Departamento de
Automática y Computación
Automatika eta
Konputazio Saila

Campus de Arrosadía
Arrosadiko Campusa
31006 Pamplona - *Iruñea*
Tfno. 948 169113, Fax. 948 168924
Email: ayc@unavarra.es

Práctica3 - Analizadores de red: Wireshark y tcpdump.

1- Objetivos

Comprender los conceptos básicos del monitoreo de tráfico de red mediante el uso del analizador de protocolos Wireshark y del sniffer tcpdump.

2- Login

En esta práctica hará uso de su cuenta `arssxy` de Linux.

3- Wireshark: Herramienta de sniffing y analizador de protocolos

Un sniffer es una herramienta que se emplea para observar los mensajes que intercambian dos entidades en comunicación a través de una red. El sniffer (literalmente "olfateador") captura las tramas a nivel de enlace que se envían/reciben a través de los interfaces de red de nuestra computadora.

Un dato importante es que un "sniffer" es un elemento pasivo: observa los mensajes que intercambian aplicaciones y protocolos, pero ni genera información por sí mismo, ni es destinatario de ésta. Las tramas que captura son siempre una copia (exacta) de las que en realidad se envían/reciben en nuestro ordenador.

Un **analizador de protocolos** es un sniffer al que se le ha dotado de funcionalidad suficiente como para entender y traducir los protocolos que se están hablando en la red. Es de utilidad para desarrollar y depurar protocolos y aplicaciones de red. Permite al ordenador capturar diversas tramas de red para analizarlas, ya sea en tiempo real o después de haberlas capturado. Por analizar se entiende que el programa puede reconocer que la trama capturada pertenece a un protocolo concreto (HTTP, TCP, ICMP,...) y mostrar al usuario la información decodificada. De esta forma, el usuario puede ver todo aquello que en un momento concreto está circulando por la red que se está analizando. Esto último es muy importante para un programador que esté desarrollando un protocolo, o cualquier programa que transmita y reciba datos en una red, ya que le permite comprobar lo que realmente hace el programa. Además de para los programadores, estos analizadores son muy útiles para todos aquellos que quieren experimentar o comprobar cómo funcionan ciertos protocolos de red, analizando la estructura y funcionalidad de las unidades de datos que se intercambian. También, gracias a estos analizadores, se puede ver la relación que hay entre diferentes protocolos, para así comprender mejor su funcionamiento.

Wireshark es un analizador de protocolos de red, con interfaz gráfico, que nos permitirá capturar las tramas que entran y salen de nuestro ordenador para luego "diseccionarlas" y estudiar el contenido de los mismas. Wireshark, emplea la misma librería de captura de paquetes (libpcap) que otros sniffers conocidos, como tcpdump, aunque es capaz de leer muchos otros tipos de formato de captura.

Además es un software de libre distribución que puede correr en distintas plataformas (Windows, Linux/Unix, y Mac). Pero, probablemente, lo más destacable sea su interfaz gráfica y la potente capacidad de filtrado que presenta.

Departamento de
Automática y Computación
Automatika eta
Konputazio Saila

Campus de Arrosadía
Arrosadiko Campusa
31006 Pamplona - *Iruñea*
Tfno. 948 169113, Fax. 948 168924
Email: ayc@unavarra.es

Nos vamos a centrar ahora en exponer unas nociones básicas de la forma en la que el analizador de protocolos Wireshark presenta la información capturada.

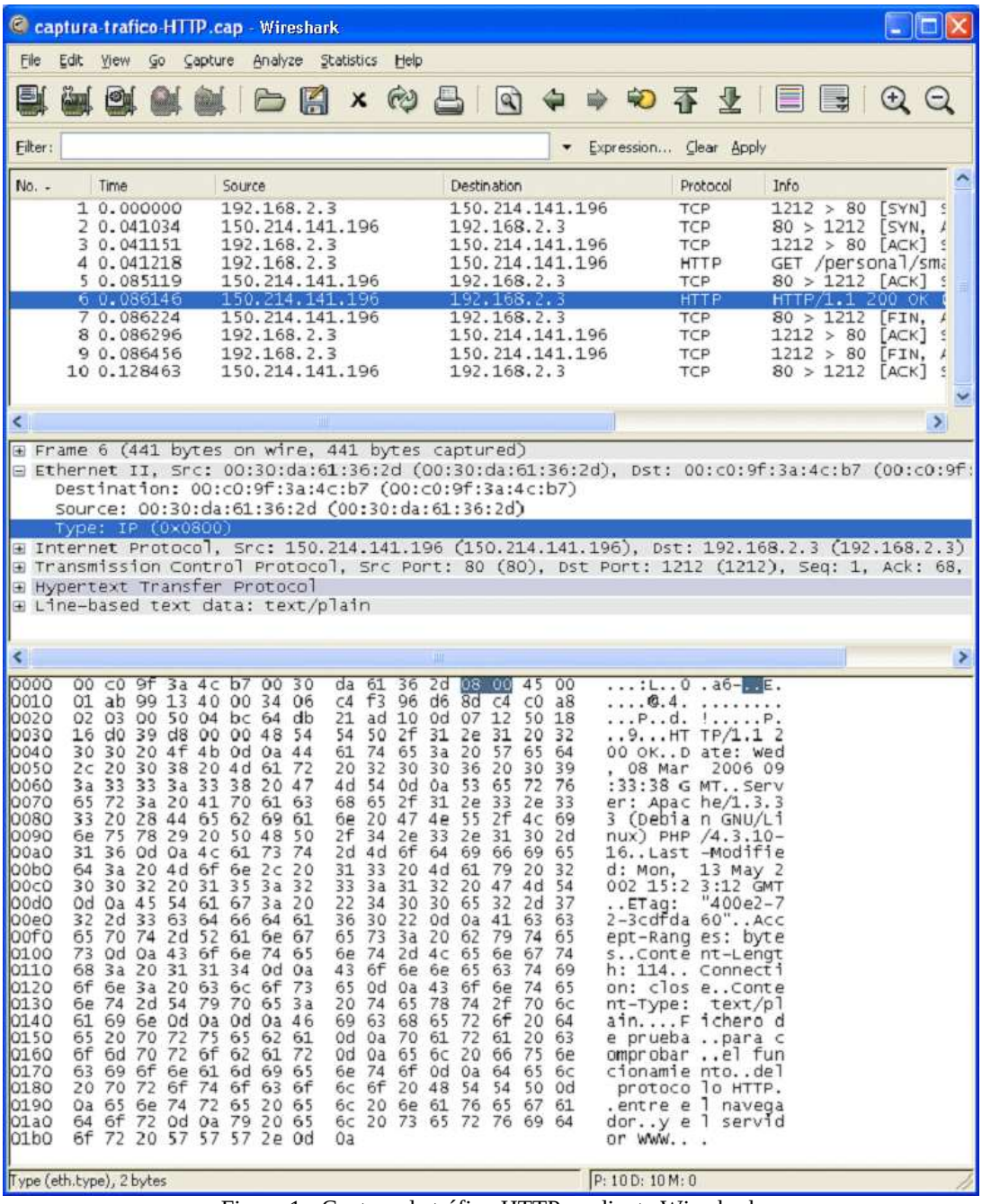


Figura 1.- Captura de tráfico HTTP mediante Wireshark

En la Fig. 1, podemos ver en funcionamiento el analizador de protocolos Wireshark. Nos está mostrando el contenido del archivo “captura-trafico-HTTP.cap”en el que hay 10 tramas previamente capturadas usando también el mismo programa, pues Wireshark sirve para ambas cosas, para capturar el tráfico de la red y para analizarlo, pudiendo salvarlo en un archivo si así se desea. Vamos a fijarnos en detalle en la forma en que Wireshark (y otros analizadores de protocolos) nos muestran las tramas capturadas.

Campus de Arrosadía
Arrosadiko Campusa
31006 Pamplona - *Iruñea*
Tfno. 948 169113, Fax. 948 168924
Email: ayc@unavarra.es

Vemos que la ventana de Wireshark se encuentra dividida en tres paneles: El superior,"Packet List" (según lo denomina el programa), el central, "Packet Details" y el inferior, "Packet Bytes". Hay que hacer notar que Wireshark llama "Packets" a las tramas capturadas. No hay que confundir estos "Packets" (llamados así quizás por razones históricas) con las PDUs de nivel 3. Nosotros nos vamos a referir siempre a los datos capturados por Wireshark como tramas.

El panel superior muestra un listado de las tramas capturadas. En este caso muestra las 10 tramas que tenemos capturadas en el archivo "captura-trafico-HTTP.cap". Nótese que hay una trama, la número 6, "resaltada" en un color más oscuro. Eso es así pues hemos hecho clic sobre ella con el ratón, seleccionándola de entre todas las del listado. La información que el listado de tramas muestra respecto a cada una de las tramas es muy limitada, por razones obvias de espacio. Se reduce a los campos más importantes de la misma y a un breve resumen que permita, de un vistazo, hacerse una idea de lo que está ocurriendo en la red.

El panel central muestra los detalles relativos al contenido de la trama que hayamos seleccionado en el panel superior, en este caso la trama ("Frame") número 6. Los detalles de la trama, que son muchos, son mostrados en forma de árbol, cuyas ramas podemos contraer o expandir, para tener una visión más general (contrayendo las ramas) o una visión más detallada (expandiéndolas). En la imagen podemos ver como hay una primera rama, que está contraída (se sabe porque aparece un signo "+ " que permitiría expandirla). Esta primera rama nos indica el número de orden de la trama con respecto a las demás (Frame 6) y diversa información relacionada con el instante en que la trama fue capturada. Es decir, la información de esta primera rama la aporta el programa Wireshark y NO es algo que tenga que ver con el contenido de la trama. Son cosas como la fecha y la hora de la captura, número de octetos que se han capturado de la trama, número de orden, etc… Las restantes ramas que van apareciendo en el panel central ya **sí** que tienen que ver con el contenido de la trama. Aparece una rama por cada cabecera que se detecta en la trama. En este caso el analizador de protocolos nos indica que en la trama 6 hay una cabecera Ethernet versión 2, luego hay una cabecera IP (Internet Protocol), luego una cabecera TCP (Transmisión Control Protocol) y así sucesivamente.

Concretamente vemos que la rama correspondiente a la cabecera Ethernet versión 2 está expandida y podemos ver los tres campos que la componen. Está resaltado en un color más oscuro el campo Tipo ("Type") y podemos ver su valor (que es 0x0800, número hexadecimal pues empieza por 0x) y su significado, en este caso IP (es correcto, pues ya sabemos que el valor 0x0800 en el campo tipo de una trama Ethernet versión 2 quiere decir que la trama contiene datos del protocolo IP).

Por último, el panel inferior muestra, sin ninguna información extra, los octetos ("bytes") de los que está compuesta la trama que se ha seleccionado en el panel superior y cuyos detalles ya estamos viendo en el panel central. Esos octetos se muestran en hexadecimal (cada octeto son dos dígitos hexadecimales) organizados en filas de 16 octetos. Como ayuda podemos ver que cada fila de 16 octetos viene precedida de un número en hexadecimal que nos indica la posición que ocupa el primero octeto de la fila en la trama. Por ejemplo, la primera fila viene precedida por el número 0000 (hexadecimal) lo que quiere decir que el primer octeto de esa fila es el que estaba en la posición primera de la trama (la cero). La segunda fila está etiquetada con el número 0010 (hexadecimal), que es el 16 en decimal. Luego el primer octeto de esa segunda fila ocupa la posición 16 en la trama. Si nos fijamos nos damos cuenta que estas etiquetas de ayuda que aparecen al principio de cada fila van de 16 en 16 (de 0010 en 0010 en hexadecimal) porque las filas tienen 16 octetos exactamente. Por comodidad, este panel inferior nos muestra también, en su parte derecha, una copia de los octetos de la trama pero en formato ASCII. Es decir, cada octeto es traducido al carácter equivalente según el

código ASCII. Los caracteres no imprimibles (los que no equivalen a letras, números o símbolos) se representan como un punto. Esto puede ser útil en ciertas tramas que contengan PDUs con datos en modo texto. Nótese que el panel inferior y el panel central son la misma cosa vista de dos modos diferentes. No hay más que ver como al haber seleccionado el campo "Type" en el panel central, en el panel inferior podemos ver resaltado un par de octetos de la trama, que son precisamente el 08 y 00, el valor que tiene dicho campo "Type". Hay que precisar que el analizador nos muestra en este panel inferior toda la trama Ethernet versión 2 o IEEE 802.3 salvo los 64 bits primeros del preámbulo. Es decir, que el octeto primero que podremos ver será el primer octeto de la dirección MAC destino. Otro campo que muchas veces no aparece será la cola de la trama (el FCS, "Frame Check Sequence) pues hay tarjetas que son incapaces de proporcionar este dato en el momento de la captura.

Finalmente, por encima de estas tres secciones aparecen otros dos elementos: los menús de comandos (menús desplegables y barra de herramientas) y el campo de filtrado de visualización.

4- ¿Qué necesito para estar en red?

Un ordenador que vaya a funcionar en red necesita una dirección para que la red pueda dirigir hacia él los datos que le envían el resto de ordenadores, es la dirección IP. Para ver la dirección IP de su ordenador (S.O. Linux) puede usar el comando `ifconfig`.

```
$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:11:2F:72:2B:9E
          inet addr:10.1.1.51  Bcast:10.1.255.255  Mask:255.255.0.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:241448 errors:0 dropped:0 overruns:0 frame:0
          TX packets:84926 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:43488888 (41.4 Mb)  TX bytes:18249053 (17.4 Mb)
          Interrupt:10 Base address:0xd800

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:3489 errors:0 dropped:0 overruns:0 frame:0
          TX packets:3489 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:2786870 (2.6 Mb)  TX bytes:2786870 (2.6 Mb)
```

Puede observar que su ordenador tiene dos interfaces:

- `eth0` es una conexión a una red de área local Ethernet y es la verdadera conexión de red de ese ordenador.
- `lo` es un interfaz ficticio llamado interfaz de `loopback`, todo lo que se envía por ese interfaz se vuelve a recibir en el ordenador. Es típico de los sistemas UNIX tener este interfaz y vale para enviarse datos a sí mismo incluso cuando el ordenador no está conectado a la red. En los sistemas UNIX muchas partes del sistema operativo funcionan como servicios de red, de ahí que el interfaz de `loopback` sea muy útil. Pero de momento no se preocupe por él.

Así pues su ordenador tiene un interfaz conectado a una red Ethernet y en dicho interfaz utiliza la dirección IP que se ve en el campo `inet addr:`. Compruebe cuál es su dirección IP. Esa dirección es

Campus de Arrosadía
Arrosadiko Campusa
31006 Pamplona - *Iruñea*
Tfno. 948 169113, Fax. 948 168924
Email: ayc@unavarra.es

suficiente para identificar a su ordenador en Internet. En nuestro caso, al estar la red del Laboratorio separada de Internet (es una Intranet) la dirección sólo le identifica entre los ordenadores del dominio del Área de Telemática (o sea este laboratorio[Telemática 1] y el de abajo[Telemática 2]). Pero para todos los efectos funciona igual que Internet.

Pruebe el comando `ping`. El comando `ping` es una utilidad que le permite comprobar si existe conectividad de red entre dos máquinas. Con ayuda de ping podremos determinar si el nivel de red funciona adecuadamente, así como los niveles de enlace y físico sobre los que descansa. Para ello la máquina que lanza el comando `ping` envía paquetes del protocolo ICMP que el sistema operativo de la máquina destino está obligada a responder al origen. El comando `ping` recibe estos paquetes y nos los muestra indicándonos también el tiempo que tardan en ir y volver (`Round Trip Time, RTT`) y contando los que se pierden. Mire la dirección IP que tiene su vecino de mesa y haga ping a su propio ordenador y al del vecino.

```
$ ping direccion_IP_de_mi_vecino
$ ping mi_direccion_IP
```

Observe la diferencia de tiempos. ¿Cómo hace ping para saber que los paquetes se pierden?

5- Utilizando Wireshark

Para ejecutar wireshark, abra un terminal y teclee; `wireshark` o bien láncelo desde el menú aplicaciones.

Por ahora, en nuestras pantallas, las distintas áreas que hemos comentado anteriormente aparecen en blanco. Capturemos los primeros paquetes y veamos qué sucede.

- ✓ Desde otro terminal terminal: `$ ping direccion_IP_de_mi_vecino`

- ✓ Para comenzar la captura desplegamos en `wireshark` el menú `Capture`, seleccionamos la opción `Interfaces`, aparecerán todos los interfaces disponibles, e iniciamos(`Start`) la captura en el interfaz `eth0`. Wireshark ya está capturando todas las tramas que traspasan nuestro interfaz de red.

- ✓ Observe que mientras `wireshark` captura, le muestra que está reconociendo paquetes de diversos protocolos. Cuando tenga algún paquete ICMP, los generados por el comando `ping`, detenga la captura y busque en estos paquetes ICMP qué dirección origen y destino llevan.

- ✓ Puede indicarle al programa `wireshark` que filtre el tráfico capturado, de forma que sólo muestre por pantalla los paquetes ICMP. Para ello en la casilla de texto junto al botón *Filter* escriba *icmp* y pulse intro. Del mismo modo puede introducir este mismo filtro en la ventana de programación de la captura de forma que sólo capture los paquetes que cumplan el filtro.

- ✓ Para finalizar la captura seleccione del menú `Capture` la opción `Stop`, o pulse el botón correspondiente en la barra de iconos.

Departamento de
Automática y Computación
Automatika eta
Konputazio Saila

Campus de Arrosadía
Arrosadiko Campusa
31006 Pamplona - *Iruñea*
Tfno. 948 169113, Fax. 948 168924
Email: ayc@unavarra.es

Analicen, a continuación, las tramas capturadas ayudándose para ello de las siguientes cuestiones.

⇒ Para la trama Ethernet que contiene el mensaje "echo request":

1. ¿Cuál es la dirección Ethernet de 48-bit del interfaz de red de tu ordenador?
2. ¿Cuál es la dirección Ethernet destino dentro de la trama Ethernet? ¿A qué dispositivo pertenece dicha dirección?
3. ¿Cuál es el valor hexadecimal del campo Tipo de Trama (Frame Type)?
4. ¿Qué tamaño tiene el campo de datos de esta trama Ethernet?

⇒ Y para la trama Ethernet que contiene el mensaje de respuesta "echo reply":

1. ¿Cuál es la dirección Ethernet origen dentro de la trama Ethernet? ¿A qué dispositivo pertenece dicha dirección?
2. ¿Cuál es la dirección destino dentro de la trama Ethernet? ¿A qué dispositivo pertenece dicha dirección?
3. ¿Cuál es el valor hexadecimal del campo Tipo de Trama (Frame Type)?
4. ¿Qué tamaño tiene el campo de datos de esta trama Ethernet?

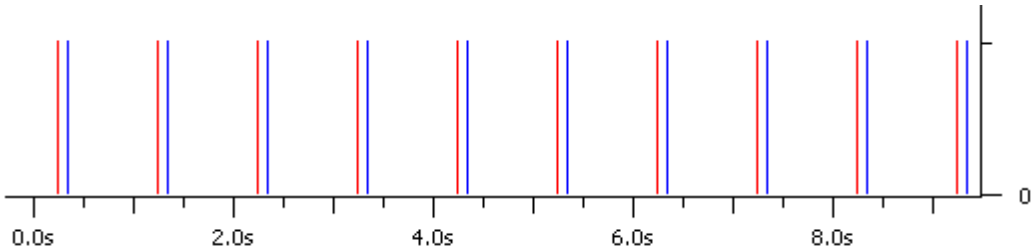
Checkpoint 3.1: Muestre al profesor de prácticas el valor del campo, dentro de la cabecera IP, que ha permitido saber al analizador que el contenido del paquete IP (campo datos) era un paquete ICMP.

A continuación nos familiarizaremos con la herramienta `IO Graphs` del menú `Statistics`. Para ello realice el siguiente ejercicio:

1. Ponga Wireshark a capturar y a continuación lance un ping de diez mensajes icmp (consulte el manual de ping; `man ping`) contra cualquier url en Internet y anote su correspondiente dirección IP. Observe el RTT de cada uno de ellos y téngalos en cuenta durante el resto del ejercicio. Pare la captura de Wireshark. ¿Cuántos mensajes icmp(request) e icmp(reply) observa?

2. Seleccione `Statistics`, `IO Graphs`. Configure la representación gráfica adecuadamente para ser capaz de determinar exactamente cada uno de los RTT devueltos por ping. Utilice para ello el estilo de representación gráfica más adecuado y ajuste las escalas necesarias en los ejes.

3. Represente en color rojo los ICMP(request) y en color azul los ICMP(reply). Aplique los filtros necesarios en cada caso. Deberá obtener una representación similar a ésta:



Nota: *Tenga en cuenta a la hora de realizar la captura, que conforme transcurre el tiempo, dado el orden de los valores del RTT, pierde resolución en el eje horizontal. Si esto le supone un problema, realice el ajuste que considere necesario a la hora de lanzar el ping. Repita el paso 1 las veces que necesite hasta obtener el resultado deseado.*

Checkpoint 3.2: Muestre al profesor de prácticas la representación gráfica del ejercicio propuesto.

Departamento de
Automática y Computación
Automatika eta
Konputazio Saila

Campus de Arrosadía
Arrosadiko Campusa
31006 Pamplona - *Iruñea*
Tfno. 948 169113, Fax. 948 168924
Email: ayc@unavarra.es

6- Empleo de tcpdump

Hemos estado revisando el contenido de paquetes desde una herramienta gráfica y de muy fácil manejo, como es Wireshark. Esta vez vamos a ver el contenido desde una herramienta de consola de comandos: `tcpdump`.

Como ya comentamos con anterioridad, la librería de captura de Wireshark (libpcap) es la misma que emplea `tcpdump`, y la sintaxis de filtrado es muy similar; pero para un mejor conocimiento de la herramienta, consulte las páginas de manual disponibles para `tcpdump`:

- En Linux → `man tcpdump`
- Online → http://www.tcpdump.org/tcpdump_man.html

Ahora analizaremos las tramas que se intercambianentre un cliente y un servidor de ftp. Para ello, mientras el programa tcpdump se esté ejecutando, realizaremos una conexión ftp a un servidor, luego haremos un GET sobre un archivo y seguidamente analizaremos la captura realizada.

Ejecute el programa `tcpdump` con la opción `-w` que permite guardar las tramas capturadas en un fichero para un posterior análisis de las mismas. Utilice la carpeta local `/tmp` para guardar este fichero de captura (indíqueselo a `tcpdump`). Es importante que en este paso no se introduzca ninguna opción (a excepción de la ya indicada), ni ningún filtro, ya que se pretende capturar todas las tramas que llegan a nuestro interfaz.

Estos son los pasos que tendremos que dar para capturar nuestras tramas:

1. Lanzar en un terminal el programa `tcpdump`. Grabaremos las tramas capturadas en el fichero `tcpdump.out`
2. Conectarse mediante ftp a la máquina `10.1.1.130` (`ftp 10.1.1.130`). En login introduciremos `anonymous` y como password `anonymous`.
3. Hacer `get RFC959FTP.txt` para "bajarse" el fichero `RFC959FTP.txt` a su ordenador.
4. Terminar la conexión tecleando `quit`.
5. Terminar la captura de tcpdump (`CONTROL-C` en el terminal donde esté lanzando).

Una vez tengamos el fichero `tcpdump.out`, realizaremos algunas consultas a las tramas capturadas mediante el propio programa `tcpdump`. Para realizar cualquier filtro sobre este fichero:

```
tcpdump -r tcpdump.out [opciones] [expresión]
```

No se permite la utilización de ningún tipo de opción a la hora de llamar a `tcpdump`, ni la del empleo de otras herramientas proporcionadas por linux. Se tendrá que hacer todo con la única ayuda de los filtros `[expresión]` de tcpdump. Se pide:

a) Mostrar sólo las tramas intercambiadas entre su ordenador y el servidor ftp. Genere el fichero `intercambio-a.out`.

```
tcpdump -r tcpdump.out [expresión] > intercambio-a.out
```

b) Mostrar sólo aquellas tramas cuyo origen sea su máquina y el destino el servidor de ftp. Genere el fichero `origendestino-b.out`.

Departamento de
Automática y Computación
Automatika eta
Konputazio Saila

Campus de Arrosadía
Arrosadiko Campusa
31006 Pamplona - *Iruñea*
Tfno. 948 169113, Fax. 948 168924
Email: ayc@unavarra.es

c) Ahora queremos quedarnos sólo con aquellas tramas que pertenezcan al protocolo ftp. Genere el fichero `protocoloftp-c.out`.

d) ¿Cómo se consigue ver las tramas de ftp cuyo origen es el servidor ftp?. Genere el fichero `tramasftpserver-d.out`.

e) ¿Sería capaz de capturar el usuario y contraseña, legibles, introducidos en el login del ftp? Genere el fichero `user&pass.out`

Checkpoint 3.3: Muestre al profesor de prácticas el contenido del fichero `user&pass.out`