Monitorización de análisis de tráfico

Lucía Asencio Martín y Rodrigo Alonso de Pool Alcántara

Pareja 10, grupo 1302

Introducción

A continuación, nos disponemos a analizar los datos obtenidos en el análisis de tráfico de la práctica 3, a través de los ejercicios propuestos en la misma. Para cada apartado, comenzaremos presentando y explicando los resultados, y después extraeremos conclusiones de los mismos.

Por orden, vamos a presentar 5 aspectos diferentes del análisis:

I. Porcentajes de distintos protocolos
II. Top de direcciones y puertos
III. Distribución del tamaño en paquetes con distintas características
IV. Distribución del *interarrival* para ciertos paquetes
V. Análisis del ancho de banda

Los datos proporcionados por el generador PCAPx64 para nuestro grupo y pareja, fueron: dirección MAC 00:11:88:CC:33:CA, dirección IP 63.161.195.170, puerto UDP 10455.

I. Porcentajes de paquetes

Primero presentaremos los porcentajes de paquetes de acuerdo a los protocolos a los que pertenecen:

```
hey@ubuntu-os:~/Desktop/redes1/practica3/codigo$ cat datos/porcentajes
No IP     0.681278 %
IP        99.3187 %

Entre los paquetes IP tenemos:
UDP       3.66295 %
TCP       61.2493 %
Otros     35.0877 %
```

Análisis a nivel 2:

Observamos que a nivel 2 el protocolo usado es predominantemente IP (versión 4) , esto no nos debe de extrañar ya que la gran mayoría de las comunicaciones en Internet se hacen utilizando dicho protocolo.

Por otro lado, el porcentaje restante podemos someterlo a análisis. Observando el fichero *tipos.tshark* y utilizando awk en la *shell* (el comando será anexado al final de apartado) podemos comprobar que todos los paquetes que no son IP tienen como tipo de protocolo el 0x86dd. Este identificador, podemos comprobar con el estándar IEEE 802 que corresponde con el protocolo IP versión 6. Una versión que pretende sustituir progresivamente a la versión 4 pero que aún representa una clara minoría como podemos apreciar en los porcentajes presentados.

Estándar IEEE 802:

https://www.iana.org/assignments/ieee-802-numbers/ieee-802-numbers.xhtml

Comando utilizado:

awk '{if($1 != "0x0800" && $2 != "0x0800" && $1 != “0x86dd”) print $1;}' tipos.tshark

(ó agregando ceros para el caso de máquinas de 64 bits)

Análisis a nivel 3:

Podemos ver que los paquetes UDP representan una pequeña porción de los paquetes IP, mientras que TCP representa la gran mayoría de estos. Esto nos indica un alto tráfico de paquetes que requieren comunicación sin pérdidas, como puede ser la descarga o subida de archivos.

Por otro lado, haciendo un análisis análogo al realizado en el nivel 2, observamos que el resto de paquetes corresponden a paquetes ICMP (Internet control message protocol). Estos paquetes cargan con información de errores ó de control, normalmente estos paquetes se ocupan de información sobre la red en sí y no sobre la comunicación entre extremos como en el caso de TCP ó UDP. Estos paquetes se pueden generar, por ejemplo, como respuesta de un router a un paquete que no pudo fragmentar, un fallo al intentar localizar una dirección ó respuestas a *pings*.

(http://www.informit.com/articles/article.aspx?p=26557&seqNum=5)

Estudiando el porcentaje de paquetes ICMP en otras trazas que tenían tráfico de usuario (navegación web, servicios de streaming, mensajería, …), observamos porcentajes de paquetes ICMP muy inferiores al dado por la traza estudiada en esta práctica. Esto nos lleva a inferir que este elevado número de paquetes ICMP podría estudiarse en más detalle para saber si corresponde a algún fallo sistemático en la red ó, quizás, a servicios como *pings* que se realizan desde dentro de la red en cuestión.

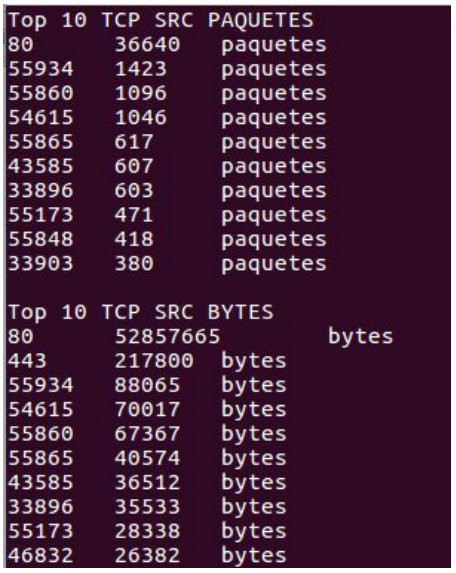
II. Top 10 direcciones y puertos

A continuación proporcionamos un total de 12 tops diferentes, que van a ser presentados en tres apartados, siguiendo este orden: (i) top puertos TCP, (ii) top puertos UDP, (iii) top direcciones IP. Cada apartado se subdividirá en cuatro secciones para distinguir puertos (o direcciones) destino y origen, y si el top viene dado por el número de paquetes o por el número de bytes.

(i) Top puertos TCP

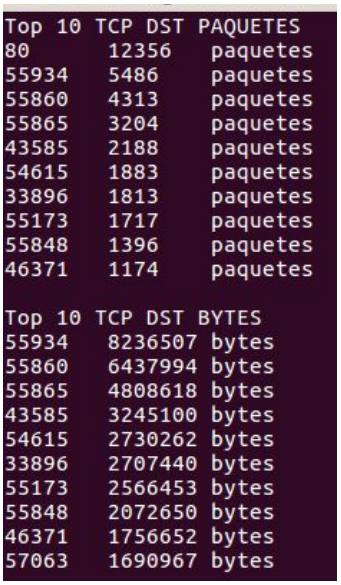
Estos son los resultados obtenidos para el top de puertos TCP de origen.

La primera lista corresponde al top por paquetes, y la segunda lista al top por bytes.



Observemos que en ambos tops coinciden 9 de los 10 puertos, manteniendo el orden. Analizaremos esto en conjunto con los puertos TCP destino.

Este es el top de los puertos TCP destino:



De nuevo, nueve de los diez puertos mostrados coinciden en el top por bytes y en el top por paquetes. En este caso, cabe destacar que el puerto 80, el de HTTP, no está en el top por bytes como puerto de destino. Como ocupa el primer puesto en el top por paquetes, es razonable que al puerto HTTP se envíen muchos paquetes de poca longitud. Analizaremos esto más adelante.

Ahora nos queda ver cómo se comportan entre sí los tops origen contra los tops destino.

En el top por paquetes, los puertos de origen coinciden en un 90% con los de destino, y no sólo eso, sino que además se conserva en casi todo caso la relación de orden que mantienen entre sí.

Sin embargo, en el top por bytes, la coincidencia entre puertos origen y destino es menor: baja a un 70%.

Estas últimas observaciones nos hacen llegar a una conclusión que podemos aplicar al caso HTTP. El hecho de que el top en paquetes se mantenga entre origen y destino, nos dice que la relación entre el número de paquetes enviados y recibidos en cada puerto es más o menos la misma para todos los puertos. Si a esto le sumamos que los puertos top en bytes difieren entre origen y destino, tiene sentido afirmar que a ciertos puertos llegan muchos paquetes pequeños, y salen muchos paquetes muy grandes.

Esta especulación se confirma en el caso del puerto 80. Guiándonos más o menos por la descripción de una transacción HTTP que encontramos aquí blog.catchpoint.com/2010/09/17/anatomyhttp/ , vemos que al puerto 80 llegan muchos *requests* de pequeño tamaño, pero cuando estas peticiones son procesadas, los paquetes que se envíen de vuelta deben tener un mayor tamaño (tiene menos carga solicitar una página web que descargarla). Esto explicaría que el

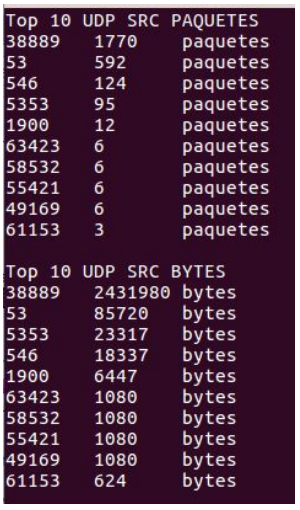
puerto 80 sea el top 1 de todos los casos menos en el top por bytes como puerto destino.

Aparte del puerto 80, sólo encontramos como puerto asignado el 443, que es puerto para conexión segura HTTPS en el top por bytes de puertos origen.

De los demás puertos, aunque había algunos como el 55934 que eran recurrentes en el top, no hemos encontrado que estuvieran reservados para ningún protocolo en particular, por lo que serán puertos que se asignen puntualmente para la comunicación entre terminales.

(ii) Top puertos UDP

De nuevo, vamos a comenzar por el top en puertos UDP de origen:

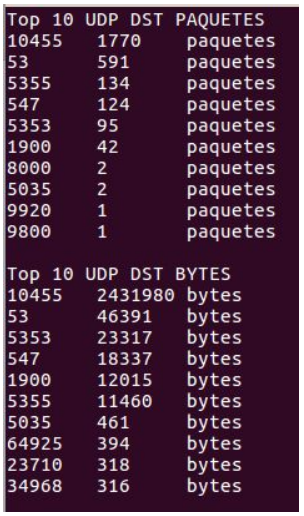


```
Top 10 UDP SRC PAQUETES
38889   1770    paquetes
53      592     paquetes
546     124     paquetes
5353    95      paquetes
1900    12      paquetes
63423   6       paquetes
58532   6       paquetes
55421   6       paquetes
49169   6       paquetes
61153   3       paquetes

Top 10 UDP SRC BYTES
38889   2431980 bytes
53      85720   bytes
5353    23317   bytes
546     18337   bytes
1900    6447    bytes
63423   1080    bytes
58532   1080    bytes
55421   1080    bytes
49169   1080    bytes
61153   624     bytes
```

Aquí, el top el conjunto de los puertos que están en el top por bytes es exactamente el mismo al del top por paquetes. Además, el orden dentro del top se conserva prácticamente intacto. Los puertos en sí los discutimos al final del apartado.

A continuación, el top en puertos UDP de destino:



```
Top 10 UDP DST PAQUETES
10455   1770    paquetes
53      591     paquetes
5355    134     paquetes
547     124     paquetes
5353    95      paquetes
1900    42      paquetes
8000    2       paquetes
5035    2       paquetes
9920    1       paquetes
9800    1       paquetes

Top 10 UDP DST BYTES
10455   2431980 bytes
53      46391   bytes
5353    23317   bytes
547     18337   bytes
1900    12015   bytes
5355    11460   bytes
5035    461     bytes
64925   394     bytes
23710   318     bytes
34968   316     bytes
```

Aunque en este caso el top por bytes y el top por paquetes sólo coinciden en un 70% (de nuevo, esto nos dice que no siempre los puertos a los que llegan muchos paquetes, llegan paquetes grandes), sí que se conserva completamente el top 6 de puertos.

Relacionando el top origen con el top destino, lo que vemos es que sólo se conservan 2 puertos: el 53, que es el puerto para DNS (usado para la resolución de dominios), y el 5353, usado para multicast DNS, que realiza la resolución de nombres en redes pequeñas.

Aparte de estos, nos llamaron la atención los puertos 546 en el top origen y 547 en el top destino. Estos dos puertos resultaron ser usados en el protocolo DHCP(v6), en el caso del 546 para los hosts y en el caso del 547 para servidores. Por lo cual, los paquetes de la traza tenían como origen un host y como destino un servidor.

Otros 2 puertos repetidos en los tops son el 1900 (*Simple Service Discovery Protocol)*, usado para recibir mensajes de difusión entre ciertos terminales (*"used by Universal Plug N' Play devices to receive broadcasted messages from other UPnP devices"*), y el 5355 (*Link-Local Multicast Name Resolution*) usado para que terminales puedan resolver los nombres de otros elementos en la misma red local (es la misma función que el multicast DNS mencionado arriba, pero mDNS es usado por Apple Computer y LLMNR por Microsoft).

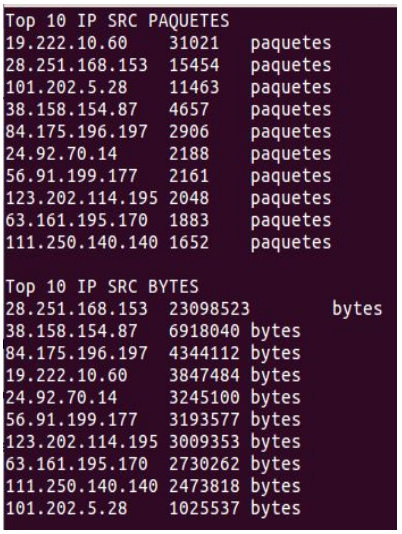
Los demás puertos que buscamos no correspondían a ningún protocolo en especial, así que habrían sido asignados a dos terminales para comunicarse.

Tanto en el top de origen como en el de destino, hemos visto diferentes puertos que son usados para trabajar con diferentes protocolos. Ahora bien, ¿tienen estos protocolos algo que ver entre sí? Hemos encontrado que muchos de estos puertos UDP se relacionan de esta manera: todos son usados dentro de Zeroconf (*Zero Configuration Networking*), con el fin de asociar IPs, nombres y hosts entre sí.
(https://en.wikipedia.org/wiki/Zero-configuration_networking /
https://es.wikipedia.org/wiki/Zeroconf).

En el estudio de los puertos TCP y UDP ha sido de gran ayuda https://www.speedguide.net/port.php

(iii) Top direcciones IP

En el caso del top direcciones IP de origen obtuvimos los siguientes resultados:

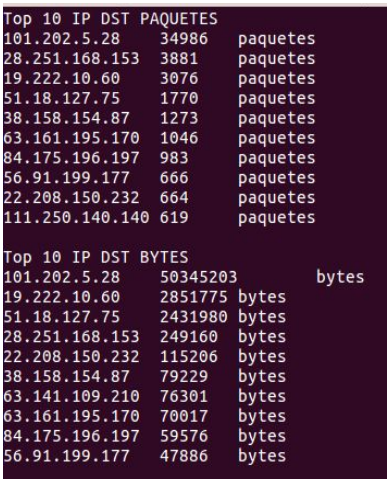


```
Top 10 IP SRC PAQUETES
19.222.10.60     31021   paquetes
28.251.168.153   15454   paquetes
101.202.5.28     11463   paquetes
38.158.154.87    4657    paquetes
84.175.196.197   2906    paquetes
24.92.70.14      2188    paquetes
56.91.199.177    2161    paquetes
123.202.114.195 2048    paquetes
63.161.195.170   1883    paquetes
111.250.140.140 1652    paquetes

Top 10 IP SRC BYTES
28.251.168.153   23098523        bytes
38.158.154.87    6918040 bytes
84.175.196.197   4344112 bytes
19.222.10.60     3847484 bytes
24.92.70.14      3245100 bytes
56.91.199.177    3193577 bytes
123.202.114.195 3009353 bytes
63.161.195.170   2730262 bytes
111.250.140.140 2473818 bytes
101.202.5.28     1025537 bytes
```

En ella, puede verse que el top en direcciones por bytes es exactamente el mismo que el top por paquetes, aunque el orden no se mantenga.

La gráfica del top en direcciones destino es esta:



```
Top 10 IP DST PAQUETES
101.202.5.28     34986   paquetes
28.251.168.153   3881    paquetes
19.222.10.60     3076    paquetes
51.18.127.75     1770    paquetes
38.158.154.87    1273    paquetes
63.161.195.170   1046    paquetes
84.175.196.197   983     paquetes
56.91.199.177    666     paquetes
22.208.150.232   664     paquetes
111.250.140.140 619     paquetes

Top 10 IP DST BYTES
101.202.5.28     50345203        bytes
19.222.10.60     2851775 bytes
51.18.127.75     2431980 bytes
28.251.168.153   249160  bytes
22.208.150.232   115206  bytes
38.158.154.87    79229   bytes
63.141.109.210   76301   bytes
63.161.195.170   70017   bytes
84.175.196.197   59576   bytes
56.91.199.177    47886   bytes
```

En el caso del top en direcciones IP de destino, la coincidencia no es total si no del 90%, y el orden tampoco se mantiene.

Entre las direcciones que son top como destino y las que son top como origen, la coincidencia es del 70%, lo que nos indica que las IP que más paquetes envían no siempre son las que más paquetes reciben. Lo mismo ocurre con el número de bytes, pero esto debería sorprendernos menos por la misma razón que discutíamos en el caso **(i) Top puertos TCP**: hay IPs que pueden estar enviando únicamente pequeños paquetes con solicitudes o asentimientos, y que a estas mismas IPs sólo lleguen grandes paquetes, por ejemplo, con páginas web. Un ejemplo de esto último es la IP 63.161.195.170: aunque el número de paquetes que recibe es similar al número de paquetes que envía, el número de bytes que recibe es 40 veces menor al que envía. Esto nos indica que podría tratarse de la IP de un servidor. Justo lo

contrario es lo que ocurre con la IP 28.251.168.153, donde el tamaño medio de paquete recibido es 23 veces mayor que el tamaño de paquete enviado, y parece ser la IP de un terminal haciendo peticiones a un servidor.

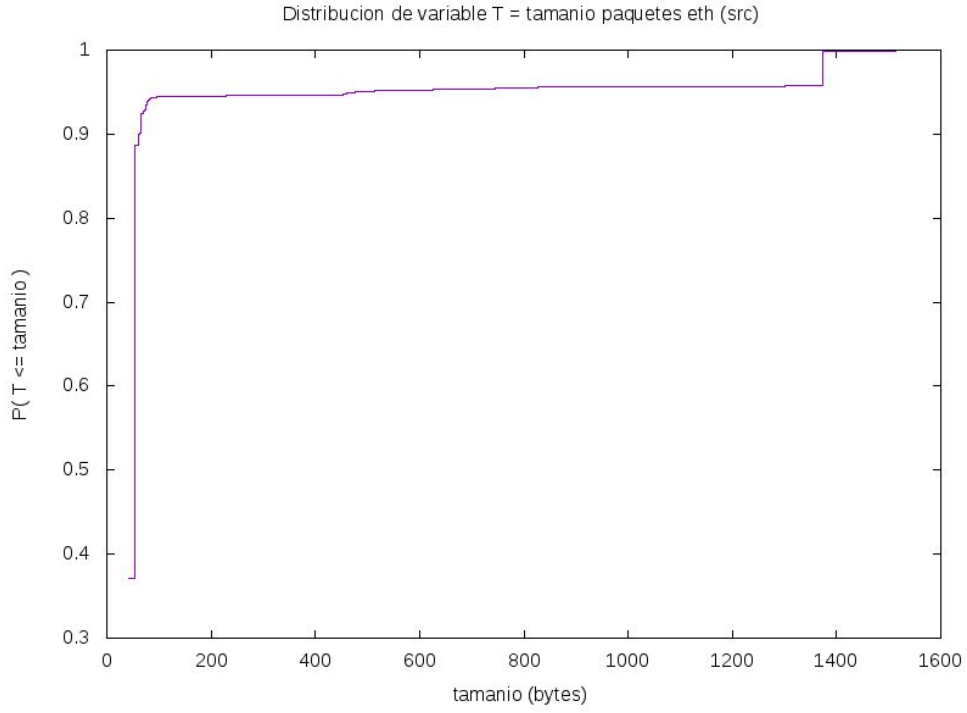
Con respecto a las direcciones, hemos comprobado que ninguna de ellas estaba en los rangos de direcciones privadas, por lo que no tendría por qué estar teniendo lugar un intercambio de paquetes dentro de una misma red local.

III. Distribución del tamaño en paquetes

Nos disponemos a comentar la distribución del tamaño de diferentes paquetes, distinguiendo entre tamaño a nivel 2, donde analizaremos (i) los paquetes desde/hacia la MAC 00:11:88:CC:33:CA, y tamaño a nivel 3, donde analizaremos (ii) los paquetes HTTP y (iii) los paquetes DNS .

(i) Paquetes a nivel 2

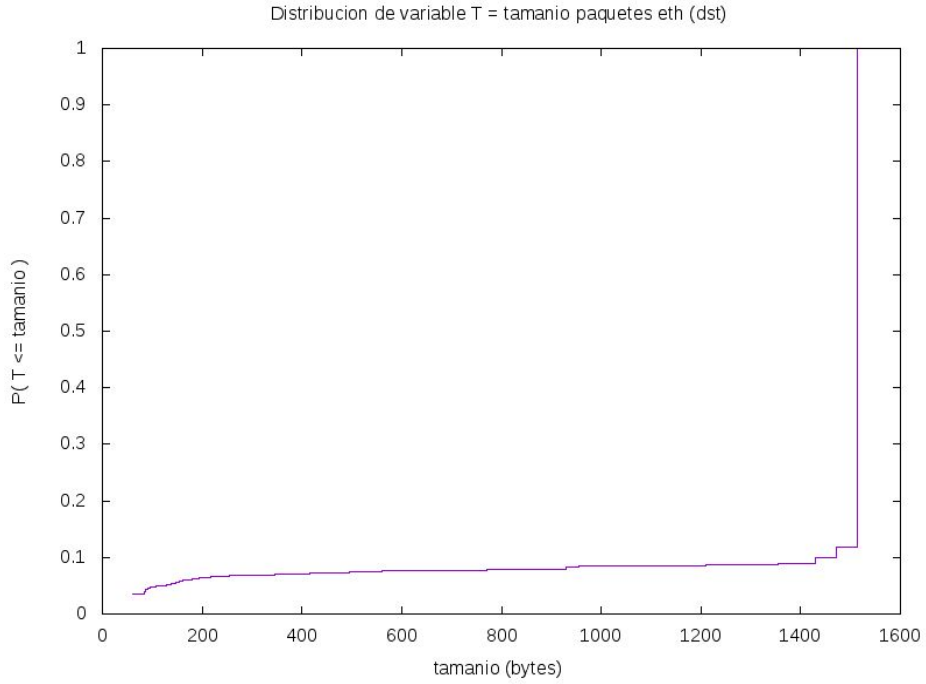
En primer lugar, filtramos todos los paquetes que tienen como **MAC origen** 00:11:88:CC:33:CA, y nos queda una función de distribución con la siguiente gráfica:



Podemos observar que casi todos las tramas enviadas desde la MAC analizada (¡más del 90%!) son muy pequeñas, con menos de 60 bytes. En particular, los ficheros que guardan los datos de la distribución (en el directorio datos) muestran que el 50% de las tramas son de 54 bytes.

El envío de tantos paquetes pequeños nos hace pensar que en estas transmisiones lo importante no son los datos que contiene el paquete a nivel de aplicación, sino el significado del paquete en sí. Además, 54 bytes es el tamaño que ocupan una cabecera ethernet, y una cabecera IP y otra TCP sin opciones. Este mensaje vacío puede estar haciendo la función de ACK en una comunicación TCP.

Ahora, nos centramos en los paquetes que tienen como **MAC destino** 00:11:88:CC:33:CA. Esta es la imagen que obtuvimos:



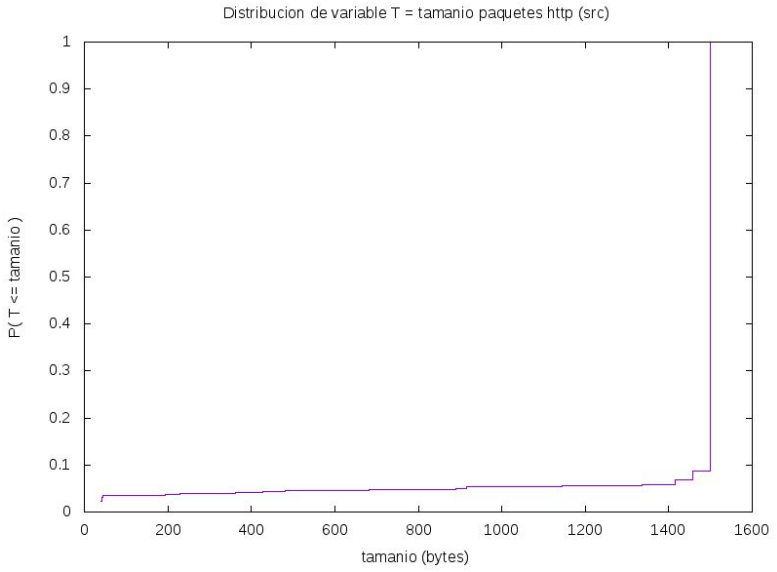
La imagen refleja la recepción de muy pocos paquetes pequeños y de un gran número de tamaño superior a 1400 bytes. De hecho, como queda reflejado en el fichero del directorio datos, ¡el 90% de las tramas son de 1514 bytes!

Lo justificamos asociando estas tramas a descargas de internet: igual que solicitar una conexión, o el acceso auna página, no requiere el envío de mucha información, la descarga de la página en cuestión sí será a través de paquetes que contengan muchos datos. Que la cifra sea 1514 y no, por ejemplo, 64K, nos indica cuál es la MTU (maximum transmission unit). Según hemos leído en https://en.wikipedia.org/wiki/Maximum_transmission_unit , 1500 es la MTU de casi todos los paquetes IP que van sobre Ethernet, y teniendo en cuenta que la cabecera Ethernet ocupa los 14 bytes restantes, ya tenemos los 1514 bytes que forman el paquete completo.

Podemos concluir que los ACKs enviados que se reflejan en la gráfica de antes, son asentimientos a la recepción de los grandes paquetes que se ven en esta gráfica. Esto es corroborado por el hecho de que haya el mismo porcentaje de paquetes pequeños enviados, que grandes paquetes recibidos.

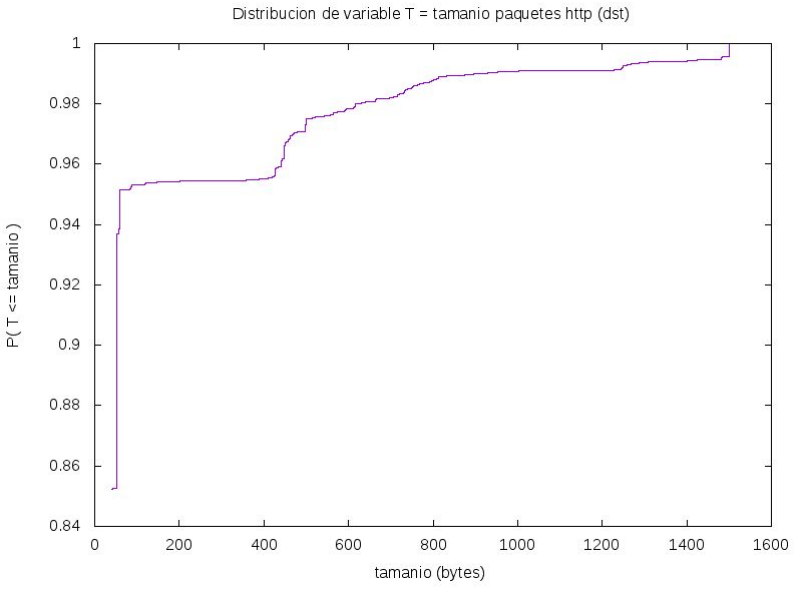
(ii) Paquetes HTTP

A continuación, vamos a analizar los paquetes HTTP (*Hyper text transport protocol)* identificado con el puerto TCP 80. Primero analizamos la gráfica con el puerto como origen:



Dado que estamos filtrando paquetes de origen HTTP, es decir, paquetes que principalmente salen de un servidor como respuesta a solicitudes, tiene sentido que los paquetes que se envíen sean de gran tamaño ya que contienen, por ejemplo, páginas webs. El número más repetido es, según los ficheros de datos, 1500 bytes, lo cual corrobora lo discutido en el apartado anterior, ya que la cifra corresponde al MTU del paquete IP de una trama Ethernet.

A continuación, echamos un vistazo a los paquetes **HTTP** (con destino el puerto 80)



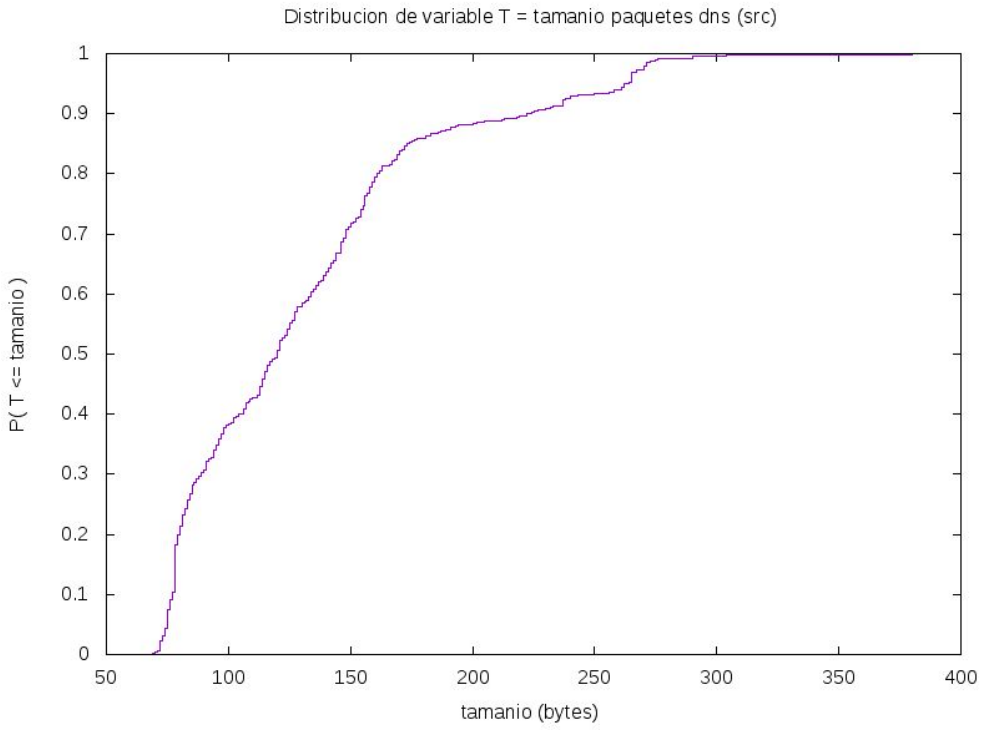
De nuevo, teniendo en cuenta que los paquetes con destino HTTP son probablemente las solicitudes o asentimientos a un servidor, tiene sentido que la mayoría de los paquetes (más del 90%) son pequeños, de menos de 52 bytes (a nivel 3). De hecho, el 80% son paquetes de 40 bytes, que es lo que obtenemos al quitar la cabecera ethernet a los paquetes de 54 bytes mencionados en el apartado (**(i) Paquetes a nivel 2**).

Después de observar las similitud entre las dos gráficas de paquetes HTTP, y las gráficas obtenidas en el apartado anterior (**(i) Paquetes a nivel 2**) podemos intuir que la dirección MAC 00:11:88:CC:33:CA es la dirección Ethernet de un terminal que está accediendo a internet.

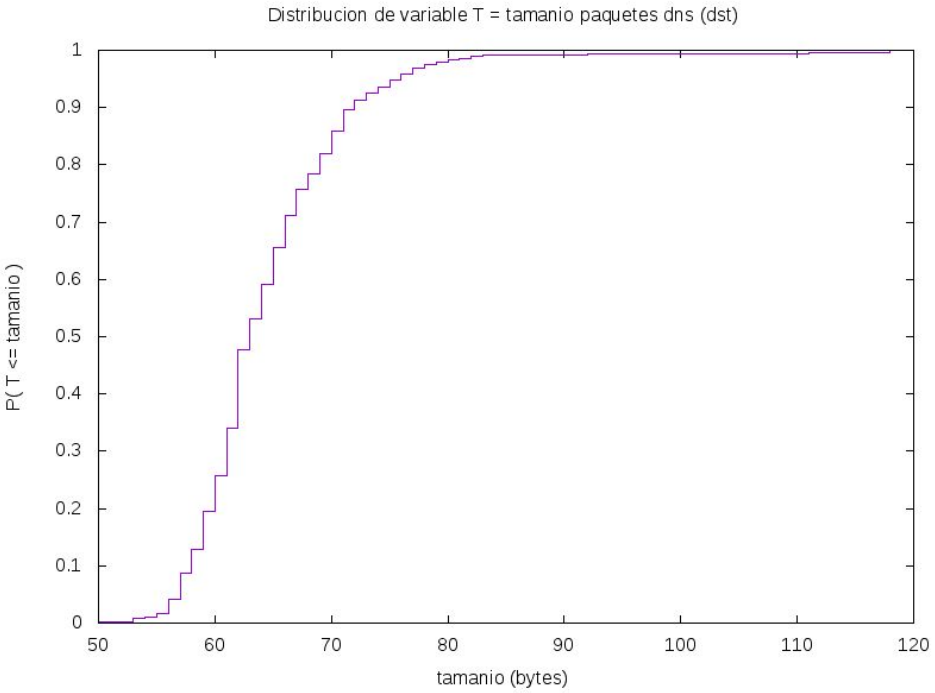
(iii) Paquetes DNS

Por último, en este apartado vamos a discutir las gráficas obtenidas para los tamaños a nivel 3 de los paquetes DNS. Vamos a tratar paralelamente los paquetes con el puerto UDP origen 53 y los paquetes con puerto UDP destino 53, ya que siguen una distribución similar.

Esta es la gráfica fijando el puerto de origen:



Y esta es la gráfica fijando el puerto de destino:



Ambas variables parecen seguir una distribución normal, y esto se debe a la naturaleza de los paquetes DNS. Según hemos leído (https://technet.microsoft.com/en-us/library/cc772774(v=ws.10).aspx), DNS es un protocolo utilizado para resolver dominios, y podemos distinguir paquetes DNS que son *querys* (para solicitar la resolución de un dominio) y *responses* (con la respuesta a la solicitud). Ambos paquetes tienen una estructura parecida, y ambos varían su tamaño en función del dominio a resolver. El hecho de que nuestra distribución sea normal es debido a que el tamaño de los dominios sigue por sí mismo una distribución normal.

Por último, intentamos justificar el hecho de que los paquetes en la gráfica DNS-origen son mayores que los de la gráfica DNS-destino. Con Wireshark, hemos podido comprobar que la mayoría de los paquetes DNS que eran *querys* tenían un tamaño de unos 80 bytes, y que entre los 90 y 400 bytes, casi todos los DNS eran *responses.* Por tanto, deducimos que los paquetes que se filtran en nuestra gráfica DNS-destino son *querys* y los que se contemplan en la gráfica DNS-origen son *responses*.

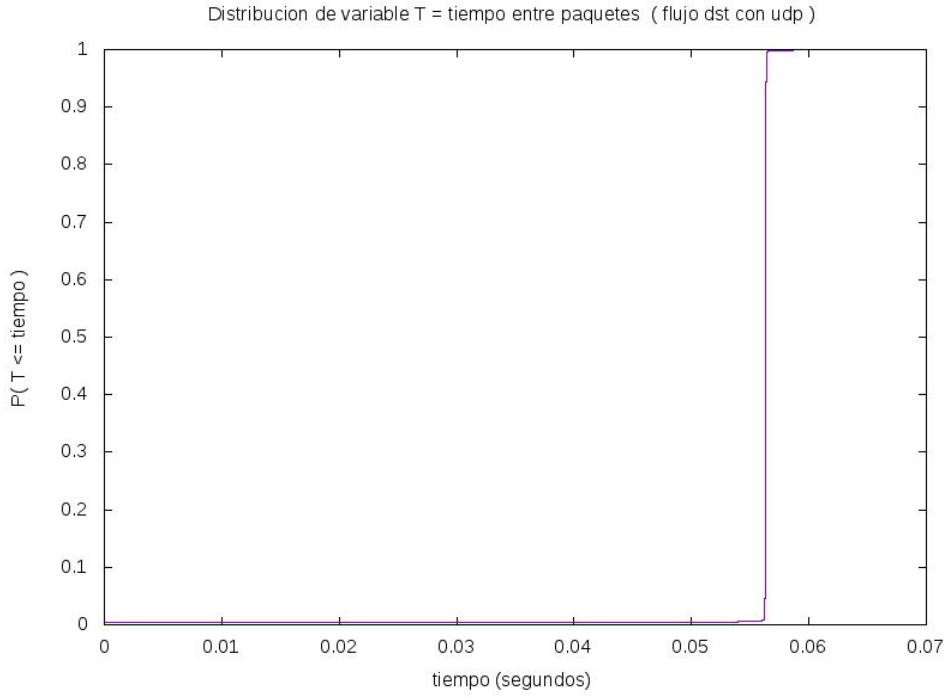
Además de los dos enlaces mencionados, también nos ha sido de utilidad en los tres apartados por separado, e incluso para relacionarlos, el enlace: blog.catchpoint.com/2010/09/17/anatomyhtt

IV. Distribución de tiempos entre paquetes

A continuación vamos a comentar la distribución del tiempo entre paquetes utilizando distintos filtros. Primero nos quedaremos solo con el tráfico UDP que tenga como puerto origen ó destino el 10455. Luego, nos quedaremos con el tráfico TCP que tenga como IP origen ó destino la dirección 63.161.195.170.

(i) Paquetes UDP

En el caso de los paquetes UDP observamos que solamente hay tráfico de bajada (el filtrado del tráfico con el puerto como origen resulta en ningún paquete). Observemos la distribución del tráfico UDP en bajada:

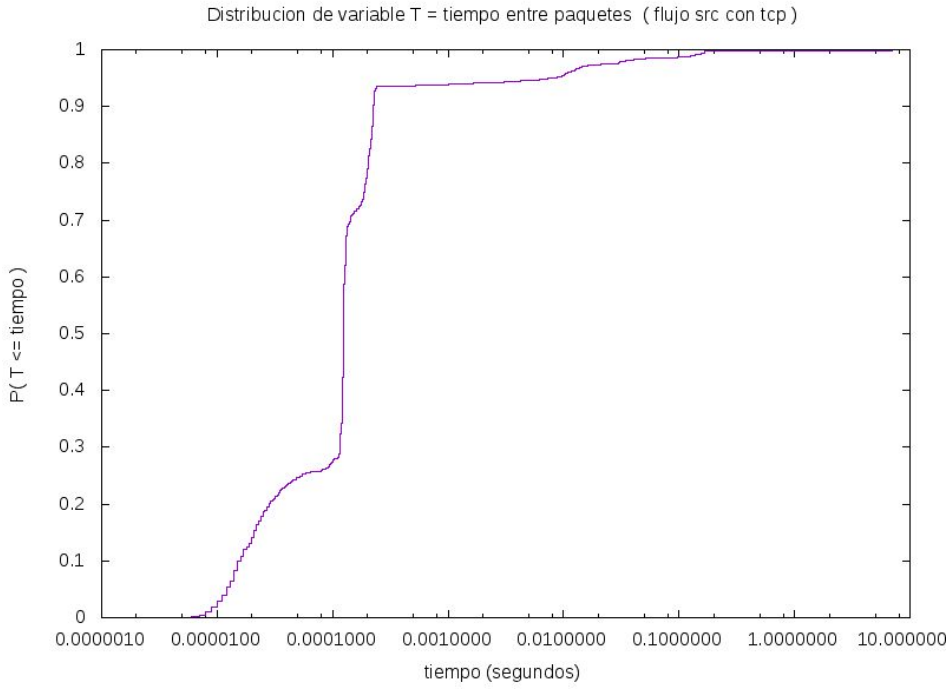


Vemos que la distribución se asemeja mucho a la de una constante. En este caso todos los paquetes llegan con un tiempo entre ellos extremadamente similar, de alrededor de 55 milisegundos. Bajo las premisas de que el tráfico es UDP, solo hay tráfico de bajada y los paquetes tienen un tiempo entre llegada muy similar, podemos inferir que el servicio que estamos observando es de *streaming* de audio/vídeo. Como estudiamos en teoría los *streamings* suelen utilizar el protocolo UDP, el hecho de que solo haya tráfico de bajada concuerda con lo inferido ya que el streaming del audio/vídeo se realizaría solo desde el servidor al cliente mientras que el cliente no requiere enviarle nada al servidor mediante el protocolo UDP ,y , por último, el hecho de que todos los paquetes lleguen en intervalos de 55 milisegundos puede tener relación con que el audio/vídeo se grabe en intervalos de 55 milisegundos y luego es mandado en un paquete, de este modo el intervalo en el que se envían los paquetes se mantiene constante y tiene el audio o vídeo correspondiente al tiempo transcurrido.

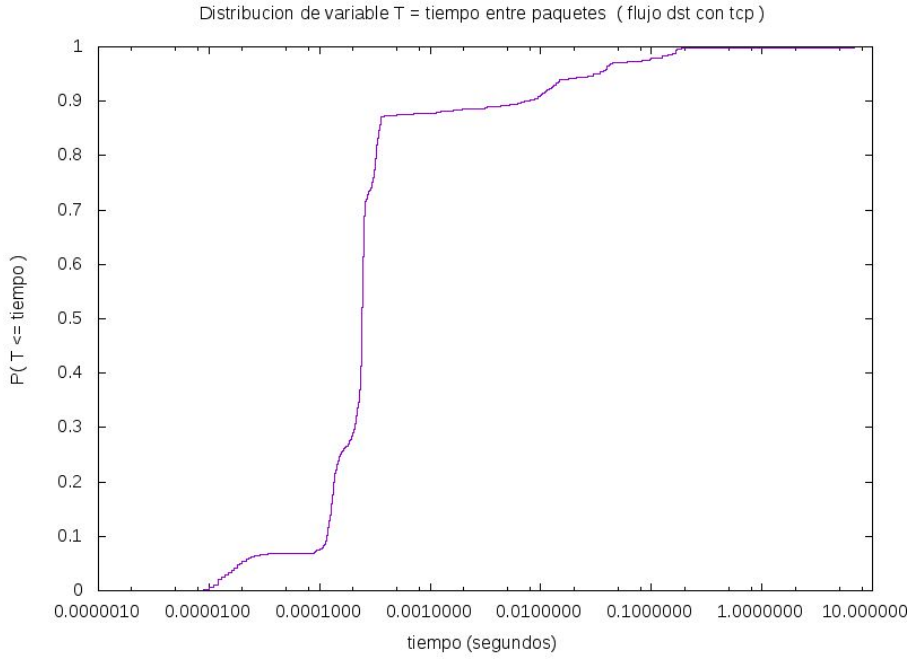
(ii) Paquetes TCP

A continuación presentamos las gráficas que corresponden a la distribución del tiempo entre paquetes del flujo TCP (filtrando con IP 63.161.195.170). Es importante destacar que para la mejor visualización de estas gráficas establecimos una **escala logarítmica en el eje x en ambos casos.**

Gráfica de subida (IP como origen) :



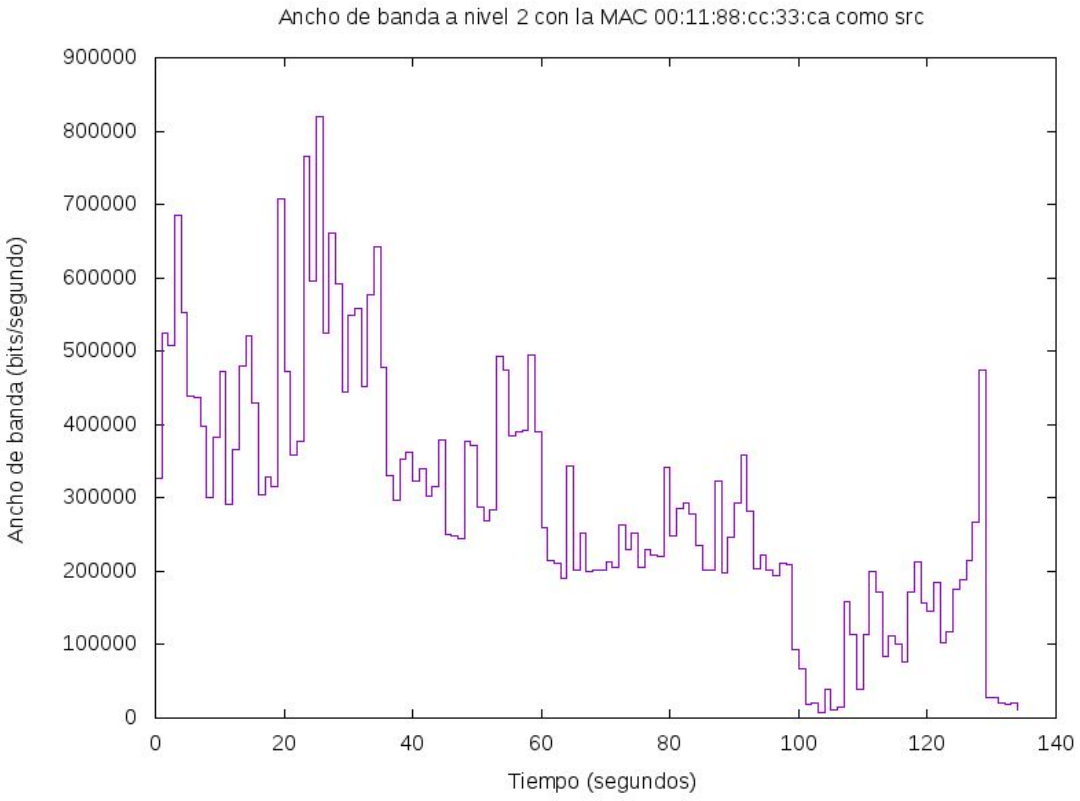
Gráfica de bajada (IP como destino):



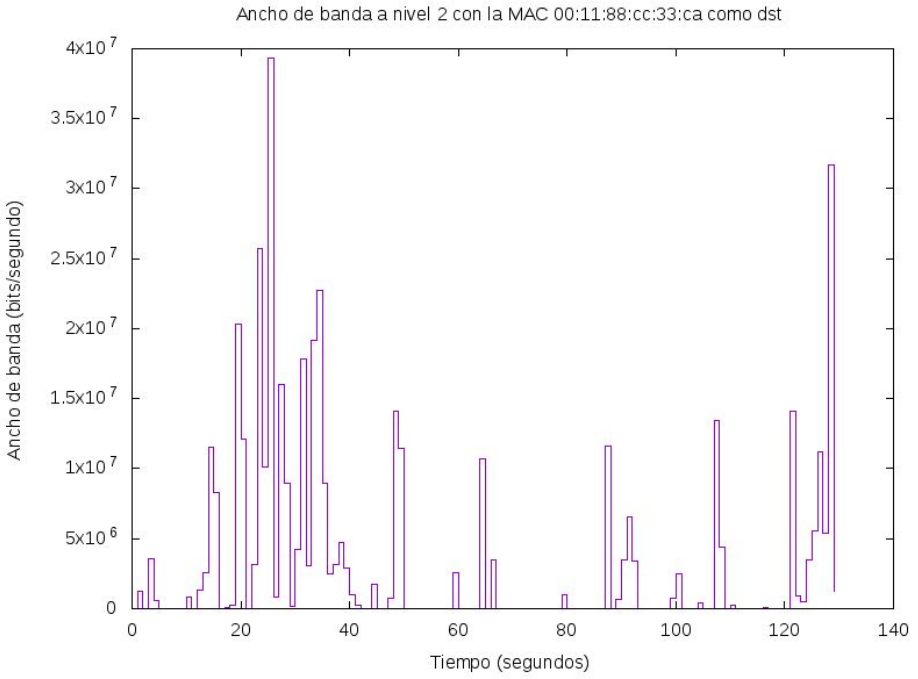
En este caso observamos distribuciones muy parecidas tanto en subida como en bajada. En ambos casos nos encontramos con que el noventa por ciento de los paquetes presentan un *interarrival* de menos de 1 milisegundo. Esto nos indica que la mayoría de los paquetes enviados utilizando TCP van prácticamente juntos. Este comportamiento es común en la descarga o subida de archivos a internet ya que el terminal que esté comunicando con TCP intentará enviar toda la información en paquetes consecutivos que justifican los bajos tiempos entre paquetes, también podría ser resultado de fragmentación de paquetes que resulta en varios paquetes TCP consecutivos.

V. Análisis del ancho de banda

A continuación analizaremos el ancho de banda consumido por el dispositivo con la dirección MAC 00:11:88:CC:33:CA tanto en subida como en bajada.
Primero tenemos la gráfica con la MAC como origen:



Y como destino:



El análisis de estas gráficas nos permite evaluar si el ancho de banda suministrado es suficiente o, si por el contrario, la red requiere de un mayor ancho de banda. En general, la presencia de mesetas en las gráficas podría indicar la necesidad de un mayor ancho de banda, sin embargo, en el caso de nuestras gráficas no existen dichas mesetas. Por tanto, para la red en cuestión, el ancho de banda suministrado (por lo menos al usuario de la MAC filtrada) es suficiente, ya que todo tráfico (en descarga o en subida) es resuelto rápidamente , lo que se ve representado por picos que duran pequeños intervalos de tiempo en la gráfica.

Otro aspecto interesante a destacar es que en la descarga observamos valores superiores en el ancho de banda que en la subida (una diferencia de dos órdenes de magnitud). Esto tiene sentido ya que el tráfico entrante, de varios terminales a la MAC filtrada, naturalmente es mayor que el tráfico saliente que es únicamente generado por un terminal (el de la MAC filtrada). Además, el tráfico entrante suele ser información enviada por servidores lo que genera paquetes mucho más grandes que las solicitudes que le pueda hacer un usuario a dichos servidores.

Conclusión

En primer lugar fuimos capaces observar que todo el tráfico de la red correspondía con el protocolo IP (el 99% con la versión 4 y el resto con la versión 6). Además, el tráfico era fundamentalmente TCP aunque observamos que pudiera haber una cantidad considerable de paquetes ICMP, que quizás debería analizarse en más detalle.

Con la información extraída de los tops, pudimos extraer, entre otras, las siguientes conclusiones: debido a la alta cantidad de tráfico HTTP capturado, comprobamos que estaban teniendo lugar muchos accesos a internet; y con respecto a los paquetes UDP, hemos podido observar el gran uso que se hace del protocolo DNS.

En cuanto al análisis de las distribuciones por tamaños concluimos: primero que, probablemente, el usuario con la MAC suministrada estaba usando servicios de navegación web; y, en segundo lugar, que las distribuciones DNS siguen un comportamiento similar a las distribuciones normales ya que mantienen relación directa con la distribución del tamaño del nombre de los dominios.

En lo referente al análisis de las distribuciones de *interarrival* fuimos capaces de observar: que la comunicación UDP era un streaming de audio/vídeo; y, detallar el comportamiento de las comunicaciones TCP.

Por último, concluimos que el ancho de banda suministrado a la MAC correspondiente suple el consumo del usuario, debido a la ausencia de mesetas en las gráficas.

A lo largo de esta práctica hemos analizado protocolos y tecnologías todavía no estudiadas en teoría, por lo que nos hemos basado en investigación a través de Internet. Debido a esto la información extraída podría no ser absolutamente certera pero esperamos haber esbozado una idea sobre el análisis de esta red.