

Taller de Wireshark

Teoría de las Comunicaciones

Departamento de Computación
Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires

08.04.2015

Objetivos

Presentar:

- Wireshark.
- ARP.
- Scapy.
- TP1.

¿Qué es Wireshark?

- Wireshark es un capturador de paquetes/protocolos de red (aka: sniffer).
- Además, parsea paquetes capturados por una interfaz y los muestra con un alto grado de detalle.
- Se usa fundamentalmente como herramienta de diagnóstico de networking: es un “debugger” de la red.
- El mejor amigo del administrador de red, analista de seguridad, programador, hacker, etc.
- Es libre, abierto y gratis.

Algunas definiciones

- ¿NIC? Network Interface Controller (wlan0, eth0, lo, prueben haciendo ifconfig).

```
$ ifconfig
eth0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
ether 3c:92:0e:33:4b:01 txqueuelen 1000 (Ethernet)
RX packets 0 bytes 0 (0.0 B)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 0 bytes 0 (0.0 B)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Algunas definiciones, cont.

Modo promiscuo

lo que significa que los paquetes con MAC destino ajena no se descartan. Suben hasta el kernel para que podamos consumir las tramas. **Igual veríamos mensajes broadcast, multicast y unicast.**

Modo monitor

lo que permite capturar tráfico por la WNIC, ya estemos asociados o no con el AP o la red Ad-Hoc, sin que este sea descartado.

Algunas definiciones, cont. 2

capabilites

Starting with kernel 2.2, Linux divides the privileges traditionally associated with superuser into distinct units, known as capabilities, which can be independently enabled and disabled. Capabilities are a per-thread attribute.

CAP_NET_ADMIN

Permite

- Allow interface configuration
- Allow modification of routing tables
- Allow setting promiscuous mode

Algunas definiciones, cont. 3

CAP_NET_RAW

Permite emitir:

Raw frames permiten escribir los headers de la capa física

Packet frames obtienen los parámetros de la capa física

Ambos permiten escribir frames con los headers de capa 2 en adelante.

Captura de paquetes, pero... ¿cómo?

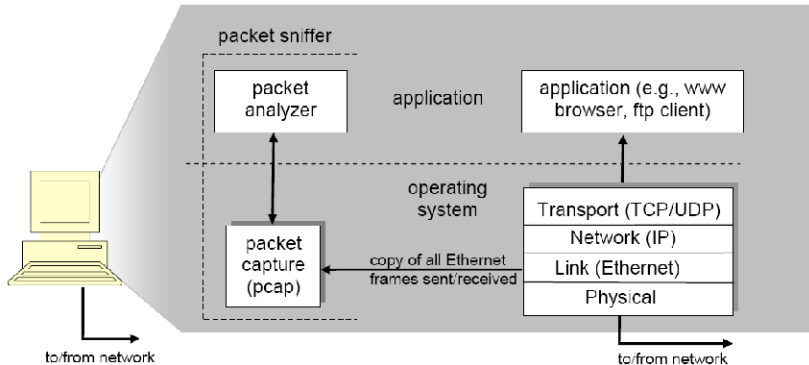


Figure 1: Packet sniffer structure

leer mas: <http://www.tcpdump.org/faq.html>

Para

Escenarios

Local

- loopback
- eth, wlan, etc

Red local

- Atrás de un hub. Todos los mensajes se floodean.
- Atrás de un switch. No podemos ver mensajes ajenos. (Salvo que...)

¿Dónde estamos parados?

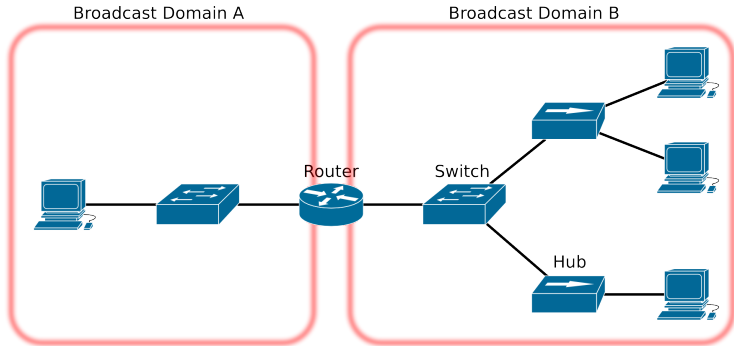
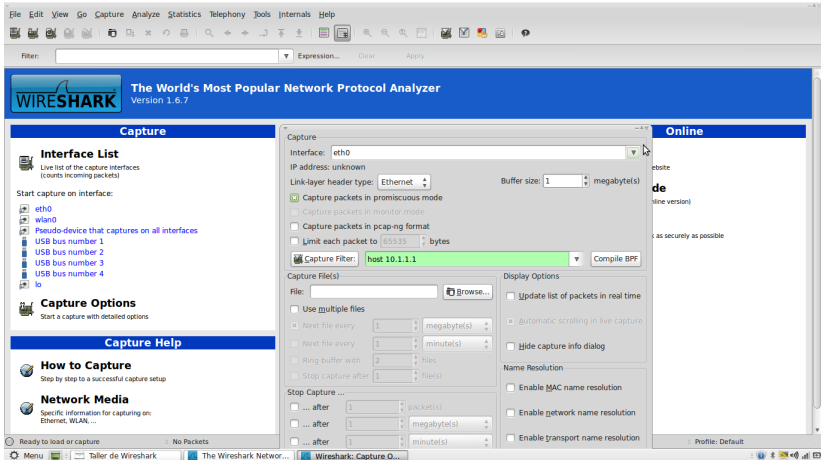


Figura : Mismo dominio de broadcast, mismo segmento de red

Wireshark 1



Filtros

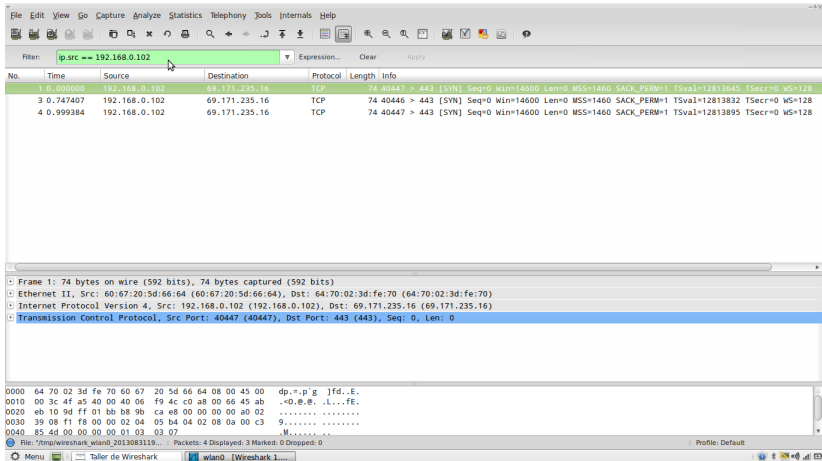
- Es demasiada información, necesitamos poder manejarla.

Ejemplos

- **broadcast ethernet:** `eth.dst == FF:FF:FF:FF:FF:FF`
- **ethernet type:** `eth.type == 0xFFFF` (2 bytes)
- **ether src ehost:** `eth.src == 90:4c:e5:bb:e0:d6`
- **ip src:** `ip.src == 192.168.1.1`
- **ip protocol:** `ip.proto == 1`
- etc. Ver secciones Expression y Filter en la barra de filtro.

Recomendado: <http://biot.com/capstats/bpf.html>

Wireshark 2





Algunas precauciones



Algunas precauciones

- Capa 2.5.
- Todavía no vimos IP.
- Nos estamos adelantando un poco.

Ethernet - MAC Address

- *Media Access Control Address.*

Ethernet - MAC Address

- *Media Access Control Address.*
- Identificador de una interfaz de red.

Ethernet - MAC Address

- *Media Access Control Address.*
- Identificador de una interfaz de red.
- 6 octetos

Ethernet - MAC Address

- *Media Access Control Address.*
- Identificador de una interfaz de red.
- 6 octetos
- 3 de OUI (Organization Unique Identifier)
`standards.ieee.org/develop/regauth/oui/public.html`

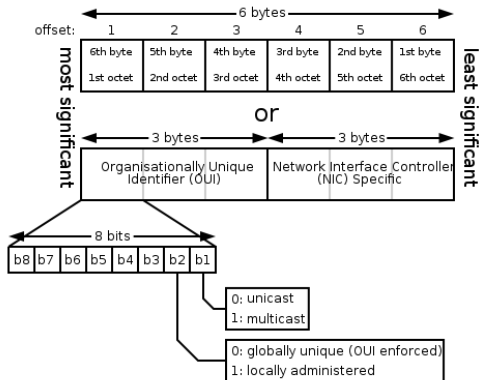
Ethernet - MAC Address

- *Media Access Control Address.*
- Identificador de una interfaz de red.
- 6 octetos
- 3 de OUI (Organization Unique Identifier)
standards.ieee.org/develop/regauth/oui/public.html
- 3 de NIC (Network Interface Controller)

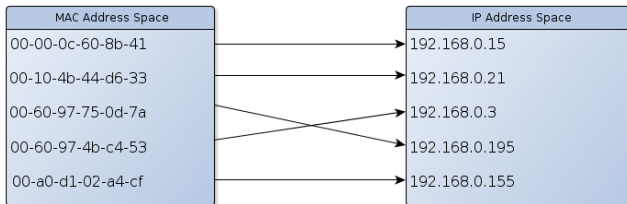
Ethernet - MAC Address

- *Media Access Control Address.*
- Identificador de una interfaz de red.
- 6 octetos
- 3 de OUI (Organization Unique Identifier)
standards.ieee.org/develop/regauth/oui/public.html
- 3 de NIC (Network Interface Controller)
- Intel Corporate: 00:1c:c0:fa:55:cc

Ethernet - MAC Address cont.



¿Perdón?



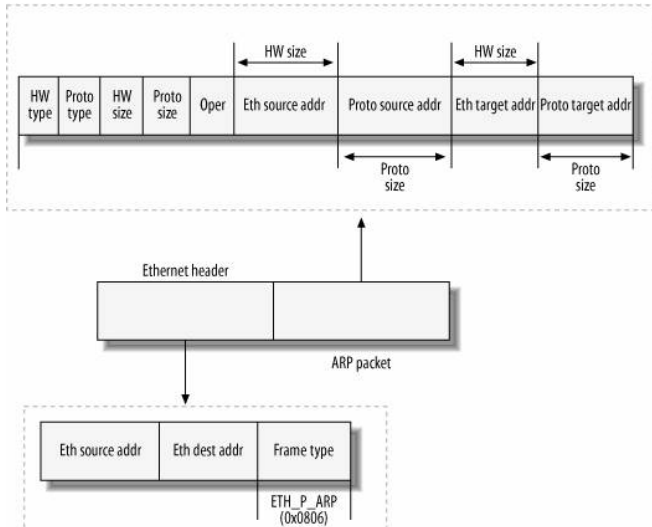
¿Qué es ARP?

- La sigla: *Address Resoution Protocol*.
- Es un protocolo que, en esencia, permite mapear direcciones de nivel de red a direcciones físicas.
- Clave e indispensable en el funcionamiento de las redes modernas.
- Especificado en el RFC 826 (circa 1982).
- No está limitado a IP + Ethernet: la especificación es general.

Tecnicismos varios

- La pregunta ARP consiste en un mensaje **broadcast** sobre la red local.
 - Recordar que no se propaga más allá de la red local!
- La respuesta, en cambio, es **unicast**.
- Optimización: se implementa una caché para guardar las direcciones resueltas (o conocidas).
 - Las entradas se agregan al resolver o bien al observar un pedido de otra máquina.
 - Cada entrada tiene un tiempo de expiración para evitar problemas.

Pormenores del paquete

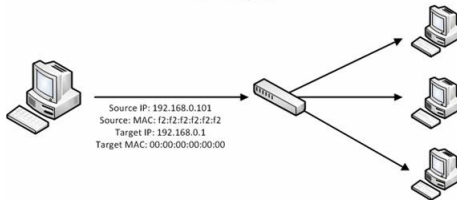


Pormenores del paquete (cont.)

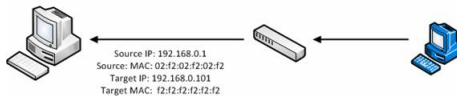
- El campo **Oper** puede tomar los valores 1 (who-has) o 2 (reply).
- Observar que la cantidad de bits asignada a las direcciones depende del valor que tomen los campos **HW size** y **Proto size**.
- Dichos campos tienen un largo de 8 bits (i.e., direcciones con un máximo de $2^8 - 1 = 255$ bits).
- **HW type** y **Proto type** indican los protocolos de nivel de enlace y de nivel de red respectivamente involucrados en la comunicación.

¿Cómo funciona?

ARP Request



ARP Response



Otro uso interesante

- Cuando una máquina bootea o se levanta una de sus interfaces, muchos SOs envían automáticamente un pedido ARP *gratuito*.
- En él, **Proto source addr == Proto target addr**.
- Objetivos:
 - Detectar IPs duplicadas en la red local: esto ocurre si se recibe una respuesta.
 - Actualizar la caché ARP de los otros hosts.

...y otro uso más: ARP Spoofing

Spoofing

- ① To deceive.
- ② To do a spoof of; satirize gently.

- De lo anterior se desprende que ARP es un protocolo **sin estado** y **sin seguridad**.
- La técnica de ARP spoofing se apoya precisamente en estas características.
- Idea: una máquina envía de la nada una respuesta ARP mapeando una IP objetivo con su propia MAC.
- \Rightarrow todo el tráfico destinado a dicha IP va a ser recibido por ella.

Intro a Scapy

- Scapy es un framework de manipulación de paquetes.

Intro a Scapy

- Scapy es un framework de manipulación de paquetes.
- Permite crear paquetes, capturar paquetes, enviar paquetes, analizar paquetes, etc.

Intro a Scapy

- Scapy es un framework de manipulación de paquetes.
- Permite crear paquetes, capturar paquetes, enviar paquetes, analizar paquetes, etc.
- Orientado a capas. `pkt = Ether() / IP() / TCP()` nos genera un paquete TCP válido.

Transmitiendo

```
#!/usr/bin/env python
# arping2tex : arpings a network and outputs a LaTeX table as a result

import sys
if len(sys.argv) != 2:
    print "Usage: _arping2tex_ <net>\n_\neg: _arping2tex_ 192.168.1.0/24"
    sys.exit(1)

from scapy.all import srp, Ether, ARP, conf
conf.verb=0
ans,unans=srp(Ether(dst="ff:ff:ff:ff:ff:ff")/ARP(pdst=sys.argv[1]),
              timeout=2)

print r"\begin{tabular}{|l|l|l|}"
print r"\hline"
print r"MAC_\_IP\_"
print r"\hline"
for snd,rcv in ans:
    print rcv.sprintf(r"%Ether.src %_\_ARP.psrc %\\")
print r"\hline"
print r"\end{tabular}"
```

Escuchando

```
#!/usr/bin/env python
from scapy.all import *
def monitor_callback(pkt):
    print pkt.show()

if __name__ == '__main__':
    sniff(prn=monitor_callback, filter="arp", store=0)
```

Trabajos Prácticos

¿Cómo son los Trabajos Prácticos?

- 2 Trabajos Prácticos
 1. TP1: Wiretapping (Information Gathering)
 2. TP2: ICMP (Rutas en Internet)
- Objetivos
 1. Experimentar con la red (no siempre es lo que parece).
 2. Analizar comportamientos esperados y no esperados.
 3. Enmarcar el análisis en un informe (o *tech rep*).

¿Qué esperamos que hagan?

- Que reflexionen sobre lo que es una red.
- Que se vayan con herramientas prácticas para hacer diagnóstico.
- Que profundicen la comprensión de los conceptos a partir de su aplicación.
- Que entreguen informes rigurosos sobre lo que experimentaron.

Dinámica de presentación y entrega.

- 3 o 4 integrantes.
- Entrega por mail. Fechas:
 - ① TP1: 29/04/2015
 - ② TP2: ?
- Pautas para los informes.
 - ① Tener en cuenta la estructura de informe científico (*introducción, experimentos, resultados, conclusiones*).
 - ② El código no es tan importante.
 - ③ Ojo con las figuras. Que sean claras y tengan leyendas.
- Template (*recomendado*):
<http://mocha-java.uccs.edu/ieee/>

¡A trabajar!: Primera consigna

- 1 Implementar una herramienta para escuchar pasivamente en la red local.
- 2 La siguiente fuente de información distingue los protocolos que se encapsulan en todos los paquetes ethernet de una red: $S = \{s_1 \cdots s_n\}$ siendo s_i el valor del campo *type* del frame de capa 2. A partir de S intentaremos caracterizar cuáles son los protocolos más utilizados en una red dada. A partir de ello se pide:

¡A trabajar!: Primera consigna

- (a) Proponga una nueva fuente S_1 con el nuevo objetivo de distinguir nodos de la red en base *solamente* al tráfico ARP. Esto significa que los símbolos de S_1 tienen que estar conformados por campos del paquete ARP.
- (b) Adapte la *tool* del inciso (a) para estimar las probabilidades de las fuentes S y S_1 en función de los paquetes observados y calcular la entropía de cada fuente.
- (c) Utilizando la herramienta realizar capturas (lo más extensas posibles) de paquetes en alguna red de acceso compartido. En la medida de lo posible, intentar capturar en una red que no sea controlada (en el trabajo, en un shopping, etc.). Debe haber exactamente una captura por cada integrante del grupo.

¡A trabajar!: Segunda consigna

Realizar un análisis que permita, para cada una de las redes estudiadas:

- (a) Determinar los protocolos distinguidos y analizar el overhead impuesto por ARP.
- (b) Determinar los nodos distinguidos.

Los resultados de esta consigna deben estar basados en la teoría de la información. O sea, deben analizar qué símbolos son estadísticamente significativos en cada LAN analizando la información de cada símbolo con respecto a la entropía de su respectiva fuente. La presentación de los resultados debe efectuarse mediante gráficos y su correspondiente análisis. Sugerimos, entre otros, histogramas (de IPs y protocolos) con cortes en los valores de entropía.

Referencias

- <http://www.tcpdump.org/papers/bpf-usenix93.pdf>
- <http://biot.com/capstats/bpf.html>
- **man capabilities**
- **man packet**