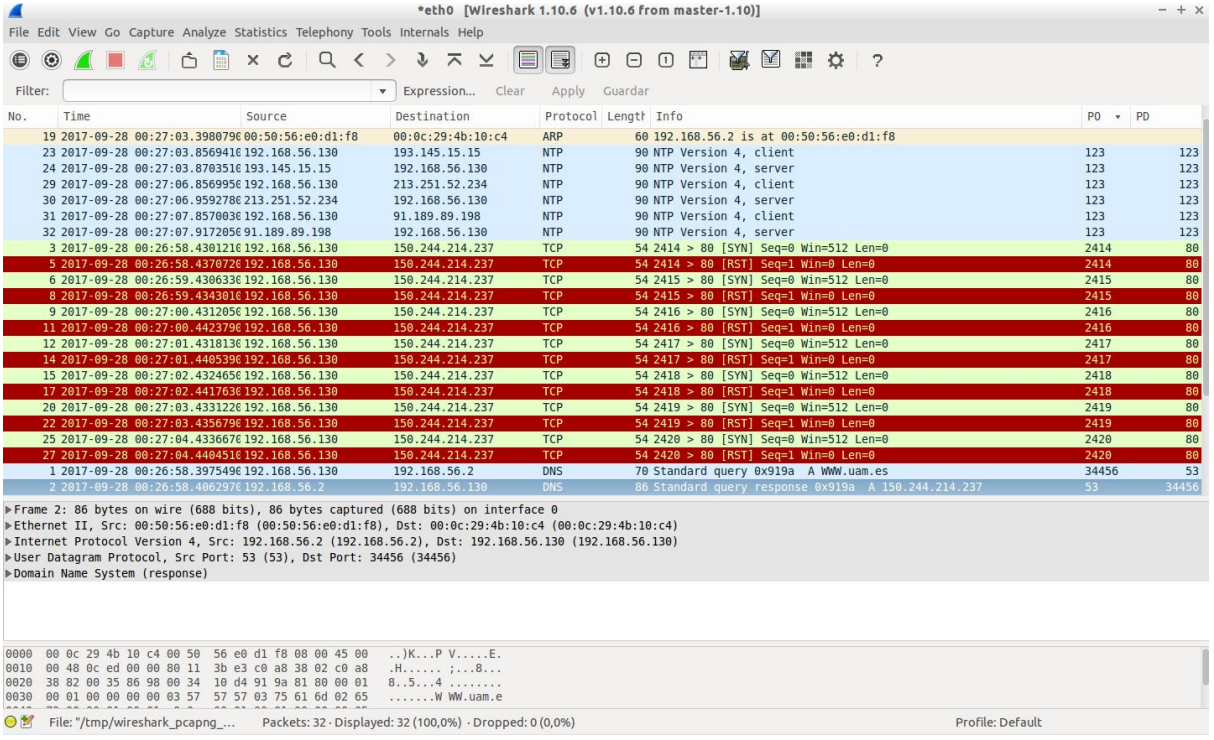
Practica 1:

Parte 0:

- Ejercicio 1:

Para añadir las columnas PO y PD seguimos las instrucciones dadas en la práctica: Edit→Preferences→Columns y en el menú seleccionamos añadir columna tomando los campos deseados. Al organizar los paquetes y ver su valor en el campo 'PO' solo encontramos un paquete con el valor 53 (seleccionado en la imagen inferior).

No se encontró inconveniente en la realización de este ejercicio. El resultado del ejercicio lo encontramos en la siguiente captura de pantalla.



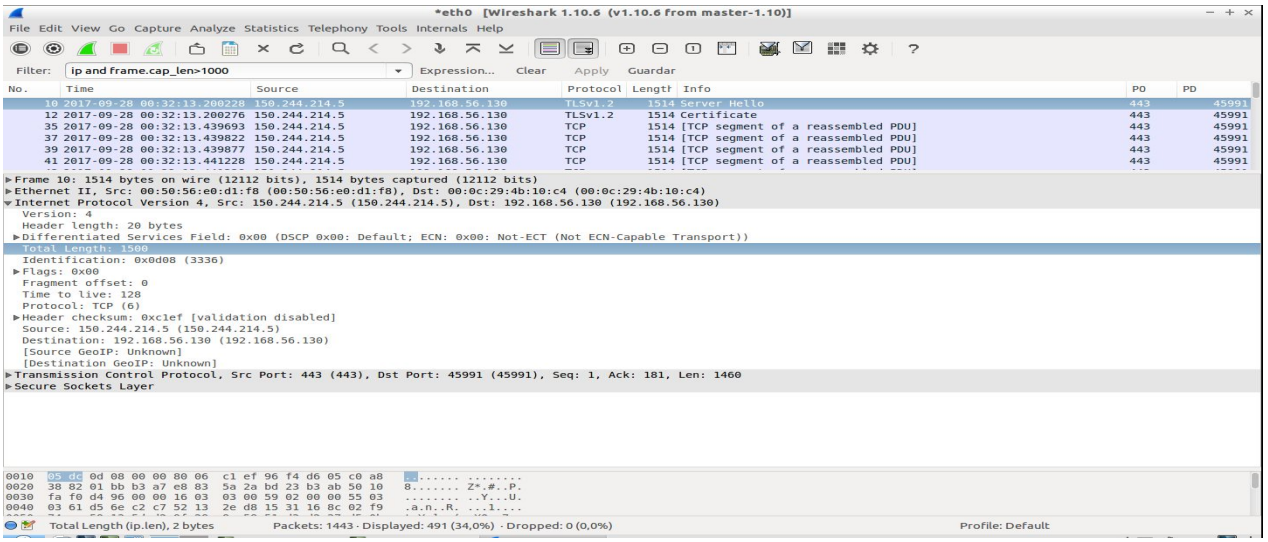
- Ejercicio 2:

Una vez terminada la captura, introducimos el filtro "ip and frame.cap_len > 1000" para obtener los paquetes solicitados por el enunciado.

Para guardar el filtro, basta con clicar en la esquina superior izquierda "File" y luego "Export specified packets".

Al revisar las longitudes de los cinco primeros paquetes (después del filtrado) todos eran de 1514 bytes, como se puede observar en la imagen inferior. Además, su longitud de protocolo IP era solo de 1500 bytes (fila seleccionada en la imagen

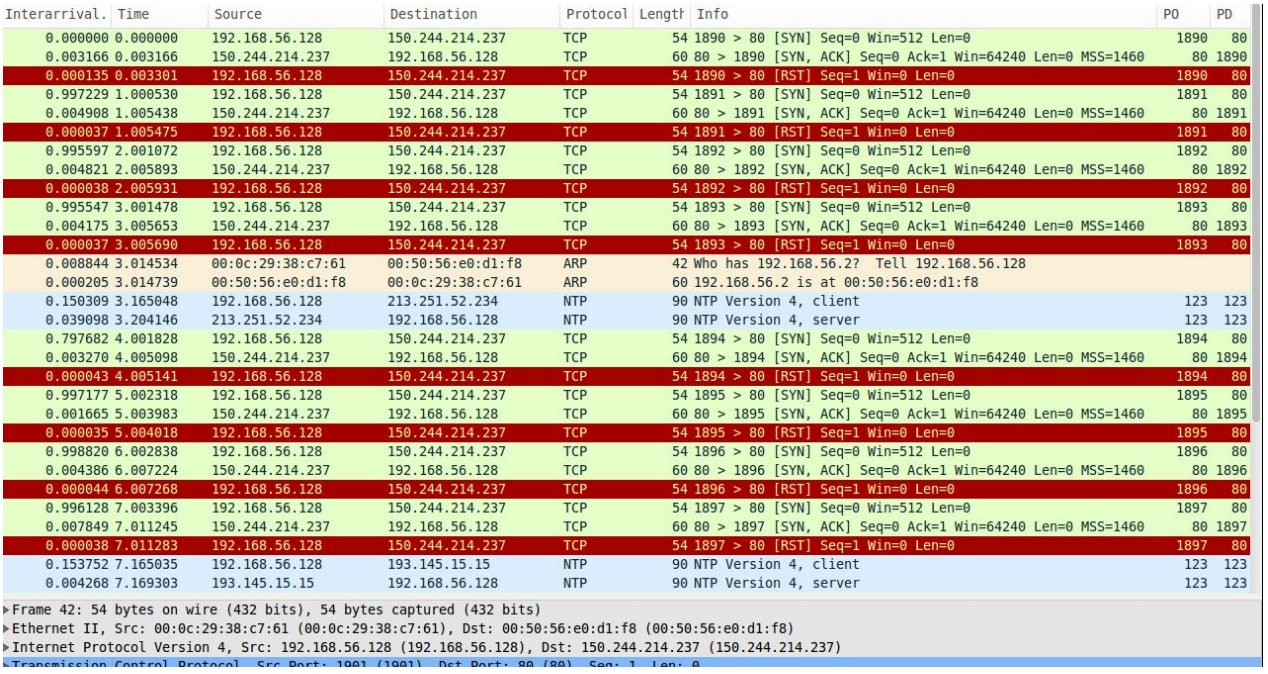
inferior), por ello inferimos que los otros 14 bytes corresponden a la cabecera del protocolo IP en la trama.



- Ejercicio 3:

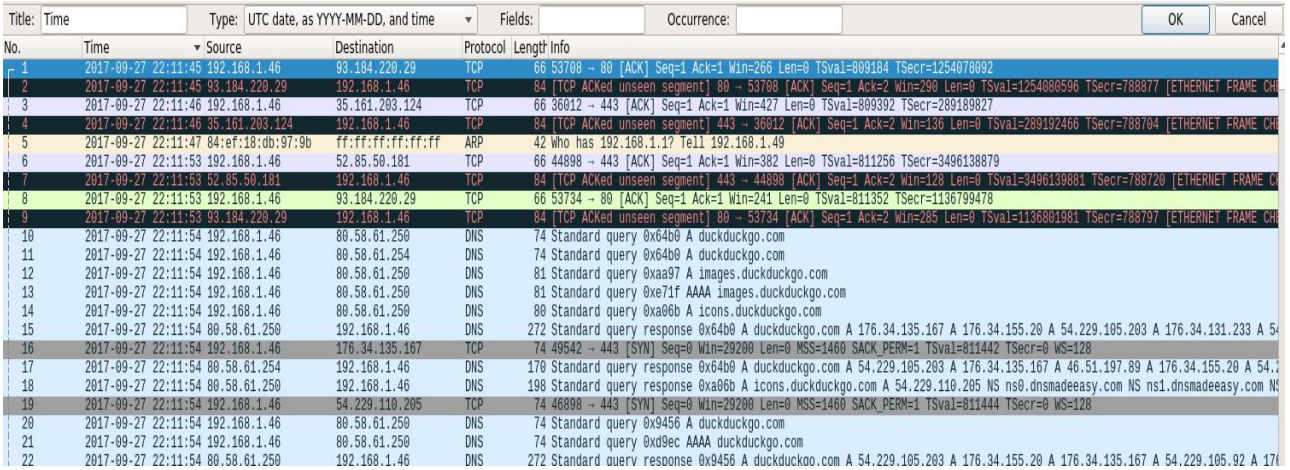
Para añadir la columna con información referente al tiempo de llegada entre paquetes utilizamos el mismo menú que en el ejercicio 1 y seleccionamos ‘Delta Time’ que muestra la información requerida.

En la siguiente captura de pantalla observamos la columna de “Interarrival” añadida al principio.



- Ejercicio 4:

Para el primer cambio, hemos clicado en la columna “Time”, hemos pulsado “Edit column”, y nos ha aparecido sobre las columnas un menú como el siguiente, donde en “Type”, eligiendo “UTC date” y clicando OK, obtenemos el tiempo en formato “humano”, como observamos a continuación (segunda columna):



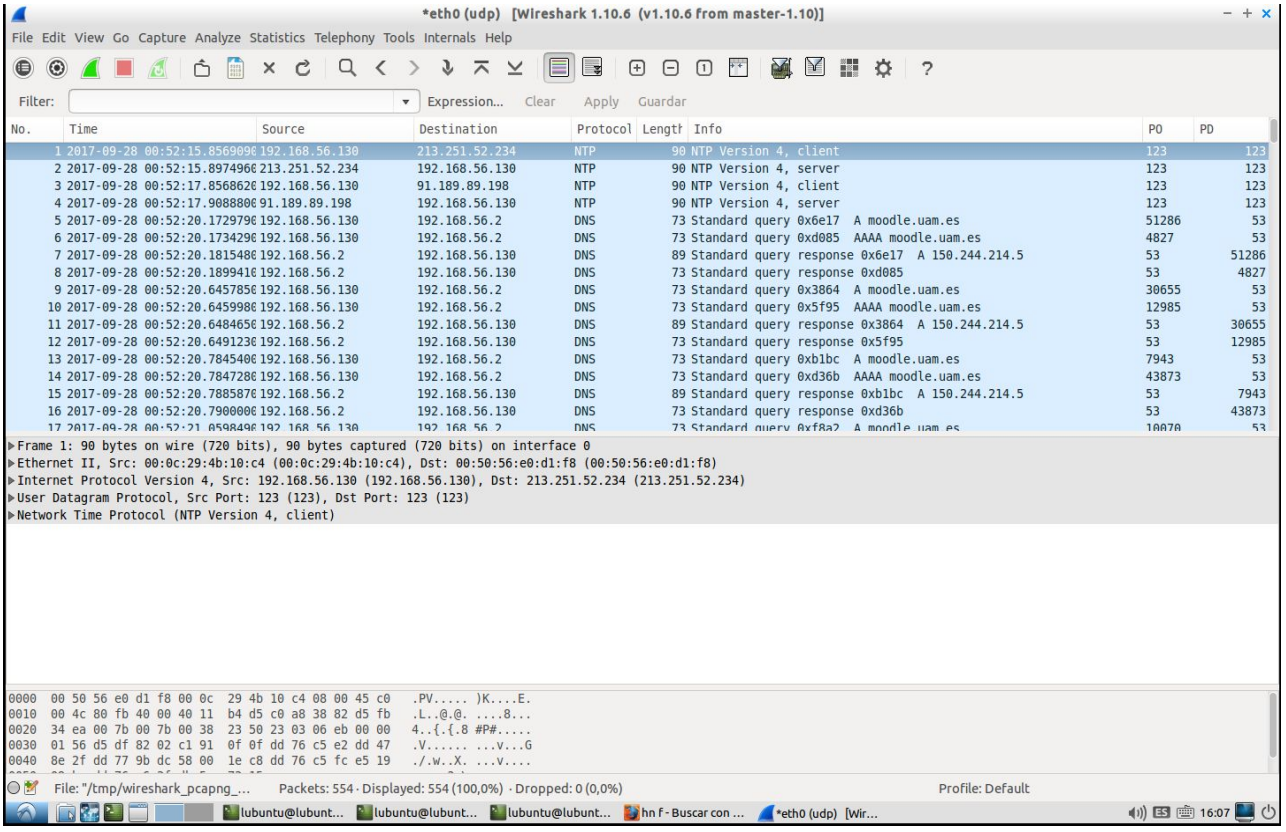
No.	Time	Source	Destination	Protocol	Length	Info
1	2017-09-27 22:11:45	192.168.1.46	93.184.220.29	TCP	66	53708 → 80 [ACK] Seq=1 Ack=1 Win=266 Len=0 TSval=809184 TSecr=1254078092
2	2017-09-27 22:11:45	93.184.220.29	192.168.1.46	TCP	84	[TCP ACKed unseen segment] 80 → 53708 [ACK] Seq=1 Ack=2 Win=290 Len=0 TSval=1254080596 TSecr=788877 [ETHERNET FRAME CH
3	2017-09-27 22:11:46	192.168.1.46	35.161.203.124	TCP	66	36012 → 443 [ACK] Seq=1 Ack=1 Win=427 Len=0 TSval=809392 TSecr=289189827
4	2017-09-27 22:11:46	35.161.203.124	192.168.1.46	TCP	84	[TCP ACKed unseen segment] 443 → 36012 [ACK] Seq=1 Ack=2 Win=136 Len=0 TSval=289192466 TSecr=788704 [ETHERNET FRAME CH
5	2017-09-27 22:11:47	84:ef:18:db:97:9b	ff:ff:ff:ff:ff:ff	ARP	42	Who has 192.168.1.1? Tell 192.168.1.49
6	2017-09-27 22:11:53	192.168.1.46	52.85.50.181	TCP	66	44898 → 443 [ACK] Seq=1 Ack=1 Win=382 Len=0 TSval=811256 TSecr=3496138879
7	2017-09-27 22:11:53	52.85.50.181	192.168.1.46	TCP	84	[TCP ACKed unseen segment] 443 → 44898 [ACK] Seq=1 Ack=2 Win=128 Len=0 TSval=3496139881 TSecr=788720 [ETHERNET FRAME C
8	2017-09-27 22:11:53	192.168.1.46	93.184.220.29	TCP	66	53734 → 80 [ACK] Seq=1 Ack=1 Win=241 Len=0 TSval=811352 TSecr=1136799478
9	2017-09-27 22:11:53	93.184.220.29	192.168.1.46	TCP	84	[TCP ACKed unseen segment] 80 → 53734 [ACK] Seq=1 Ack=2 Win=285 Len=0 TSval=1136801981 TSecr=788797 [ETHERNET FRAME CH
10	2017-09-27 22:11:54	192.168.1.46	80.58.61.250	DNS	74	Standard query 0x64b0 A duckduckgo.com
11	2017-09-27 22:11:54	192.168.1.46	80.58.61.254	DNS	74	Standard query 0x64b0 A duckduckgo.com
12	2017-09-27 22:11:54	192.168.1.46	80.58.61.250	DNS	81	Standard query 0xaa97 A images.duckduckgo.com
13	2017-09-27 22:11:54	192.168.1.46	80.58.61.250	DNS	81	Standard query 0xe71f AAAA images.duckduckgo.com
14	2017-09-27 22:11:54	192.168.1.46	80.58.61.250	DNS	80	Standard query 0xa06b A icons.duckduckgo.com
15	2017-09-27 22:11:54	80.58.61.250	192.168.1.46	DNS	272	Standard query response 0x64b0 A duckduckgo.com A 176.34.135.167 A 176.34.155.20 A 54.229.105.203 A 176.34.131.233 A 5
16	2017-09-27 22:11:54	192.168.1.46	176.34.135.167	TCP	74	49542 → 443 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=811442 TSecr=0 WS=128
17	2017-09-27 22:11:54	80.58.61.254	192.168.1.46	DNS	170	Standard query response 0x64b0 A duckduckgo.com A 54.229.105.203 A 176.34.135.167 A 46.51.197.89 A 176.34.155.20 A 54.
18	2017-09-27 22:11:54	80.58.61.250	192.168.1.46	DNS	198	Standard query response 0xa06b A icons.duckduckgo.com A 54.229.110.205 NS ns0.dnsmadeeasy.com NS ns1.dnsmadeeasy.com N
19	2017-09-27 22:11:54	192.168.1.46	54.229.110.205	TCP	74	46898 → 443 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=811444 TSecr=0 WS=128
20	2017-09-27 22:11:54	192.168.1.46	80.58.61.250	DNS	74	Standard query 0x9456 A duckduckgo.com
21	2017-09-27 22:11:54	192.168.1.46	80.58.61.250	DNS	74	Standard query 0xd9ec AAAA duckduckgo.com
22	2017-09-27 22:11:54	80.58.61.250	192.168.1.46	DNS	272	Standard query response 0x9456 A duckduckgo.com A 54.229.105.203 A 176.34.155.20 A 176.34.135.167 A 54.229.105.92 A 17

Para poner el tiempo en formato Unix hemos tenido que clickar en “View” en el menú superior, luego en “Time Display Format”, y seleccionar “Seconds since 1970-01-01” y obtenemos un escenario como el siguiente (segunda columna):

No.	Time	Source	Destination	Protocol	Length	Info
1	1506550305	192.168.1.46	93.184.220.29	TCP	66	53708 → 80 [ACK] Seq=1 Ack=1 Win=266 Len=0 TSval=809184 TSecr=1254078092
2	1506550305	93.184.220.29	192.168.1.46	TCP	84	[TCP ACKed unseen segment] 80 → 53708 [ACK] Seq=1 Ack=2 Win=290 Len=0 TSval=1254080596 TSecr=788877 [ETHERNET FRAME CH
3	1506550306	192.168.1.46	35.161.203.124	TCP	66	36012 → 443 [ACK] Seq=1 Ack=1 Win=427 Len=0 TSval=809392 TSecr=289189827
4	1506550306	35.161.203.124	192.168.1.46	TCP	84	[TCP ACKed unseen segment] 443 → 36012 [ACK] Seq=1 Ack=2 Win=136 Len=0 TSval=289192466 TSecr=788704 [ETHERNET FRAME CH
5	1506550307	84:ef:18:db:97:9b	ff:ff:ff:ff:ff:ff	ARP	42	Who has 192.168.1.1? Tell 192.168.1.49
6	1506550313	192.168.1.46	52.85.50.181	TCP	66	44898 → 443 [ACK] Seq=1 Ack=1 Win=382 Len=0 TSval=811256 TSecr=3496138879
7	1506550313	52.85.50.181	192.168.1.46	TCP	84	[TCP ACKed unseen segment] 443 → 44898 [ACK] Seq=1 Ack=2 Win=128 Len=0 TSval=3496139881 TSecr=788720 [ETHERNET FRAME C
8	1506550313	192.168.1.46	93.184.220.29	TCP	66	53734 → 80 [ACK] Seq=1 Ack=1 Win=241 Len=0 TSval=811352 TSecr=1136799478
9	1506550313	93.184.220.29	192.168.1.46	TCP	84	[TCP ACKed unseen segment] 80 → 53734 [ACK] Seq=1 Ack=2 Win=285 Len=0 TSval=1136801981 TSecr=788797 [ETHERNET FRAME CH
10	1506550314	192.168.1.46	80.58.61.250	DNS	74	Standard query 0x64b0 A duckduckgo.com
11	1506550314	192.168.1.46	80.58.61.254	DNS	74	Standard query 0x64b0 A duckduckgo.com
12	1506550314	192.168.1.46	80.58.61.250	DNS	81	Standard query 0xaa97 A images.duckduckgo.com
13	1506550314	192.168.1.46	80.58.61.250	DNS	81	Standard query 0xe71f AAAA images.duckduckgo.com
14	1506550314	192.168.1.46	80.58.61.250	DNS	80	Standard query 0xa06b A icons.duckduckgo.com
15	1506550314	80.58.61.250	192.168.1.46	DNS	272	Standard query response 0x64b0 A duckduckgo.com A 176.34.135.167 A 176.34.155.20 A 54.229.105.203 A 176.34.131.233 A 5
16	1506550314	192.168.1.46	176.34.135.167	TCP	74	49542 → 443 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=811442 TSecr=0 WS=128
17	1506550314	80.58.61.254	192.168.1.46	DNS	170	Standard query response 0x64b0 A duckduckgo.com A 54.229.105.203 A 176.34.135.167 A 46.51.197.89 A 176.34.155.20 A 54.
18	1506550314	80.58.61.250	192.168.1.46	DNS	198	Standard query response 0xa06b A icons.duckduckgo.com A 54.229.110.205 NS ns0.dnsmadeeasy.com NS ns1.dnsmadeeasy.com N
19	1506550314	192.168.1.46	54.229.110.205	TCP	74	46898 → 443 [SYN] Seq=0 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=811444 TSecr=0 WS=128
20	1506550314	192.168.1.46	80.58.61.250	DNS	74	Standard query 0x9456 A duckduckgo.com
21	1506550314	192.168.1.46	80.58.61.250	DNS	74	Standard query 0xd9ec AAAA duckduckgo.com
22	1506550314	80.58.61.250	192.168.1.46	DNS	272	Standard query response 0x9456 A duckduckgo.com A 54.229.105.203 A 176.34.155.20 A 176.34.135.167 A 54.229.105.92 A 17
23	1506550314	80.58.61.250	192.168.1.46	DNS	130	Standard query response 0xd9ec AAAA duckduckgo.com SOA ns0.dnsmadeeasy.com
24	1506550314	80.58.61.250	192.168.1.46	DNS	219	Standard query response 0xaa97 A images.duckduckgo.com CNAME icons.duckduckgo.com A 54.229.110.205 NS ns4.dnsmadeeasy.
25	1506550314	80.58.61.250	192.168.1.46	DNS	157	Standard query response 0xe71f AAAA images.duckduckgo.com CNAME icons.duckduckgo.com SOA ns0.dnsmadeeasy.com
26	1506550314	54.229.110.205	192.168.1.46	TCP	74	443 → 46898 [SYN, ACK] Seq=0 Ack=1 Win=26847 Len=0 MSS=1452 SACK_PERM=1 TSval=1189312994 TSecr=811444 WS=128
27	1506550314	192.168.1.46	54.229.110.205	TCP	66	46898 → 443 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TSval=811453 TSecr=1189312994
28	1506550314	192.168.1.46	54.229.110.205	TLSv1.2	583	Client Hello

- Ejercicio 5:

La captura la iniciamos con la configuración Capture Filter a “udp”, dejando pasar solo paquetes del protocolo “User Data Protocol”. Con esta configuración no se capturaban los paquetes generados por el comando hping3. A continuación vemos el resultado:



Parte 1:

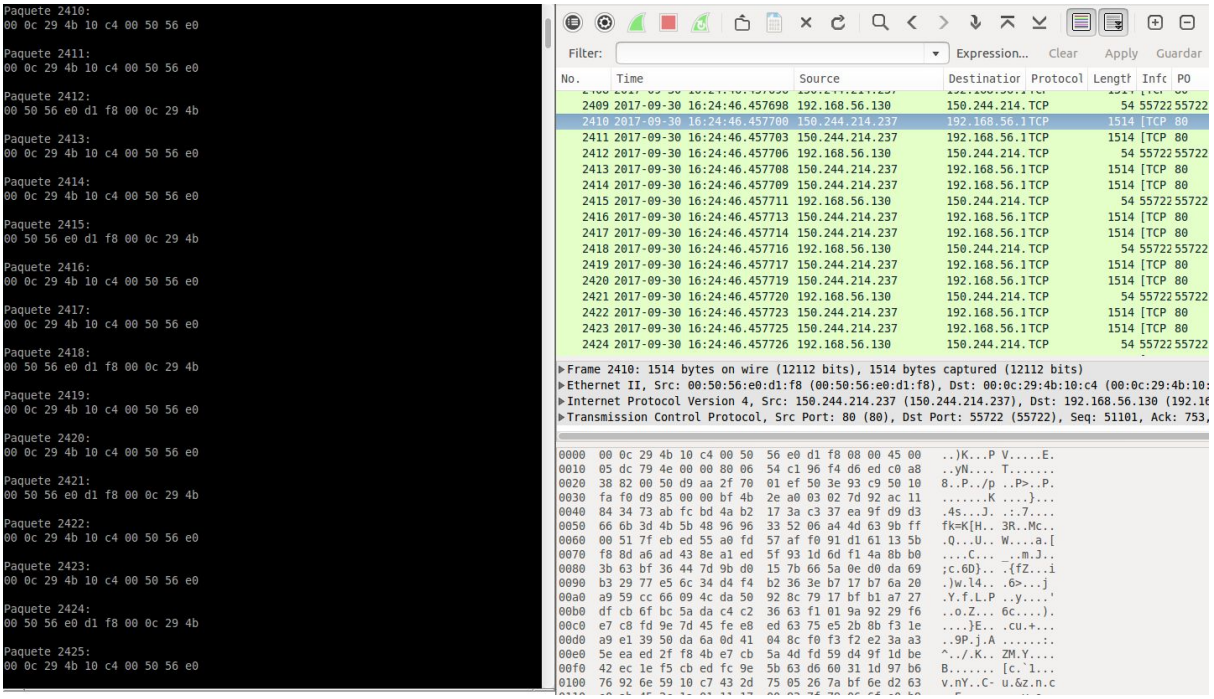
El programa de la práctica fue probado tanto con Valgrind como comprobando las salidas con Wireshark.

A continuación adjuntamos capturas de pantalla en las que mostramos el correcto funcionamiento del programa implementado.

Redes de comunicación 1
Autores: Lucía Asencio, Rodrigo De Pool

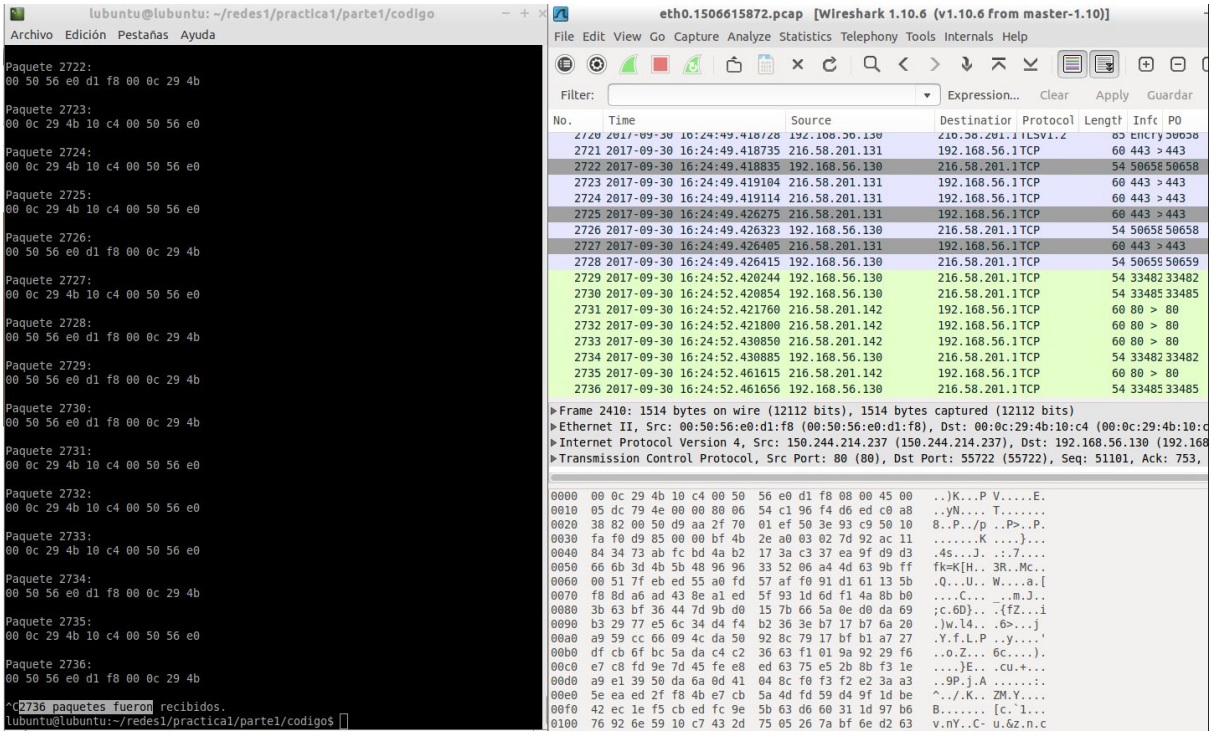
Capturas de ejecución con un argumento:

Solicitamos que se mostraran por pantalla 10 bytes.



Observamos que, en efecto, los 10 primeros bytes del paquete 2410 coinciden en la salida del programa con Wireshark.

En la siguiente captura se puede ver que el número total de paquetes capturados (los que muestra Wireshark) coincide con los paquetes contados por el programa.



Generando un archivo .pcap a traves de la captura de paquetes en wireshark, hemos podido comprobar el funcionamiento de nuestro programa al introducir 2 argumentos (siendo este fichero pcap el segundo).

Efectivamente, el numero de paquetes analizados coincide con el de wireshark, asi como los 10 primeros Bytes de cada uno. En la imagen adjunta se puede apreciar como esto ocurre con el ultimo paquete capturado.