Win=65535 Len=0 MSS=1450 SACK_PERM=1 WS=256
 42338 54.053271 192.168.0.158    13.224.117.23    TCP        54 63031 → 80 [ACK] Seq=1 Ack=1 Win=131840 Len=0
 42339 54.057869 192.168.0.158    150.253.224.159  RTCP      142 Receiver Report   Source description
 42340 54.060454 150.253.224.159  192.168.0.158    UDP       144 9000 → 59364 Len=102
 42341 54.060454 150.253.224.159  192.168.0.158    UDP       135 9000 → 59364 Len=93
 42342 54.060754 150.253.224.159  192.168.0.158    UDP       190 9000 → 59364 Len=148
 42343 54.060754 150.253.224.159  192.168.0.158    UDP       190 9000 → 59364 Len=148
 42344 54.069611 192.168.0.158    150.253.224.159  UDP       166 54910 → 9000 Len=124
 42345 54.069782 13.224.117.23    192.168.0.158    TCP        56 80 → 63030 [ACK] Seq=1 Ack=440 Win=66816 Len=0
 42346 54.069782 213.163.86.53    192.168.0.158    UDP      1028 50007 → 50572 Len=986
 42347 54.069782 213.163.86.53    192.168.0.158    UDP      1028 50007 → 50572 Len=986
 42348 54.069782 213.163.86.53    192.168.0.158    UDP      1029 50007 → 50572 Len=987
 42349 54.069782 213.163.86.53    192.168.0.158    UDP        85 50007 → 50572 Len=43
 42350 54.070142 13.224.117.23    192.168.0.158    TCP       548 80 → 63030 [PSH, ACK] Seq=1 Ack=440 Win=66816 Len=494 [TCP segment of a reassembled PDU]
 42351 54.073885 213.163.86.53    192.168.0.158    UDP        85 50007 → 50572 Len=43
 42352 54.085414 192.168.0.158    172.217.17.13    TLSv1.3   118 Change Cipher Spec, Application Data
 42353 54.085725 192.168.0.158    172.217.17.13    TLSv1.3   146 Application Data
 42354 54.086037 192.168.0.158    172.217.17.13    TCP      1484 63029 → 443 [ACK] Seq=674 Ack=2665 Win=131328 Len=1430 [TCP segment of a reassembled PDU]
 42355 54.086037 192.168.0.158    172.217.17.13    TLSv1.3   423 Application Data
 42356 54.086097 192.168.0.158    172.217.17.13    TLSv1.3    86 Application Data
 42357 54.086192 213.163.86.53    192.168.0.158    UDP      1027 50007 → 50572 Len=985
 42358 54.086192 213.163.86.53    192.168.0.158    UDP      1028 50007 → 50572 Len=986
 42359 54.086192 213.163.86.53    192.168.0.158    UDP      1028 50007 → 50572 Len=986
 42360 54.086192 161.69.165.24    192.168.0.158    TLSv1.2   147 Application Data
 42361 54.086946 192.168.0.158    216.58.211.206   TCP        66 63032 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM=1
 42362 54.087304 192.168.0.158    213.163.86.53    UDP        90 50572 → 50007 Len=48
 42363 54.091416 192.168.0.158    172.217.171.234  TLSv1.3   118 Change Cipher Spec, Application Data
 42364 54.091624 192.168.0.158    172.217.171.234  TLSv1.3   146 Application Data
 42365 54.091768 192.168.0.158    172.217.171.234  TLSv1.3   327 Application Data
 42366 54.091813 192.168.0.158    172.217.171.234  TLSv1.3   315 Application Data
 42367 54.098605 213.163.86.53    192.168.0.158    UDP        85 50007 → 50572 Len=43
 42368 54.101848 62.109.250.23    192.168.0.158    TLSv1.2   203 Application Data

> Frame 1: 102 bytes on wire (816 bits), 102 bytes captured (816 bits) on interface \Device\NPF_{18D7AB59-5FA1-4D06-8252-DA6BB45360CB}, id 0
> Ethernet II, Src: IntelCor_c7:f2:ae (44:03:2c:c7:f2:ae), Dst: Sagemcom_0b:7b:44 (28:9e:fc:0b:7b:44)
> Internet Protocol Version 4, Src: 192.168.0.158, Dst: 213.163.86.74
> User Datagram Protocol, Src Port: 56861, Dst Port: 50010
> Real-time Transport Control Protocol (Sender Report)

0000  28 9e fc 0b 7b 44 44 03  2c c7 f2 ae 08 00 45 00   (···{DD· ,·····E·
0010  00 58 68 f4 00 00 80 11  e4 6c c0 a8 00 9e d5 a3   ·Xh····· ·l······
0020  56 4a de 1d c3 5a 00 44  b4 6d 80 c8 00 06 00 0f   VJ···Z·D ·m······
0030  47 a8 aa 99 cc 09 a2 b2  c4 6f 9a 4b 65 6b 9b 6b   G······· ·o·Kek·k
0040  d8 06 6d c1 7d 1f 11 73  56 63 e5 b1 da 29 59 20   ··m·}··s Vc···)Y 
0050  21 e2 4e f4 42 b3 22 61  90 27 ed 35 89 c0 9f 95   !·N·B·"a ·'·5····
0060  23 f2 90 12 06 80                                  #·····

wireshark_Wi-Fi3TJMV0.pcapng     Paquetes: 56166 · Mostrado: 56166 (100.0%) · Perdido: 0 (0.0%)     Perfil: Default
```



```
Microsoft Windows [Versión 10.0.19041.685]
(c) 2020 Microsoft Corporation. Todos los derechos reservados.

C:\Users\borja>ping www.iaju.org

Haciendo ping a www.iaju.org [23.185.0.4] con 32 bytes de datos:
Respuesta desde 23.185.0.4: bytes=32 tiempo=32ms TTL=59
Respuesta desde 23.185.0.4: bytes=32 tiempo=19ms TTL=59
Respuesta desde 23.185.0.4: bytes=32 tiempo=18ms TTL=59
Respuesta desde 23.185.0.4: bytes=32 tiempo=17ms TTL=59

Estadísticas de ping para 23.185.0.4:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 17ms, Máximo = 32ms, Media = 21ms

C:\Users\borja>
```

When we make the ping to [www.iaju.org](http://www.iaju.org), we are making ping to the IP 23.185.0.4. This direction must appear in the packet capture. In the next screenshot we can see the ping.



```
No.   Time      Source          Destination     Protocol Length Info
49428 77.432602 192.168.0.158   212.166.132.104 DNS        72 Standard query 0xa701 A www.iaju.org
49429 77.436442 213.163.86.53   192.168.0.158   UDP        85 50007 → 50572 Len=43
49430 77.442423 212.166.132.104 192.168.0.158   DNS        88 Standard query response 0xa701 A www.iaju.org A 23.185.0.4
49431 77.443985 213.163.86.53   192.168.0.158   UDP      1143 50007 → 50572 Len=1101
49432 77.443985 213.163.86.53   192.168.0.158   UDP      1143 50007 → 50572 Len=1101
49433 77.443985 213.163.86.53   192.168.0.158   UDP      1144 50007 → 50572 Len=1102
49434 77.449377 192.168.0.158   213.163.86.53   UDP        94 50572 → 50007 Len=52
49435 77.457772 213.163.86.53   192.168.0.158   UDP        85 50007 → 50572 Len=43
49436 77.457772 62.109.250.23   192.168.0.158   TLSv1.2   203 Application Data
49437 77.457951 192.168.0.158   62.109.250.23   TCP        54 62262 → 443 [ACK] Seq=1 Ack=11399 Win=1019 Len=0
49438 77.463321 150.253.224.159 192.168.0.158   UDP       160 9000 → 59364 Len=118
49439 77.463321 150.253.224.159 192.168.0.158   UDP       206 9000 → 59364 Len=164
49440 77.463353 192.168.0.158   23.185.0.4      ICMP       74 Echo (ping) request  id=0x0001, seq=228/58368, ttl=128 (reply in 49446)
49441 77.470215 150.253.224.159 192.168.0.158   UDP       151 9000 → 59364 Len=109
49442 77.478217 213.163.86.53   192.168.0.158   UDP        85 50007 → 50572 Len=43
49443 77.478217 213.163.86.53   192.168.0.158   UDP      1152 50007 → 50572 Len=1110
49444 77.478217 213.163.86.53   192.168.0.158   UDP      1152 50007 → 50572 Len=1110
49445 77.478217 213.163.86.53   192.168.0.158   UDP      1152 50007 → 50572 Len=1110
49446 77.483258 23.185.0.4      192.168.0.158   ICMP       74 Echo (ping) reply    id=0x0001, seq=228/58368, ttl=59 (request in 49440)
49447 77.499669 213.163.86.53   192.168.0.158   UDP        85 50007 → 50572 Len=43
49448 77.500343 192.168.0.158   213.163.86.53   UDP        90 50572 → 50007 Len=48
49449 77.501500 212.166.132.104 192.168.0.158   DNS        88 Standard query response 0xa701 A www.iaju.org A 23.185.0.4
49450 77.510806 213.163.86.53   192.168.0.158   UDP       967 50007 → 50572 Len=925
```

## Question 2: What information looks familiar to you?

There are several protocols that looks familiar to me because we learnt it in class.

a. ***UDP** (User Datagram Protocol)*: is one of the tow protocols that the Internet has, this is a *connectionless* protocol. It sends packets between applications, letting applications build their own protocols on top as needed.

b. ***TCP** (Transmission Control Protocol)*: is the other protocol that the Internet has, this is a *connection-oriented* protocol. It makes connections and adds reliability with retransmissions, along with flow control and congestion control.

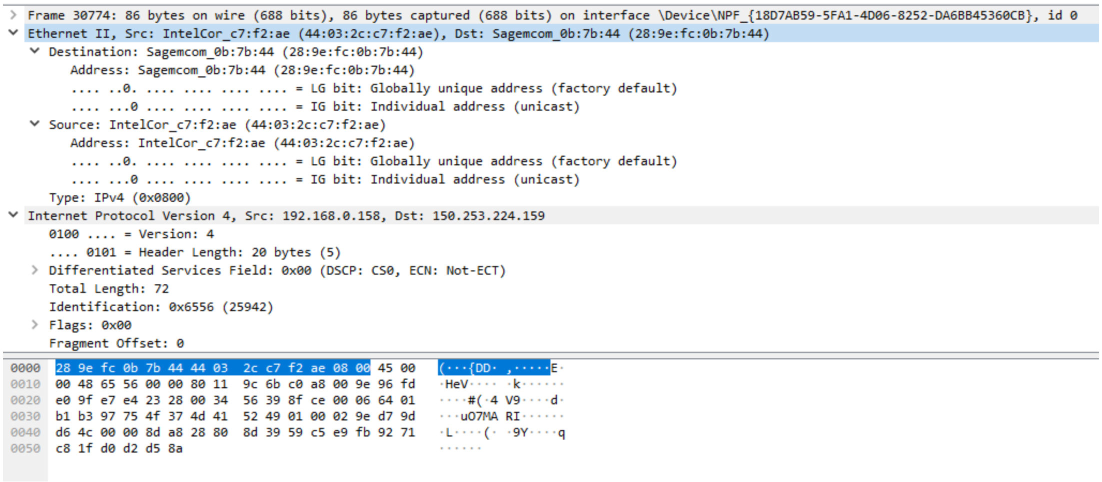
c. ***DNS** (Domain Name System)*: this protocol translate the computers names to IP addresses.

d. ***RTCP** (Real-Time Transport Control Protocol)*: this protocol provide feedback on delay, it does not transport any media samples and handles feedback, synchronization, and the user interface.

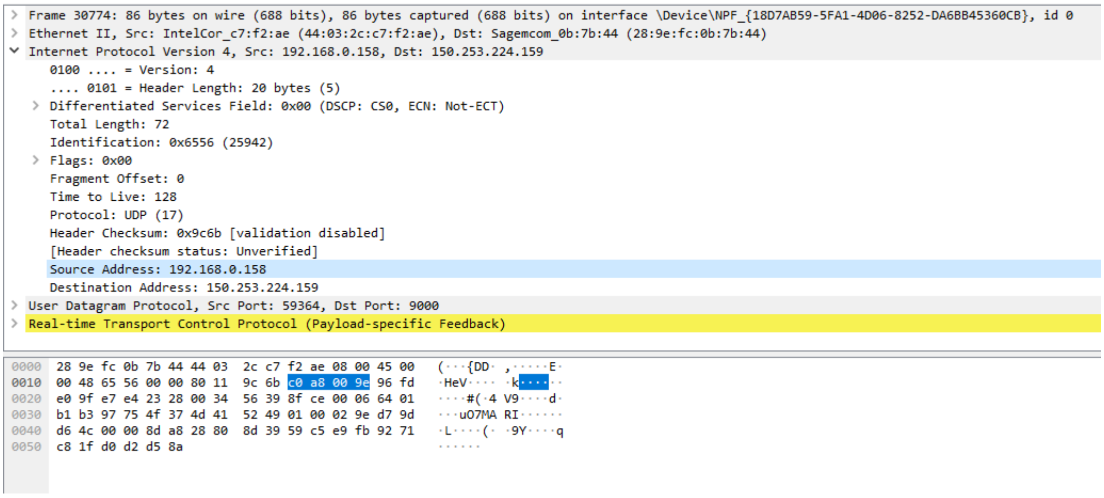
## Question 3: What information does not look familiar?

The protocols that we do not now are the *ARP, ICMP, ICMPv6, IGMPv6, MDNS, STUN* and *TLSv1.2.*

## Question 4:

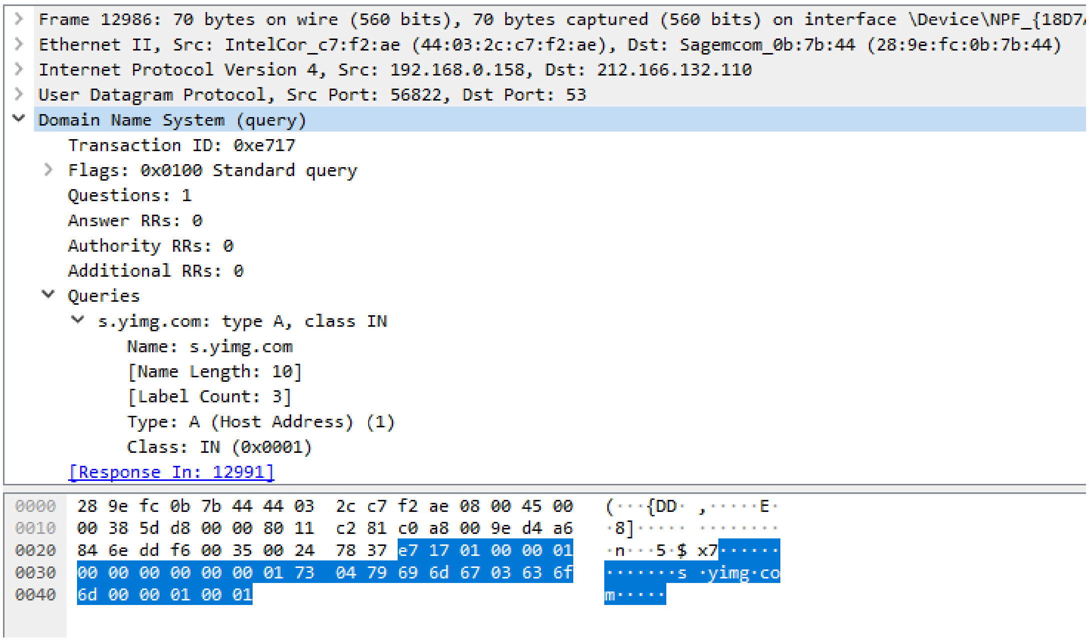


```
> Frame 30774: 86 bytes on wire (688 bits), 86 bytes captured (688 bits) on interface \Device\NPF_{18D7AB59-5FA1-4D06-8252-DA6BB45360CB}, id 0
v Ethernet II, Src: IntelCor_c7:f2:ae (44:03:2c:c7:f2:ae), Dst: Sagemcom_0b:7b:44 (28:9e:fc:0b:7b:44)
  v Destination: Sagemcom_0b:7b:44 (28:9e:fc:0b:7b:44)
      Address: Sagemcom_0b:7b:44 (28:9e:fc:0b:7b:44)
      .... ..0. .... .... .... .... = LG bit: Globally unique address (factory default)
      .... ...0 .... .... .... .... = IG bit: Individual address (unicast)
  v Source: IntelCor_c7:f2:ae (44:03:2c:c7:f2:ae)
      Address: IntelCor_c7:f2:ae (44:03:2c:c7:f2:ae)
      .... ..0. .... .... .... .... = LG bit: Globally unique address (factory default)
      .... ...0 .... .... .... .... = IG bit: Individual address (unicast)
    Type: IPv4 (0x0800)
v Internet Protocol Version 4, Src: 192.168.0.158, Dst: 150.253.224.159
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 72
    Identification: 0x6556 (25942)
  > Flags: 0x00
    Fragment Offset: 0

0000  28 9e fc 0b 7b 44 44 03  2c c7 f2 ae 08 00 45 00   (···{DD· ,·····E·
0010  00 48 65 56 00 00 80 11  9c 6b c0 a8 00 9e 96 fd   ·HeV····  ·k······
0020  e0 9f e7 e4 23 28 00 34  56 39 8f ce 00 06 64 01   ····#(·4 V9····d·
0030  b1 b3 97 75 4f 37 4d 41  52 49 01 00 02 9e d7 9d   ···uO7MA RI······
0040  d6 4c 00 00 8d a8 28 80  8d 39 59 c5 e9 fb 92 71   ·L····(·  ·9Y····q
0050  c8 1f d0 d2 d5 8a                                  ······
```



```
> Frame 30774: 86 bytes on wire (688 bits), 86 bytes captured (688 bits) on interface \Device\NPF_{18D7AB59-5FA1-4D06-8252-DA6BB45360CB}, id 0
> Ethernet II, Src: IntelCor_c7:f2:ae (44:03:2c:c7:f2:ae), Dst: Sagemcom_0b:7b:44 (28:9e:fc:0b:7b:44)
v Internet Protocol Version 4, Src: 192.168.0.158, Dst: 150.253.224.159
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 72
    Identification: 0x6556 (25942)
  > Flags: 0x00
    Fragment Offset: 0
    Time to Live: 128
    Protocol: UDP (17)
    Header Checksum: 0x9c6b [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 192.168.0.158
    Destination Address: 150.253.224.159
> User Datagram Protocol, Src Port: 59364, Dst Port: 9000
> Real-time Transport Control Protocol (Payload-specific Feedback)

0000  28 9e fc 0b 7b 44 44 03  2c c7 f2 ae 08 00 45 00   (···{DD· ,·····E·
0010  00 48 65 56 00 00 80 11  9c 6b c0 a8 00 9e 96 fd   ·HeV···· ·k······
0020  e0 9f e7 e4 23 28 00 34  56 39 8f ce 00 06 64 01   ····#(·4 V9····d·
0030  b1 b3 97 75 4f 37 4d 41  52 49 01 00 02 9e d7 9d   ···uO7MA RI······
0040  d6 4c 00 00 8d a8 28 80  8d 39 59 c5 e9 fb 92 71   ·L····(· ·9Y····q
0050  c8 1f d0 d2 d5 8a                                  ······
```

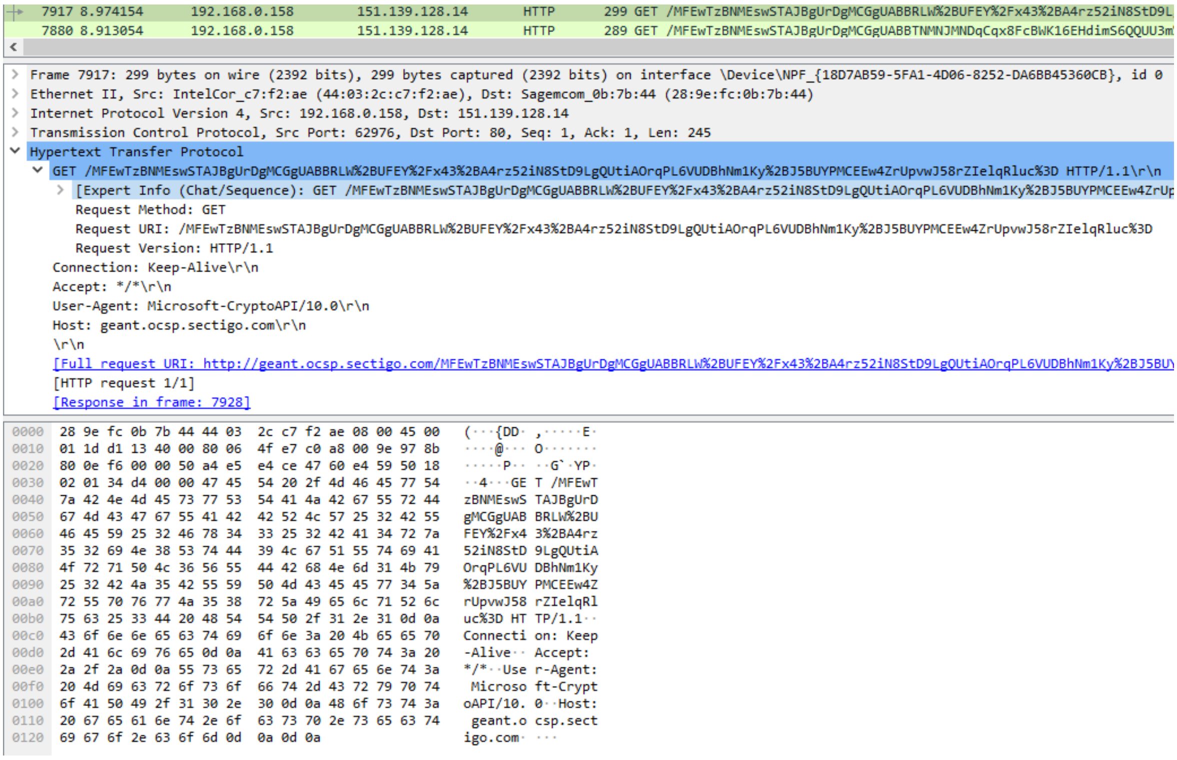
**<u>Question 5:</u> What is the DNS name being requested? What is DNS record type is being requested? What is the length of the request (bytes on wire)? Provide a screenshot of the DNS query analysed.**



```
> Frame 12986: 70 bytes on wire (560 bits), 70 bytes captured (560 bits) on interface \Device\NPF_{18D7
> Ethernet II, Src: IntelCor_c7:f2:ae (44:03:2c:c7:f2:ae), Dst: Sagemcom_0b:7b:44 (28:9e:fc:0b:7b:44)
> Internet Protocol Version 4, Src: 192.168.0.158, Dst: 212.166.132.110
> User Datagram Protocol, Src Port: 56822, Dst Port: 53
v Domain Name System (query)
    Transaction ID: 0xe717
  > Flags: 0x0100 Standard query
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 0
  v Queries
    v s.yimg.com: type A, class IN
        Name: s.yimg.com
        [Name Length: 10]
        [Label Count: 3]
        Type: A (Host Address) (1)
        Class: IN (0x0001)
    [Response In: 12991]

0000  28 9e fc 0b 7b 44 44 03  2c c7 f2 ae 08 00 45 00   (···{DD· ,·····E·
0010  00 38 5d d8 00 00 80 11  c2 81 c0 a8 00 9e d4 a6   ·8]····· ········
0020  84 6e dd f6 00 35 00 24  78 37 e7 17 01 00 00 01   ·n···5·$ x7······
0030  00 00 00 00 00 00 01 73  04 79 69 6d 67 03 63 6f   ·······s ·yimg·co
0040  6d 00 00 01 00 01                                  m·····
```

The name of the DNS (this protocol has been seen in class before) that has been requested is *s.xymg.com,* we can find this name in the properties of the Queries. The type that is requested is A, this means is for IPv4 address. The last thing that we have been asked is the length of the request, the value of the length is 70 bytes.

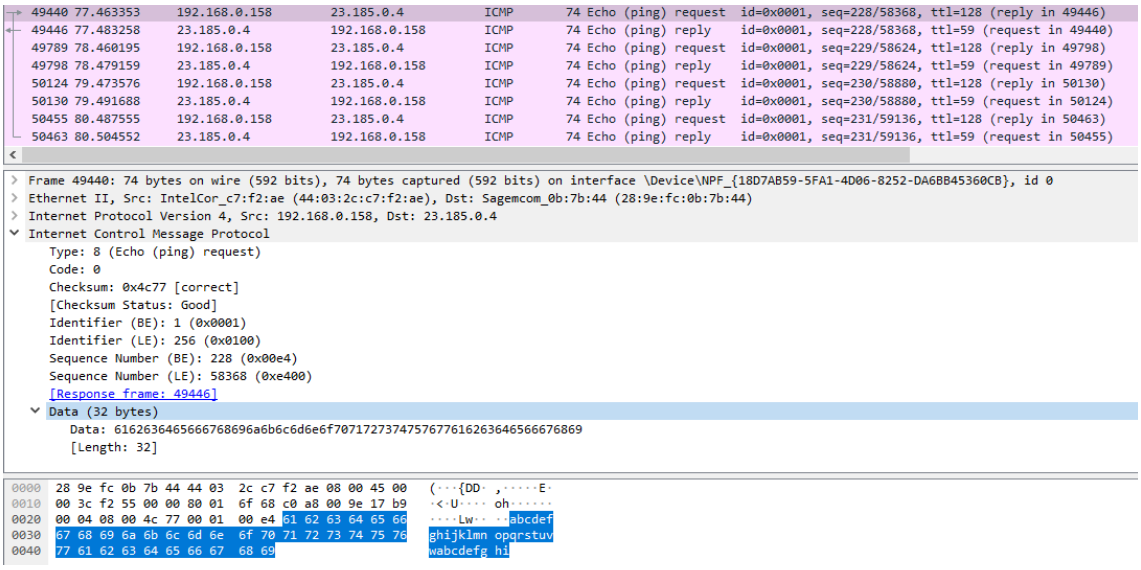
**<u>Question 6:</u> What is the Host: you are requesting data from? What is the Request URI? What is the Request Version? Provide a screenshot.**



```
  7917 8.974154     192.168.0.158       151.139.128.14      HTTP      299 GET /MFEwTzBNMEswSTAJBgUrDgMCGgUABBRLW%2BUFEY%2Fx43%2BA4rz52iN8StD9L
  7880 8.913054     192.168.0.158       151.139.128.14      HTTP      289 GET /MFEwTzBNMEswSTAJBgUrDgMCGgUABBTNMNJMNDqCqx8FcBWK16EHdimS6QQUU3m
<

> Frame 7917: 299 bytes on wire (2392 bits), 299 bytes captured (2392 bits) on interface \Device\NPF_{18D7AB59-5FA1-4D06-8252-DA6BB45360CB}, id 0
> Ethernet II, Src: IntelCor_c7:f2:ae (44:03:2c:c7:f2:ae), Dst: Sagemcom_0b:7b:44 (28:9e:fc:0b:7b:44)
> Internet Protocol Version 4, Src: 192.168.0.158, Dst: 151.139.128.14
> Transmission Control Protocol, Src Port: 62976, Dst Port: 80, Seq: 1, Ack: 1, Len: 245
v Hypertext Transfer Protocol
  v GET /MFEwTzBNMEswSTAJBgUrDgMCGgUABBRLW%2BUFEY%2Fx43%2BA4rz52iN8StD9LgQUtiAOrqPL6VUDBhNm1Ky%2BJ5BUYPMCEEw4ZrUpvwJ58rZIelqRluc%3D HTTP/1.1\r\n
    > [Expert Info (Chat/Sequence): GET /MFEwTzBNMEswSTAJBgUrDgMCGgUABBRLW%2BUFEY%2Fx43%2BA4rz52iN8StD9LgQUtiAOrqPL6VUDBhNm1Ky%2BJ5BUYPMCEEw4ZrUp
      Request Method: GET
      Request URI: /MFEwTzBNMEswSTAJBgUrDgMCGgUABBRLW%2BUFEY%2Fx43%2BA4rz52iN8StD9LgQUtiAOrqPL6VUDBhNm1Ky%2BJ5BUYPMCEEw4ZrUpvwJ58rZIelqRluc%3D
      Request Version: HTTP/1.1
    Connection: Keep-Alive\r\n
    Accept: */*\r\n
    User-Agent: Microsoft-CryptoAPI/10.0\r\n
    Host: geant.ocsp.sectigo.com\r\n
    \r\n
    [Full request URI: http://geant.ocsp.sectigo.com/MFEwTzBNMEswSTAJBgUrDgMCGgUABBRLW%2BUFEY%2Fx43%2BA4rz52iN8StD9LgQUtiAOrqPL6VUDBhNm1Ky%2BJ5BUY
    [HTTP request 1/1]
    [Response in frame: 7928]

0000  28 9e fc 0b 7b 44 44 03  2c c7 f2 ae 08 00 45 00   (···{DD· ,·····E·
0010  01 1d d1 13 40 00 80 06  4f e7 c0 a8 00 9e 97 8b   ····@··· O·······
0020  80 0e f6 00 00 50 a4 e5  e4 ce 47 60 e4 59 50 18   ·····P·· ··G`·YP·
0030  02 01 34 d4 00 00 47 45  54 20 2f 4d 46 45 77 54   ··4···GE T /MFEwT
0040  7a 42 4e 4d 45 73 77 53  54 41 4a 42 67 55 72 44   zBNMEswS TAJBgUrD
0050  67 4d 43 47 67 55 41 42  42 52 4c 57 25 32 42 55   gMCGgUAB BRLW%2BU
0060  46 45 59 25 32 46 78 34  33 25 32 42 41 34 72 7a   FEY%2Fx4 3%2BA4rz
0070  35 32 69 4e 38 53 74 44  39 4c 67 51 55 74 69 41   52iN8StD 9LgQUtiA
0080  4f 72 71 50 4c 36 56 55  44 42 68 4e 6d 31 4b 79   OrqPL6VU DBhNm1Ky
0090  25 32 42 4a 35 42 55 59  50 4d 43 45 45 77 34 5a   %2BJ5BUY PMCEEw4Z
00a0  72 55 70 76 77 4a 35 38  72 5a 49 65 6c 71 52 6c   rUpvwJ58 rZIelqRl
00b0  75 63 25 33 44 20 48 54  54 50 2f 31 2e 31 0d 0a   uc%3D HT TP/1.1··
00c0  43 6f 6e 6e 65 63 74 69  6f 6e 3a 20 4b 65 65 70   Connecti on: Keep
00d0  2d 41 6c 69 76 65 0d 0a  41 63 63 65 70 74 3a 20   -Alive·· Accept: 
00e0  2a 2f 2a 0d 0a 55 73 65  72 2d 41 67 65 6e 74 3a   */*··Use r-Agent:
00f0  20 4d 69 63 72 6f 73 6f  66 74 2d 43 72 79 70 74    Microso ft-Crypt
0100  6f 41 50 49 2f 31 30 2e  30 0d 0a 48 6f 73 74 3a   oAPI/10. 0··Host:
0110  20 67 65 61 6e 74 2e 6f  63 73 70 2e 73 65 63 74    geant.o csp.sect
0120  69 67 6f 2e 63 6f 6d 0d  0a 0d 0a                  igo.com· ···
```

The name of the Host that are requesting data from is *geant.ocsp.sectigo.com\r\n*. The Request URI (*Uniform Resource Identifier*), identifies a resource by name, location or both and indicated if an identified resource is available and where is it, is */MFEwTzBNMEswSTAJ…*, the Request Version *HTTP/1.1*, persistent connections are enabled by default and work well with proxies. It also allows the client to send multiple requests at the same time through the same connection (pipelining, this can send 2 requests before the first request has arrived) which makes it possible to eliminate the Round-Trip delay time for each request.

**<u>Question 7:</u> Now find the reply that matches your request. What are the differences between the ICMP echo request and the ICMP echo reply packets?**

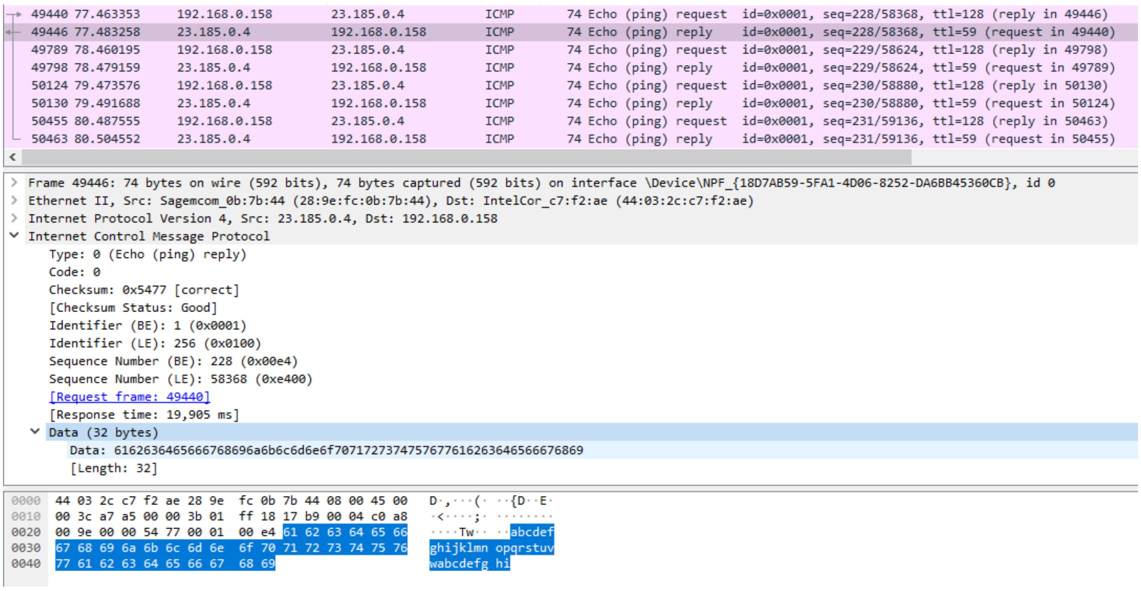


```
49440 77.463353   192.168.0.158   23.185.0.4      ICMP   74 Echo (ping) request  id=0x0001, seq=228/58368, ttl=128 (reply in 49446)
49446 77.483258   23.185.0.4      192.168.0.158   ICMP   74 Echo (ping) reply    id=0x0001, seq=228/58368, ttl=59 (request in 49440)
49789 78.460195   192.168.0.158   23.185.0.4      ICMP   74 Echo (ping) request  id=0x0001, seq=229/58624, ttl=128 (reply in 49798)
49798 78.479159   23.185.0.4      192.168.0.158   ICMP   74 Echo (ping) reply    id=0x0001, seq=229/58624, ttl=59 (request in 49789)
50124 79.473576   192.168.0.158   23.185.0.4      ICMP   74 Echo (ping) request  id=0x0001, seq=230/58880, ttl=128 (reply in 50130)
50130 79.491688   23.185.0.4      192.168.0.158   ICMP   74 Echo (ping) reply    id=0x0001, seq=230/58880, ttl=59 (request in 50124)
50455 80.487555   192.168.0.158   23.185.0.4      ICMP   74 Echo (ping) request  id=0x0001, seq=231/59136, ttl=128 (reply in 50463)
50463 80.504552   23.185.0.4      192.168.0.158   ICMP   74 Echo (ping) reply    id=0x0001, seq=231/59136, ttl=59 (request in 50455)
```

```
> Frame 49440: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface \Device\NPF_{18D7AB59-5FA1-4D06-8252-DA6BB45360CB}, id 0
> Ethernet II, Src: IntelCor_c7:f2:ae (44:03:2c:c7:f2:ae), Dst: Sagemcom_0b:7b:44 (28:9e:fc:0b:7b:44)
> Internet Protocol Version 4, Src: 192.168.0.158, Dst: 23.185.0.4
v Internet Control Message Protocol
    Type: 8 (Echo (ping) request)
    Code: 0
    Checksum: 0x4c77 [correct]
    [Checksum Status: Good]
    Identifier (BE): 1 (0x0001)
    Identifier (LE): 256 (0x0100)
    Sequence Number (BE): 228 (0x00e4)
    Sequence Number (LE): 58368 (0xe400)
    [Response frame: 49446]
  v Data (32 bytes)
      Data: 6162636465666768696a6b6c6d6e6f7071727374757677616263646566676869
      [Length: 32]
```

```
0000  28 9e fc 0b 7b 44 44 03  2c c7 f2 ae 08 00 45 00   (···{DD· ,·····E·
0010  00 3c f2 55 00 00 80 01  6f 68 c0 a8 00 9e 17 b9   ·<·U···· oh······
0020  00 04 08 00 4c 77 00 01  00 e4 61 62 63 64 65 66   ····Lw·· ··abcdef
0030  67 68 69 6a 6b 6c 6d 6e  6f 70 71 72 73 74 75 76   ghijklmn opqrstuv
0040  77 61 62 63 64 65 66 67  68 69                     wabcdefg hi
```

Request.



```
49440 77.463353   192.168.0.158   23.185.0.4      ICMP   74 Echo (ping) request  id=0x0001, seq=228/58368, ttl=128 (reply in 49446)
49446 77.483258   23.185.0.4      192.168.0.158   ICMP   74 Echo (ping) reply    id=0x0001, seq=228/58368, ttl=59 (request in 49440)
49789 78.460195   192.168.0.158   23.185.0.4      ICMP   74 Echo (ping) request  id=0x0001, seq=229/58624, ttl=128 (reply in 49798)
49798 78.479159   23.185.0.4      192.168.0.158   ICMP   74 Echo (ping) reply    id=0x0001, seq=229/58624, ttl=59 (request in 49789)
50124 79.473576   192.168.0.158   23.185.0.4      ICMP   74 Echo (ping) request  id=0x0001, seq=230/58880, ttl=128 (reply in 50130)
50130 79.491688   23.185.0.4      192.168.0.158   ICMP   74 Echo (ping) reply    id=0x0001, seq=230/58880, ttl=59 (request in 50124)
50455 80.487555   192.168.0.158   23.185.0.4      ICMP   74 Echo (ping) request  id=0x0001, seq=231/59136, ttl=128 (reply in 50463)
50463 80.504552   23.185.0.4      192.168.0.158   ICMP   74 Echo (ping) reply    id=0x0001, seq=231/59136, ttl=59 (request in 50455)
```

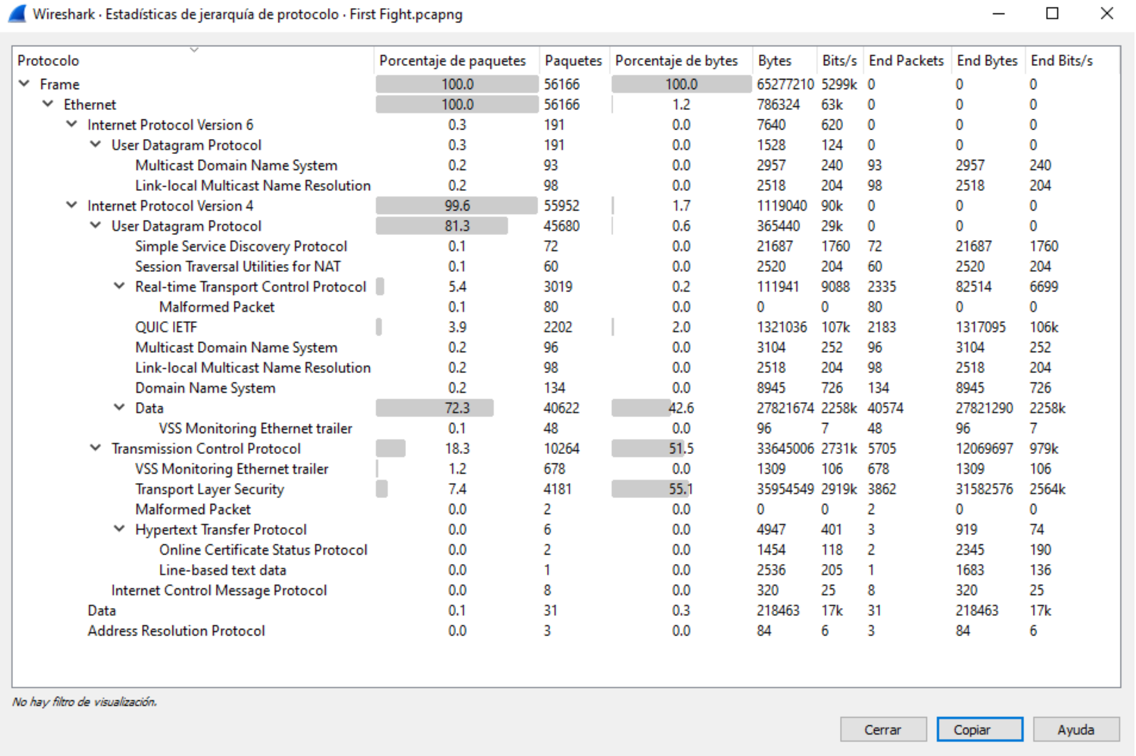
```
> Frame 49446: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface \Device\NPF_{18D7AB59-5FA1-4D06-8252-DA6BB45360CB}, id 0
> Ethernet II, Src: Sagemcom_0b:7b:44 (28:9e:fc:0b:7b:44), Dst: IntelCor_c7:f2:ae (44:03:2c:c7:f2:ae)
> Internet Protocol Version 4, Src: 23.185.0.4, Dst: 192.168.0.158
v Internet Control Message Protocol
    Type: 0 (Echo (ping) reply)
    Code: 0
    Checksum: 0x5477 [correct]
    [Checksum Status: Good]
    Identifier (BE): 1 (0x0001)
    Identifier (LE): 256 (0x0100)
    Sequence Number (BE): 228 (0x00e4)
    Sequence Number (LE): 58368 (0xe400)
    [Request frame: 49440]
    [Response time: 19,905 ms]
  v Data (32 bytes)
      Data: 6162636465666768696a6b6c6d6e6f7071727374757677616263646566676869
      [Length: 32]
```

```
0000  44 03 2c c7 f2 ae 28 9e  fc 0b 7b 44 08 00 45 00   D·,···(· ··{D··E·
0010  00 3c a7 a5 00 00 3b 01  ff 18 17 b9 00 04 c0 a8   ·<····;· ········
0020  00 9e 00 00 54 77 00 01  00 e4 61 62 63 64 65 66   ····Tw·· ··abcdef
0030  67 68 69 6a 6b 6c 6d 6e  6f 70 71 72 73 74 75 76   ghijklmn opqrstuv
0040  77 61 62 63 64 65 66 67  68 69                     wabcdefg hi
```

Reply.

The difference between ICMP echo request and the ICMP echo reply packets are the type, in the ICMP request the type is 8, this mean that is a request. In the ICMP reply the type is 0, this mean that is a reply. Another difference is the checksum. In the ICMP request is 0x4c77 and in the ICMP reply is 0x5477.

## Question 8: What is it used for?



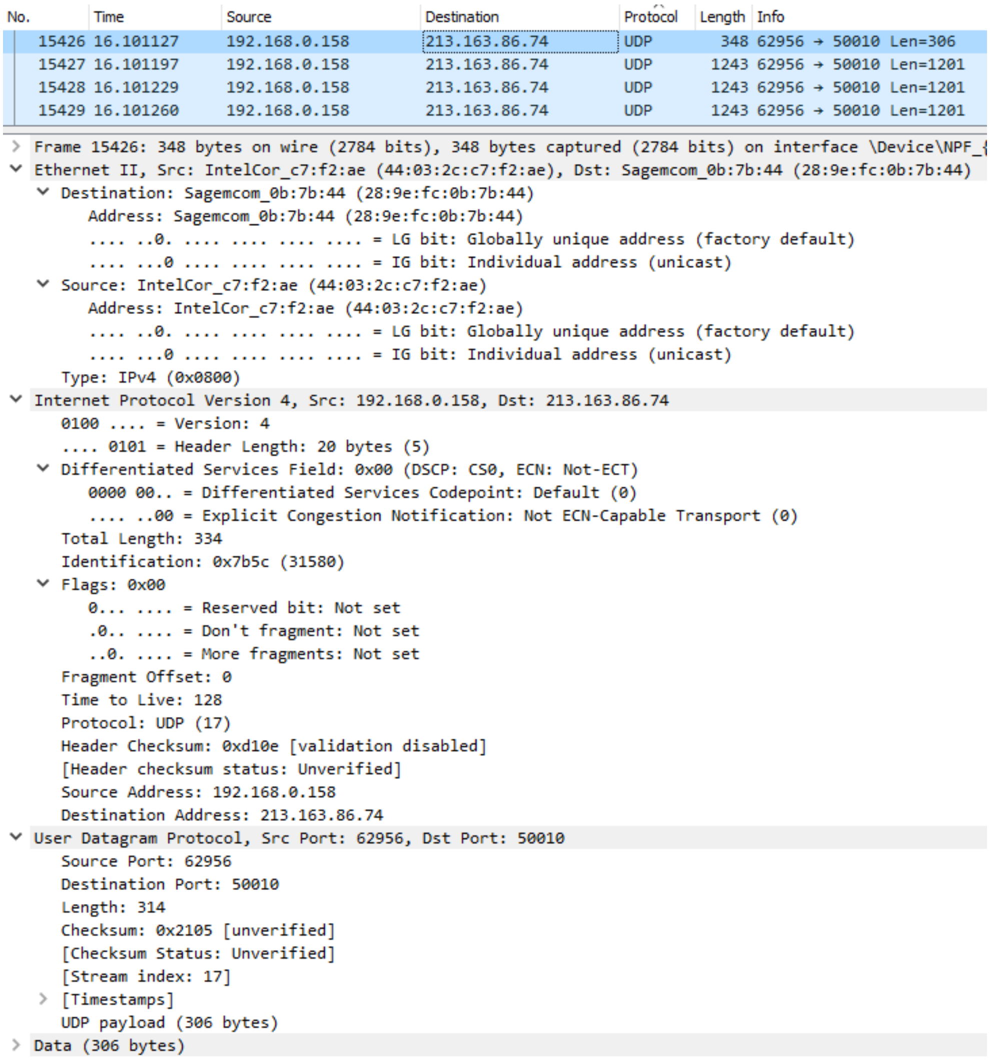
Wireshark · Estadísticas de jerarquía de protocolo · First Fight.pcapng

| Protocolo | Porcentaje de paquetes | Paquetes | Porcentaje de bytes | Bytes | Bits/s | End Packets | End Bytes | End Bits/s |
|---|---|---|---|---|---|---|---|---|
| Frame | 100.0 | 56166 | 100.0 | 65277210 | 5299k | 0 | 0 | 0 |
| Ethernet | 100.0 | 56166 | 1.2 | 786324 | 63k | 0 | 0 | 0 |
| Internet Protocol Version 6 | 0.3 | 191 | 0.0 | 7640 | 620 | 0 | 0 | 0 |
| User Datagram Protocol | 0.3 | 191 | 0.0 | 1528 | 124 | 0 | 0 | 0 |
| Multicast Domain Name System | 0.2 | 93 | 0.0 | 2957 | 240 | 93 | 2957 | 240 |
| Link-local Multicast Name Resolution | 0.2 | 98 | 0.0 | 2518 | 204 | 98 | 2518 | 204 |
| Internet Protocol Version 4 | 99.6 | 55952 | 1.7 | 1119040 | 90k | 0 | 0 | 0 |
| User Datagram Protocol | 81.3 | 45680 | 0.6 | 365440 | 29k | 0 | 0 | 0 |
| Simple Service Discovery Protocol | 0.1 | 72 | 0.0 | 21687 | 1760 | 72 | 21687 | 1760 |
| Session Traversal Utilities for NAT | 0.1 | 60 | 0.0 | 2520 | 204 | 60 | 2520 | 204 |
| Real-time Transport Control Protocol | 5.4 | 3019 | 0.2 | 111941 | 9088 | 2335 | 82514 | 6699 |
| Malformed Packet | 0.1 | 80 | 0.0 | 0 | 0 | 80 | 0 | 0 |
| QUIC IETF | 3.9 | 2202 | 2.0 | 1321036 | 107k | 2183 | 1317095 | 106k |
| Multicast Domain Name System | 0.2 | 96 | 0.0 | 3104 | 252 | 96 | 3104 | 252 |
| Link-local Multicast Name Resolution | 0.2 | 98 | 0.0 | 2518 | 204 | 98 | 2518 | 204 |
| Domain Name System | 0.2 | 134 | 0.0 | 8945 | 726 | 134 | 8945 | 726 |
| Data | 72.3 | 40622 | 42.6 | 27821674 | 2258k | 40574 | 27821290 | 2258k |
| VSS Monitoring Ethernet trailer | 0.1 | 48 | 0.0 | 96 | 7 | 48 | 96 | 7 |
| Transmission Control Protocol | 18.3 | 10264 | 51.5 | 33645006 | 2731k | 5705 | 12069697 | 979k |
| VSS Monitoring Ethernet trailer | 1.2 | 678 | 0.0 | 1309 | 106 | 678 | 1309 | 106 |
| Transport Layer Security | 7.4 | 4181 | 55.1 | 35954549 | 2919k | 3862 | 31582576 | 2564k |
| Malformed Packet | 0.0 | 2 | 0.0 | 0 | 0 | 2 | 0 | 0 |
| Hypertext Transfer Protocol | 0.0 | 6 | 0.0 | 4947 | 401 | 3 | 919 | 74 |
| Online Certificate Status Protocol | 0.0 | 2 | 0.0 | 1454 | 118 | 2 | 2345 | 190 |
| Line-based text data | 0.0 | 1 | 0.0 | 2536 | 205 | 1 | 1683 | 136 |
| Internet Control Message Protocol | 0.0 | 8 | 0.0 | 320 | 25 | 8 | 320 | 25 |
| Data | 0.1 | 31 | 0.3 | 218463 | 17k | 31 | 218463 | 17k |
| Address Resolution Protocol | 0.0 | 3 | 0.0 | 84 | 6 | 3 | 84 | 6 |

*No hay filtro de visualización.*

The protocol hierarchy is a nested list of all protocols used in any of the captured packets. Each row contains the statistical values of one protocol. It used for detecting anomalies such as a UDP flooding attack. In the first column appears the protocol's name, the next column is the percentage of protocol packets. The packets column is the absolute number of packets. The Bytes column is the absolute number of bytes. The MBit/s column is the bandwidth of the protocol, relative to the capture time. The End Packets is the absolute number of packets of his protocol with the highest protocol to decode. The End Bytes is the absolute number of bytes of this protocol with the highest protocol to decode. End MBit/s is the bandwidth of his protocol, relative to the capture time with the highest protocol to decode.

**Question 9:** **What data is stored in the Ethernet header? What data is in the network layer (IP) header? What data is in the transport layer header (either TCP or UDP)?**



```
No.     Time            Source            Destination       Protocol  Length Info
  15426 16.101127       192.168.0.158     213.163.86.74     UDP        348 62956 → 50010 Len=306
  15427 16.101197       192.168.0.158     213.163.86.74     UDP       1243 62956 → 50010 Len=1201
  15428 16.101229       192.168.0.158     213.163.86.74     UDP       1243 62956 → 50010 Len=1201
  15429 16.101260       192.168.0.158     213.163.86.74     UDP       1243 62956 → 50010 Len=1201

› Frame 15426: 348 bytes on wire (2784 bits), 348 bytes captured (2784 bits) on interface \Device\NPF_{
⌄ Ethernet II, Src: IntelCor_c7:f2:ae (44:03:2c:c7:f2:ae), Dst: Sagemcom_0b:7b:44 (28:9e:fc:0b:7b:44)
  ⌄ Destination: Sagemcom_0b:7b:44 (28:9e:fc:0b:7b:44)
      Address: Sagemcom_0b:7b:44 (28:9e:fc:0b:7b:44)
      .... ..0. .... .... .... .... = LG bit: Globally unique address (factory default)
      .... ...0 .... .... .... .... = IG bit: Individual address (unicast)
  ⌄ Source: IntelCor_c7:f2:ae (44:03:2c:c7:f2:ae)
      Address: IntelCor_c7:f2:ae (44:03:2c:c7:f2:ae)
      .... ..0. .... .... .... .... = LG bit: Globally unique address (factory default)
      .... ...0 .... .... .... .... = IG bit: Individual address (unicast)
    Type: IPv4 (0x0800)
⌄ Internet Protocol Version 4, Src: 192.168.0.158, Dst: 213.163.86.74
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  ⌄ Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
      0000 00.. = Differentiated Services Codepoint: Default (0)
      .... ..00 = Explicit Congestion Notification: Not ECN-Capable Transport (0)
    Total Length: 334
    Identification: 0x7b5c (31580)
  ⌄ Flags: 0x00
      0... .... = Reserved bit: Not set
      .0.. .... = Don't fragment: Not set
      ..0. .... = More fragments: Not set
    Fragment Offset: 0
    Time to Live: 128
    Protocol: UDP (17)
    Header Checksum: 0xd10e [validation disabled]
    [Header checksum status: Unverified]
    Source Address: 192.168.0.158
    Destination Address: 213.163.86.74
⌄ User Datagram Protocol, Src Port: 62956, Dst Port: 50010
    Source Port: 62956
    Destination Port: 50010
    Length: 314
    Checksum: 0x2105 [unverified]
    [Checksum Status: Unverified]
    [Stream index: 17]
  › [Timestamps]
    UDP payload (306 bytes)
› Data (306 bytes)
```

The data that is store in the Ethernet is the destination, in which we can find the address. The source, in which we can find the address. And the type.

The data that is store in the IPv4 is the differentiated Services Field, in which we can find the total length and the identification. The flags, in which we can find the fragment Offset, Time to Live, Protocol, Header Checksum, Source Address and Destination Address.

The data that is stored in the UDP is the Source Port, Length, Checksum, Stream index, Timestamps and UDP payload.

**Question 10: What percentage of your network traffic was IPv4? What about IPv6? and TCP vs UDP?**

There are two types of percentage. In IPv4, the packet's percentage is 99,6% and the byte's percentage is 1,7%. In IPv6, the packet's percentage is 0,3%, and the byte's percentage is 0%. In TCP, the packet's percentage is 18,3% and the byte's percentage is 51,5%. And in UDP, the packet's percentage is 81,3% and the byte's percentage is 0,6%.