

# Redes 1 – Práctica 1

WIRESHARK

Alfonso Camblor – Ingeniería Informática – UAM

# Contenido

Ejercicio 1 ..... 2

Ejercicio 2 ..... 3

Ejercicio 3 ..... 4

Ejercicio 4 ..... 5

Ejercicio 5 ..... 6

# Ejercicio 1

Abierta la consola o Shell, ejecuto el comando

“sudo wireshark-gtk”

Para abrir el programa Wireshark con funciones de captura de paquetes habilitadas.

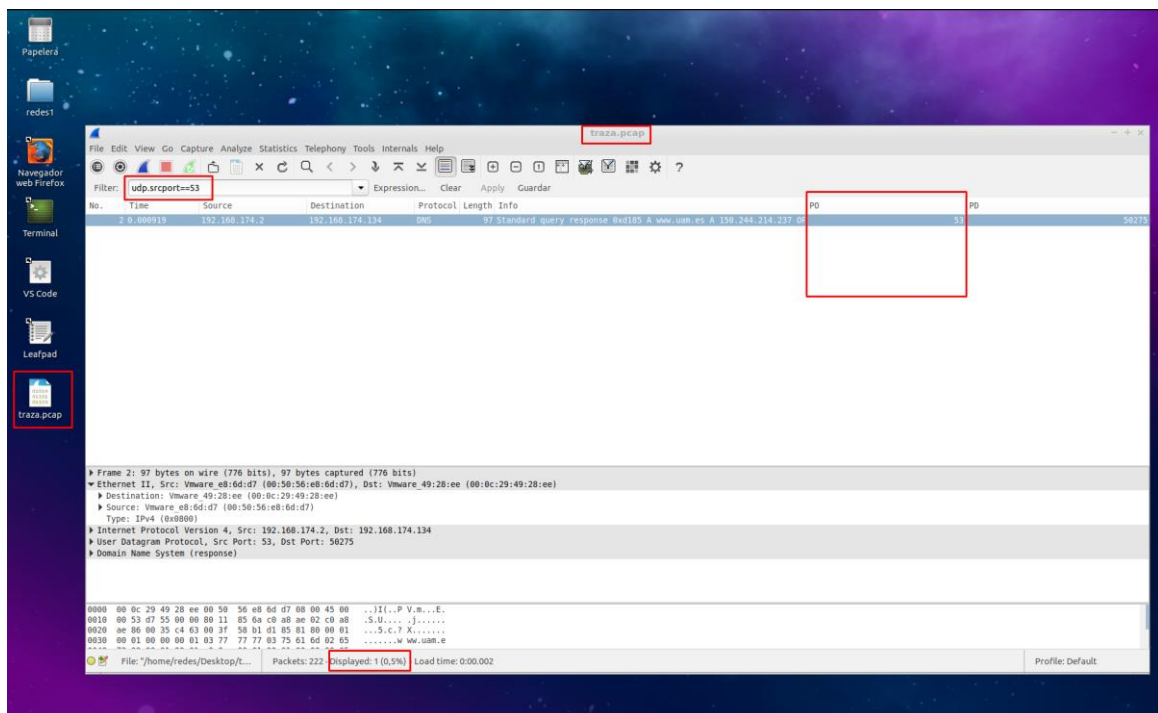
Hecho esto, comienzo la captura de la interfaz “ens33” y simultáneamente en otra terminal introduzco el comando

“sudo hping3 -S -p 80 [www.uam.es](http://www.uam.es)”

El resultado de estas acciones es la captura de gran cantidad de paquetes por parte de Wireshark. El ejercicio nos pide guardar la traza en un formato que no sea “pcap-ng”, por lo que fue usado el formato “pcap”. Reiniciamos Wireshark y abrimos el fichero guardado.

Anteriormente habíamos creado las columnas P0 y PD, asignándolas al valor de puerto de origen y el del puerto destino correspondientemente.

Una vez ordenamos en función de P0 de forma descendente, encontramos:



Contabilizamos 1 paquete, pues es indicado en la parte inferior del programa (Displayed).

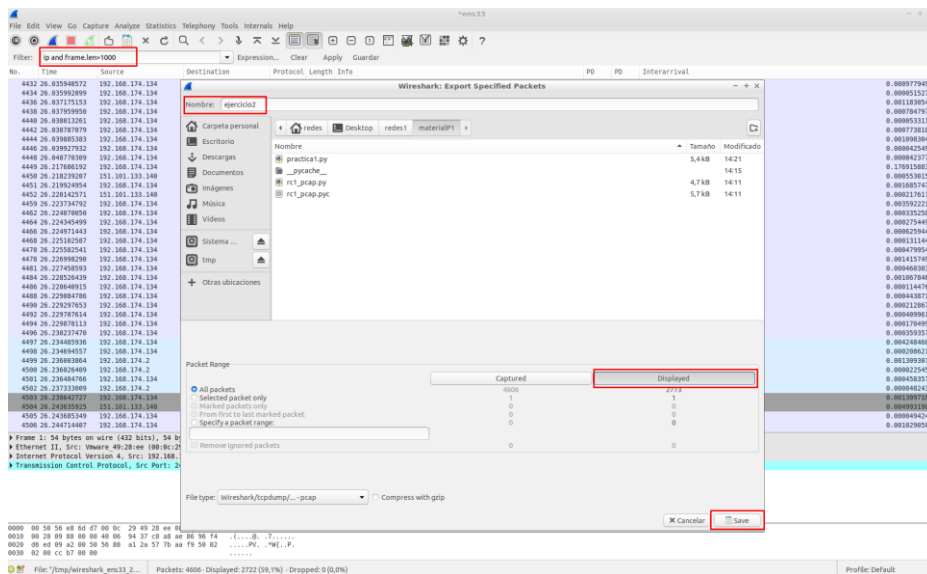
# Ejercicio 2

Empezamos activando la captura de tráfico y abriendo “[www.reddit.com](http://www.reddit.com)” en el navegador.

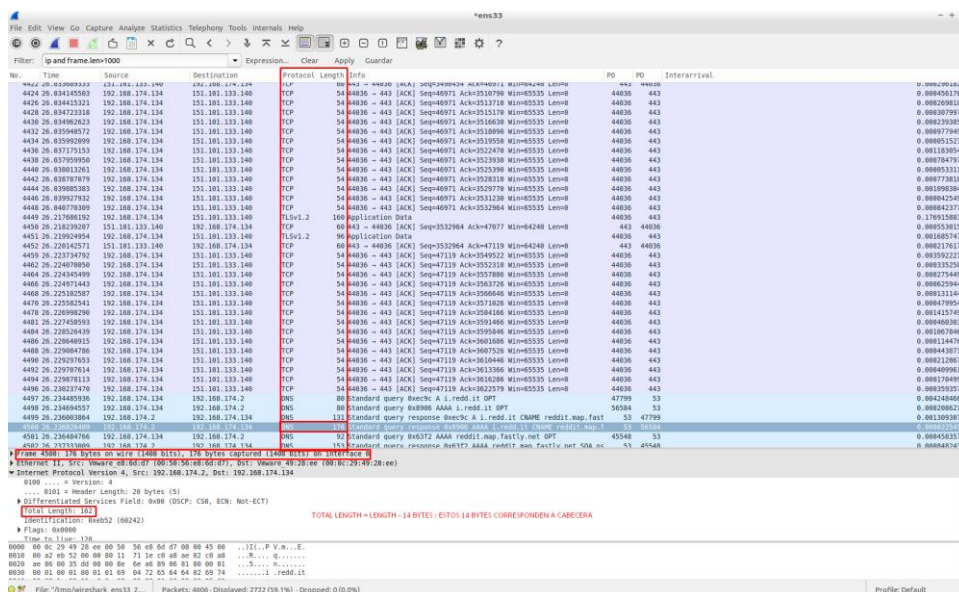
Capturamos paquetes durante unos 20 segundos.

Introducimos el filtro “ip and frame.len > 1000” para asegurarnos de filtrar los paquetes de tipo IP y cuyo tamaño sea MAYOR a 1000bytes.

Guardamos una traza con los paquetes filtrados:



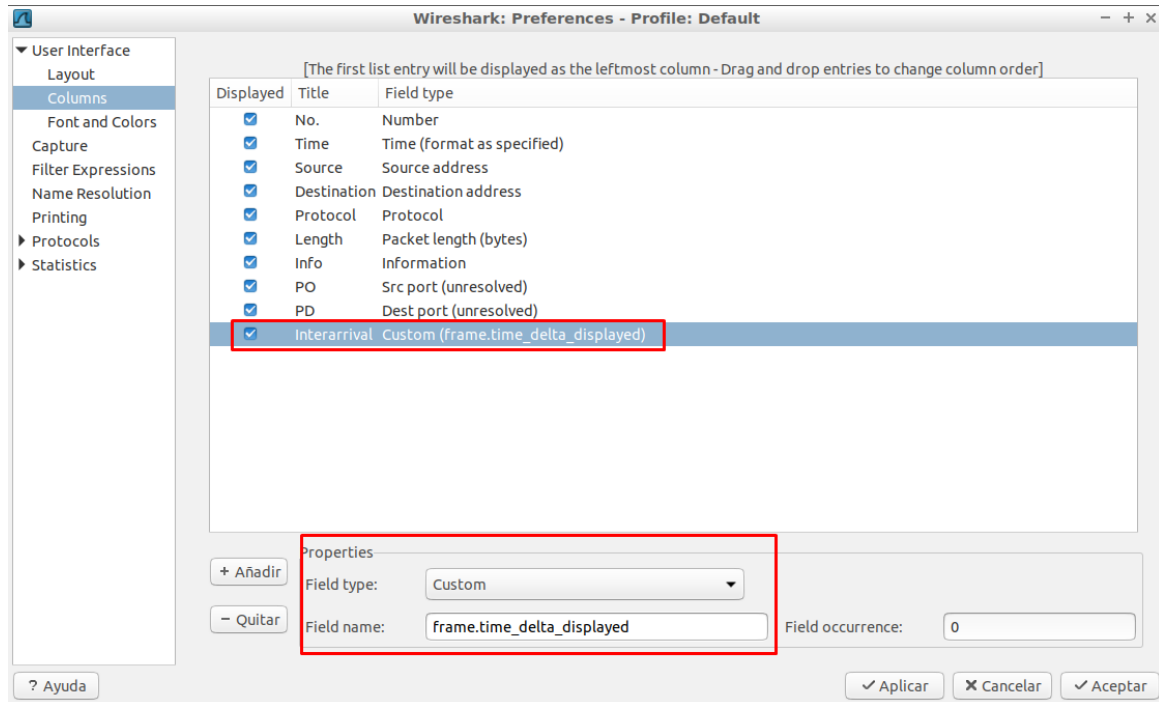
Determinamos que los 14 primeros bytes de cada paquete corresponden a la cabecera del paquete:



## Ejercicio 3

Para añadir la columna, empezamos creando una nueva y declarándola de tipo custom, con el comando:

“frame.time\_delta\_displayed” en Field name.



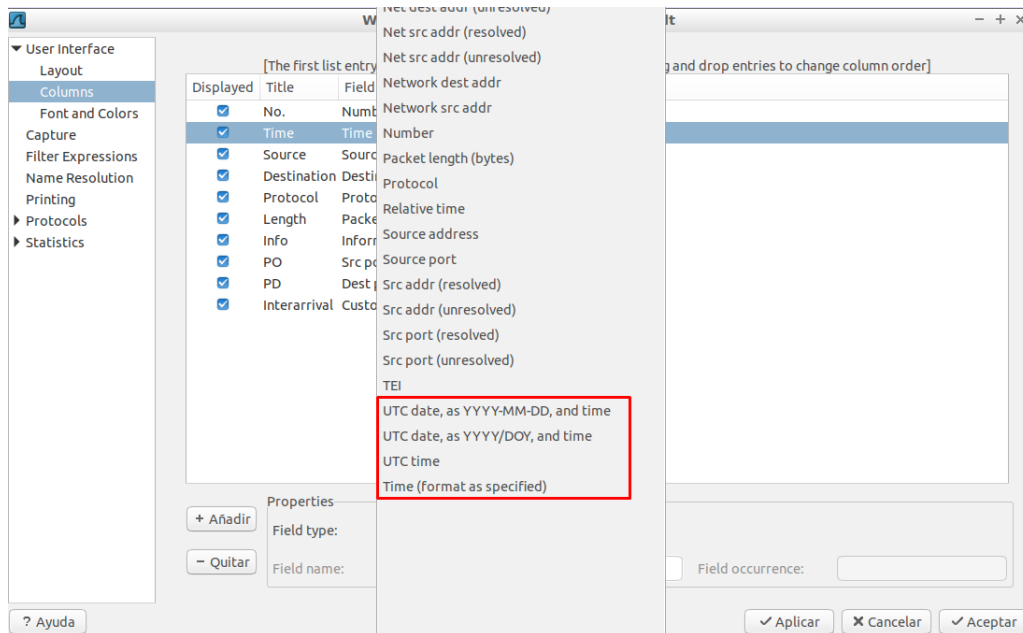
Apreciamos la nueva columna (Primer paquete tiene delta\_time=0 dado que anteriormente no se captura ninguno):

No.	Time	Source	Destination	Protocol	Length	Info	PO	PO	Interarrival
1	0.000000000	192.168.174.134	13.224.119.228	TLSv1.2	85	Encrypted Alert	57868	443	0.000000000
2	0.000133599	192.168.174.134	13.224.119.228	TCP	54	57868 → 443 [FIN, ACK] Seq=32 Ack=1 Win=62780 Len=0	57868	443	0.000133599
3	0.000297365	13.224.119.228	192.168.174.134	TCP	60	443 → 57868 [ACK] Seq=1 Ack=32 Win=64240 Len=0	443	57868	0.000163766
4	0.000379361	13.224.119.228	192.168.174.134	TCP	60	443 → 57868 [ACK] Seq=1 Ack=33 Win=64239 Len=0	443	57868	0.000001996
5	0.000575855	192.168.174.134	13.224.119.228	TLSv1.2	85	Encrypted Alert	57866	443	0.000496494
6	0.000655584	192.168.174.134	13.224.119.228	TCP	54	57866 → 443 [FIN, ACK] Seq=32 Ack=1 Win=62780 Len=0	57866	443	0.000099648
7	0.001042021	13.224.119.228	192.168.174.134	TCP	60	443 → 57866 [ACK] Seq=1 Ack=32 Win=64240 Len=0	443	57866	0.000076517
8	0.001358666	13.224.119.228	192.168.174.134	TCP	60	443 → 57866 [ACK] Seq=1 Ack=33 Win=64239 Len=0	443	57866	0.000398845
9	0.000354593	13.224.119.228	192.168.174.134	TCP	60	443 → 57866 [FIN, PSH, ACK] Seq=1 Ack=33 Win=64239 Len=0	443	57866	0.000603727
10	0.000395219	192.168.174.134	13.224.119.228	TCP	54	57866 → 443 [ACK] Seq=33 Ack=2 Win=62780 Len=0	57866	443	0.000040626
11	0.000440356	13.224.119.228	192.168.174.134	TCP	60	443 → 57866 [FIN, PSH, ACK] Seq=1 Ack=33 Win=64239 Len=0	443	57866	0.000045137
12	0.000454041	192.168.174.134	13.224.119.228	TCP	54	57868 → 443 [ACK] Seq=33 Ack=2 Win=62780 Len=0	57868	443	0.000013605
13	1.000756397	192.168.174.134	93.184.220.29	TCP	54	39076 → 80 [FIN, ACK] Seq=1 Ack=1 Win=64240 Len=0	39076	80	0.994302356
14	1.001038587	93.184.220.29	192.168.174.134	TCP	60	80 → 39076 [ACK] Seq=1 Ack=2 Win=64239 Len=0	80	39076	0.000202190
15	1.000229498	93.184.220.29	192.168.174.134	TCP	60	80 → 39076 [FIN, PSH, ACK] Seq=1 Ack=2 Win=64239 Len=0	80	39076	0.007190911
16	1.000272477	192.168.174.134	93.184.220.29	TCP	54	39076 → 80 [ACK] Seq=2 Ack=2 Win=64240 Len=0	39076	80	0.000042979
17	2.700972482	192.168.174.134	151.101.133.140	TCP	74	39824 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 TSval=558698092 TSecr=0 WS=128	39824	443	1.692706065
18	2.707132780	151.101.133.140	192.168.174.134	TCP	60	443 → 39824 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460	443	39824	0.006160306
19	2.707216334	192.168.174.134	151.101.133.140	TCP	54	39824 → 443 [ACK] Seq=1 Ack=1 Win=64240 Len=0	39824	443	0.000003546
20	2.710389499	192.168.174.134	151.101.133.140	TLSv1.2	571	Client Hello	39824	443	0.003173165
21	2.710883203	151.101.133.140	192.168.174.134	TCP	60	443 → 39824 [ACK] Seq=1 Ack=518 Win=64240 Len=0	443	39824	0.000493704
22	2.710747517	192.168.174.134	151.101.133.140	TCP	54	39824 → 443 [ACK] Seq=518 Ack=3943 Win=61320 Len=0	39824	443	0.007864314

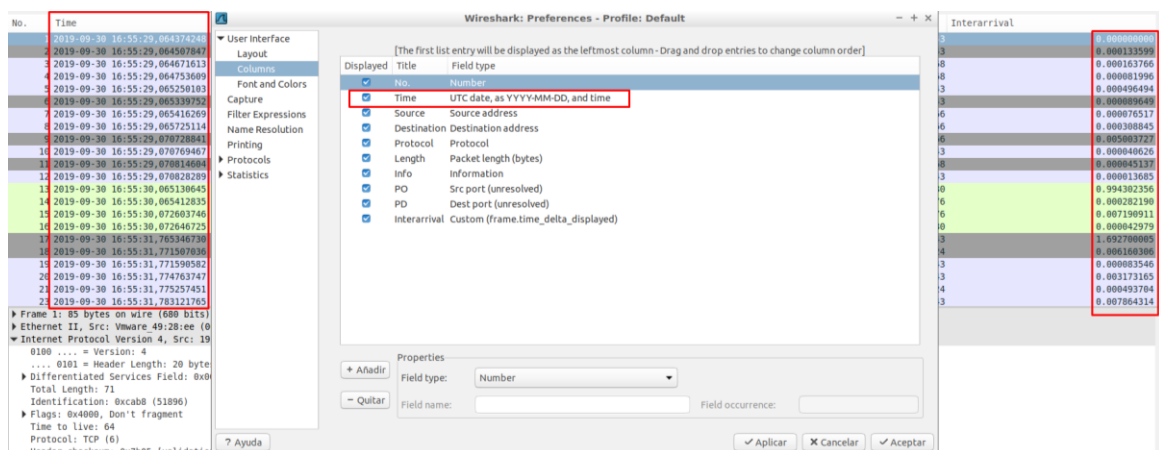
Frame 1: 85 bytes on wire (680 bits), 85 bytes captured (680 bits) on interface 0  
Ethernet II, Src: Vmware 49:28:ee (00:0c:29:49:28:ee), Dst: Vmware e8:6d:d7 (00:50:56:e8:6d:d7)  
Internet Protocol Version 4, Src: 192.168.174.134, Dst: 13.224.119.228  
0180 ... = Version: 4  
... 0181 = Header Length: 20 bytes (5)  
Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)  
Total Length: 71  
Identification: 0xcab8 (51896)  
Flags: 0x4000, Don't Fragment  
Time to live: 64  
Protocol: TCP (6)  
Header checksum: 0x7b05 [validation disabled]  
0000 00 50 56 e8 6d d7 00 0c 29 49 28 ee 00 00 45 00 .PV.m...I{(...E.  
0010 00 47 ca b8 40 00 40 06 7b 05 c0 a8 ae 8d e0 .G..@.B. {.....  
0020 77 e4 e2 0c 01 b0 c2 28 ea 6a 36 0a 8b 47 50 18 W.....(.J6.GP.  
0030 f5 3c f5 2c 00 00 15 03 03 0a 1a 00 00 00 00 .C.....  
.....  
File: /tmp/wireshark\_ens3\_2... Packets: 13661 - Displayed: 5629 (41.2%) - Dropped: 0 (0.0%) Profile: Default

## Ejercicio 4

Entramos en las opciones de columna y modificamos la columna Time. Tendremos las siguientes opciones:



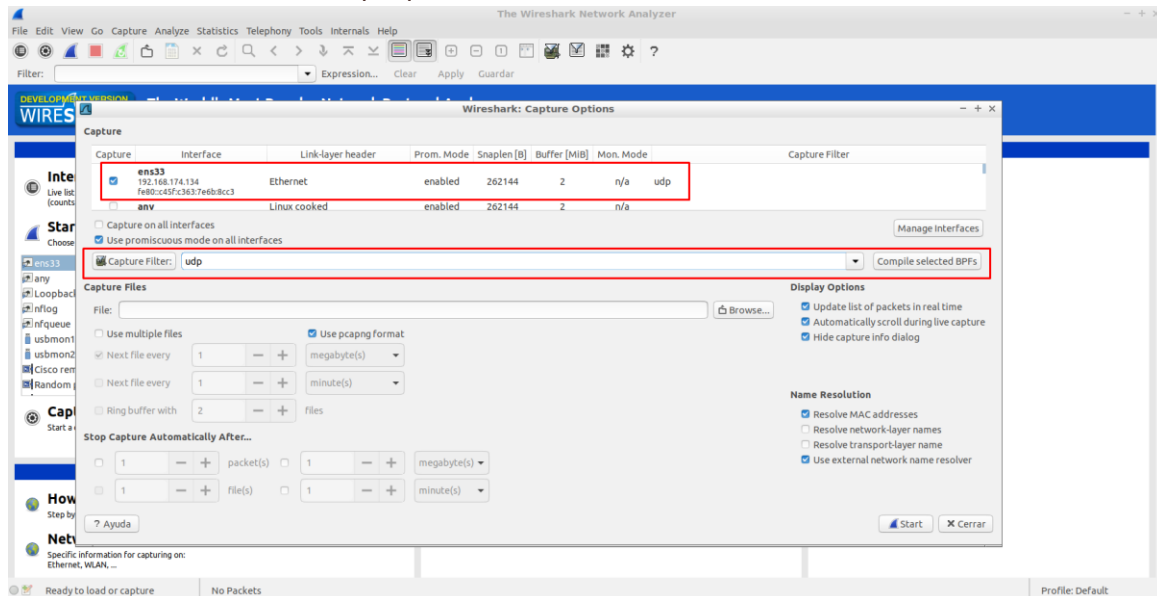
De estas, seleccionamos la mostrada (Se puede apreciar la corrección al corresponder el tiempo entre paquetes a la columna Interarrival que introdujimos en el ejercicio 3):



# Ejercicio 5

Para realizar este ejercicio, comenzamos preparando Wireshark para capturar paquetes en la interfaz “ens33”, una terminal para ejecutar “sudo hping3 -S -p 80 [www.uam.es](http://www.uam.es)”, y el navegador de internet listo para abrir “[www.reddit.com](http://www.reddit.com)”.

En Wireshark configuramos la captura de paquetes para que únicamente sean los paquetes UDP:



Procedemos entonces a iniciar el ping, abrir Reddit y capturar el tráfico generado por estas acciones, comprobando que una vez acaba la captura, al introducir un filtro que muestre únicamente paquetes UDP, se muestran todos los paquetes capturados sin excepción:

