

INGENIERÍA INFORMÁTICA
Escuela Politécnica Superior
Universidad Autónoma De Madrid

Introducción a Libpcap y Wireshark

Práctica 1

David Teófilo Garitagoitia Romero
Daniel Cerrato Sanchez

Pareja 9 Grupo 1322
10/2/2021

Índice de Contenidos

1. Ejercicio 1.....	2
2. Ejercicio 2.....	4
3. Ejercicio 3.....	6
4. Ejercicio 4.....	7
5. Ejercicio 5.....	8

1. Ejercicio 1

1. Abra una consola o shell, y déjela abierta en espera de ejecutar algún comando.
2. Ejecute Wireshark y seleccione y configure el interfaz por el que se capturará el tráfico (habitualmente será eth0) Acuérdesse de seleccionar las opciones de visualización que más le convenga.
3. Inicie la captura de tráfico pulsando en el botón 'Start'.
4. Vuelva a la consola y ejecute el siguiente comando (tecléelo y pulse): `$ sudo hping3 -S -p 80 www.uam.es`
5. Detenga la captura de tráfico mediante el botón 'Stop'.
6. Analice el tráfico capturado (aunque no lo entienda en detalle)
7. Guarde la traza en un fichero (Importante: no utilizar el formato pcap-ng).
8. Cierre Wireshark, y vuelva a abrirlo.
9. Abra el fichero almacenado y compruebe que se almacenó correctamente.
10. Utilizando las columnas que se han añadido durante el tutorial, ordene con respecto al campo 'PO' en sentido descendente y contabilice el número de paquetes en el que este campo tiene valor 53.

Describe el proceso realizado y discuta los problemas que haya encontrado durante la realización del ejercicio.

Procedemos a realizar los pasos como indica el enunciado.

Como podemos ver, al parar la capturar tráfico de la red tras esperar un rato desde el envío del comando `$ sudo hping3 -S -p 80 www.uam.es` se puede ver como se han enviado paquetes por el protocolo TCP al puerto 80 de la IP destino de la uam.

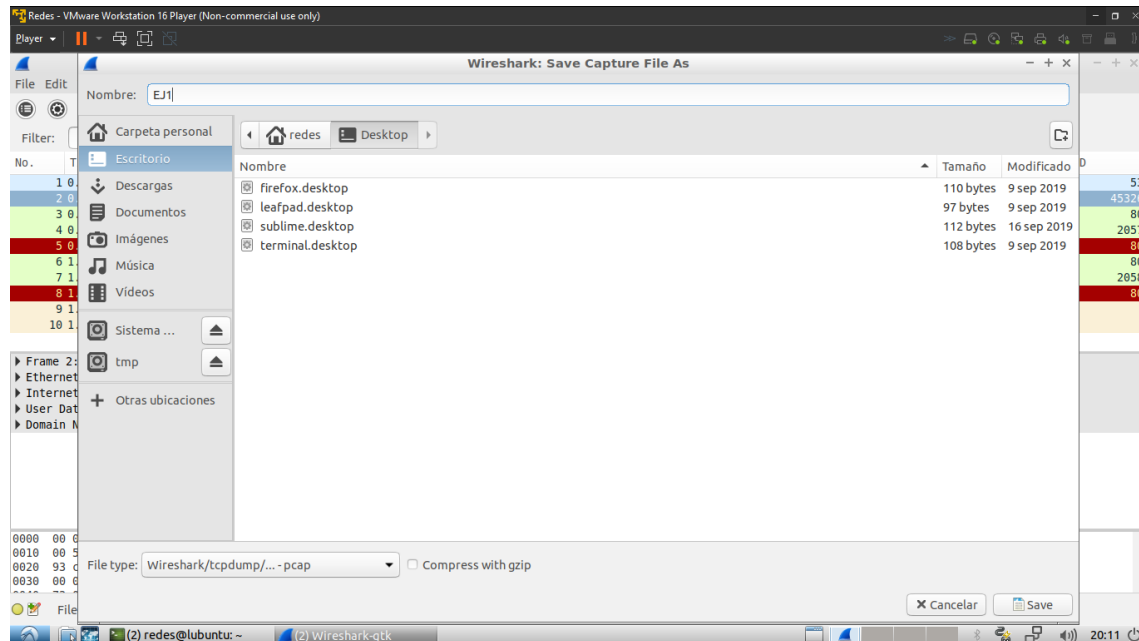
The screenshot displays the Wireshark network protocol analyzer interface. The top menu bar includes File, Edit, View, Go, Capture, Analyze, Statistics, Telephony, Tools, Internals, and Help. The main window is divided into three panes: Packet List, Packet Details, and Packet Bytes.

Packet List: Shows a list of captured packets. The selected packet is No. 97, a DNS Standard query response from 192.168.147.2 to 192.168.147.237. Other visible packets include TCP SYN packets and an ARP request.

Packet Details: Shows the hierarchical structure of the selected packet. It includes Ethernet II, Internet Protocol Version 4, User Datagram Protocol, and Domain Name System (response).

Packet Bytes: Shows the raw data of the selected packet in hexadecimal and ASCII format.

Guardamos la traza como se nos indica en el enunciado, en este caso lo guardaremos en el fichero EJ1 (evitando el uso del formato pcapng)



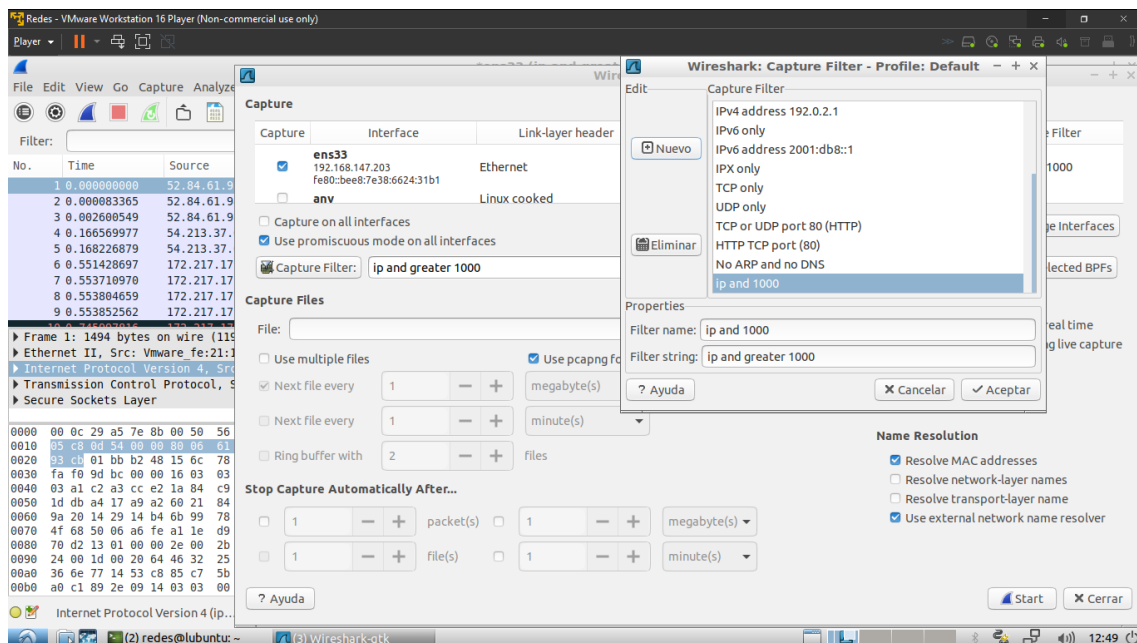
Tras cerrar Wireshark y volver a abrirlo para leer la traza guardada en el paso anterior, podemos ver como se conservan los datos, por último, ordenamos con respecto a la columna indicada (aclarar que wireshark ordena como si fuera texto y no número por lo que $12 < 2$)
Tras ordenarlos observamos como el puerto 53 tiene una única ocurrencia (el segundo paquete que se corresponde a la respuesta de la query)

2. Ejercicio 2

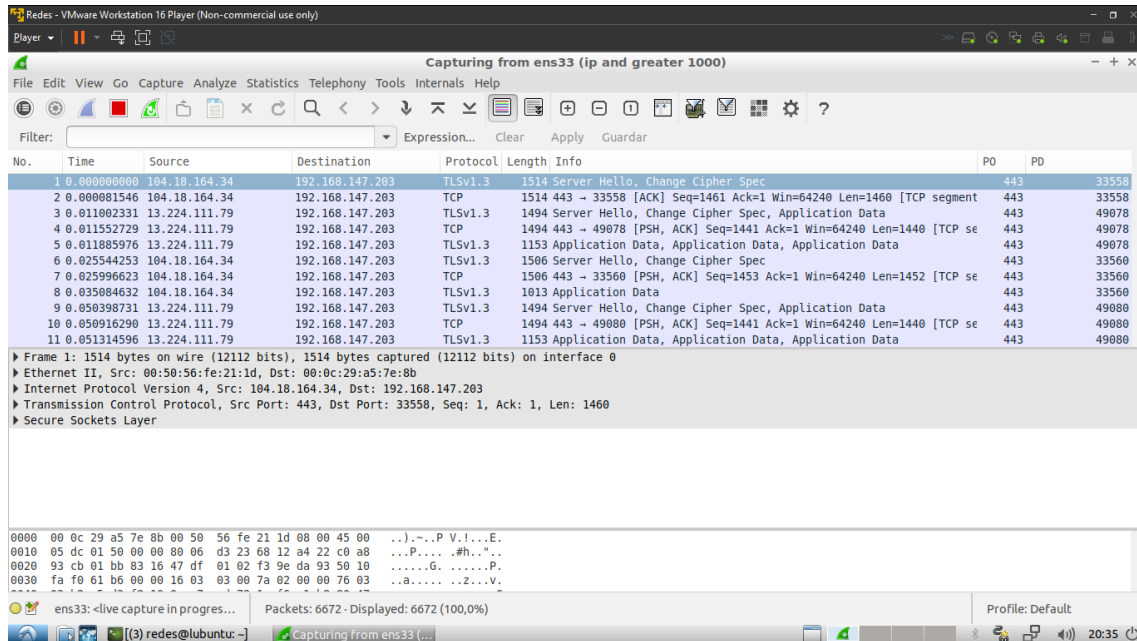
Tras haber leído la documentación online facilitada, empiece a capturar tráfico. Abra un navegador y genere tráfico a partir de la visualización de páginas web. Pare la captura, y añada un filtro en el interfaz de modo que solo se visualicen paquetes que sean de tipo IP y que tengan un tamaño de paquete mayor a 1000 Bytes.

1. Copie el filtro realizado.
 2. ¿Cómo almacenaría en una captura solo los paquetes mostrados?
 3. Compare el tamaño del primer paquete IP, y el campo 'length' del protocolo IP de este.
- Repita para los primeros 5 paquetes, ¿qué relación encuentra?

El filtro a usar es `ip and greater 1000`, tras estar un rato en firefox obtenemos los siguientes resultados (se puede comprobar como efectivamente el filtro logra lo esperado)



Para almacenarlos es tan fácil como al inicio, simplemente guardar la traza.



La diferencia de tamaño entre el tamaño del paquete IP y el campo 'length' del protocolo IP de este es de $1514 - 1460 = 54$; esta diferencia de tamaño se debe a las cabeceras de los diferentes niveles.

- El nivel 2 o nivel de enlace corresponde con Ethernet podemos ver los 14 bytes de su cabecera (6 que corresponden al destino, otros 6 que corresponden al origen y finalmente 2 bytes para el tipo)
- El nivel 3, o nivel de transporte con sus 20 bytes como indica wireshark
- El nivel de transporte que es de protocolo TCP que son otros 20 bytes

Sumando todo ello, obtenemos 54 bytes entre todas las cabeceras

Al restar esos 54 bytes al total de 1514 obtenemos efectivamente lo 1460 bytes

Lo mismo ocurre con el resto, la diferencia entre el tamaño del paquete y el campo length del protocolo son 54 bytes

3. Ejercicio 3

Añada una columna llamada interarrival que muestre el tiempo entre paquetes consecutivos. Explique brevemente qué menús y opciones ha seleccionado.

Simplemente pinchamos en edit preferences->user interface->columns le damos a añadir una nueva (que será la interarrival, en field type colocamos delta time y lo tenemos. (Delta time is the time between packets)

No.	Time	Source	Destination	Protocol	Length	interarrival	Info
1	0.000000000	52.84.61.98	192.168.147.203	TLSv1.3	1494	0.000000000	Server Hello, Change Cipher Spec, Application Data
2	0.000083365	52.84.61.98	192.168.147.203	TCP	1494	0.000083365	443 → 45640 [PSH, ACK] Seq=1441 Ack=1 Win=64240 Len=1440 [TCP segment of...
3	0.002600549	52.84.61.98	192.168.147.203	TLSv1.3	1121	0.002517184	Application Data, Application Data, Application Data
4	0.166569977	54.213.37.69	192.168.147.203	TLSv1.2	1506	0.163969428	Server Hello
5	0.168226879	54.213.37.69	192.168.147.203	TCP	1506	0.001656902	443 → 58810 [PSH, ACK] Seq=1453 Ack=1 Win=64240 Len=1452 [TCP segment...
6	0.551420897	172.217.17.14	192.168.147.203	TLSv1.3	1484	0.383201818	Server Hello, Change Cipher Spec
7	0.553710970	172.217.17.14	192.168.147.203	TCP	1514	0.002282273	443 → 57044 [ACK] Seq=1431 Ack=1 Win=64240 Len=1460 [TCP segment of ...]
8	0.553804659	172.217.17.14	192.168.147.203	TCP	1514	0.000093689	443 → 57044 [ACK] Seq=2891 Ack=1 Win=64240 Len=1460 [TCP segment of ...]
9	0.553852562	172.217.17.14	192.168.147.203	TCP	1514	0.000047983	443 → 57044 [ACK] Seq=4351 Ack=1 Win=64240 Len=1460 [TCP segment of ...]
10	0.745997816	172.217.17.14	192.168.147.203	TLSv1.3	1484	0.192145254	[TCP Previous segment not captured], Application Data
11	0.746029642	172.217.17.14	192.168.147.203	TLSv1.3	1484	0.000031826	Application Data
12	0.746305341	172.217.17.14	192.168.147.203	TLSv1.3	1484	0.000275609	Application Data

Frame 1: 1494 bytes on wire (11952 bits), 1494 bytes captured (11952 bits) on interface 0
 Ethernet II, Src: Vmware fe:21:1d (00:50:56:fe:21:1d), Dst: Vmware a5:7e:8b (00:0c:29:a5:7e:8b)
 Internet Protocol Version 4, Src: 52.84.61.98, Dst: 192.168.147.203
 Transmission Control Protocol, Src Port: 443, Dst Port: 45640, Seq: 1, Ack: 1, Len: 1440
 Secure Sockets Layer

File: /tmp/wireshark_ens33_2... Packets: 2032 · Displayed: 2032 (100,0%) · Dropped: 0 (0,0%) Profile: Default

4. Ejercicio 4

Modifique la forma en que Wireshark muestra la información en la columna 'Time' de cada paquete. En concreto muestre los tiempos en formato para humanos, y en tiempo Unix con resolución en segundos. Explique brevemente los pasos realizados.

Cambiando el field type de time a absolute date and time, vemos como se muestra en la columna tanto la fecha como la hora en la que se capturó el paquete.

The screenshot shows the Wireshark interface with the following details:

- Filter:** *ens33 (ip and greater 1000)
- Packet List:**

No.	Time	Source	Destination	Protocol	Length	Interarrival	Info
1	2021-09-19 07:49:07.106642235	52.84.61.98	192.168.147.203	TLSv1.3	1494	0.000000000	Server Hello, Change Cipher Spec, Application Data
2	2021-09-19 07:49:07.106725600	52.84.61.98	192.168.147.203	TCP	1494	0.000003365	443 → 45640 [PSH, ACK] Seq=1441 Ack=1 Win=64240 Len=14
3	2021-09-19 07:49:07.109242784	52.84.61.98	192.168.147.203	TLSv1.3	1121	0.002517184	Application Data, Application Data, Application Data
4	2021-09-19 07:49:07.273212212	54.213.37.69	192.168.147.203	TLSv1.2	1506	0.163969428	Server Hello
5	2021-09-19 07:49:07.274869114	54.213.37.69	192.168.147.203	TCP	1506	0.001656902	443 → 58810 [PSH, ACK] Seq=1453 Ack=1 Win=64240 Len=14
6	2021-09-19 07:49:07.658070932	172.217.17.14	192.168.147.203	TLSv1.3	1484	0.383201818	Server Hello, Change Cipher Spec
7	2021-09-19 07:49:07.660353205	172.217.17.14	192.168.147.203	TCP	1514	0.002282273	443 → 57044 [ACK] Seq=1431 Ack=1 Win=64240 Len=1460 [T
8	2021-09-19 07:49:07.660446894	172.217.17.14	192.168.147.203	TCP	1514	0.000093689	443 → 57044 [ACK] Seq=2891 Ack=1 Win=64240 Len=1460 [T
9	2021-09-19 07:49:07.660494797	172.217.17.14	192.168.147.203	TCP	1514	0.000047903	443 → 57044 [ACK] Seq=4351 Ack=1 Win=64240 Len=1460 [T
10	2021-09-19 07:49:07.852640051	172.217.17.14	192.168.147.203	TLSv1.3	1484	0.192145254	[TCP Previous segment not captured], Application Data
11	2021-09-19 07:49:07.852671877	172.217.17.14	192.168.147.203	TLSv1.3	1484	0.000031826	Application Data

- Packet Details:**
 - Frame 1: 1494 bytes on wire (11952 bits), 1494 bytes captured (11952 bits) on interface 0
 - Ethernet II, Src: Vmware fe:21:1d (00:50:56:fe:21:1d), Dst: Vmware a5:7e:8b (00:0c:29:a5:7e:8b)
 - Internet Protocol Version 4, Src: 52.84.61.98, Dst: 192.168.147.203
 - Transmission Control Protocol, Src Port: 443, Dst Port: 45640, Seq: 1, Ack: 1, Len: 1440
 - Secure Sockets Layer
- Packet Bytes:**

```
0000 00 0c 29 a5 7e 8b 00 50 56 fe 21 1d 08 00 45 00  ...P V!...E.
0010 05 c8 0d 54 00 00 00 06 61 b2 34 54 3d 62 c0 a8  ...T... a.4T=b..
0020 93 cb 01 bb b2 48 15 6c 78 1e 03 e8 74 d5 50 18  ....H.l X...t.P.
0030 fa f0 9d bc 00 00 16 03 03 00 7a 02 00 00 76 03  ....Z...v.
```

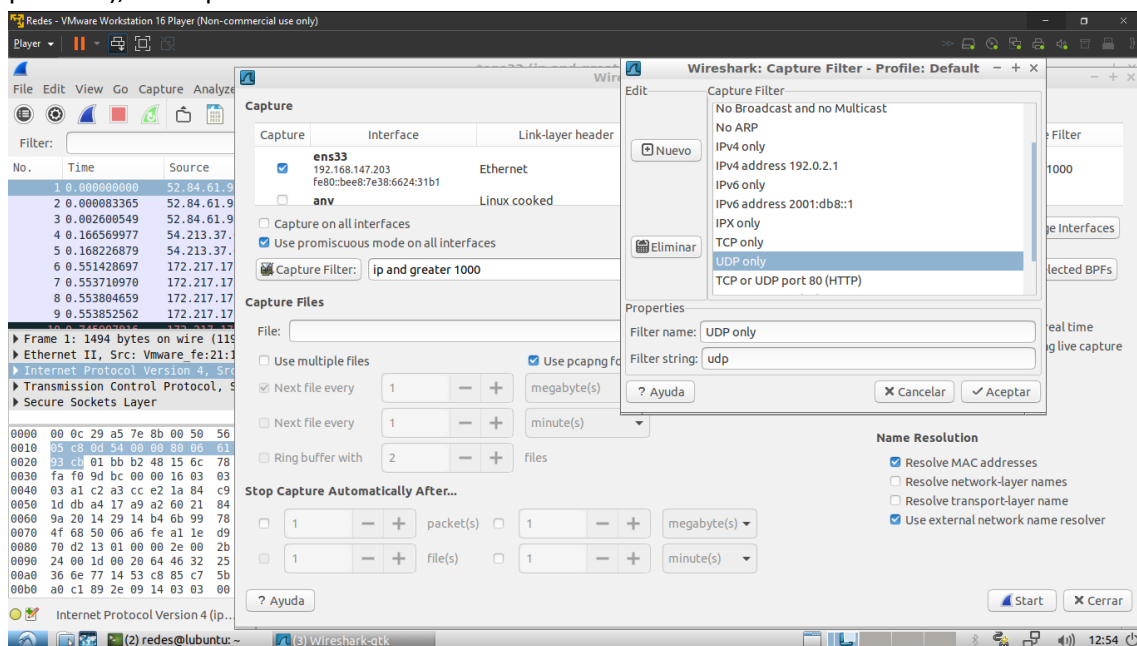
File: /tmp/wireshark_ens33_2... Packets: 2032 - Displayed: 2032 (100,0%) - Dropped: 0 (0,0%) Profile: Default

5. Ejercicio 5

Inicie una captura en Wireshark, pero aplicando filtros de captura, en concreto solo queremos capturar tráfico UDP. Mientras captura tráfico, genere durante algunos instantes tráfico a partir de la visualización de páginas web, y ejecute al mismo tiempo en una consola el comando `$ sudo hping3 -S -p 80 www.uam.es`. Compruebe que solo se capturan paquetes UDP, y describa brevemente los pasos realizados.

Para ello aplicamos el filtro UDP only.

Tras aplicarlo se puede ver como solo aparecen paquetes con protocolos UDP (user diagram protocol), como pueden ser los DNS o SSDP.



No aparecen paquetes relacionados con el comando de terminal `hping3` ya que este envía paquetes TCP por lo que wireshark con la configuración empleada no los captura.

redes@lubuntu: ~
t=15.1 ms
len=46 ip=150.244.214.237 ttl=128 id=55516 sport=80 flags=...
t=16.2 ms
len=46 ip=150.244.214.237 ttl=128 id=55517 sport=80 flags=...
t=17.7 ms
len=46 ip=150.244.214.237 ttl=128 id=55519 sport=80 flags=...
t=13.2 ms
--- www.uam.es hping statistic ---
15 packets transmitted, 15 packets received, 0% packet loss
round-trip min/avg/max = 8.8/80.1/1010.6 ms
redes@lubuntu:~\$

Mozilla Firefox
Nueva pestaña x Firefox Privacy Notice-- x +
youtube
Bienvenido a Firefox
Tiene el navegador.
Conozca el resto de Firefox.
Únase a

*ens33 (udp)
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Filter: Expression... Clear Apply Guardar
No. Time Interarrival Protocol Source Destination Length Ir
1 2021-10-02 15:34:26,58 0.000000000 DNS 192.168.147.203 192.168.147.2 81 St
2 2021-10-02 15:34:26,59 0.006683502 DNS 192.168.147.2 192.168.147.203 97 St
3 2021-10-02 15:34:30,17 3.582925743 DNS 192.168.147.203 192.168.147.2 85 St
4 2021-10-02 15:34:30,17 0.000264306 DNS 192.168.147.203 192.168.147.2 85 St
5 2021-10-02 15:34:30,18 0.005735832 DNS 192.168.147.2 192.168.147.203 101 St
6 2021-10-02 15:34:30,18 0.000032043 DNS 192.168.147.2 192.168.147.203 113 St
7 2021-10-02 15:34:30,23 0.049637042 DNS 192.168.147.203 192.168.147.2 84 St
8 2021-10-02 15:34:30,23 0.000379807 DNS 192.168.147.203 192.168.147.2 84 St
9 2021-10-02 15:34:30,24 0.007867475 DNS 192.168.147.2 192.168.147.203 135 St
10 2021-10-02 15:34:30,24 0.000043725 DNS 192.168.147.2 192.168.147.203 147 St
Frame 1: 81 bytes on wire (648 bits), 81 bytes captured (648 bits) on interface 0
Ethernet II, Src: Vmware_a5:7e:8b (00:0c:29:a5:7e:8b), Dst: Vmware_fe:21:1d (00:50:56:fe:21:1d)
Internet Protocol Version 4, Src: 192.168.147.203, Dst: 192.168.147.2
User Datagram Protocol, Src Port: 39797, Dst Port: 53
Domain Name System (query)
0000 00 50 56 fe 21 1d 00 0c 29 a5 7e 8b 00 00 45 00 .PV.).-...E.
0010 00 43 da 9f 40 00 40 11 47 ec c0 a8 93 cb c0 a8 .CJ.@.@.G.....
0020 93 02 9b 75 00 35 00 2f 82 6e ba 1b 01 00 00 01 ...U.S./..n.....
0030 00 00 00 00 00 01 03 77 77 77 03 75 61 6d 02 65w.w.uam.e
File: "/tmp/wireshark_ens33_2..." Packets: 15 - Displayed: 15 (100,0%) - Droppe... Profile: Default

[FINAL DE DOCUMENTO]