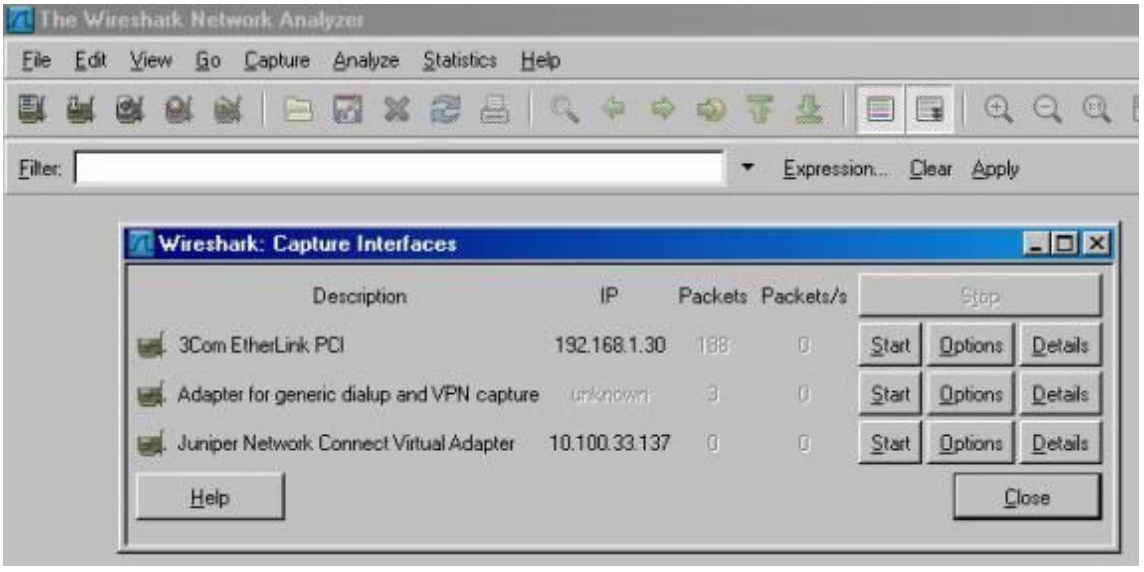
Wireshark

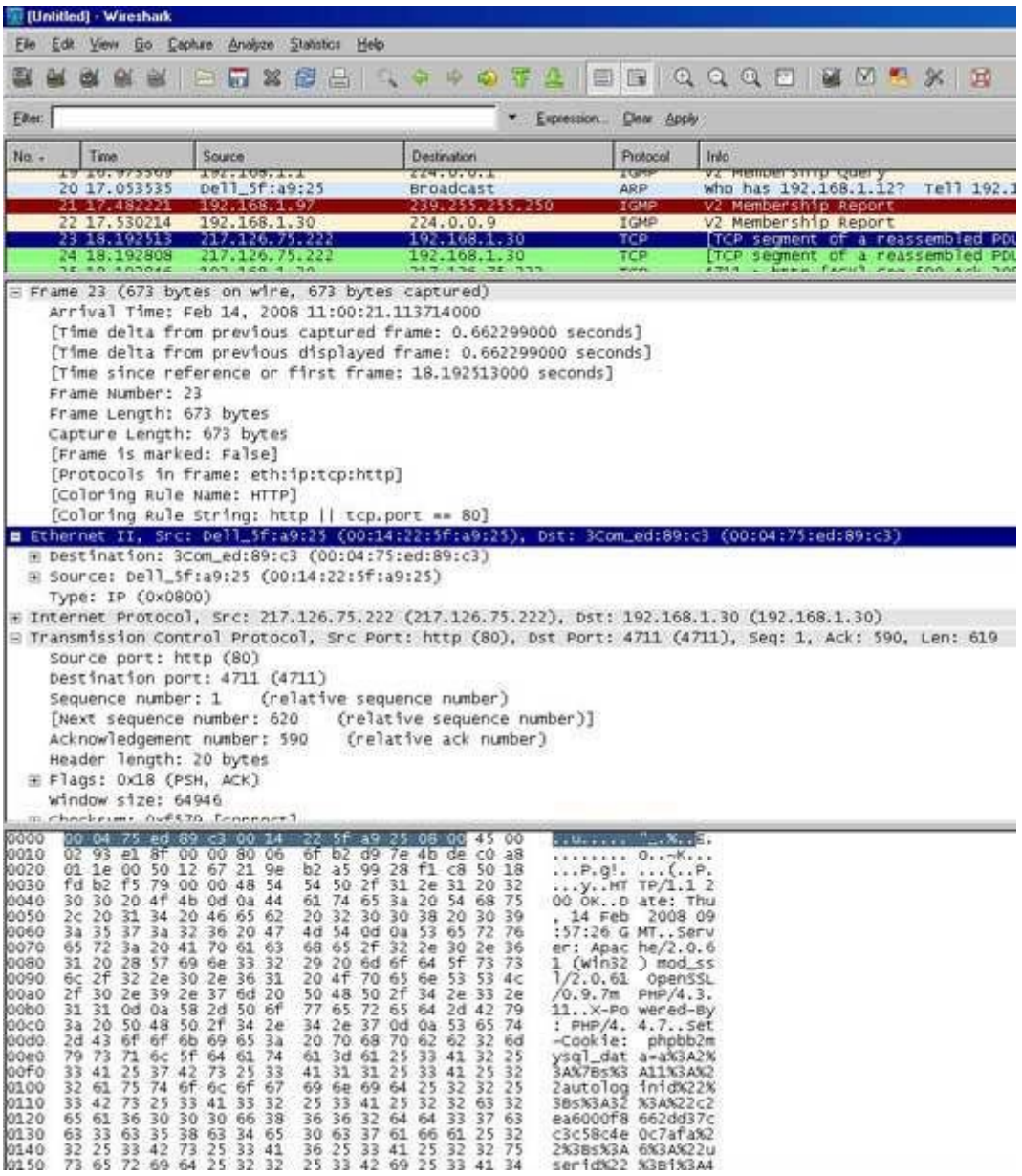
Wireshark es una herramienta de código abierto, multiplataforma, de análisis de red.

Nos permite capturar y analizar el tráfico que llega a nuestro sistema. Funciona al igual que lo puede hacer cualquier otro sniffer tal como Windump, TCPDump ó dsniff. Pero, al contrario de estos, lo hace mostrando los datos a través de un entorno gráfico y de forma más amigable y entendible.

Antes que nada, tras arrancar Wireshark, el menú Capture > Interfaces…. nos muestra la siguiente pantalla:



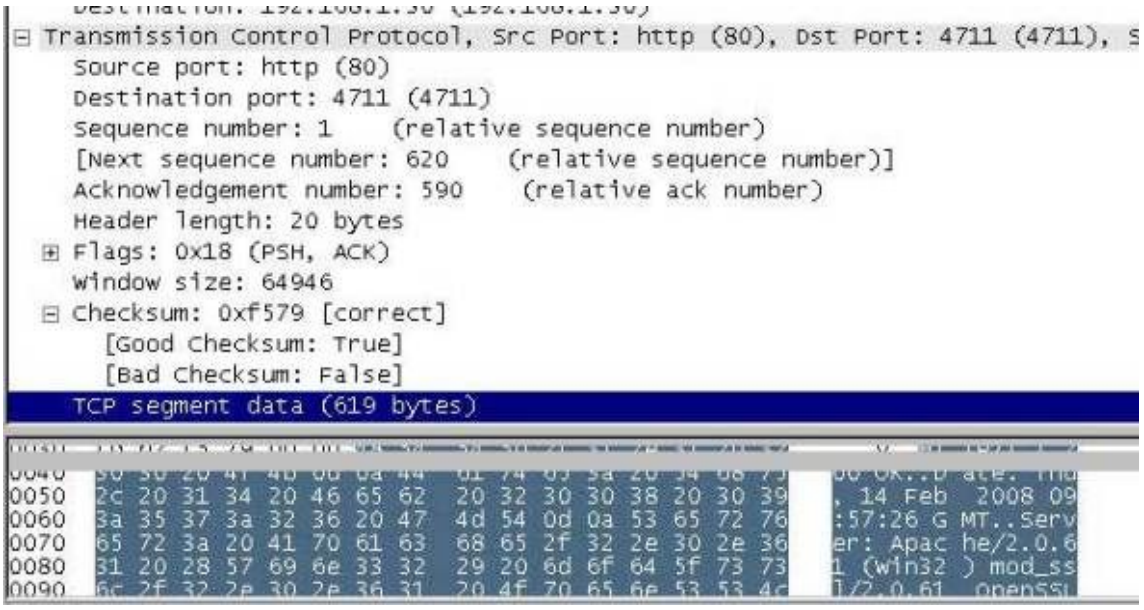
Solo tendremos que pulsar en Start para capturar a través de la interface que nos interese. Inmediatamente Wireshark comienza a capturar.



Se establecen 3 zonas de datos. La primera es la zona de listado de los paquetes capturados con información del Número de Frame, tiempo en segundos de la captura, Origen, Destino, protocolo involucrado y por último un campo de información extra que previamente Wireshark a decodificado.

La segunda zona muestra los datos del Frame capturado.

- Frame 23 (los numera secuencialmente). Nos da información sobre la hora de llegada el tamaño, etc.
- Ethernet II nos muestra la cabecera Ethernet II que a su vez pertenece a la capa de enlace de datos. Donde podemos identificar MAC origen y MAC destino.
- Transmission Control Protocol. (TCP): Puerto origen, puerto destino, flags, etc.
- TCP Segment Data, con todo el contenido del campo Data del segmento TCP.



Filtros

Wireshark contempla dos tipos de Filtros. Filtros de captura y Filtros de visualización. En Wireshark para los filtros de captura podemos hacer uso de los mismos filtros de TCPDump / Windump, ya que usa la misma librería pcap.

Los filtros de captura (Capture Filter) son los que se establecen para definir qué paquetes vamos a capturar.

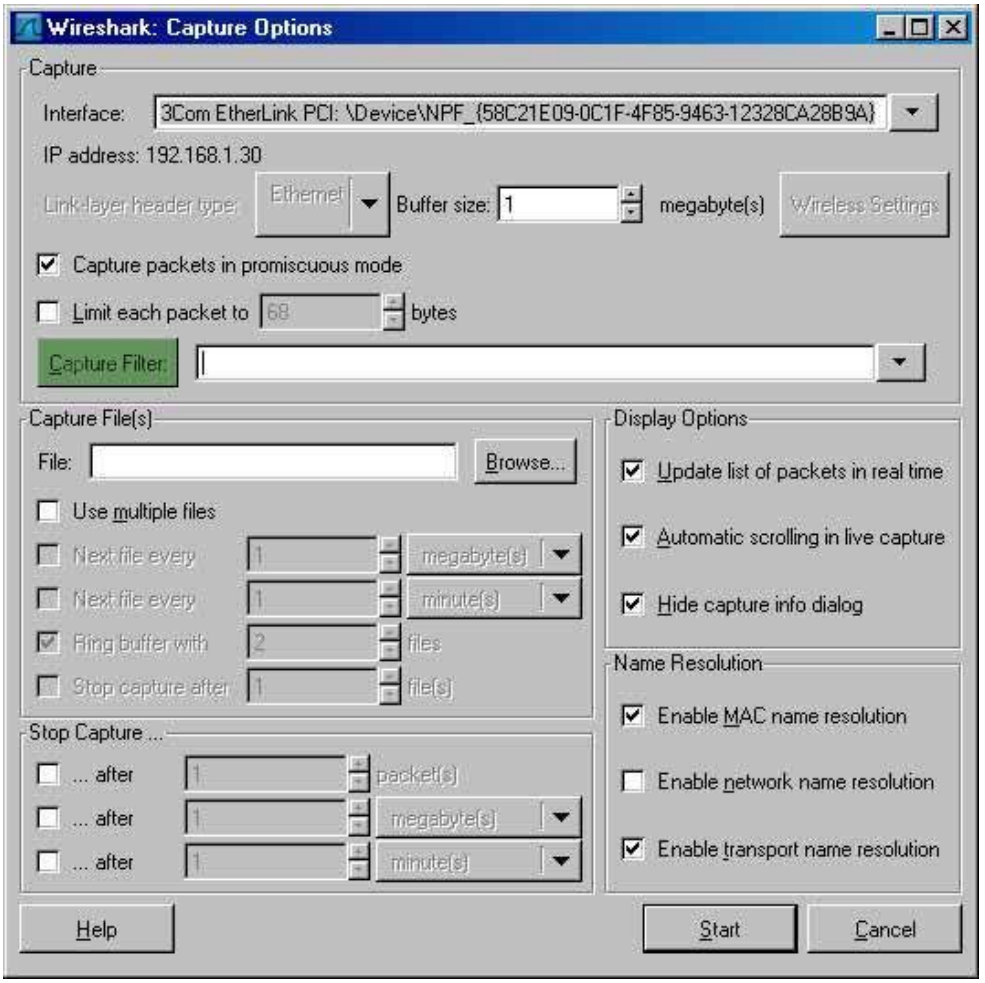
Los filtros de visualización (Display Filer) establecen un criterio de filtro sobre las paquetes capturados y que estamos visualizando en la pantalla principal de Wireshark. Estos filtros son más flexibles y potentes.

Filtros de Captura

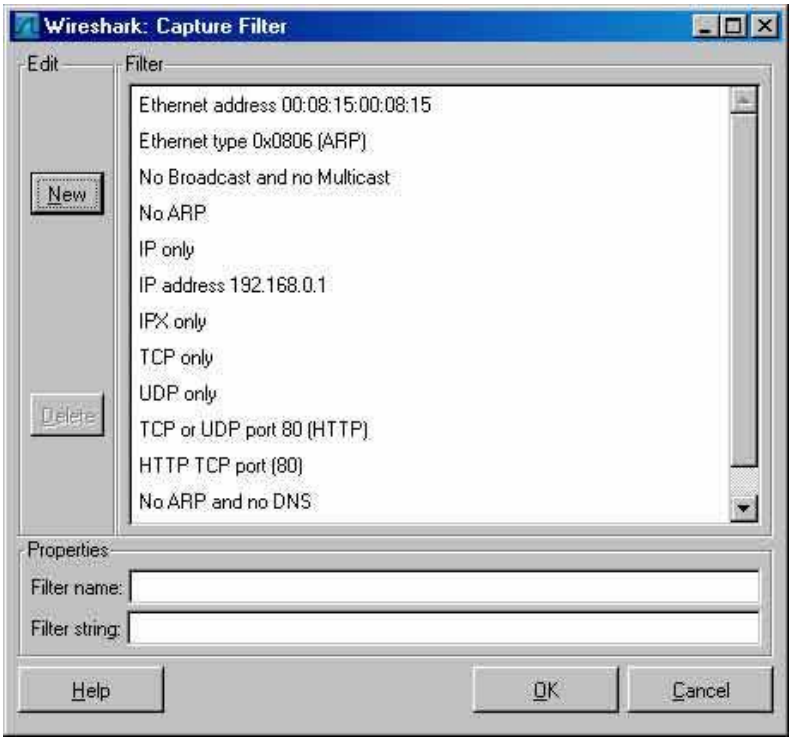
Estos filtros están basados en las librerías pcap. Los filtros son los mismos que podemos aplicar para Windump / TCPDump.

Si no establecemos ninguno, Wireshark capturará todo el tráfico y lo presentará en la pantalla principal. Aún así podremos establecer filtros de visualización (display filter) para que nos muestre solo el tráfico deseado.

Se aplican en Capture > Options:



En el campo Capture Filter introducimos el filtro o pulsamos el botón Capture Filter para filtros predefinidos:



Combinación de Filtros.

Podemos combinar las primitivas de los filtros de la siguiente forma:

- ❏ Negación: ! ó not
- ❏ Unión o Concatenación: && ó and
- ❏ Alternancia:|| ó or

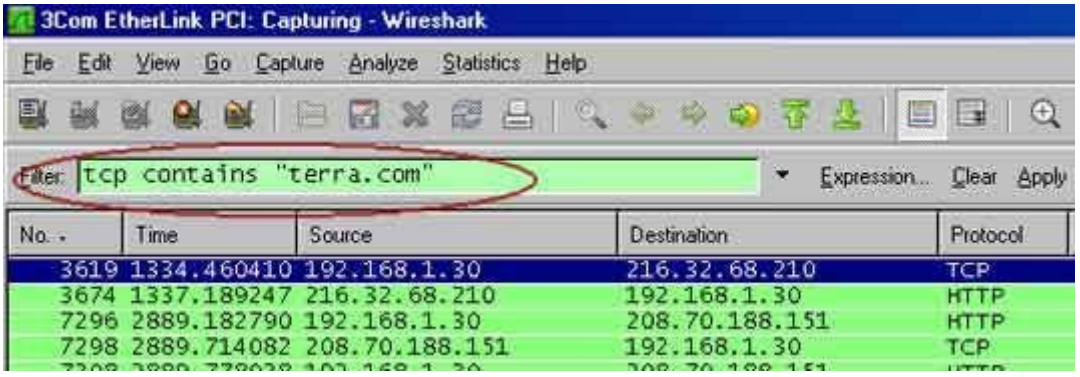
Vamos ahora a los filtros:

Filtros basados en hosts	
Ejemplo	Significado
host 192.168.1.20	Paquetes con origen o destino 192.168.1.20
src host 192.168.1.1	Paquetes con origen 192.1681.1
dst host 192.168.1.1	Paquetes con destino 192.168.1.1
Filtros basados en puertos	
Ejemplo	Significado
port 21	Paquetes con puerto origen o destino 21
src port 21	Paquetes con puerto origen 21
not port 21 and not port 80	Paquetes excepto origen y destino puertos 21 y 80
portrange 1-1024	Paquetes con puerto origen y destino en un rango de puertos 1 a 1024
dst portrange 1-1024	Captura todos los paquetes con puerto destino en un rango de puertos 1 a 1024
Filtros protocolos Ethernet / IP	
Ejemplos	
ip	Captura todo el tráfico IP
tcp	Captura todos los segmentos TCP
arp	Captura todo el tráfico ARP
Filtros basados en red	
Ejemplos	
net 192.168.1.0	Tráfico con origen y destino subred 1.0
net 192.168.1.0/24	Tráfico para la subred 1.0 máscara 255.0
dst net 192.168.2.0	Tráfico con destino para la subred 2.0
net 192.168.2.0 and port 21	Tráfico origen y destino puerto 21 en subred 2.0
broadcast	Solo el tráfico broadcast
not broadcast and not multicast	Todo el tráfico excepto broadcast y el multicast

Filtros de Visualización (Display Filter)

Los filtros de visualización establecen un criterio de filtro sobre las paquetes que estamos visualizando en la pantalla principal de Wireshark. Al aplicar el filtro en la pantalla principal de Wireshark aparecerá solo el tráfico filtrado a través del filtro de visualización.

Comparación		
Forma 1	Forma 2	Significado
==	eq	Igual a
!=	neq	Distinto a
>	gt	Mayor a
<	lt	Menor a
>=	ge	Mayor o igual a
<=	le	Menor o igual a
Combinación		
!	not	Negación (no)
&&	and	Unión (y)
||	or	Alternancia (uno u otro)
Otros operadores		
contains	Buscar una determinada cadena de caracteres	



Ejemplos	
Sintaxis	Significado
ip.addr == 192.168.1.40	Tráfico por host 192.168.1.40
ip.addr != 192.168.1.25	Todo excepto host 192.168.1.25
ip.dst == 192.168.1.30	Host destino 192.168.1.30

ip.src == 192.168.1.30	Host origen 192.168.1.30
ip	Todo el tráfico IP
tcp.port ==143	Origen o destino puerto TCP/143
ip.addr == 8.8.8.8 and tcp.port == 80	Origen y destino puerto TCP/80 y host 8.8.8.8
http contains "www.terra.com"	Paquetes HTTP que contienen www.terra.com
frame contains "@miempresa.es"	Correos con origen o destino al dominio miempresa.es.
icmp[0:1] == 08	Tráfico icmp de tipo echo request
ip.ttl == 1	Paquetes IP cuyo campo TTL sea igual a 1
tcp.windows_size != 0	Paquetes cuyos campo Tamaño de Ventana del segmento TCP sea distinto de 0
ip.tos == x	Paquetes IP cuyo campo TOS sea igual a x
ip.flags.df == x	Paquetes IP cuyo campo DF sea igual a x
udp.port == 53	Visualiza todo el tráfico UDP puerto 53
tcp contains "terra.com"	Segmentos TCP conteniendo la cadena terra.com

---

Autor: Fabian Martinez Portantier

Fuentes:

- http://incibe.es
- http://seguridadyredes.wordpress.com
- http://nmap.org