Prueba - Análisis de Seguridad en Redes de Datos

Autor: Carlos Aliendres

Descripción
La empresa Desafío Latam ha implementado una red corporativa en donde deberás realizar
una serie de análisis de tráfico y pruebas de conectividad que permita evaluar el comportamiento de la red desde el punto de vista de seguridad, así como la identificación
de patrones de tráfico y posibles anomalías.
Requerimientos de la Prueba:

Análisis de Tráfico con hping3 en Kali Linux
Utilizando Kali Linux, deberás realizar las siguientes actividades con la herramienta hping3:

- a) Crear un archivo de texto plano con información personal (nombre, curso, fecha)

  Prueba Modulo 7.txt

- b) Utilizar hping3 para enviar diferentes tipos de paquetes a un host objetivo (puede
  ser google.com o la puerta de enlace local):
  - ICMP ping normal
    - hping3 -1 -c 5 google.com
  - TCP SYN a puerto 80
    - hping3 -S -p 80 -c 5 google.com
  - UDP a puerto 53
    - hping3 -2 -p 53 -c 1 google.com
  - TCP con datos personalizados
    - hping3 -c 5 -S -p 443 google.com --data "Hola, Carlos Aliendres"
- c) Documentar los comandos utilizados y explicar las diferencias en las respuestas

```
┌──(root㉿kali)-[/home/kali]
└─# hping3 -1 -c 5 google.com
HPING google.com (eth0 142.251.0.113): icmp mode set, 28 headers + 0 data byt
es
len=46 ip=142.251.0.113 ttl=107 id=0 icmp_seq=0 rtt=7.8 ms
len=46 ip=142.251.0.113 ttl=107 id=0 icmp_seq=1 rtt=13.0 ms
len=46 ip=142.251.0.113 ttl=107 id=0 icmp_seq=2 rtt=14.1 ms
len=46 ip=142.251.0.113 ttl=107 id=0 icmp_seq=3 rtt=10.9 ms
len=46 ip=142.251.0.113 ttl=107 id=0 icmp_seq=4 rtt=6.7 ms

── google.com hping statistic ──
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 6.7/10.5/14.1 ms
```

Este comando nos permite verificar la conectividad a un host, medir la latencia y determinar si el host se encuentra activo.

```
┌──(root㉿kali)-[/home/kali]
└─# hping3 -S -p 80 -c 1 google.com
HPING google.com (eth0 142.251.0.113): S set, 40 headers + 0 data bytes
len=46 ip=142.251.0.113 ttl=121 DF id=0 sport=80 flags=SA seq=0 win=65535 rtt
=12.8 ms

── google.com hping statistic ──
1 packets transmitted, 1 packets received, 0% packet loss
round-trip min/avg/max = 12.8/12.8/12.8 ms
```

```
┌──(root㉿kali)-[/home/kali]
└─# hping3 -S -p 80 -c 5 google.com
HPING google.com (eth0 142.251.0.139): S set, 40 headers + 0 data bytes
len=46 ip=142.251.0.139 ttl=122 DF id=0 sport=80 flags=SA seq=0 win=65535 rtt
=7.8 ms
len=46 ip=142.251.0.139 ttl=122 DF id=0 sport=80 flags=SA seq=1 win=65535 rtt
=7.7 ms
len=46 ip=142.251.0.139 ttl=121 DF id=0 sport=80 flags=SA seq=2 win=65535 rtt
=6.9 ms
len=46 ip=142.251.0.139 ttl=121 DF id=0 sport=80 flags=SA seq=3 win=65535 rtt
=11.9 ms
len=46 ip=142.251.0.139 ttl=122 DF id=0 sport=80 flags=SA seq=4 win=65535 rtt
=15.5 ms

── google.com hping statistic ──
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 6.9/10.0/15.5 ms
```

Este comando nos permite identificar que puertos TCP están abiertos en un host y mapear que servicios se ejecutan en el puerto 80

```
┌──(root㉿kali)-[/home/kali]
└─# hping3 -2 -p 53 -c 1 google.com
HPING google.com (eth0 142.251.0.139): udp mode set, 28 headers + 0 data byte
s

── google.com hping statistic ──
1 packets transmitted, 0 packets received, 100% packet loss
round-trip min/avg/max = 0.0/0.0/0.0 ms
```

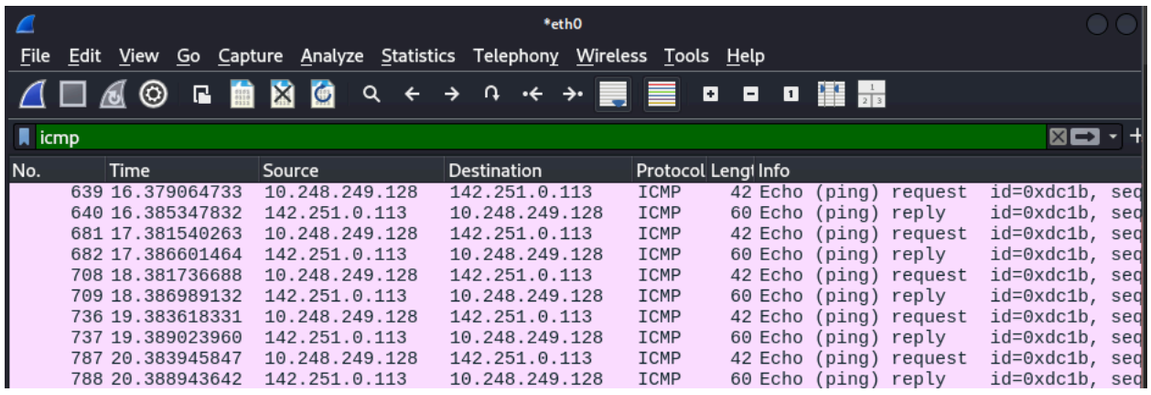
Este comando se usa para realizar escaneos de puertos UDP, este no recibe confirmación de recepción por eso el 100% de paquetes perdidos

```
┌──(root㉿kali)-[/home/kali]
└─# hping3 -c 5 -S -p 443 google.com --data "Hola, Carlos Aliendres"
HPING google.com (eth0 142.251.0.100): S set, 40 headers + 0 data bytes
len=46 ip=142.251.0.100 ttl=121 DF id=0 sport=443 flags=SA seq=0 win=65535 rt
t=12.2 ms
len=46 ip=142.251.0.100 ttl=122 DF id=0 sport=443 flags=SA seq=1 win=65535 rt
t=15.6 ms
len=46 ip=142.251.0.100 ttl=121 DF id=0 sport=443 flags=SA seq=2 win=65535 rt
t=13.9 ms
len=46 ip=142.251.0.100 ttl=122 DF id=0 sport=443 flags=SA seq=3 win=65535 rt
t=12.3 ms
len=46 ip=142.251.0.100 ttl=122 DF id=0 sport=443 flags=SA seq=4 win=65535 rt
t=17.5 ms

── google.com hping statistic ──
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max = 12.2/14.3/17.5 ms
```

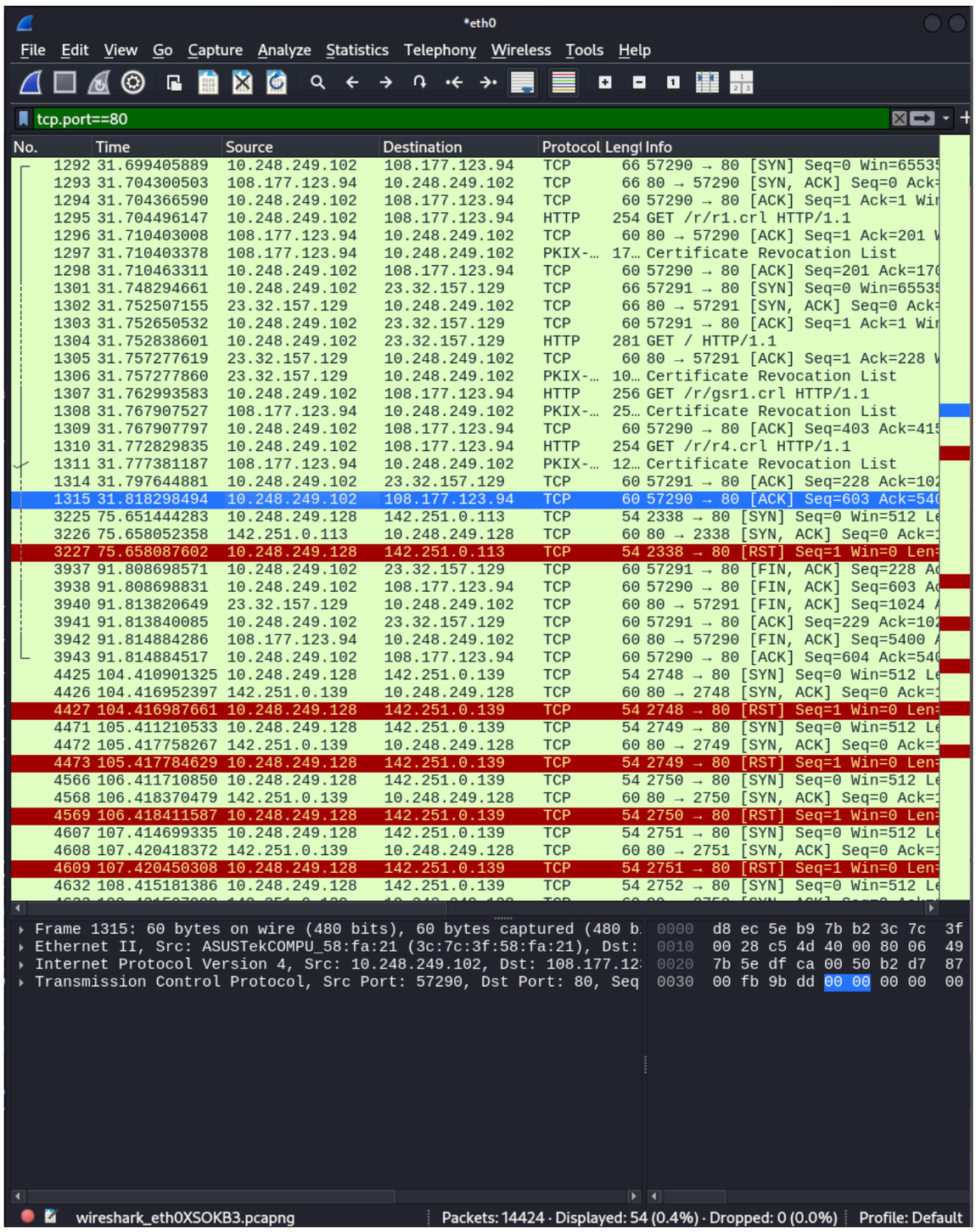
Este comando permite probar si el servicio responde a datos personalizados.

- d) Capturar el tráfico generado con Wireshark durante las pruebas

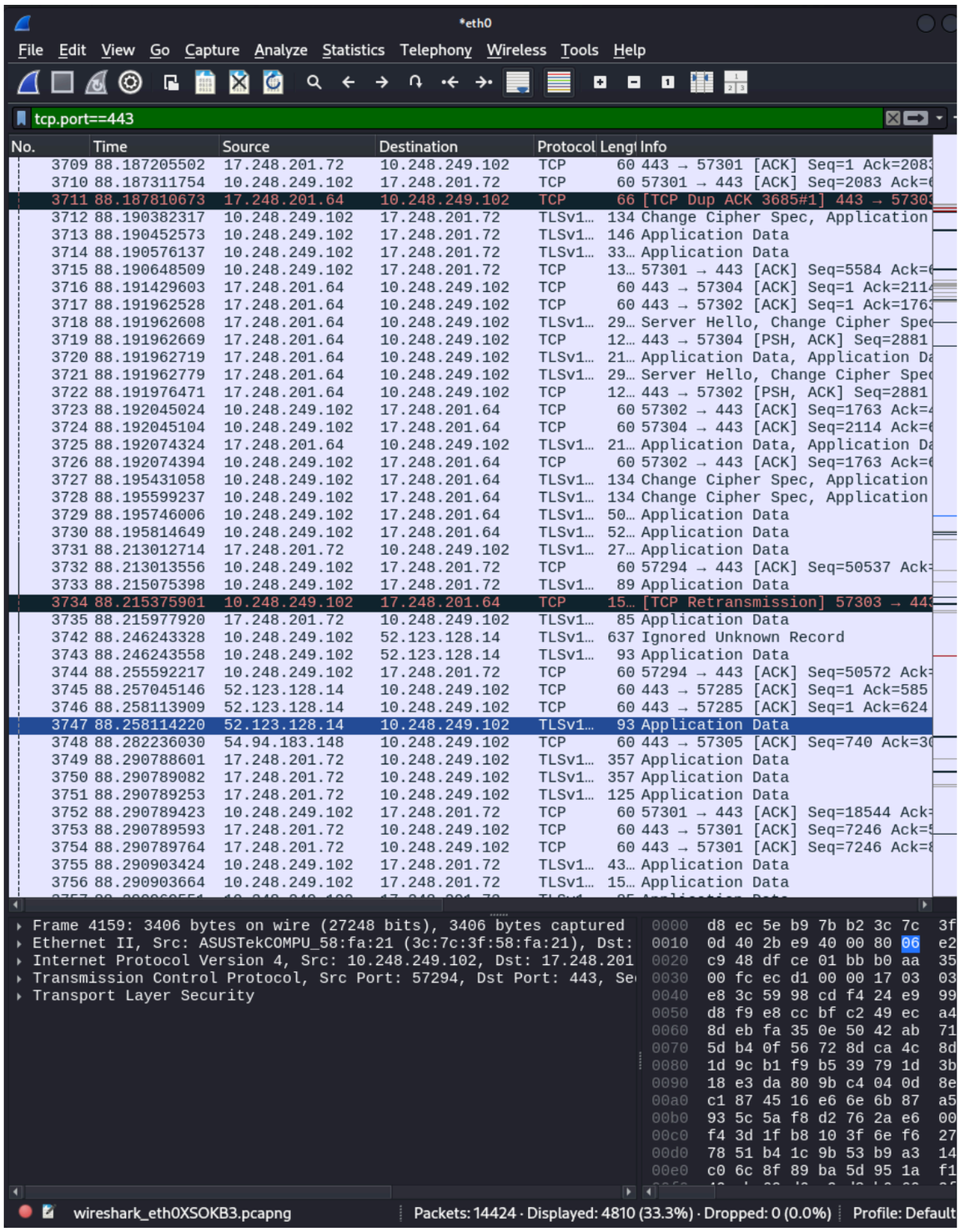
ICMP



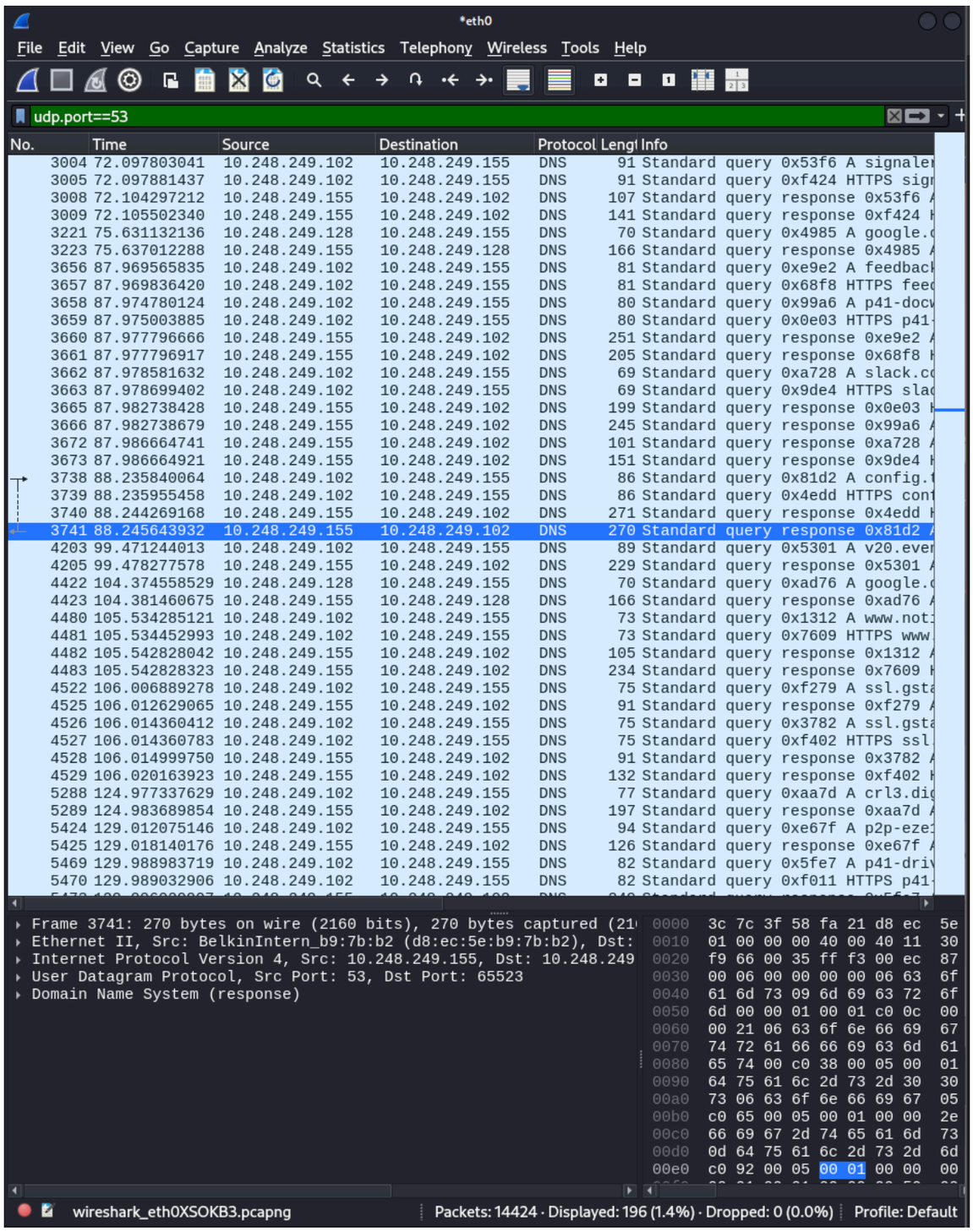
TCP.PORT==80


*eth0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
tcp.port==80
No. Time Source Destination Protocol Leng Info
1292 31.699405889 10.248.249.102 108.177.123.94 TCP 66 57290 → 80 [SYN] Seq=0 Win=6553
1293 31.704300503 108.177.123.94 10.248.249.102 TCP 66 80 → 57290 [SYN, ACK] Seq=0 Ack=
1294 31.704366590 10.248.249.102 108.177.123.94 TCP 60 57290 → 80 [ACK] Seq=1 Ack=1 Wi
1295 31.704496147 10.248.249.102 108.177.123.94 HTTP 254 GET /r/r1.crl HTTP/1.1
1296 31.710403008 108.177.123.94 10.248.249.102 TCP 60 80 → 57290 [ACK] Seq=1 Ack=201
1297 31.710403378 108.177.123.94 10.248.249.102 PKIX-… 17… Certificate Revocation List
1298 31.710463311 10.248.249.102 108.177.123.94 TCP 60 57290 → 80 [ACK] Seq=201 Ack=17
1301 31.748294661 10.248.249.102 23.32.157.129 TCP 66 57291 → 80 [SYN] Seq=0 Win=6553
1302 31.752507155 23.32.157.129 10.248.249.102 TCP 66 80 → 57291 [SYN, ACK] Seq=0 Ack=
1303 31.752650532 10.248.249.102 23.32.157.129 TCP 60 57291 → 80 [ACK] Seq=1 Ack=1 Wi
1304 31.752838601 10.248.249.102 23.32.157.129 HTTP 281 GET / HTTP/1.1
1305 31.757277619 23.32.157.129 10.248.249.102 TCP 60 80 → 57291 [ACK] Seq=1 Ack=228
1306 31.757277860 23.32.157.129 10.248.249.102 PKIX-… 10… Certificate Revocation List
1307 31.762993583 10.248.249.102 108.177.123.94 HTTP 256 GET /r/gsr1.crl HTTP/1.1
1308 31.767907527 108.177.123.94 10.248.249.102 PKIX-… 25… Certificate Revocation List
1309 31.767907797 10.248.249.102 108.177.123.94 TCP 60 57290 → 80 [ACK] Seq=403 Ack=41
1310 31.772829835 10.248.249.102 108.177.123.94 HTTP 254 GET /r/r4.crl HTTP/1.1
1311 31.777381187 108.177.123.94 10.248.249.102 PKIX-… 12… Certificate Revocation List
1314 31.797644881 10.248.249.102 23.32.157.129 TCP 60 57291 → 80 [ACK] Seq=228 Ack=10
1315 31.818298494 10.248.249.102 108.177.123.94 TCP 60 57290 → 80 [ACK] Seq=603 Ack=54
3225 75.651444283 10.248.249.128 142.251.0.113 TCP 54 2338 → 80 [SYN] Seq=0 Win=512 Le
3226 75.658052358 142.251.0.113 10.248.249.128 TCP 60 80 → 2338 [SYN, ACK] Seq=0 Ack=
3227 75.658087602 10.248.249.128 142.251.0.113 TCP 54 2338 → 80 [RST] Seq=1 Win=0 Len
3937 91.808698571 10.248.249.102 23.32.157.129 TCP 60 57291 → 80 [FIN, ACK] Seq=228 A
3938 91.808698831 10.248.249.102 108.177.123.94 TCP 60 57290 → 80 [FIN, ACK] Seq=603 A
3940 91.813820649 23.32.157.129 10.248.249.102 TCP 60 80 → 57291 [FIN, ACK] Seq=1024
3941 91.813840085 10.248.249.102 23.32.157.129 TCP 60 57291 → 80 [ACK] Seq=229 Ack=10
3942 91.814884286 108.177.123.94 10.248.249.102 TCP 60 80 → 57290 [FIN, ACK] Seq=5400
3943 91.814884517 10.248.249.102 108.177.123.94 TCP 60 57290 → 80 [ACK] Seq=604 Ack=54
4425 104.410901325 10.248.249.128 142.251.0.139 TCP 54 2748 → 80 [SYN] Seq=0 Win=512 Le
4426 104.416952397 142.251.0.139 10.248.249.128 TCP 60 80 → 2748 [SYN, ACK] Seq=0 Ack=
4427 104.416987661 10.248.249.128 142.251.0.139 TCP 54 2748 → 80 [RST] Seq=1 Win=0 Len
4471 105.411210533 10.248.249.128 142.251.0.139 TCP 54 2749 → 80 [SYN] Seq=0 Win=512 Le
4472 105.417758267 142.251.0.139 10.248.249.128 TCP 60 80 → 2749 [SYN, ACK] Seq=0 Ack=
4473 105.417784629 10.248.249.128 142.251.0.139 TCP 54 2749 → 80 [RST] Seq=1 Win=0 Len
4566 106.411710850 10.248.249.128 142.251.0.139 TCP 54 2750 → 80 [SYN] Seq=0 Win=512 Le
4568 106.418370479 142.251.0.139 10.248.249.128 TCP 60 80 → 2750 [SYN, ACK] Seq=0 Ack=
4569 106.418411587 10.248.249.128 142.251.0.139 TCP 54 2750 → 80 [RST] Seq=1 Win=0 Len
4607 107.414699335 10.248.249.128 142.251.0.139 TCP 54 2751 → 80 [SYN] Seq=0 Win=512 Le
4608 107.420418372 142.251.0.139 10.248.249.128 TCP 60 80 → 2751 [SYN, ACK] Seq=0 Ack=
4609 107.420450308 10.248.249.128 142.251.0.139 TCP 54 2751 → 80 [RST] Seq=1 Win=0 Len
4632 108.415181386 10.248.249.128 142.251.0.139 TCP 54 2752 → 80 [SYN] Seq=0 Win=512 Le
Frame 1315: 60 bytes on wire (480 bits), 60 bytes captured (480 b
Ethernet II, Src: ASUSTekCOMPU_58:fa:21 (3c:7c:3f:58:fa:21), Dst:
Internet Protocol Version 4, Src: 10.248.249.102, Dst: 108.177.12
Transmission Control Protocol, Src Port: 57290, Dst Port: 80, Seq
0000 d8 ec 5e b9 7b b2 3c 7c 3f
0010 00 28 c5 4d 40 00 80 06 49
0020 7b 5e df ca 00 50 b2 d7 87
0030 00 fb 9b dd 00 00 00 00 00
wireshark_eth0XSOKB3.pcapng
Packets: 14424 · Displayed: 54 (0.4%) · Dropped: 0 (0.0%)
Profile: Default

TCP.PORT== 443

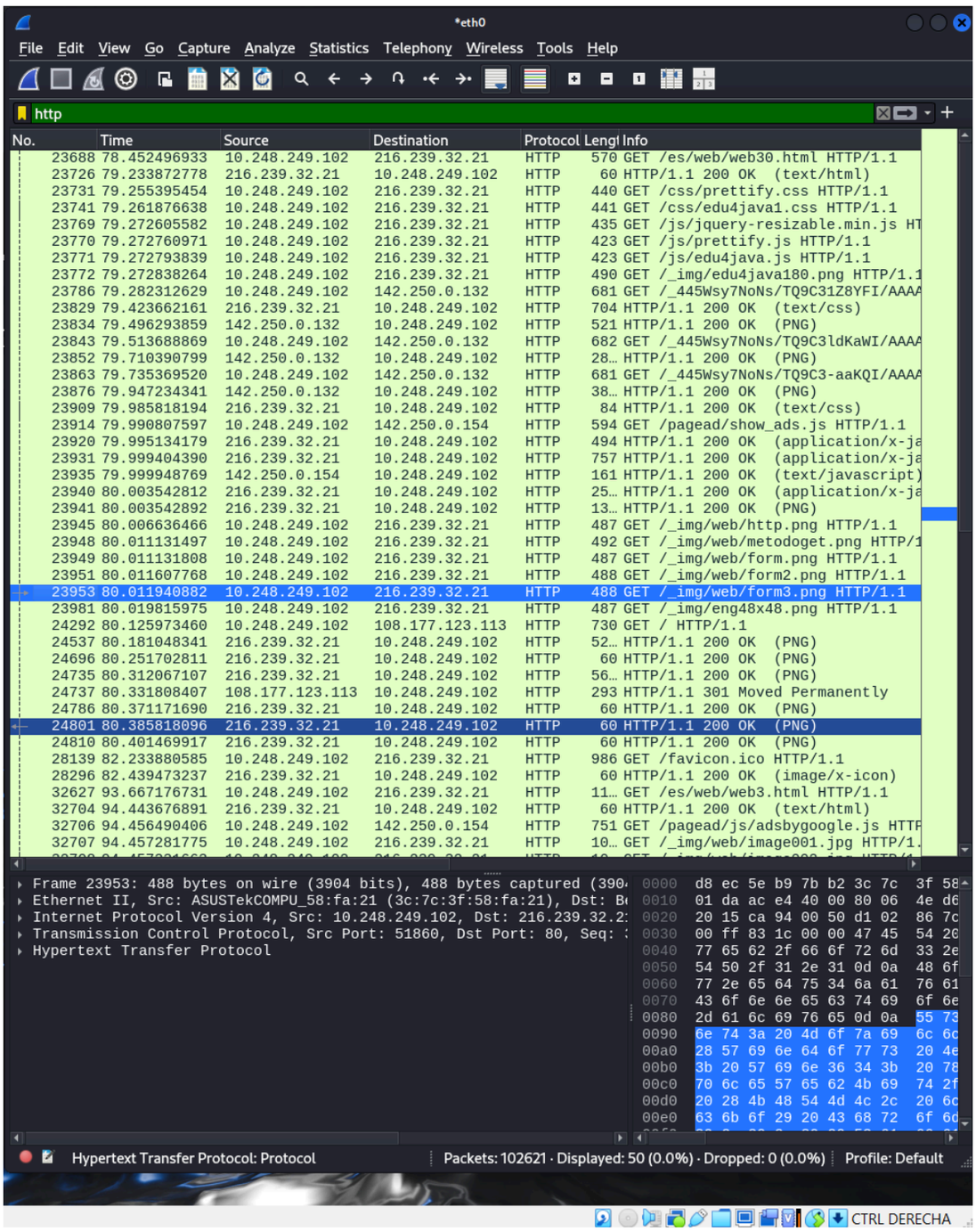

*eth0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
tcp.port==443
No. Time Source Destination Protocol Leng Info
3709 88.187205502 17.248.201.72 10.248.249.102 TCP 60 443 → 57301 [ACK] Seq=1 Ack=208
3710 88.187311754 10.248.249.102 17.248.201.72 TCP 60 57301 → 443 [ACK] Seq=2083 Ack=
3711 88.187810673 17.248.201.64 10.248.249.102 TCP 66 [TCP Dup ACK 3685#1] 443 → 5730
3712 88.190382317 10.248.249.102 17.248.201.72 TLSv1… 134 Change Cipher Spec, Application
3713 88.190452573 10.248.249.102 17.248.201.72 TLSv1… 146 Application Data
3714 88.190576137 10.248.249.102 17.248.201.72 TLSv1… 33… Application Data
3715 88.190648509 10.248.249.102 17.248.201.72 TCP 13… 57301 → 443 [ACK] Seq=5584 Ack=
3716 88.191429603 17.248.201.64 10.248.249.102 TCP 60 443 → 57304 [ACK] Seq=1 Ack=211
3717 88.191962528 17.248.201.64 10.248.249.102 TCP 60 443 → 57302 [ACK] Seq=1 Ack=176
3718 88.191962608 17.248.201.64 10.248.249.102 TLSv1… 29… Server Hello, Change Cipher Spec
3719 88.191962669 17.248.201.64 10.248.249.102 TCP 12… 443 → 57304 [PSH, ACK] Seq=2881
3720 88.191962719 17.248.201.64 10.248.249.102 TLSv1… 21… Application Data, Application Da
3721 88.191962779 17.248.201.64 10.248.249.102 TLSv1… 29… Server Hello, Change Cipher Spec
3722 88.191976471 17.248.201.64 10.248.249.102 TCP 12… 443 → 57302 [PSH, ACK] Seq=2881
3723 88.192045024 10.248.249.102 17.248.201.64 TCP 60 57302 → 443 [ACK] Seq=1763 Ack=
3724 88.192045104 10.248.249.102 17.248.201.64 TCP 60 57304 → 443 [ACK] Seq=2114 Ack=
3725 88.192074324 17.248.201.64 10.248.249.102 TLSv1… 21… Application Data, Application Da
3726 88.192074394 10.248.249.102 17.248.201.64 TCP 60 57302 → 443 [ACK] Seq=1763 Ack=
3727 88.195431058 10.248.249.102 17.248.201.64 TLSv1… 134 Change Cipher Spec, Application
3728 88.195599237 10.248.249.102 17.248.201.64 TLSv1… 134 Change Cipher Spec, Application
3729 88.195746006 10.248.249.102 17.248.201.64 TLSv1… 50… Application Data
3730 88.195814649 10.248.249.102 17.248.201.64 TLSv1… 52… Application Data
3731 88.213012714 17.248.201.72 10.248.249.102 TLSv1… 27… Application Data
3732 88.213013556 10.248.249.102 17.248.201.72 TCP 60 57294 → 443 [ACK] Seq=50537 Ack=
3733 88.215075398 10.248.249.102 17.248.201.72 TLSv1… 89 Application Data
3734 88.215375901 10.248.249.102 17.248.201.64 TCP 15… [TCP Retransmission] 57303 → 44
3735 88.215977920 17.248.201.72 10.248.249.102 TLSv1… 85 Application Data
3742 88.246243328 10.248.249.102 52.123.128.14 TLSv1… 637 Ignored Unknown Record
3743 88.246243558 10.248.249.102 52.123.128.14 TLSv1… 93 Application Data
3744 88.255592217 10.248.249.102 17.248.201.72 TCP 60 57294 → 443 [ACK] Seq=50572 Ack=
3745 88.257045146 52.123.128.14 10.248.249.102 TCP 60 443 → 57285 [ACK] Seq=1 Ack=585
3746 88.258113909 52.123.128.14 10.248.249.102 TCP 60 443 → 57285 [ACK] Seq=1 Ack=624
3747 88.258114220 52.123.128.14 10.248.249.102 TLSv1… 93 Application Data
3748 88.282236030 54.94.183.148 10.248.249.102 TCP 60 443 → 57305 [ACK] Seq=740 Ack=3
3749 88.290788601 17.248.201.72 10.248.249.102 TLSv1… 357 Application Data
3750 88.290789082 17.248.201.72 10.248.249.102 TLSv1… 357 Application Data
3751 88.290789253 17.248.201.72 10.248.249.102 TLSv1… 125 Application Data
3752 88.290789423 10.248.249.102 17.248.201.72 TCP 60 57301 → 443 [ACK] Seq=18544 Ack=
3753 88.290789593 17.248.201.72 10.248.249.102 TCP 60 443 → 57301 [ACK] Seq=7246 Ack=
3754 88.290789764 17.248.201.72 10.248.249.102 TCP 60 443 → 57301 [ACK] Seq=7246 Ack=
3755 88.290903424 10.248.249.102 17.248.201.72 TLSv1… 43… Application Data
3756 88.290903664 10.248.249.102 17.248.201.72 TLSv1… 15… Application Data
Frame 4159: 3406 bytes on wire (27248 bits), 3406 bytes captured
Ethernet II, Src: ASUSTekCOMPU_58:fa:21 (3c:7c:3f:58:fa:21), Dst:
Internet Protocol Version 4, Src: 10.248.249.102, Dst: 17.248.201
Transmission Control Protocol, Src Port: 57294, Dst Port: 443, Se
Transport Layer Security
0000 d8 ec 5e b9 7b b2 3c 7c 3f
0010 0d 40 2b e9 40 00 80 06 e2
0020 c9 48 df ce 01 bb b0 aa 35
0030 00 fc ec d1 00 00 17 03 03
0040 e8 3c 59 98 cd f4 24 e9 99
0050 d8 f9 e8 cc bf c2 49 ec a4
0060 8d eb fa 35 0e 50 42 ab 71
0070 5d b4 0f 56 72 8d ca 4c 8d
0080 1d 9c b1 f9 b5 39 79 1d 3b
0090 18 e3 da 80 9b c4 04 0d 8e
00a0 c1 87 45 16 e6 6e 6b 87 a5
00b0 93 5c 5a f8 d2 76 2a e6 00
00c0 f4 3d 1f b8 10 3f 6e f6 27
00d0 78 51 b4 1c 9b 53 b9 a3 14
00e0 c0 6c 8f 89 ba 5d 95 1a f1
wireshark_eth0XSOKB3.pcapng
Packets: 14424 · Displayed: 4810 (33.3%) · Dropped: 0 (0.0%)
Profile: Default

UDP.PORT==53

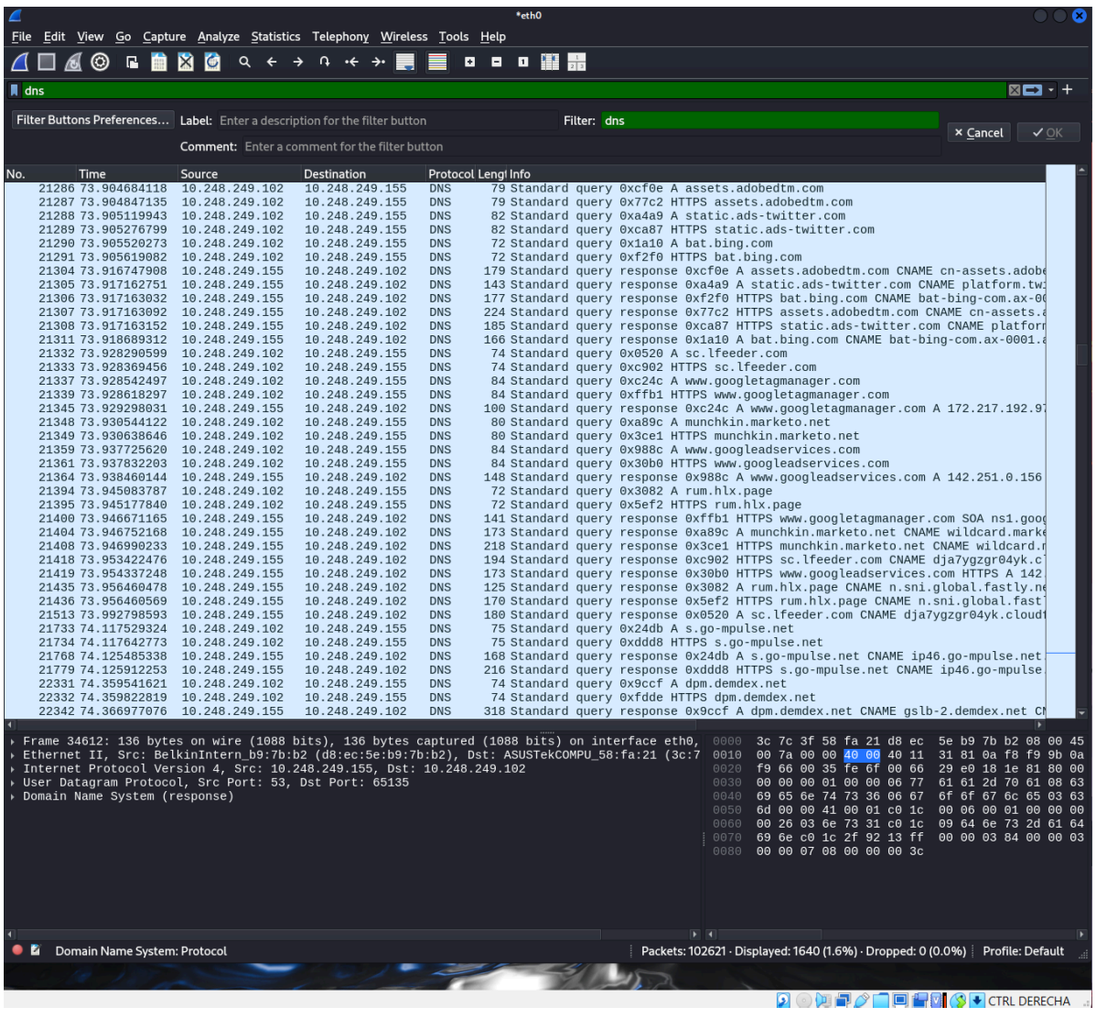


Captura y Análisis de Tráfico con Wireshark
Realizar una sesión de captura de tráfico de red durante 10-15 minutos mientras navegas
por diferentes sitios web:

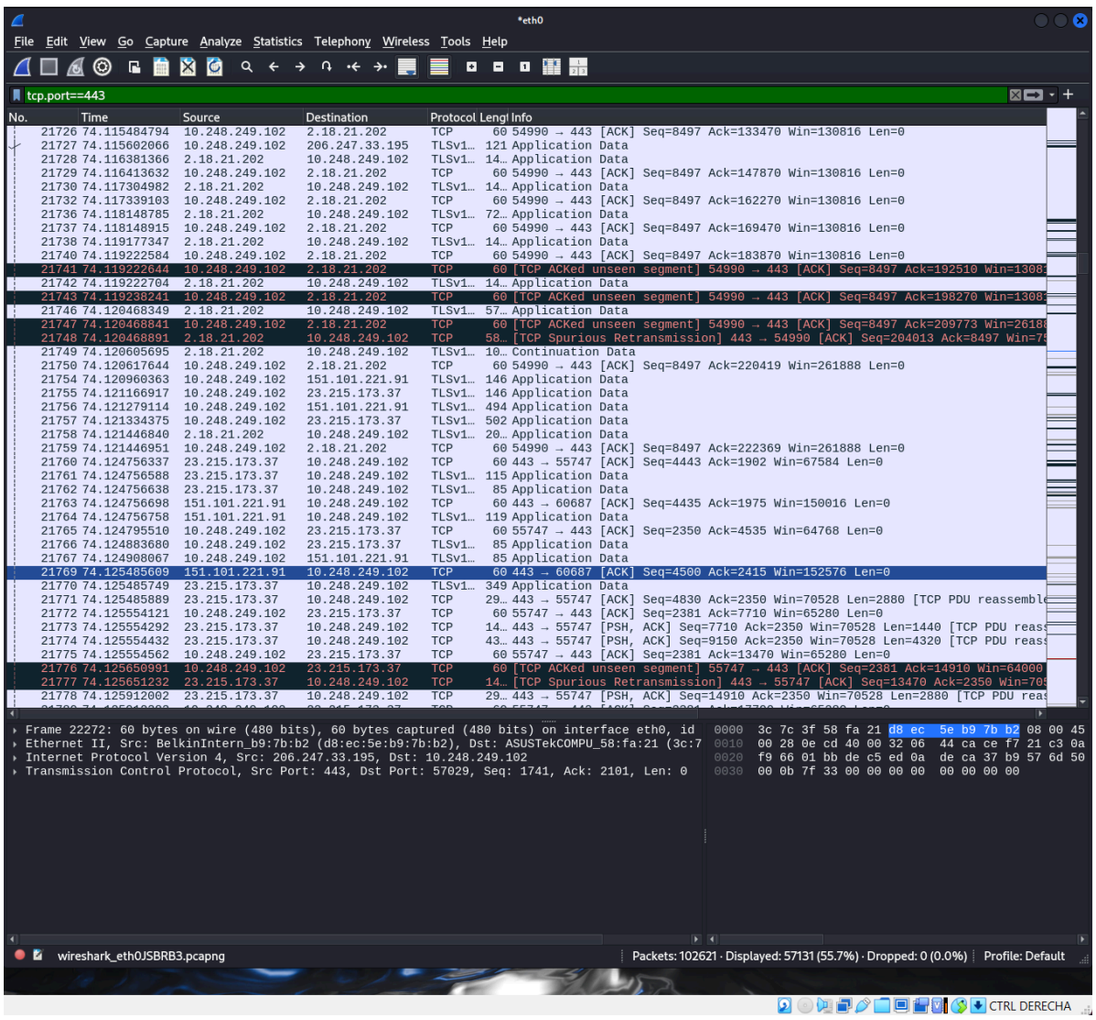
- a) Acceder a al menos 5 sitios web diferentes (incluir HTTP y HTTPS)
- b) Realizar una descarga de archivo pequeño
- c) Enviar un correo electrónico o usar una aplicación de mensajería
- d) Aplicar los siguientes filtros en Wireshark y documentar los resultados:
  - http - Tráfico HTTP



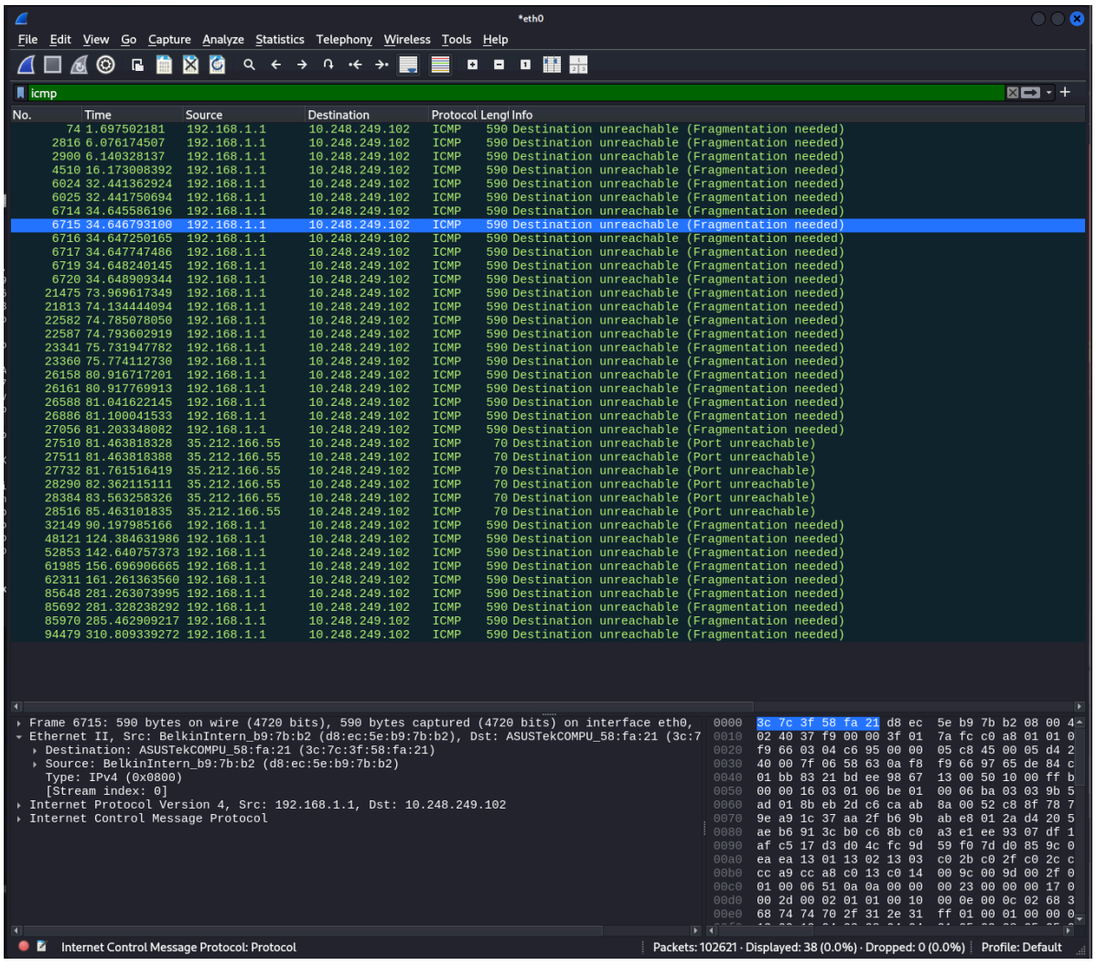
- dns - Consultas DNS



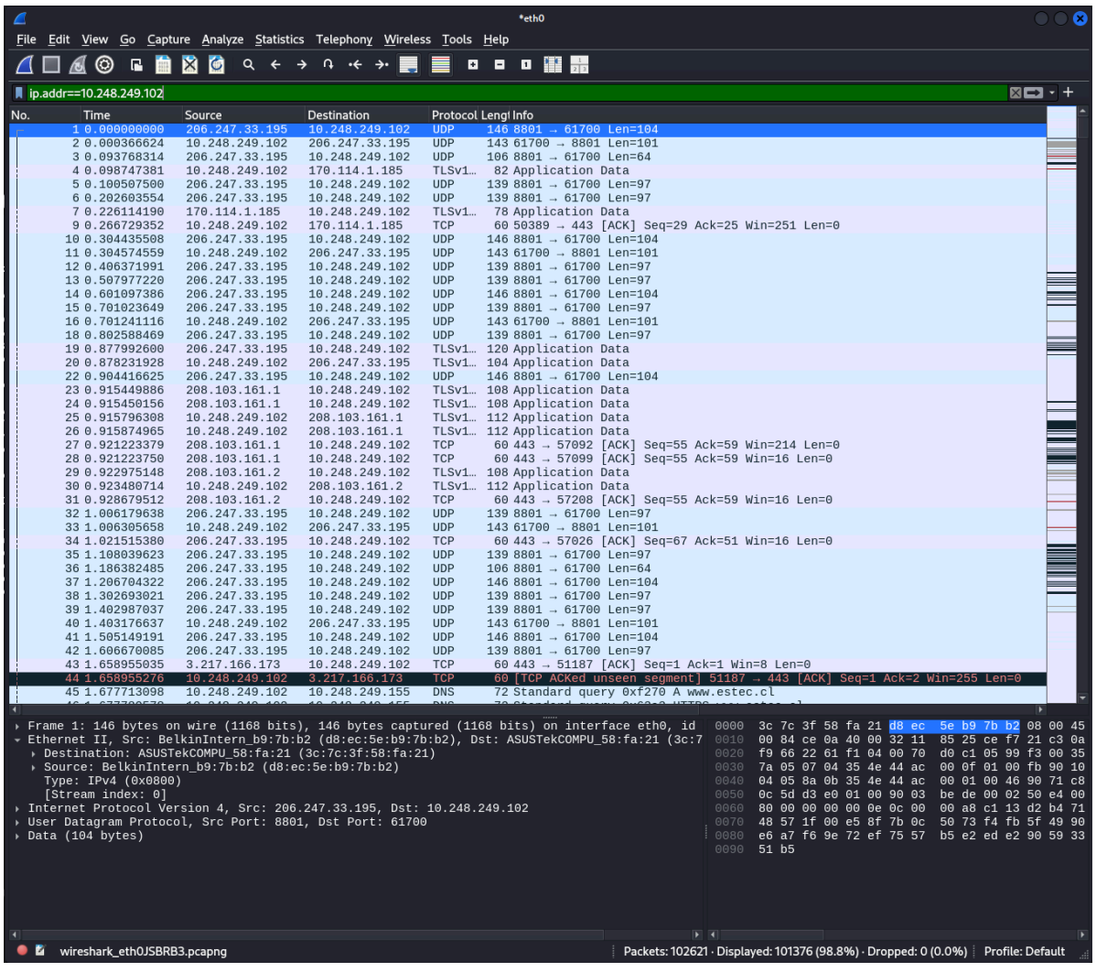
- tcp.port == 443 - Tráfico HTTPS



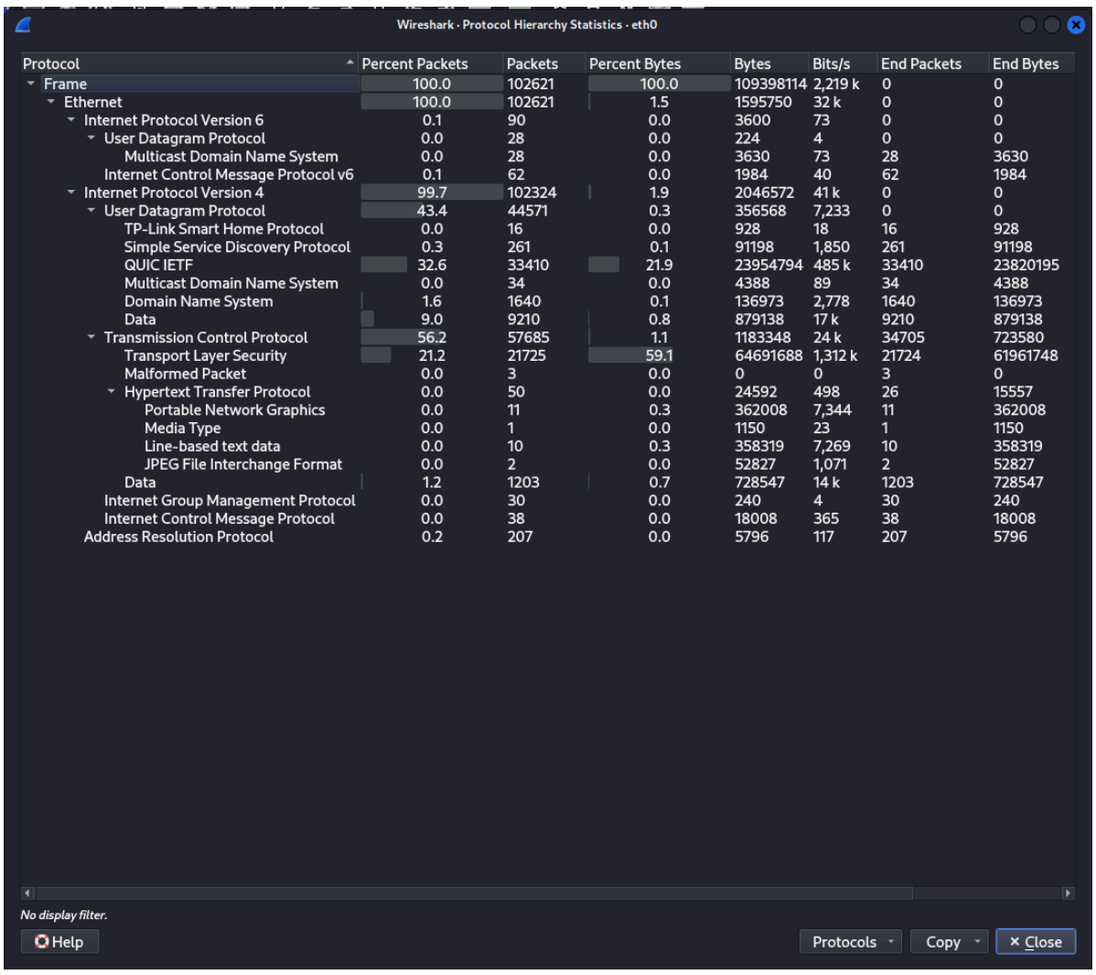
- icmp - Tráfico ICMP



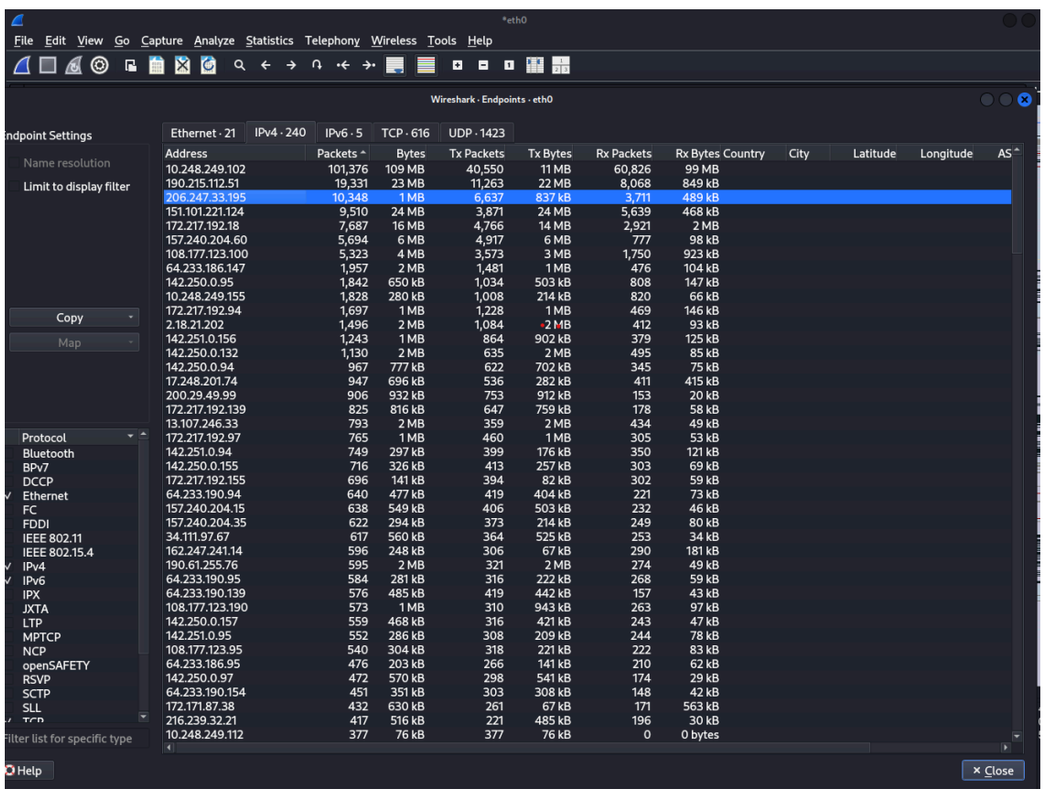
- ip.addr == [10.248.249.102] - Todo el tráfico de tu máquina

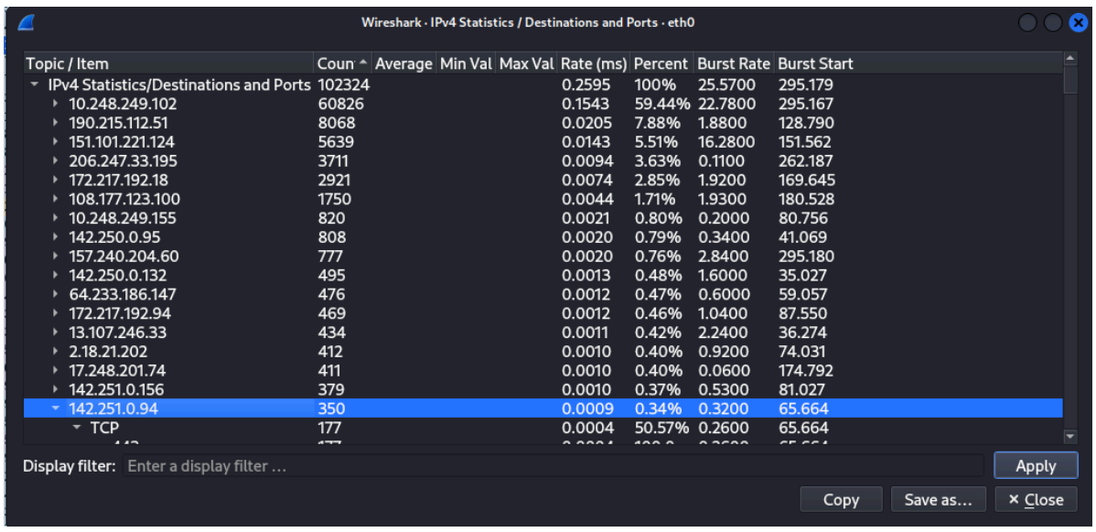


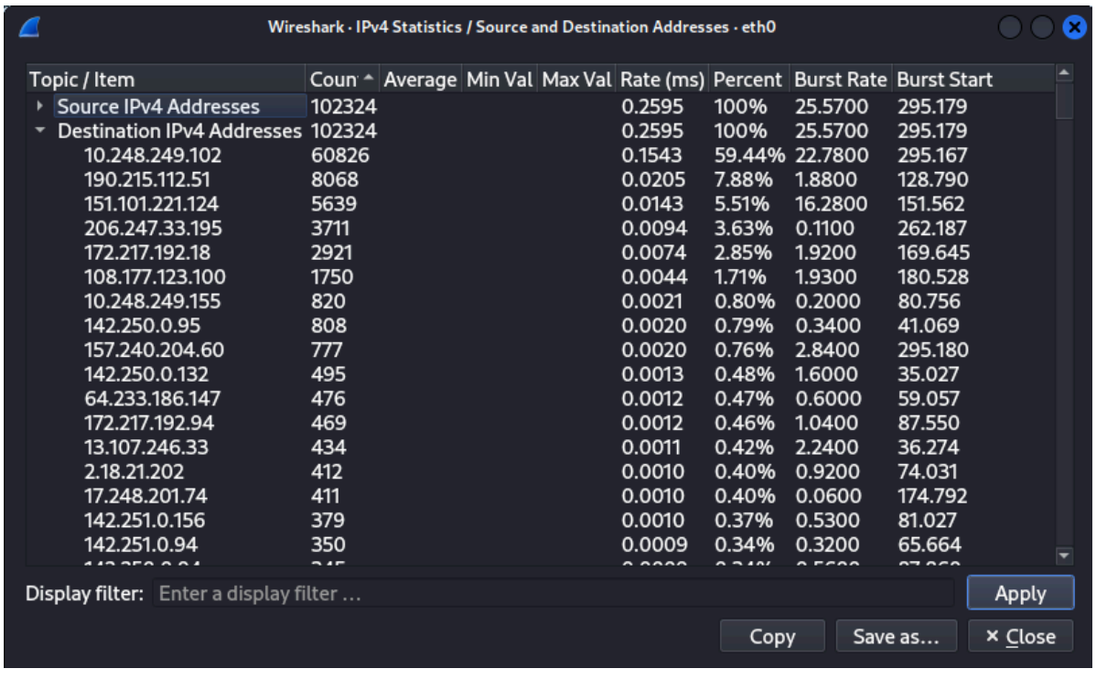
- e) Identificar y explicar:
  - Protocolos más utilizados



- Direcciones IP de destino más frecuentes

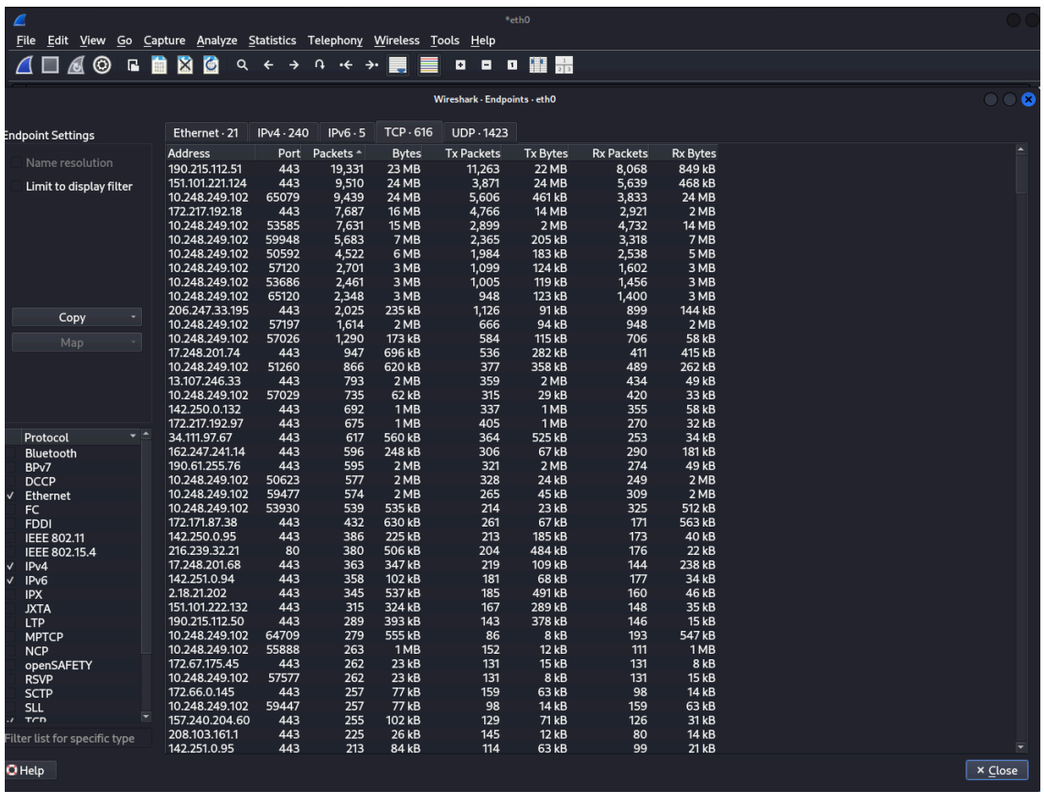




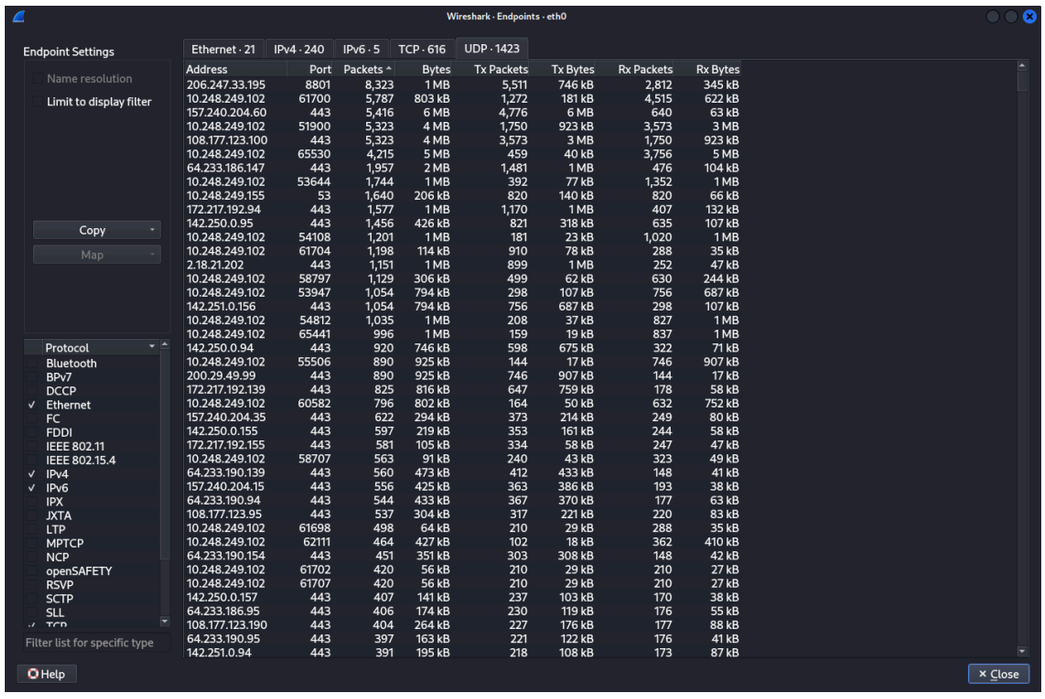


- Puertos más utilizados

  Puertos TCP

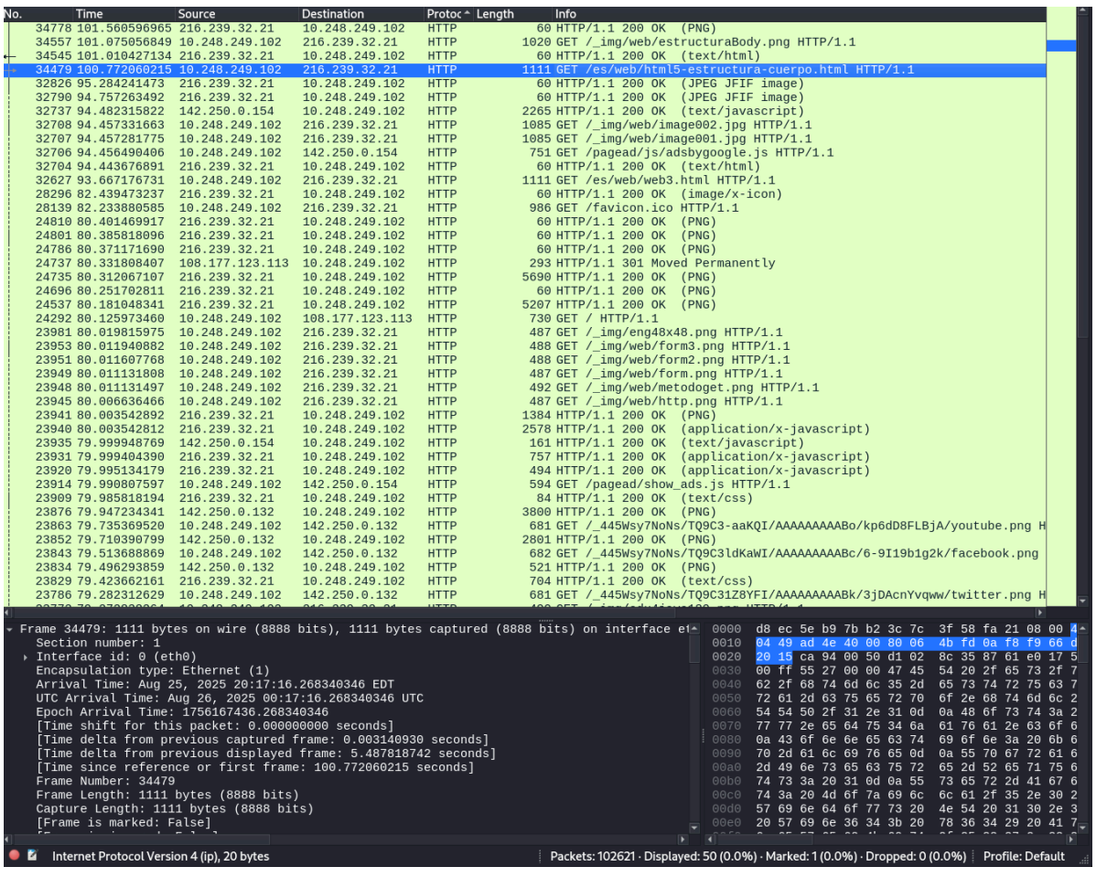


Puertos UDP



- Posibles vulnerabilidades observadas (tráfico no cifrado, etc.)

Tendría que revisar a detalle el trafico generado para determinar si hay vulnerabilidades mayores, a simple vista en el trafico http se logran identificar los nombres de algunos archivos y sus posibles rutas los cuales no se encuentran cifrados.



Análisis de Conectividad y Respuesta de Red
Utilizando tanto hping3 como Wireshark:

- a) Realizar un análisis de conectividad a diferentes puertos de un servidor remoto:
  - Puerto 22 (SSH)

```
┌──(root㉿kali)-[/home/kali]
└─# hping3 -S -p 22 -c 1 scanme.nmap.org
HPING scanme.nmap.org (eth0 45.33.32.156): S set, 40 headers + 0 data bytes
len=46 ip=45.33.32.156 ttl=45 DF id=0 sport=22 flags=SA seq=0 win=64240 rtt=1
93.0 ms

--- scanme.nmap.org hping statistic ---
1 packets transmitted, 1 packets received, 0% packet loss
round-trip min/avg/max = 193.0/193.0/193.0 ms
```

  - Puerto 80 (HTTP)

```
┌──(root㉿kali)-[/home/kali]
└─# hping3 -S -p 80 -c 1 scanme.nmap.org
HPING scanme.nmap.org (eth0 45.33.32.156): S set, 40 headers + 0 data bytes
len=46 ip=45.33.32.156 ttl=45 DF id=0 sport=80 flags=SA seq=0 win=64240 rtt=1
83.8 ms

--- scanme.nmap.org hping statistic ---
1 packets transmitted, 1 packets received, 0% packet loss
round-trip min/avg/max = 183.8/183.8/183.8 ms
```

  - Puerto 443 (HTTPS)

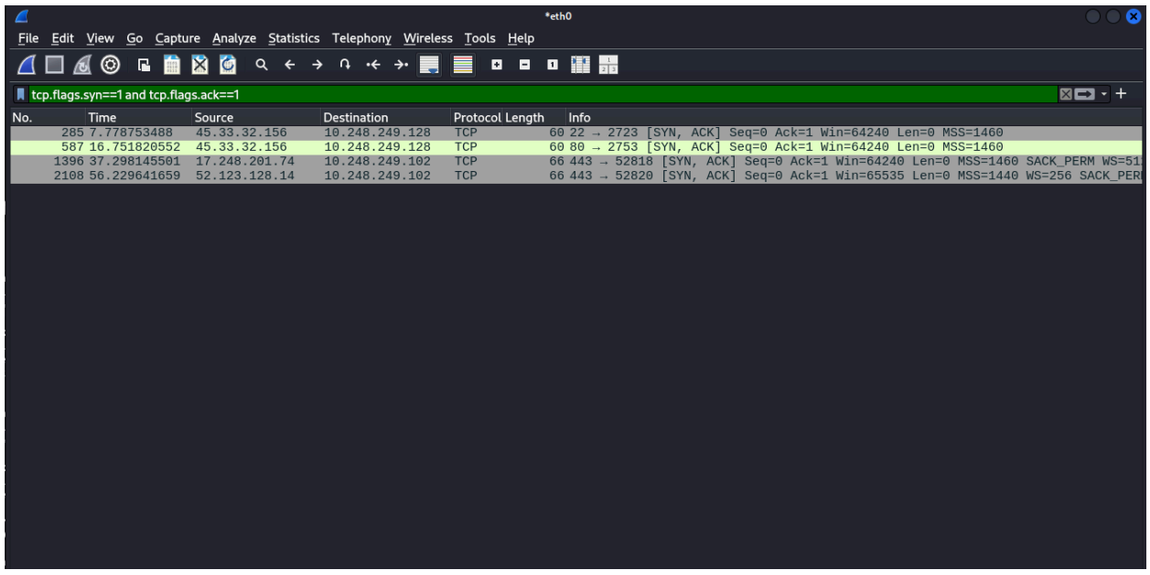
```
┌──(root㉿kali)-[/home/kali]
└─# hping3 -S -p 443 -c 1 scanme.nmap.org
HPING scanme.nmap.org (eth0 45.33.32.156): S set, 40 headers + 0 data bytes
len=46 ip=45.33.32.156 ttl=45 DF id=0 sport=443 flags=RA seq=0 win=0 rtt=175.
8 ms

--- scanme.nmap.org hping statistic ---
1 packets transmitted, 1 packets received, 0% packet loss
round-trip min/avg/max = 175.8/175.8/175.8 ms
```

- Puerto 21 (FTP)

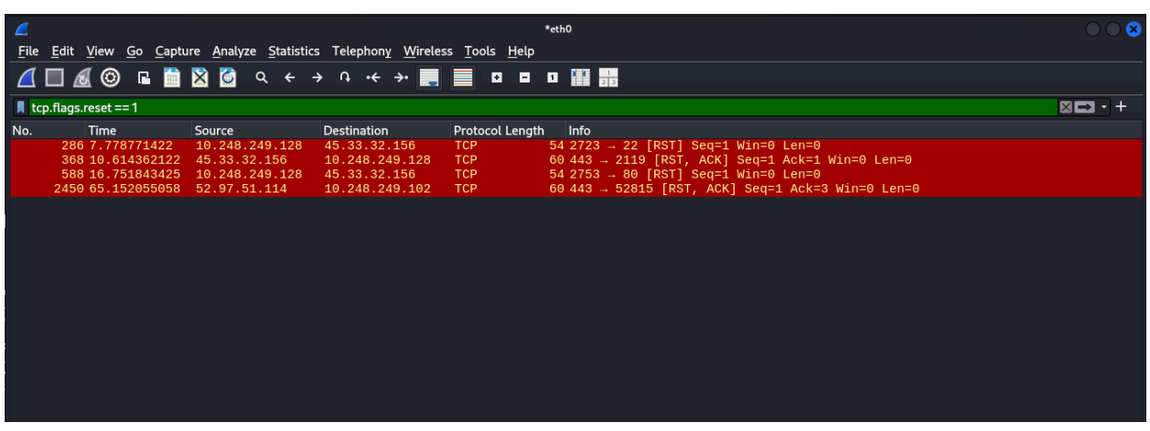
```
┌──(root㉿kali)-[/home/kali]
└─# hping3 -2 -p 21 -c 1 scanme.nmap.org
HPING scanme.nmap.org (eth0 45.33.32.156): udp mode set, 28 headers + 0 data
bytes
ICMP Port Unreachable from ip=45.33.32.156 name=scanme.nmap.org
status=0 port=1576 seq=0

--- scanme.nmap.org hping statistic ---
1 packets transmitted, 1 packets received, 0% packet loss
round-trip min/avg/max = 575.1/575.1/575.1 ms

┌──(root㉿kali)-[/home/kali]
└─#
```

- b) Documentar qué puertos están abiertos, cerrados o filtrados

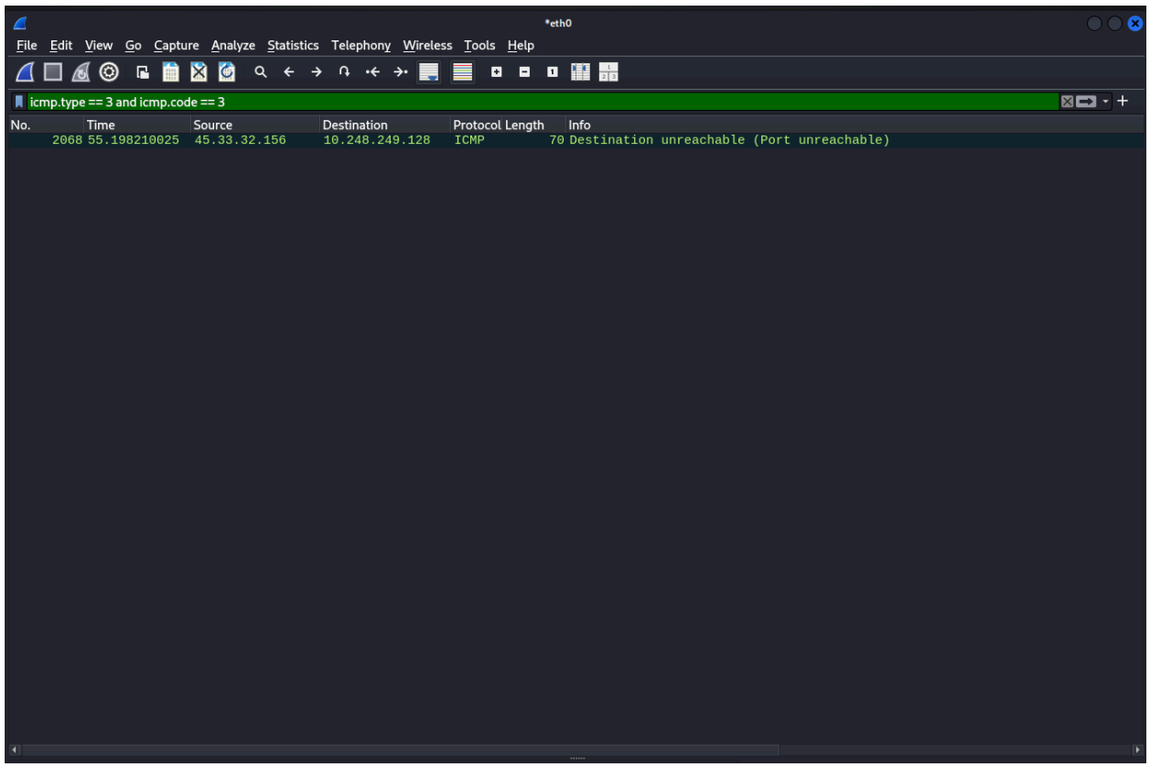
Abiertos



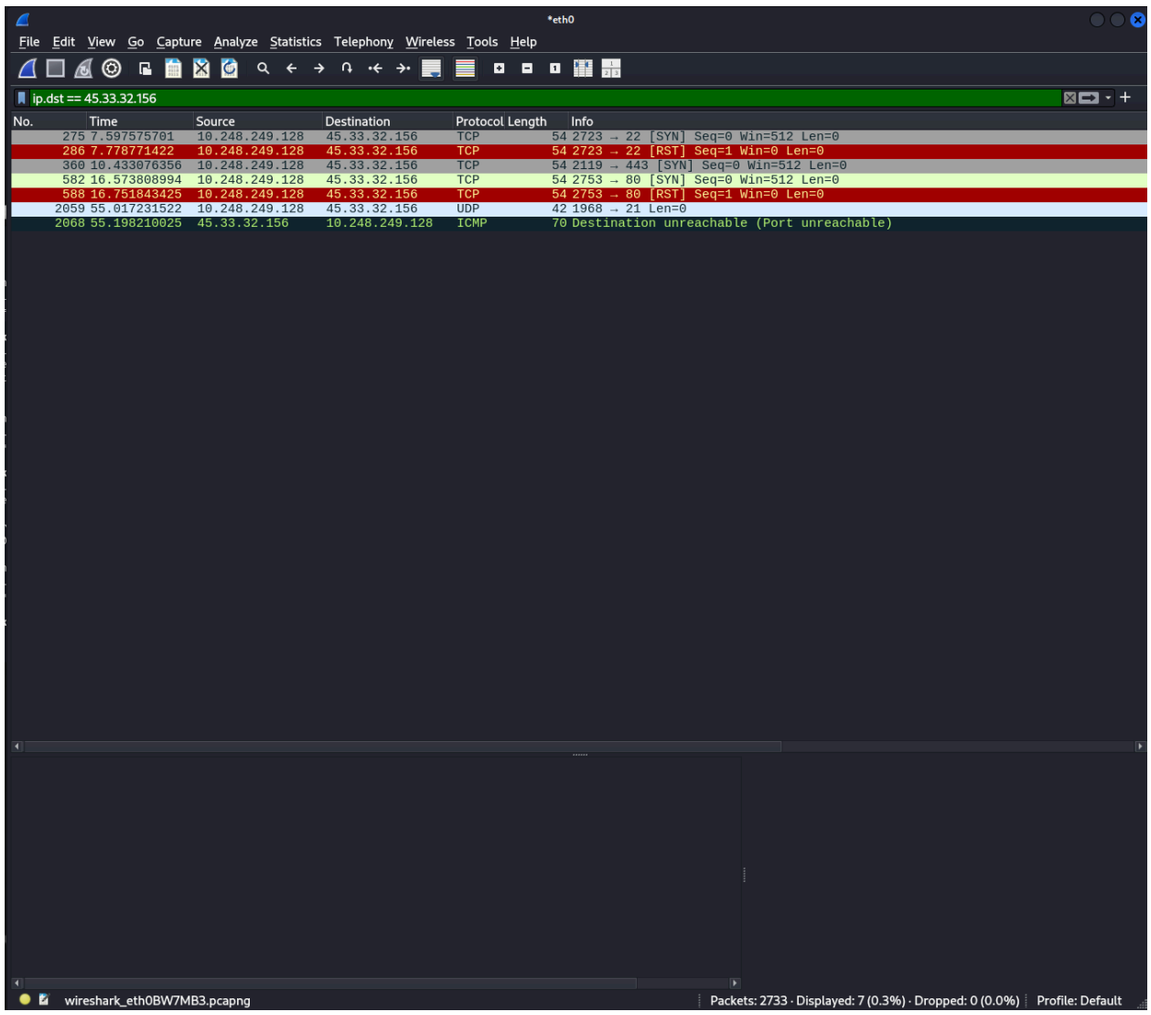
Cerrados



Puerto UDP



Filtrados



- c) Analizar los tiempos de respuesta y patrones de conectividad

  Puerto 22: Tiempo de respuesta promedio 193.0 ms

  Puerto 80: Tiempo de respuesta promedio 183.8 ms

  Puerto 443: Tiempo de respuesta promedio 175.8 ms

  Puerto 21: Tiempo de respuesta promedio 575.1 ms