

Práctica2 – Observando la red

1- Objetivos

El objetivo principal que se persigue en esta práctica es ser capaz de observar el tráfico de red mediante un analizador de protocolos como Wireshark y comprender los conceptos básicos de uso de unos protocolos sobre otros.

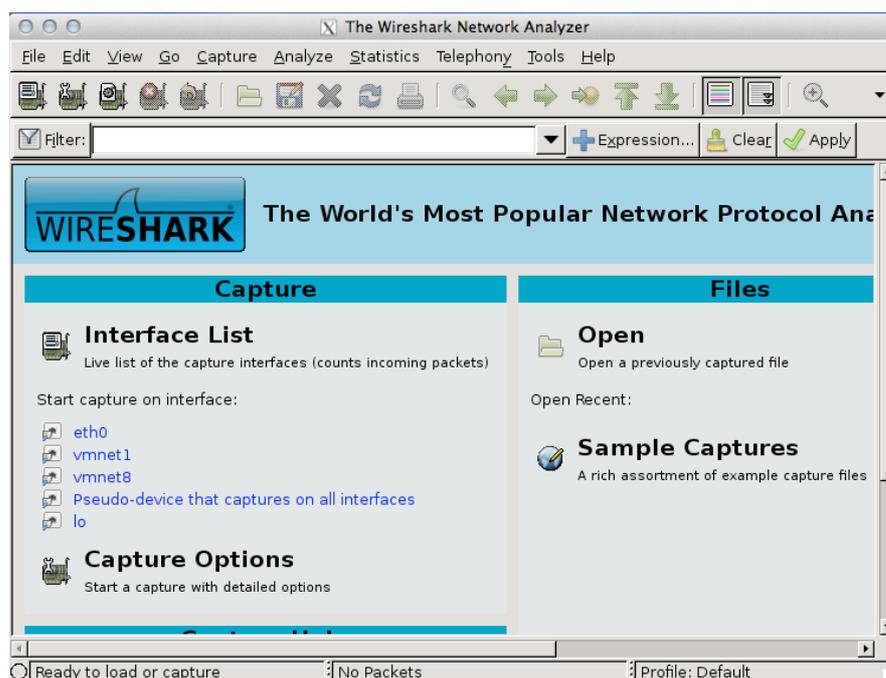
2- Usando Wireshark

Lance el programa wireshark desde el menú de aplicaciones o bien escribiendo en un terminal

```
$ wireshark &
```

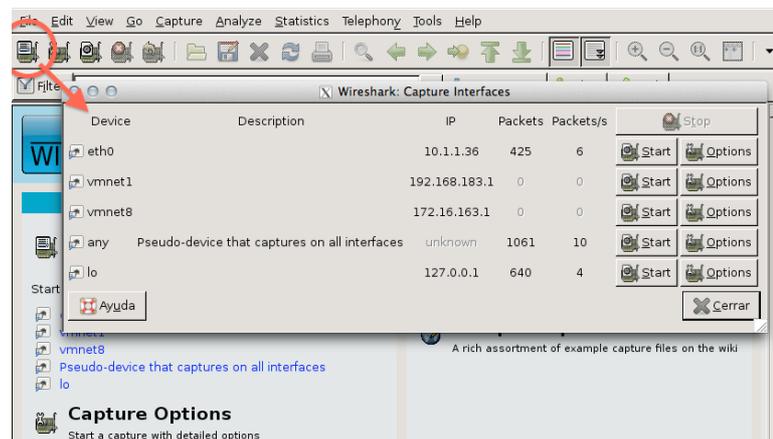
El programa se lanzará y abrirá una ventana. Wireshark es un analizador de protocolos. Su función principal es mostrar los paquetes observados en un interfaz y ayudar a diseccionarlos identificando cada cabecera de protocolo y los campos con información incluidos en la cabecera. Puede hacer esto directamente observando la red (o las redes) a las que está enchufado su ordenador pero también permite grabar los paquetes que ha visto en un fichero para su posterior análisis. A estas grabaciones las llamamos normalmente ficheros de captura o trazas. Wireshark también es capaz de abrir un fichero de traza previamente grabado aunque sea en otra máquina y hacer su análisis sobre los paquetes observados en la grabación.

Observe que desde la ventana inicial de Wireshark puede iniciar una captura de la red o abrir un fichero para su análisis. También es posible desde el menú o los iconos arriba a la izquierda.



Para iniciar una captura debe elegir en cuál de los interfaces que unen un ordenador a la red quiere capturar. Observe la lista eligiendo el primer icono de arriba. ¿Su ordenador tiene varios interfaces de red? Aunque es posible tener varias tarjetas de red y estar conectado a varias redes, en el caso del laboratorio la mayoría de los interfaces que ve son virtuales y no representan en realidad una salida a una red física.

El interfaz eth0 es el correspondiente a la tarjeta Ethernet de su ordenador. Puede observar el cable por detrás que lo une al punto de red de la mesa. Eso es eth0. Del resto de los interfaces que ve, los vmnet son interfaces virtuales para conectar maquinas virtuales (empleando VMWare). El interfaz lo es el llamado interfaz de loopback que vale para que programas de este ordenador puedan hablarse usando protocolos de red aunque el ordenador esté desconectado o no posea un interfaz físico. El interfaz any no es un interfaz sino la manera en que Wireshark permite observar todos a la vez. En esta práctica queremos observar la red Ethernet del laboratorio así que elija siempre eth0.



Una vez elegido el interfaz a usar, podemos capturar directamente sin pensar mucho (eligiendo start) o bien configurar algunas opciones en la captura (eligiendo options). Esto mismo se podía hacer desde las opciones de la pantalla inicial o con los iconos de arriba que dejan iniciar captura rápido desde el ultimo interfaz seleccionado o ir directamente a las opciones.

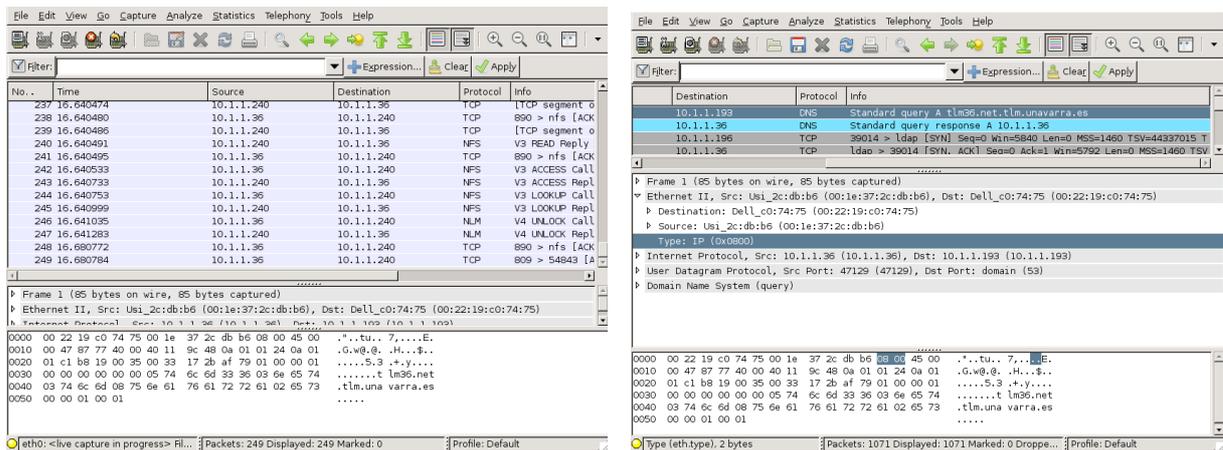
Elija options en el interfaz eth0 (o el icono de capture options). Aquí puede configurar algunas cosas de interés. Las que no se expliquen puede dejarlas como están.

- Interfaz : puede elegir entre los que hemos visto antes. Elija eth0 si no está ya seleccionado
- El link-layer header type es para elegir el tipo de trama que se envía en ese interfaz. En nuestro caso es Ethernet
- El modo promiscuo indica si queremos capturar todas las tramas que se vean en ese interfaz o solo las que haya enviado o vayan dirigidas a este ordenador. Elija de momento todas activando el promiscuous mode
- Se puede limitar que no se capturen paquetes enteros sino solo el principio de cada paquete para ahorrar memoria/disco ya que normalmente lo que queremos ver es solo las cabeceras. De momento no capturaremos muchos paquetes así que déjelo en no limitar.

- El filtro de captura (capture filter) nos permite decidir de una manera flexible qué paquetes queremos capturar y cuáles no. Es una expresión de texto en un lenguaje de reglas que pone condiciones a todos los paquetes se reciben. Si un paquete cumple la regla se captura y se muestra o se guarda y si no la cumple se descarta. En esta primera prueba asegúrese de que el campo está vacío, lo que significa que queremos capturar cualquier cosa
- Las opciones de más abajo de momento déjelas como están y elija start.

Wireshark empezará a capturar paquetes y mostrar una lista de los que ha capturado. En cuanto haya capturado algún paquete detenga la captura con el icono de stop.

Si selecciona un paquete de la lista puede ver detalles sobre ese paquete en los paneles de abajo. Desactive el botón de colorize packets para que sea más claro. Abajo del todo se ve el contenido del paquete completo y en el medio el análisis que hace Wireshark del contenido del mismo. Podemos desplegar cada una de las cabeceras y seleccionar los campos de cada cabecera y en el display de abajo se muestra dónde está situado ese campo.



El nivel inferior del análisis, *Frame*, muestra los datos de la captura del paquete y no es propiamente una cabecera. Dentro de *Frame* verá tramas Ethernet y dentro de las tramas Ethernet vera paquetes IP o ARP. IP a su vez puede transportar paquetes TCP o UDP. Por ejemplo elija un paquete cualquiera y vea con ayuda del análisis dónde están situadas las direcciones origen y destino de Ethernet. Observe también como los paquetes de diferentes tipos IP o ARP tienen diferente valor en el campo Type

Busque en su captura paquetes que transporten protocolos TCP o UDP sobre IP y observe el encapsulado de unos paquetes dentro de otros.

Utilice el comando `ifconfig` para averiguar la dirección de su interfaz Ethernet. Puede observar la dirección MAC (direcciónHW) y la dirección IP (Direc. Inet).

```
$ ifconfig eth0
eth0      Link encap:Ethernet  direcciónHW 00:1e:37:2c:db:b2
          Direc. inet:10.1.1.26  Difus.:10.1.255.255  Másc:255.255.0.0
          ...
```

Utilice la dirección IP para localizar un paquete enviado por su ordenador. Fíjese que en la lista de paquetes puede aplicar un filtro para seleccionar algunos. Esto se llama display filter y se escribe en un lenguaje de reglas. Puede editar reglas y añadirlas pulsando en el botón expression (tendrá que elegir en la lista los protocolos sobre los que aplicar las reglas, busque Ethernet por ahí). Por ejemplo puede ver solo los paquetes que tengan una dirección IP o una dirección MAC concreta eligiendo reglas como estas (averigüe qué hacen). Pulse en el botón apply para aplicar el filtro de display.

```
eth.addr==00:1e:37:2c:db:b2 (ponga su dirección MAC)
```

```
eth.src==00:1e:37:2c:db:b2
```

```
ip.src==10.1.1.26
```

En este paquete examine la cabecera de Ethernet para comprobar las direcciones MAC origen y destino de la trama Ethernet. Compruebe que la dirección MAC de origen coincide con la de su ordenador (que puede ver también haciendo `ifconfig` en un terminal). Observe que las direcciones IP origen y destino del paquete aparecen en la cabecera de IP que va dentro de la trama Ethernet. Borre el filtro de display con clear para volver a observar todos los paquetes.

Guarde la traza de paquetes capturados en disco para su posterior análisis, use el formato wireshark/tcpdump libpcap. Observe que puede guardar solo los paquetes seleccionados o los que cumplen el display filter si es necesario. Cierre el Wireshark y pruebe a volver a abrirlo y abrir el fichero para volver a examinar la traza.

Por ultimo, en esta visión general del uso de Wireshark, vuelva a iniciar el diálogo de opciones para capturar. Elija el interfaz eth0 y fíjese en el filtro de captura. Puede indicar a Wireshark que no capture todos los paquetes que vea sino que solo elija algunos. Esto se llama capture filter. Consiste también en una serie de condiciones en un lenguaje de reglas. Por razones históricas este lenguaje no es el mismo que el del display filter. Por ejemplo las reglas de antes en capture filter son

```
ether host 00:1e:37:2c:db:b2
```

```
ether src 00:1e:37:2c:db:b2
```

```
ip src 10.1.1.26
```

En el botón capture filter tiene algunos ejemplos mas de reglas pregrabadas.

3- Observando y localizando el tráfico

Una vez se tiene un control básico de Wireshark observemos el tráfico en la red. La idea es poner a Wireshark a capturar los paquetes que intercambia su maquina con otro ordenador concreto utilizando un capture filter. No debería observar demasiados paquetes de forma que puede dejar la captura en tiempo real y observar lo que va apareciendo. Para ello obtenga la dirección MAC de otro ordenador de su mesa del laboratorio y realice una captura que vea solo el tráfico entre estos ordenadores con un filtro de captura. Deberá usar una de estas dos opciones en el capture filter. Pruebe las dos y decida cuál es la correcta. Las dos significan cosas diferentes y debería ser capaz de interpretar la diferencia

```
ether src midireccionMAC and ether dst direccionMACdelOtro  
ether host midireccionMAC and ether host direccionMACdelOtro
```

Una vez capturando, haga un ping al otro ordenador, lo que le enviará paquetes al otro para obtener respuestas verificando que está funcionando como se vio en la práctica anterior.

```
$ ping ordenadorvecino
```

```
$ ping -c 2 ordenadorvecino      (puede poner con -c que solo envíe 2 veces)
```

Observe los paquetes que aparecen en la red como resultado del ping. Seguidamente haga algo que genere intercambio de información entre esos dos ordenadores como utilizar un ssh para obtener un acceso remoto de uno en el otro y observe el tráfico que genera.

```
$ ssh ordenadorvecino
```

Pruebe en el menú statistics la opción resumen (Summary) que resume los parámetros principales de una captura. Pruebe también la descomposición del tráfico en protocolos (Protocol Hierarchy) para ver cuántos paquetes y qué porcentaje de la captura corresponde a cada protocolo.

Pruebe también que puede graficar los parámetros en función del tiempo usando IO Graphs y utilícelo para ver cuántos paquetes por segundo y bits por segundo está generando. Observe que en la grafica puede elegir el intervalo de tiempo en el que van a promediarse las medidas (tick interval). Con eso puede calcular por ejemplo el throughput (Mbps) en intervalos de 1s o de 10s.

Puede usar estas estadísticas sobre una captura en tiempo real o sobre un fichero ya capturado.

Puede probar esto también con un ordenador externo. Para ello elija una pagina web que no sea google. Utilice el comando host para obtener la dirección del servidor. Por ejemplo:

```
$ host www.tlm.unavarra.es  
www.tlm.unavarra.es is an alias for pluto.tlm.unavarra.es.  
pluto.tlm.unavarra.es has address 130.206.164.68
```

Lance Wireshark capturando solo los paquetes que vayan entre su maquina y el servidor. Para ello también puede poner condiciones sobre protocolos que no sean Ethernet por ejemplo la dirección IP del servidor

```
ether src midireccionMAC and ip dst direccionIPservidor  
ether host midireccionMAC and ip host direccionIPservidor
```

Una vez capturando pida una pagina web al servidor mientras mantiene la captura y observe los paquetes causados por la petición web. Pruebe a buscar el contenido de la petición y de la pagina pedida en los paquetes capturados.

4- Analizando una captura en fichero

Descargue el fichero p2_capture.cap de la página web de la asignatura que contiene una captura realizada previamente. La captura se ha realizado en la maquina con dirección MAC 00:1e:37:2c:db:b2. Analícela para responder a las siguientes preguntas

¿Cuántos paquetes capturados hay en el fichero?

¿Durante cuánto tiempo se ha estado capturando esa traza?

¿Cuál es la velocidad media de captura? ¿Cuál es el tamaño medio del paquete capturado?

¿En que instante de tiempo aparece por primera vez un tráfico sostenido de mas de 5Mbps?

¿Durante cuánto tiempo se mantiene un tráfico sostenido de mas de 7Mbps?

¿El primer paquete de la traza ha sido enviado o recibido por la maquina en la que se ha obtenido la captura?

¿En el paquete número 2000 cuál es la dirección MAC de origen y destino?

¿Cuántos paquetes IP hay en la captura? ¿Cuántos paquetes TCP? ¿Cuántos UDP?

¿Cuál es la velocidad media en paquetes por segundo de la traza?

¿A qué velocidad aproximada se han enviado datos hacia la dirección MAC 00:1e:37:2c:dd:04?

¿Durante cuánto tiempo ha recibido datos a esa velocidad la dirección MAC 00:1e:37:2c:dd:04?

¿Las direcciones IP 10.1.1.24, 10.1.1.25, 10.1.1.26 aparecen en la traza?

5- Analizadores de tráfico de línea de comandos [opcional]

Wireshark es fácil y cómodo de utilizar pero en ocasiones queremos capturar rápidamente en línea de comandos simplemente para ver el contenido de un paquete o si hay determinado tipo de tráfico rápidamente. Para ello existen programas capaces de capturar trafico en línea de comandos.

Pruebe los programas tcpdump y tshark para mostrar los paquetes que se vean en el interfaz eth0 de su maquina. Tcpdump es el clásico, tshark es la herramienta de línea de comandos de wireshark. Los dos son similares y se basan en la librería de captura de tráfico llamada libpcap. Pruebe al menos estas opciones.

```
tcpdump -i eth0          capturar todo en el interfaz eth0
tshark -i eth0           capturar todo en el interfaz eth0
tshark -V -i eth0        volcando además los detalles de cada paquete
```

También puede usar un capture filter indicándolo al final de la línea de comandos

```
tshark -i eth0 ether host 00:1e:37:2c:db:b2

tcpdump -i eth0 ether host 00:1e:37:2c:db:b2
```

Mostrar el contenido de los paquetes

```
tcpdump -i eth0 -XX ether host 00:1e:37:2c:db:b2
tshark -i eth0 -x ether host 00:1e:37:2c:db:b2
```

Grabar y cargar ficheros

```
tcpdump -i eth0 -w captura.cap          guarda los paquetes en captura.cap
tcpdump -r captura.cap                  lee de captura.cap en lugar del capturar
```