UNIVERSIDADE DA CORUÑA

Sniffing

LSI

2012-2013

Contenido

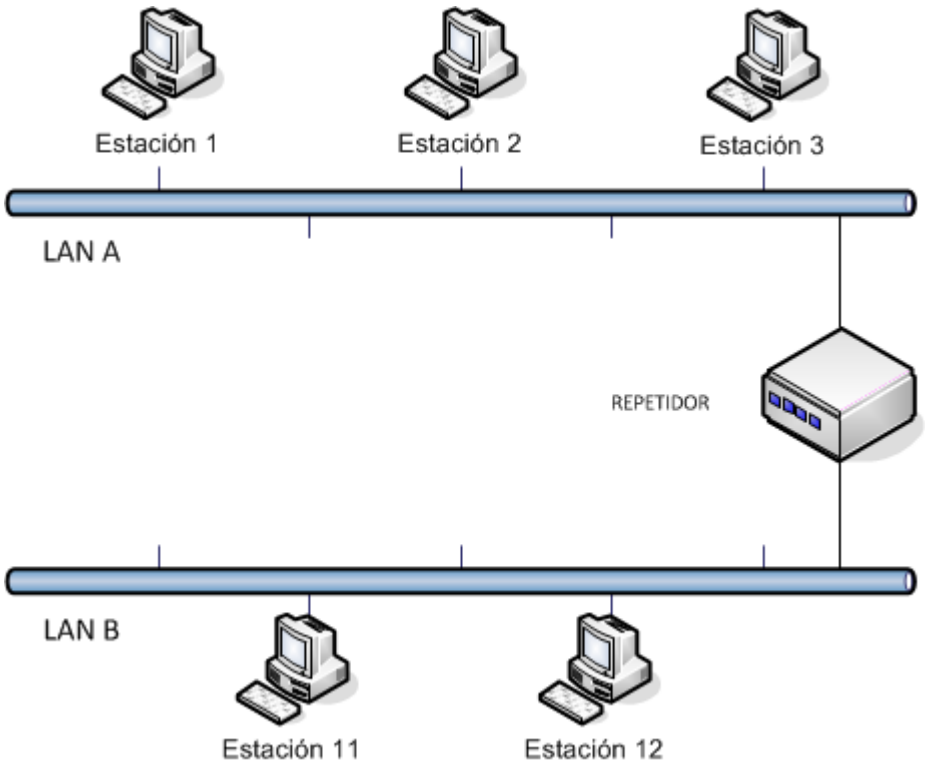
- Sniffing
  - Conceptos básicos
  - Sniffing en redes de medio compartido
    - Funcionamiento
    - Protección
  - Sniffing en redes de medio conmutado
    - Funcionamiento y métodos
    - Protección

Conceptos básicos

- Legacy Ethernet
  - Implementaciones antiguas de Ethernet basadas en cable coaxial y protocolo de acceso al medio CSMA / CD.

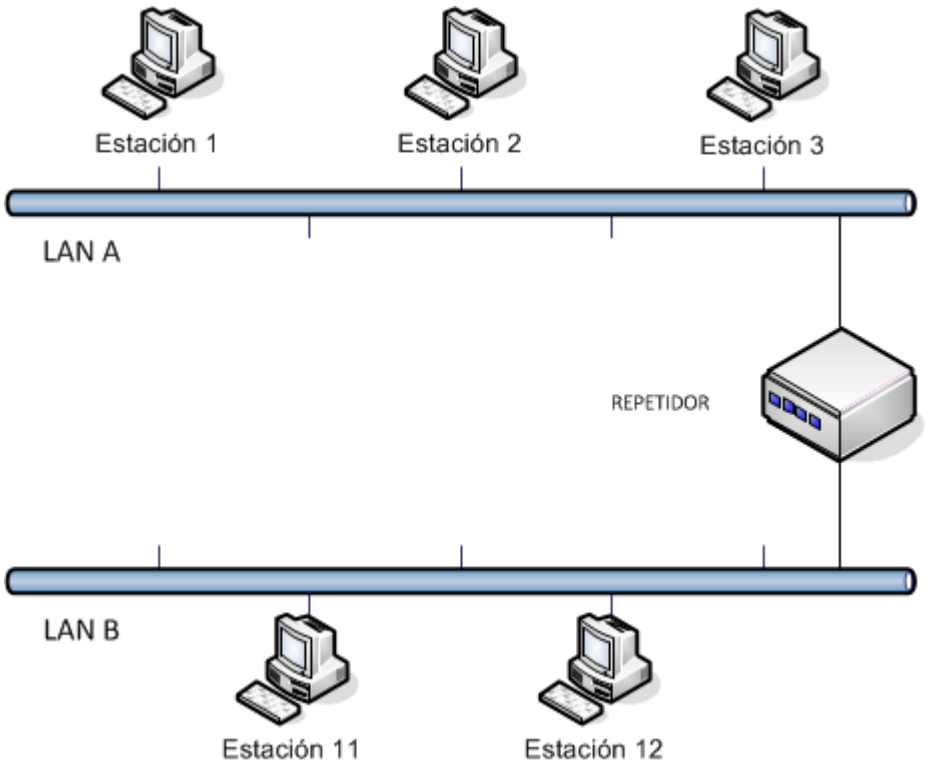
Conceptos básicos

- Legacy Ethernet
  - Repetidor
  - Concentrador (HUB) o repetidor multipuerto



Conceptos básicos

- Legacy Ethernet
  - Repetidor
  - Concentrador (HUB) o repetidor multipuerto



Dominio de colisión: Es el conjunto de dispositivos que comparten un medio físico de transmisión y que, por lo tanto, compiten por él.

¿Y Wi-Fi?

- Medio compartido.
- Se utiliza **Carrier sense multiple access with collision avoidance** (**CSMA/CA**)
- Se rigen por el estándar 802.11
- Especialmente vulnerables
  - Cualquiera puede escuchar
- Seguridad:
  - WEP
  - WPA
  - WPA2

WEP (Wired Equivalent Privacy)

- Se basa en clave única y estática
- El equipo necesita la clave para autenticarse ante el punto de acceso (AP)
- La comunicación se cifra usando esta clave más un Vector de Inicialización (IV):
  - clave de cifrado = clave WEP + IV
- Al enviar la trama, se envía el IV, para que el receptor pueda descifrarla, si conoce la clave WEP.
- RC4 como algoritmo de cifrado.
- Problemas:
  - IV es demasiado pequeño (24 bits). Acaba por repetirse después de un número no muy grande de tramas => capturando tramas con el mismo IV, se puede descifrar la clave WEP
  - RC4, en la forma en que lo usa WEP, puede romperse

WPA (Wi-Fi Protected Access)

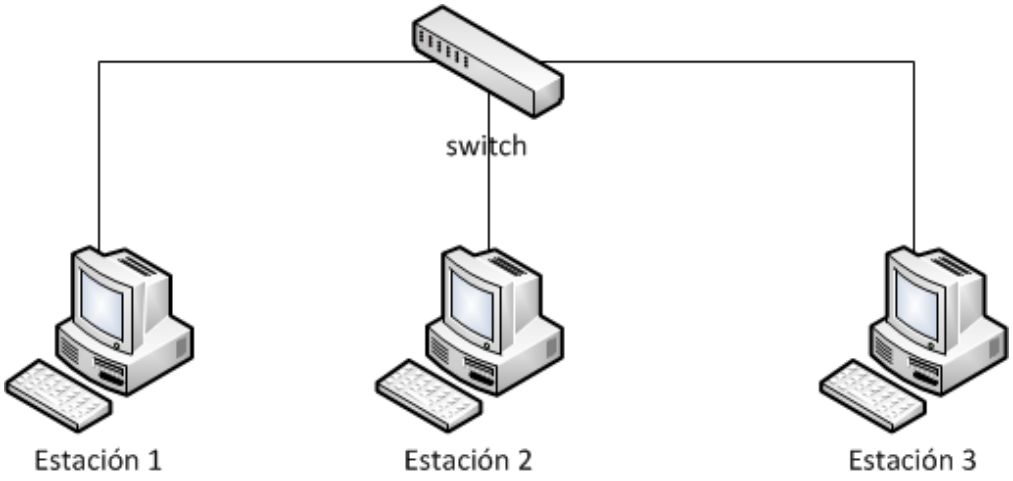
- Usa claves dinámicas en lugar de clave estática.
  - Algoritmo TKIP (Temporary Key Integrity Protocol) <= roto
- RC4 como algoritmo de cifrado (corrigiendo las deficiencias de WEP).
- Compatible con equipos existentes.

WPA2 (802.11i)

- AES (**A**dvanced **E**ncryption **S**tandard) - CCMP (**C**ounter Mode **C**BC **M**AC **P**rotocol) como algoritmo de cifrado.
  - Counter Mode: dificulta encontrar patrones
  - CBC-MAC (Cipher Block Chaining-Message Authentication Code): proporciona integridad
- Requiere mucha más carga de computación => nuevo HW.
- Desde 2006, todos los productos Wi-Fi Certified deben usar WPA2

Ethernet Conmutado

- Nuevos dispositivos: Puente (bridge) y conmutador (switch)
  - Trabajan en capa 2, tomando decisiones sobre el envío de tramas en base a direcciones MAC.
  - Cada dirección MAC conocida está asociada a un puerto.
  - Varios dominios de colisión (cada puerto es un dominio de colisión).
  - Un dominio de difusión o broadcast.



Dominio de difusión o dominio de broadcast

- Es el conjunto de dispositivos pertenecientes a una red tal que si uno de ellos genera un mensaje de difusión (e.g. IP de destino de un paquete IP 255.255.255.255) todos los demás equipos de dicho dominio reciben y procesan el mensaje.
- Este tipo de tráfico es utilizando en muchas ocasiones para facilitar las tareas de comunicación entre equipos:
  - Peticiones ARP, peticiones DHCP, …
- Un dominio de difusión o de broadcast se define habitualmente como "red lógica" por lo que habitualmente se asocia una dirección IP de red/subred lógica a cada dominio de difusión o broadcast.
  - Los equipos que pertenecen a un mismo dominio de difusión, y por lo tanto pertenecen a la misma red lógica, pueden comunicarse directamente entre sí
  - Los equipos que pertenecen a distintos dominios de difusión necesitan los servicios de las pasarelas o puertas de enlace (que habitualmente son routers) para intercambiar información entre distintos dominios de difusión

Ethernet Conmutado

- Nuevos dispositivos: encaminador (router)
  - Dispositivo de interconexión de redes de capa 3, cuyo objetivo es conectar redes lógicas diferentes. Estos dispositivos realizan dos funciones fundamentales:
    - Determinación de la ruta (enrutamiento).
    - Transmisión del paquete por la interfaz adecuada (conmutación).
  - Varios dominios de difusión o broadcast.

Repaso de conceptos fundamentales

- En una red de área local, formada por hubs, switches y dispositivos finales, únicamente existe un dominio de difusión, por lo que todos los equipos estarán además configurados en la misma red IP
  - Inconvenientes:
    - Saturación de tráfico de difusión en la red
    - Falta de control interno en las comunicaciones
- ¿Qué se puede hacer para segmentar un dominio difusión en varios más pequeños, con el tráfico de difusión de cada uno aislado?
  - Utilización de routers
  - Creación de VLAN

VLAN

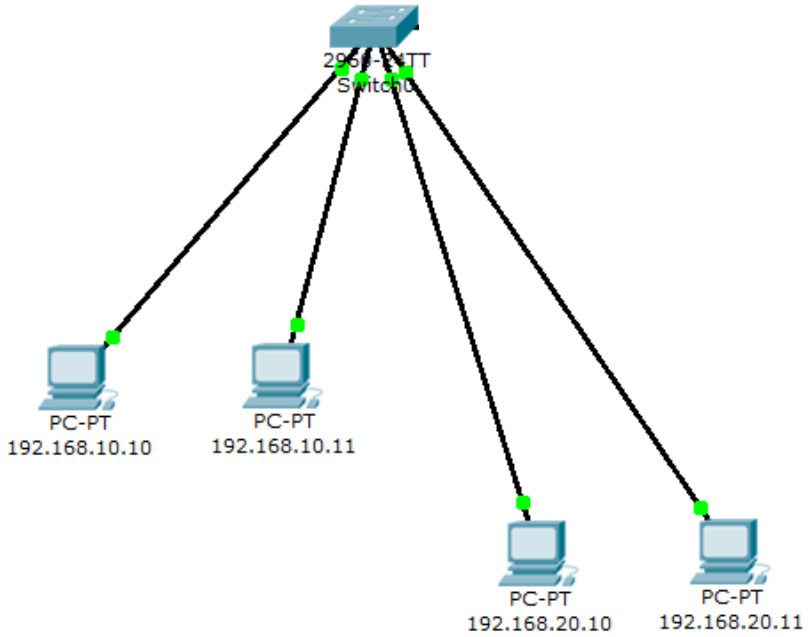
- ¿Qué es una VLAN?
  - Es una agrupación lógica de dispositivos que se basa en la configuración de switches, de tal modo que se pueden crear en un switch (o conjunto de switches) diferentes dominios de difusión, asignando cada puerto del switch a una agrupación (VLAN) concreta

VLAN

- ¿Qué es una VLAN?
  - Es una agrupación lógica de dispositivos que se basa en la configuración de switches, de tal modo que se pueden crear en un switch (o conjunto de switches) diferentes dominios de difusión, asignando cada puerto del switch a una agrupación (VLAN) concreta
  - Los criterios que permiten determinar a que VLAN está asignado un puerto pueden ser muy diferentes:
    - Configuración estática del puerto
    - En función de la dirección IP del dispositivo conectado al puerto
    - En función de la dirección MAC del dispositivo conectado al puerto
    - En función del usuario conectado al puerto (IEEE 802.1x)
    - ...

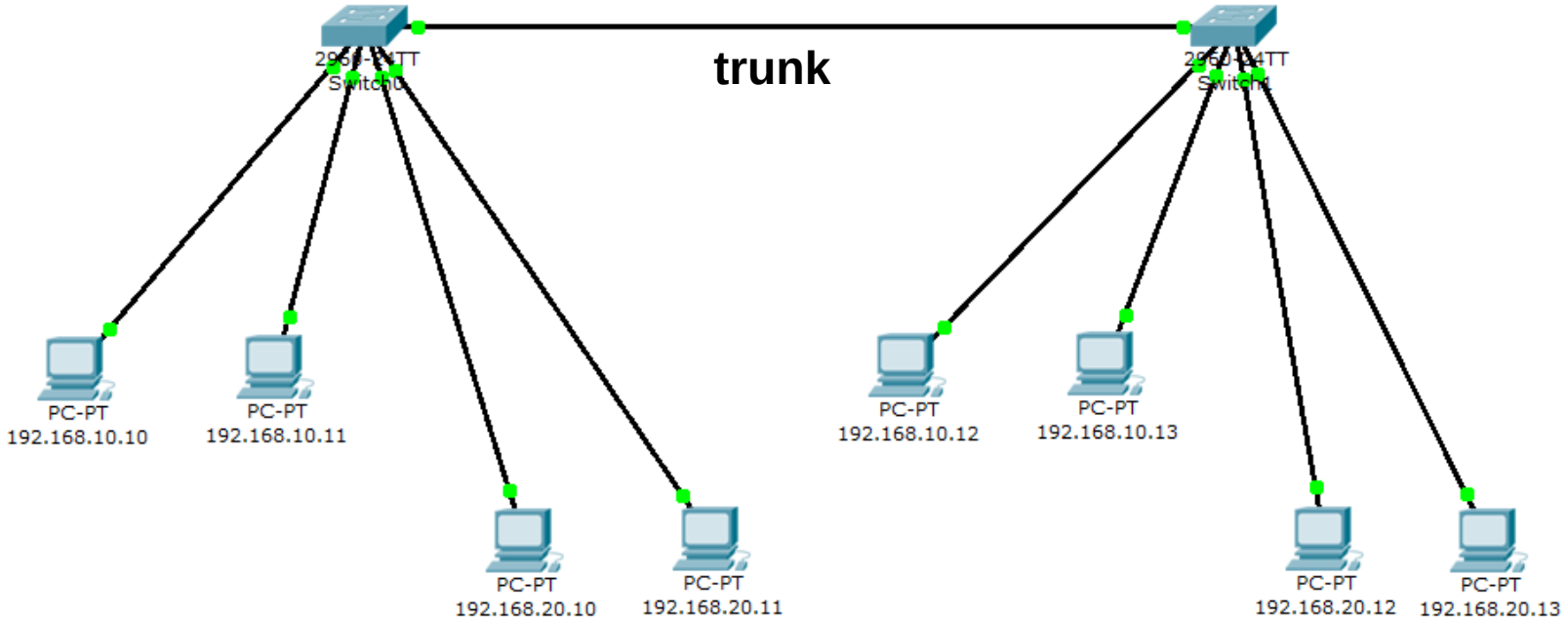
VLAN

- Ejemplo:
  - Los dispositivos de la VLAN 1 pertenecen a la red 192.168.10.0/24
  - Los dispositivos de la VLAN 2 pertenecen a la red 192.168.20.0/24



VLAN

- Ejemplo:
  - Los dispositivos de la VLAN 1 pertenecen a la red 192.168.10.0/24
  - Los dispositivos de la VLAN 2 pertenecen a la red 192.168.20.0/24



- Para unir VLANs que están definidas en varios switches se puede crear un enlace especial llamado **trunk**, por el que fluye tráfico de varias VLANs.

VLAN

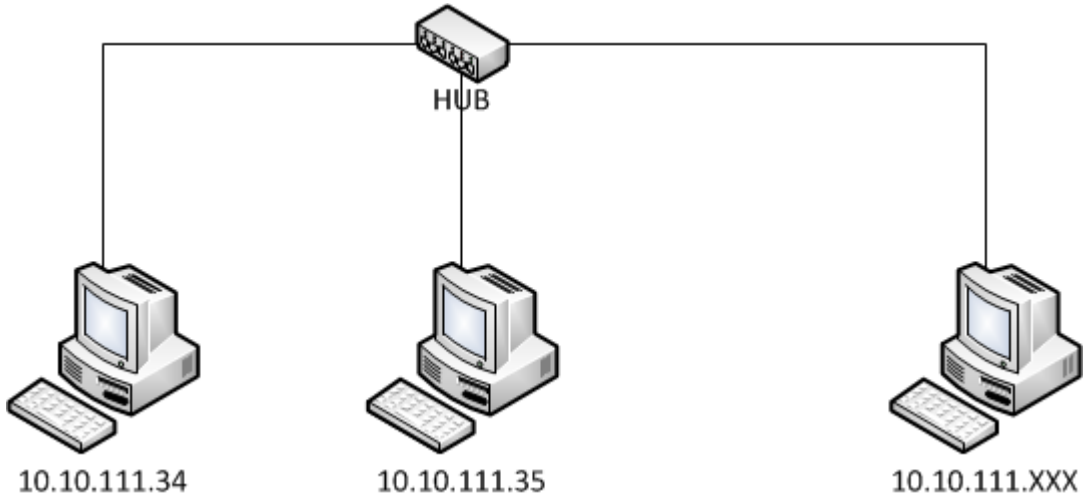
- Los puertos que están asignados a una única VLAN, se conocen como puertos de acceso
- Los puertos que están asignados a varias VLANs (enlace trunk), se conocen como puertos troncales
  - ¿Cómo sabe un switch a que VLAN pertenece una trama cuando la recibe por un puerto troncal?
    - Las tramas se etiquetan antes de ser transmitidas por el enlace troncal
    - El estándar IEEE 802.1Q permite añadir una etiqueta de 4 bytes a la cabecera de las tramas Ethernet, en donde se incluye el nº de VLAN al que pertenece dicha trama

Ventajas de las VLAN

- Gran flexibilidad => Permite segmentar por proyecto, función, departamento, …
- Reducción del tráfico de broadcast => mayor ancho de banda
- Mayor seguridad

Sniffing en redes de medio compartido

- Cada paquete se envía a todos los puertos, de modo que todos los equipos reciben todos los paquetes



- Las redes Wi-Fi se "asemejan" a este escenario (red de cable con topología HUB)

Modo "promiscuo"

- En un medio compartido, la NIC, en modo normal sólo "acepta" los paquetes destinados a su dir. MAC
- En modo "promiscuo", la NIC "acepta" todos los paquetes, correspondan o no con su dir. MAC

- En Wi-Fi, existe el modo "monitor", que permite a la NIC capturar paquetes sin asociarse con el punto de acceso

Herramientas

- Tcpdump
- Wireshark
- Ettercap
- …

Herramientas. Tcpdump

- Herramienta de línea de comandos cuya utilidad principal es analizar el tráfico que circula por la red.
- Funciona en la mayoría de los sistemas operativos UNIX, utilizando libpcap.
  - Hay una adaptación para Windows, WinDump, que utiliza WinPcap
- Web:
  - http://www.tcpdump.org/
- Instalación (Linux):
  - apt-get install tcpdump

Herramientas. Tcpdump

- Ejemplos:

```
-- captura tráfico tcp e imprime cada paquete en hexadecimal y ASCII (-X)
-- vv: verbose
-- protocolos soportados:  fddi, tr, wlan, ip, ip6, arp, rarp, decnet, tcp y udp
tcpdump tcp -X -vv
```

Herramientas. Tcpdump

- Ejemplos:

```
-- captura tráfico tcp e imprime cada paquete en hexadecimal y ASCII (-X)
-- vv: verbose
-- protocolos soportados:  fddi, tr, wlan, ip, ip6, arp, rarp, decnet, tcp y udp
tcpdump tcp -X -vv

-- captura tráfico del puerto 80
tcpdump port http
```

Herramientas. Tcpdump

- Ejemplos:

```
-- captura tráfico tcp e imprime cada paquete en hexadecimal y ASCII (-X)
-- vv: verbose
-- protocolos soportados:  fddi, tr, wlan, ip, ip6, arp, rarp, decnet, tcp y udp
tcpdump tcp -X -vv

-- captura tráfico del puerto 80
tcpdump port http

-- envía la captura a un archivo de log
tcpdump -w capture.pcap
```

Herramientas. Tcpdump

- Ejemplos:

```
-- captura tráfico tcp e imprime cada paquete en hexadecimal y ASCII (-X)
-- vv: verbose
-- protocolos soportados:  fddi, tr, wlan, ip, ip6, arp, rarp, decnet, tcp y udp
tcpdump tcp -X -vv

-- captura tráfico del puerto 80
tcpdump port http

-- envía la captura a un archivo de log
tcpdump -w capture.pcap

-- lee un archivo de log (tb. se puede util. Wireshark)
tcpdump -r capture.pcap
```

Herramientas. Tcpdump

- Ejemplos:

```
-- captura tráfico tcp e imprime cada paquete en hexadecimal y ASCII (-X)
-- vv: verbose
-- protocolos soportados:  fddi, tr, wlan, ip, ip6, arp, rarp, decnet, tcp y udp
tcpdump tcp -X -vv

-- captura tráfico del puerto 80
tcpdump port http

-- envía la captura a un archivo de log
tcpdump -w capture.pcap

-- lee un archivo de log (tb. se puede util. Wireshark)
tcpdump -r capture.pcap

-- mostrar los paquetes con destino www.openmaniak.com
tcpdump host www.openmaniak.com
```

Herramientas. Tcpdump

- Ejemplos:

```
-- captura tráfico tcp e imprime cada paquete en hexadecimal y ASCII (-X)
-- vv: verbose
-- protocolos soportados:  fddi, tr, wlan, ip, ip6, arp, rarp, decnet, tcp y udp
tcpdump tcp -X -vv

-- captura tráfico del puerto 80
tcpdump port http

-- envía la captura a un archivo de log
tcpdump -w capture.pcap

-- lee un archivo de log (tb. se puede util. Wireshark)
tcpdump -r capture.pcap

-- mostrar los paquetes con destino www.openmaniak.com
tcpdump host www.openmaniak.com

-- mostrar los paquetes ftp con el origen y destino indicados
tcpdump src 192.168.1.100 and dst 192.168.1.2 and port ftp
```

Herramientas. Tcpdump

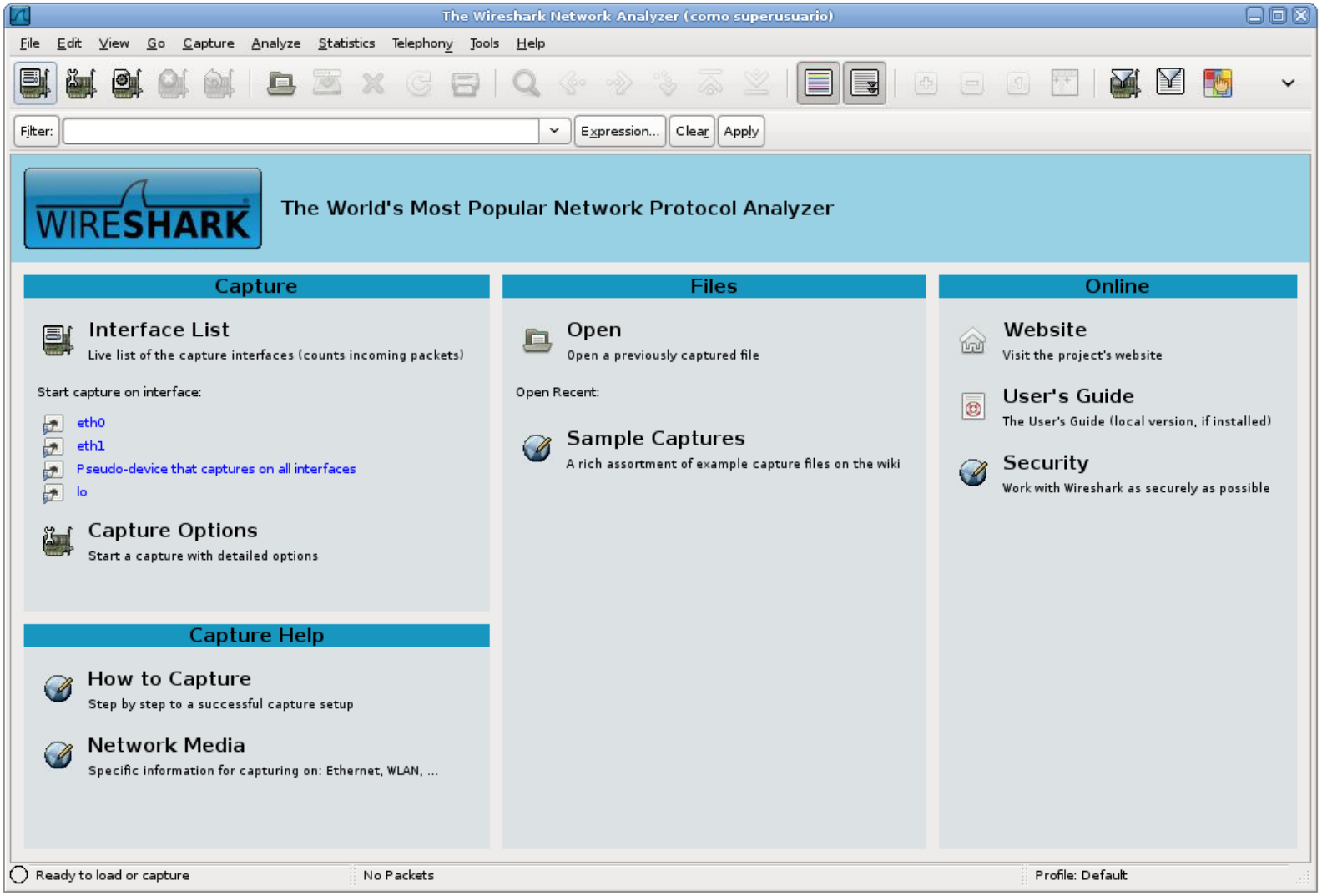
- Ejemplos:

```
-- captura tráfico tcp e imprime cada paquete en hexadecimal y ASCII (-X)
-- vv: verbose
-- protocolos soportados:  fddi, tr, wlan, ip, ip6, arp, rarp, decnet, tcp y udp
tcpdump tcp -X -vv

-- captura tráfico del puerto 80
tcpdump port http

-- envía la captura a un archivo de log
tcpdump -w capture.pcap

-- lee un archivo de log (tb. se puede util. Wireshark)
tcpdump -r capture.pcap

-- mostrar los paquetes con destino www.openmaniak.com
tcpdump host www.openmaniak.com

-- mostrar los paquetes ftp con el origen y destino indicados
tcpdump src 192.168.1.100 and dst 192.168.1.2 and port ftp

-- capturar los paquetes dirigidos a un puerto que lleguen por la interfaz eth0
tcpdump -i eth0 port 22
```

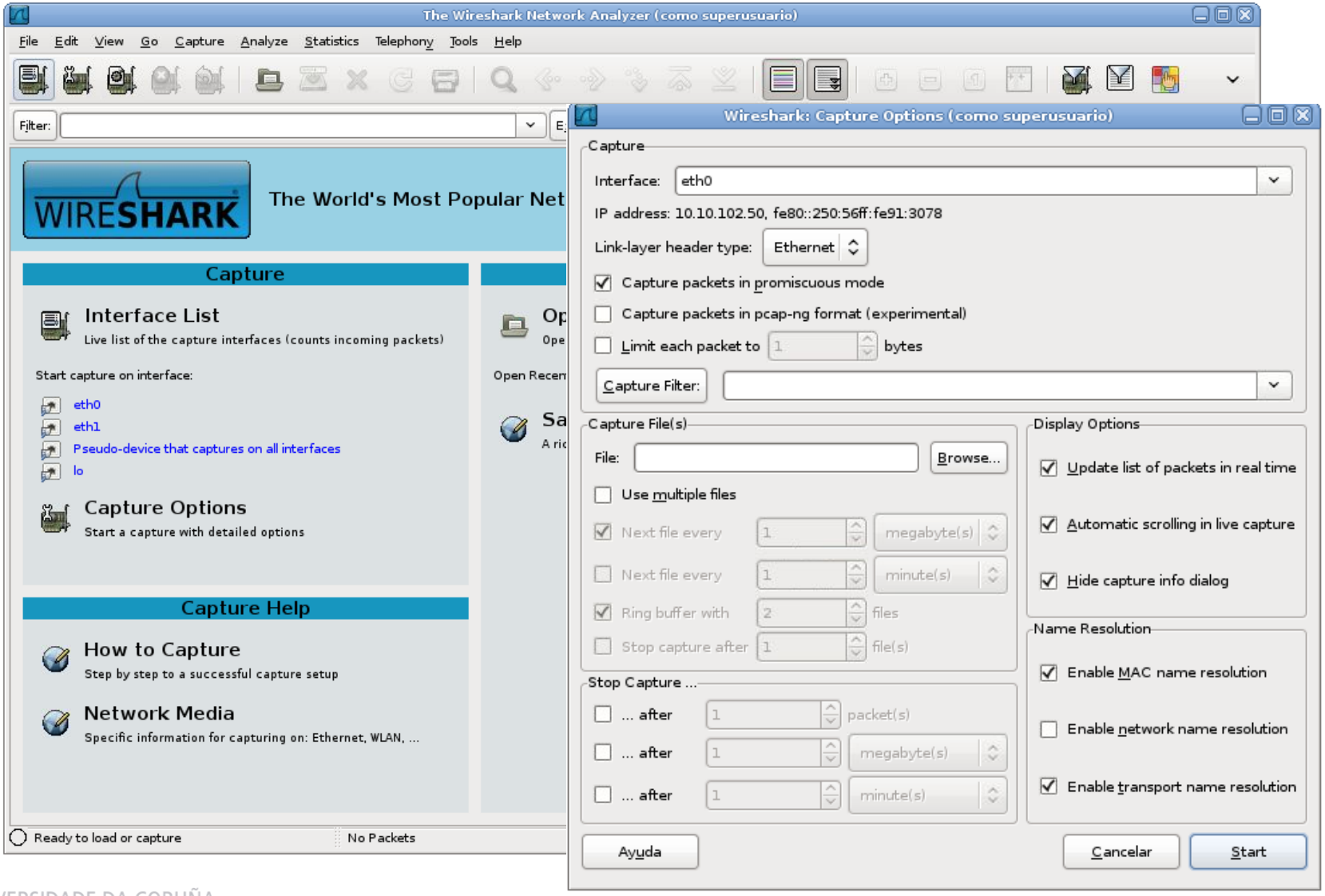
Herramientas. Wireshark

- Herramienta de análisis de tráfico de red.
- Cuenta con una muy buena interfaz gráfica.
- Incluye soporte para multitud de protocolos.
- Filtros, seguimiento de comunicaciones, análisis automático del tráfico,…
- Web:
  - http://www.wireshark.org/
    - Multitud de tutoriales, videos, etc.
- Instalación (Linux):
  - apt-get install tcpdump

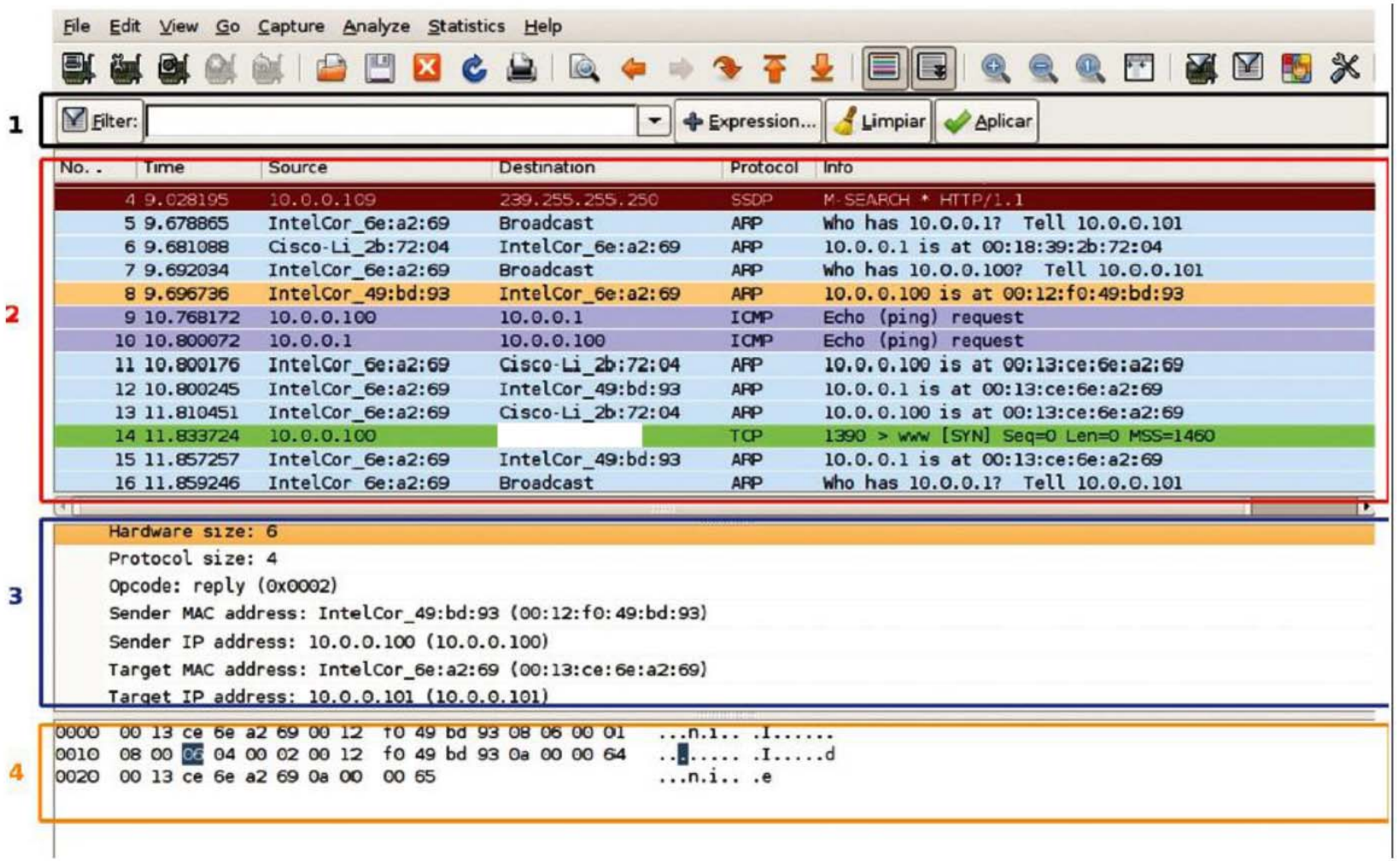
Herramientas. Wireshark



Herramientas. Wireshark

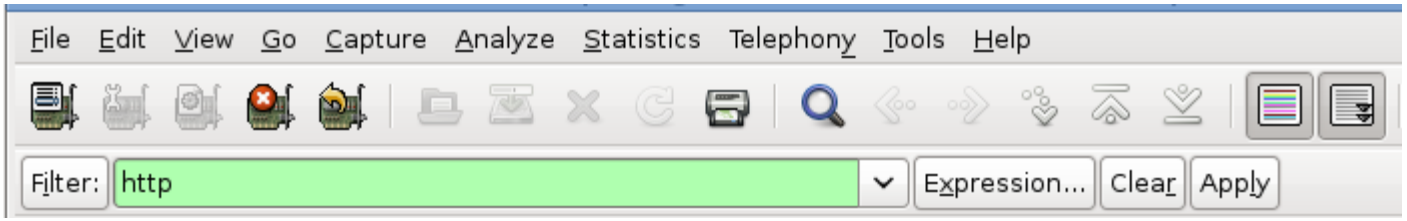


Herramientas. Wireshark



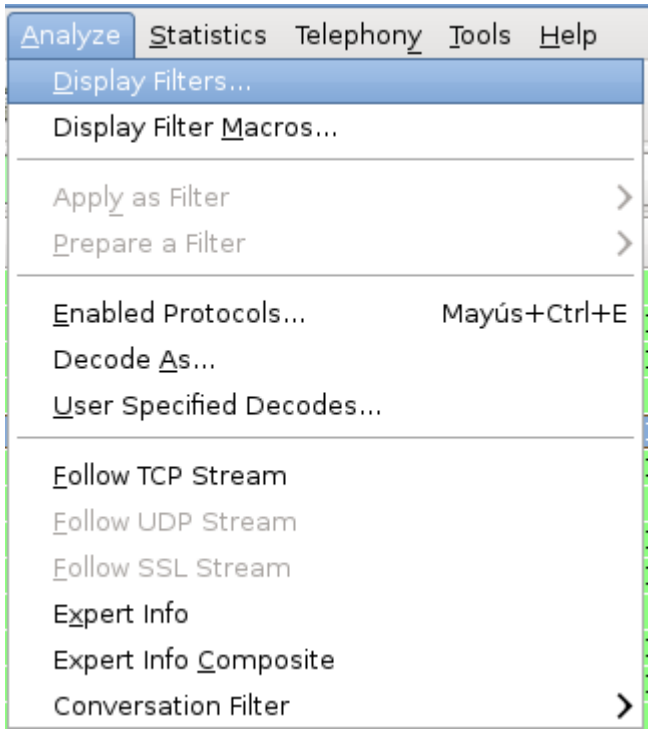
Herramientas. Wireshark

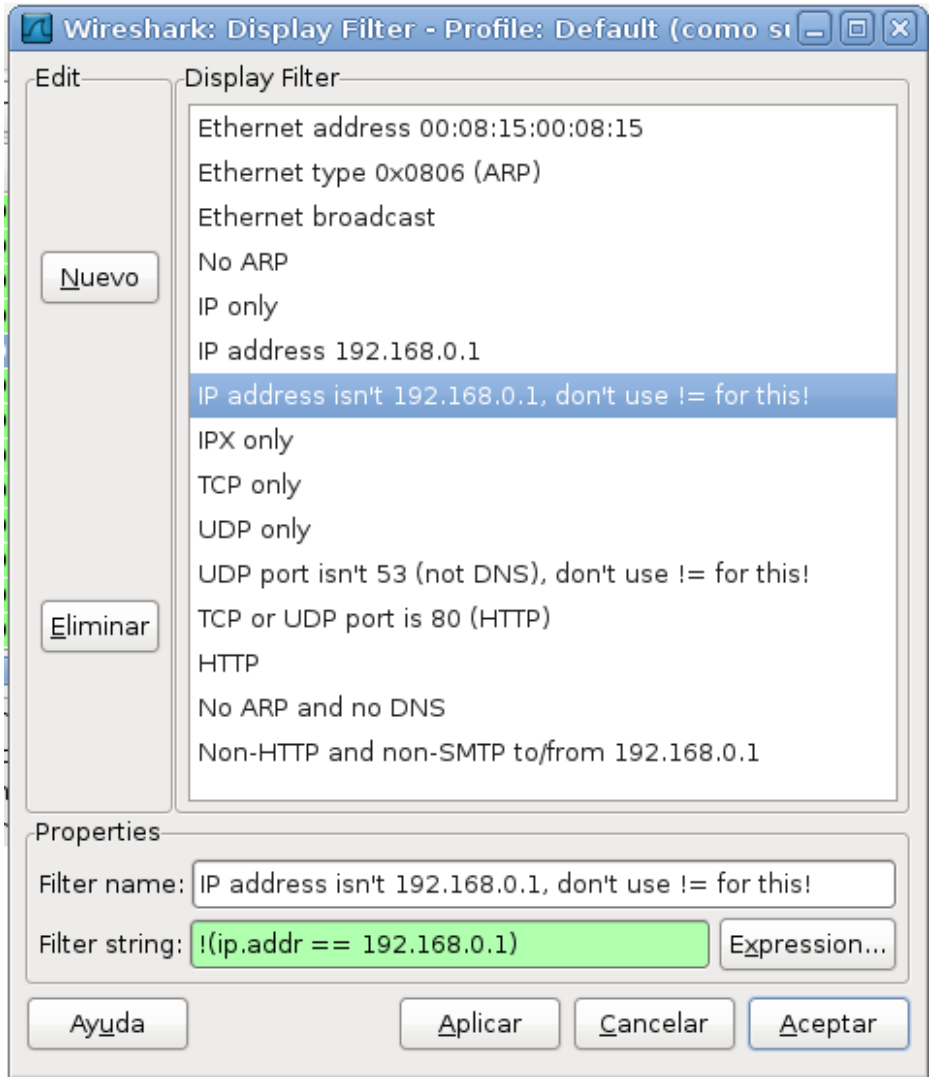
- Ejemplos
  - Filtrado por protocolo



  - Filtrado por IP
    - ip.addr == 192.168.0.1 / !(ip.addr == 192.168.0.1)
    - Filtros de ejemplo ->

Herramientas. Wireshark



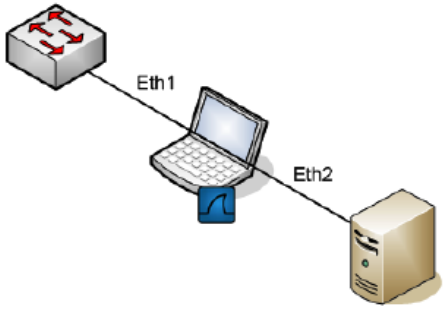


Herramientas. Wireshark

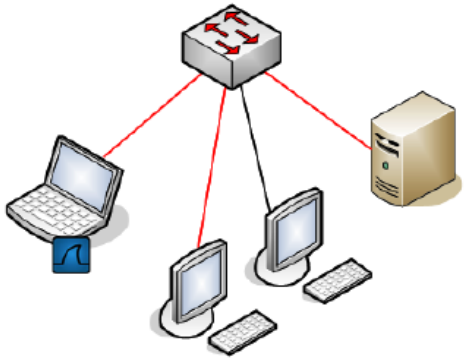
- Otras opciones útiles:
  - Follow TCP stream
  - Expert Info
  - Búsqueda de una cadena de texto
    - Edit > Find packet > String

Sniffing en medio conmutado

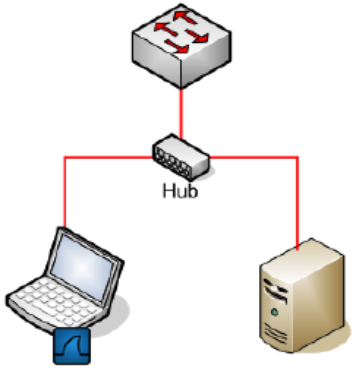
Modo Bridge



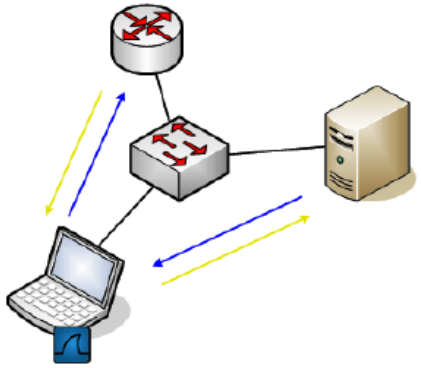
Port Mirroring



Hub



Arp-Spoof



ARP (Address Resolution Protocol)

- Protocolo de la capa de enlace de datos responsable de encontrar la dirección MAC que corresponde a una determinada dirección IP.
- Funcionamiento:
  - Se envía un paquete (***ARP request***) a la dirección de difusión de la red que contiene la dirección IP por la que se pregunta, y se espera a que esa máquina (u otra) responda (***ARP reply***) con la dirección Ethernet que le corresponde.

Source	Destination	Protocol	Info
HitronTe_44:55:66	Broadcast	ARP	Who has 192.168.0.5? Tell 192.168.0.1
HewlettP_b7:e9:28	HitronTe_44:55:66	ARP	192.168.0.15 is at 00:15:60:b7:e9:28

ARP (Address Resolution Protocol)

- Cada máquina mantiene una caché con las direcciones traducidas para reducir el retardo y la carga.
  - Las entradas de la tabla se borran cada cierto tiempo, ya que las direcciones físicas de la red pueden cambiar.

```
root@debian:/home/lsi# arp -a
? (10.10.102.4) at 00:90:fb:22:ff:95 [ether] on eth0
? (10.10.102.5) at 00:90:fb:22:ff:95 [ether] on eth0
? (10.10.102.27) at 00:1d:09:14:1e:7c [ether] on eth0
```

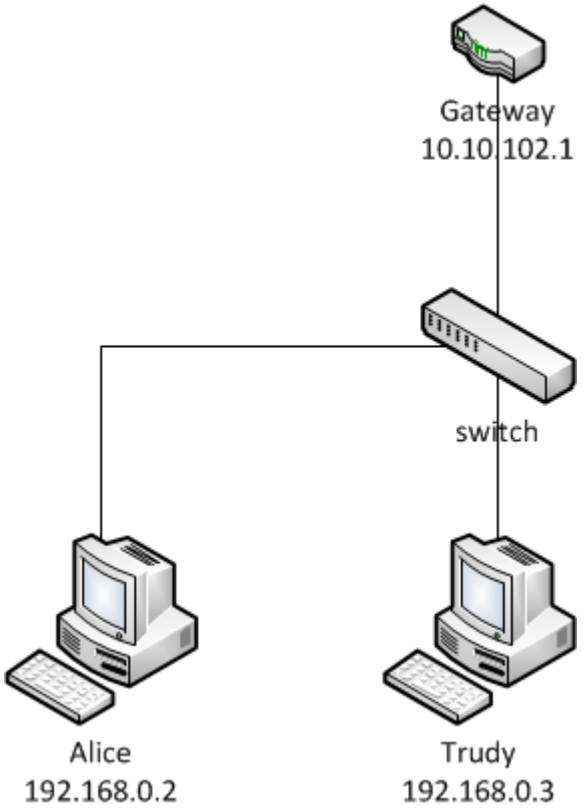
ARP Spoofing

- También se conoce como ***ARP Poisoning*** o ***ARP Poison Routing.***
- Técnica para infiltrarse en red Ethernet conmutada.
- Permite al atacante leer paquetes de datos en la LAN, modificar el tráfico o detenerlo.
- El atacante intenta asociar su MAC con la IP de la víctima.

ARP Spoofing

- Trudy lanza un *ARP request* a la dir. *broadcast* preguntando por la MAC de la IP 192.168.0.1 (*Gateway*)
- El *GW* contesta con *ARP reply* indicando cuál es su dir. MAC.
- Trudy lanza un *ARP request* a la dir. *broadcast* preguntando por la MAC de la IP 192.168.0.2 *(Alice*)
- Alice contesta con su dir. MAC.

Proceso normal



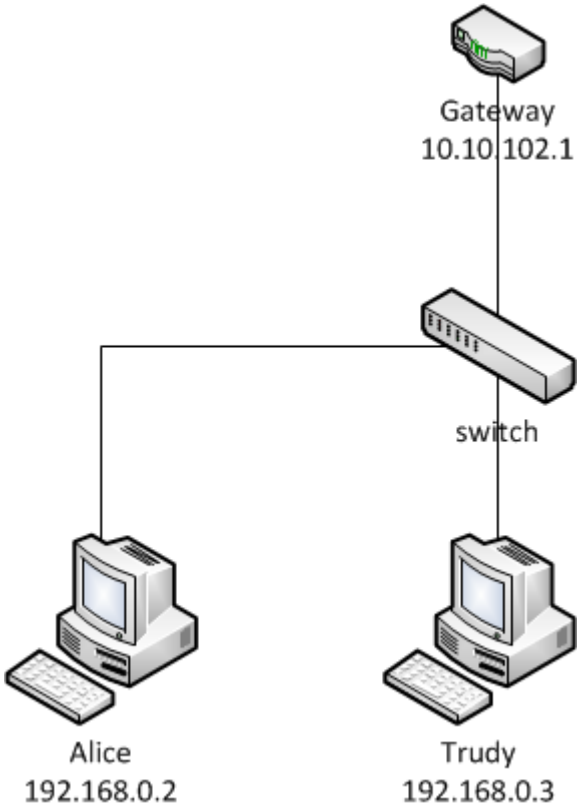
ARP Spoofing

- Trudy lanza un *ARP request* a la dir. *broadcast* preguntando por la MAC de la IP 192.168.0.1 (*Gateway*)
- El *GW* contesta con *ARP reply* indicando cuál es su dir. MAC.
- Trudy lanza un *ARP request* a la dir. *broadcast* preguntando por la MAC de la IP 192.168.0.2 *(Alice*)
- Alice contesta con su dir. MAC.

Proceso normal

- Trudy envía reiteradamente *ARP reply* falsos, a Alice y al *GW*, asociando la IP de ambos con su propia MAC.
  - A Alice le hace creer que él es el *GW*
  - Al *GW* le hace creer que él es Alice
  - Todo el tráfico que transite entre el *GW* y Alice pasará a través de Trudy

spoof



Herramientas que permiten ARP spoof

- Ettercap
- Cain y Abel
- suit Dsniff

ettercap

- Herramienta de seguridad libre y de código abierto.
- Puede usarse para análisis de protocolos de red y para auditorías de seguridad.
- Permite realizar ataques *man-in-the-middle* en una LAN.
- **Instalación** en Debian
  - apt-get install ettercap
  - apt-get install ettercap-gtk // con la interfaz gráfica

ettercap. Funcionalidades.

- Soporte para SSH1.
- Soporte para SSL.
- Inyección de caracteres en una conexión establecida.
- Filtrado/borrado de paquetes.
- Soporte para plug-ins.
- Recolector de contraseñas para multitud de protocolos:
  - TELNET, FTP, POP, RLOGIN, SSH1, ICQ, SMB, MySQL, HTTP, NNTP, X11, NAPSTER, IRC, RIP, BGP, SOCKS 5, IMAP 4, VNC, LDAP, NFS, SNMP, HALF LIFE, QUAKE 3, MSN, YMSG
- OS fingerprinting pasivo.
- Terminación de conexiones.

ettercap

- Especificación de cada objetivo: MAC/IPs/PUERTOs
  - "//80" cq MAC, cq IP y sólo puerto 80
  - "/10.0.0.1/" cq MAC, sólo IP 10.0.0.1, y cq puerto
  - "/192.168.0.100,192.168.0.105-7/" <- se pueden especificar varios objetivos y rangos
  - /192.168.0.100/21-23
  - ettercap -P list
    - Muestra la lista de plugins disponibles

ettercap

- Ayuda en modo interactivo: se activa pulsado "h"

```
[vV]      - change the visualization mode
[pP]      - activate a plugin
[lL]      - print the hosts list
[oO]      - print the profiles list
[cC]      - print the connections list
[sS]      - print interfaces statistics
[<space>] - stop/cont printing packets
[qQ]      - quit
```

ettercap

- ip_forwarding disabled. Lo hace ettercap.
- -M, --mitm <METHOD:ARGS>
  - Ataques disponibles:
    - ARP Spoofing: arp ([remote],[oneway])
    - Port Stealing: port ([remote],[tree])
    - …

ettercap. ARP Spoofing

- arp ([remote],[oneway])
  - Implementa ***ARP poisoning mitm attack***
  - Se envían ARP request/replies a las víctimas para envenenar su caché ARP.
  - Una vez que la caché ha sido envenenada, las víctimas enviarán todos los paquetes al atacante que, podrá modificarlos y reenviarlos al destino real.
  - "remote" es opcional. Se debe especificar si se quiere capturar tráfico de una dir. IP remota envenenando un GW. Si se especifica una víctima y el GW en los "targets", ettercap capturará sólo la conexión entre ellos, pero para permitir a ettercap capturar conexiones que pasan a través del GW, hay que usar este parámetro.
  - "oneway" forzará a ettercap a envenenar sólo desde TARGET1 a TARGET2. Útil si tu quieres envenenar sólo el cliente y no el router (donde puede haber un monitor de ARP).

ettercap. ARP Spoofing

- arp ([remote],[oneway])
  - Ejemplo:
    - ettercap -T -M arp:oneway,remote / 192.168.1.2 / / 192.168.1.1/
      - Realiza ARP poisoning contra el host 2 en la LAN y el GW
  - IMPORTANTE: detener el ataque (q en modo consola)

ARP Spoofing. Detección

- Herramientas que permiten detectar ARP spoof:
  - Arpwatch
  - Snort
  - Nast
  - NEPED
  - …

ARP Spoofing. Detección

- Arpwatch (apt-get install arpwatch)
  - Primero inspecciona la red y anota las MACs.
  - Luego monitoriza y genera alertas si hay cambios.

```
root@Mordor:~# arpwatch -n 192.168.254.0/24 -i eth0
root@Mordor:~# tail -f /var/log/syslog | grep -i arpwatch
Oct 19 09:16:42 Mordor arpwatch: listening on eth0
Oct 19 09:16:56 Mordor arpwatch: flip flop 192.168.254.254 08:00:27:f3:b1:0b (00:0e:0c:c6:c5:82) eth0
Oct 19 09:16:56 Mordor arpwatch: flip flop 192.168.254.254 08:00:27:f3:b1:0b (00:0e:0c:c6:c5:82) eth0
Oct 19 09:17:02 Mordor arpwatch: flip flop 192.168.254.245 08:00:27:f3:b1:0b (00:15:58:e8:50:0e) eth0
Oct 19 09:17:02 Mordor arpwatch: flip flop 192.168.254.245 08:00:27:f3:b1:0b (00:15:58:e8:50:0e) eth0
Oct 19 09:17:07 Mordor arpwatch: ethernet mismatch 192.168.254.254 08:00:27:f3:b1:0b (00:0e:0c:c6:c5:82) eth0
```

  - la MAC 08:00:27:f3:b1:0b, perteneciente al atacante, está intentando usurpar la MAC 0:0e:0c:c6:c5:82, que pertenece al *gateway* legítimo, mediante peticiones ARP fraudulentas.

ARP Spoofing. Detección

- Snort
  - Descomentar la siguiente línea en snort.conf:

```
#preprocessor arpspoof
```

  - Añadir la relación de MACs/IPs a monitorizar:

```
preprocessor arpspoof_detect_host: 192.168.254.254 00:0e:0c:c6:c5:82
```

    - Implica trabajo de gestión. Problema en redes grandes.

ARP Spoofing. Detección

- Snort
  - Lanzar Snort:

```
root@Mordor:~# snort -d -h 192.168.254.0/24 -A full-c /etc/snort/snort.conf
```

```
root@Mordor: /var/log/snort 159x3
root@Mordor:/var/log/snort# tail -f /var/log/snort/alert

[**] [112:4:1] (spp_arpspoof) Attempted ARP cache overwrite attack [**]
10/19-13:30:49.671380

[**] [112:4:1] (spp_arpspoof) Attempted ARP cache overwrite attack [**]
10/19-13:30:50.689457

[**] [112:4:1] (spp_arpspoof) Attempted ARP cache overwrite attack [**]
10/19-13:30:51.699448

[**] [112:4:1] (spp_arpspoof) Attempted ARP cache overwrite attack [**]
10/19-13:30:52.711415
```

ARP Spoofing. Detección

- Nast (apt-get install nast)
  - -P, --check-sniffers
    - Busca tarjetas en modo "promiscuo"

```
root@Mordor:~# nast -P all

Nast V. 0.2.0

This check can have false response, pay attention!
Probe for hosts...done

192.168.254.1 (192.168.254.1)   --------> Not found
192.168.254.3 (192.168.254.3)   ---------> Found!
192.168.254.6 (192.168.254.6)   ---------> Not found
192.168.254.32 (192.168.254.32)   ---------> Not found
```

  - -c, --check-arp-poisoning
    - Cuando se inicia realiza una asociación de todas las MACs de la LAN
    - Luego permanece a la escucha, por si alguna cambia

ARP Spoofing. Detección

- NEPED (Network Promiscuous Ethernet Detector)
  - Pequeño programa en C
    - http://downloads.securityfocus.com/tools/neped.c
  - Se basa en la siguiente técnica (test ARP):
    - Realiza petición ARP para cada IP a diagnosticar pero en lugar de dirigirla a la dirección de broadcast (FF:FF:FF:FF:FF:FF) lo hace a una aleatoria e inexistente.
    - Sólo las interfaces en modo promiscuo verán estos paquetes, luego sólo estas interfaces contestarán a estas peticiones.

ARP Spoofing. Detección

- Con ettercap
  - ettercap -T // -P search_promisc
  - ettercap -T // -P arp_cop
  - ettercap -T // -P scan_poisoner
- Con Wireshark
  - Wireshark normalmente detecta el ataque por arp-spoofing y lanza el mensaje (duplicate use of 192.168.1.11 detect!)"
  - También se puede identificar el ataque filtrando por:
    - Protocolo ARP: arp
    - ARP request: arp.opcode==0x0001
    - ARP reply: arp.opcode==0x0002
- Otras: Sentinel, AntiSniff, SniffDet, ArpOn, …

ARP Spoofing. Detección

- Algunos switches disponen de *Dynamic Arp Inspection* y *DHCP Snooping*
  - Detectan el ataque y pueden parar automáticamente el puerto del atacante

Port flooding (1)

- Descripción: Consiste en enviar múltiples tramas falsificadas (flood) a través de un puerto, con el objetivo de llenar la tabla de asignación del switch.
- Otros nombres: CAM Flooding, CAM Table Overflow, MAC flooding (+ general)
- ¿Qué es la tabla de asignación del switch?
  - Los switches mantienen una tabla que mapea direcciones MAC a puertos físicos de switch.
  - Esto es lo que se conoce como tabla de asignación o tabla CAM (Content-Addressable Memory) del switch
  - Esto permite al switch dirigir datos sólo al puerto físico en el que se encuentra el destinatario (a diferencia de un HUB)

Port flooding (2)

- ¿Cómo se rellena CAM?
  - Cuando una trama llega a puerto físico del switch, se añade una entrada, especificando la MAC del equipo que envió la trama junto con el puerto por el que entra.
    - De esta forma, cuando el switch recibe una trama dirigida a ese equipo sabrá por qué puerto debe enviarla.
  - CCNA How Switches Learn MAC Addresses: http://www.youtube.com/watch?v=WqjpBn-0oI4&feature=related

Port flooding (3)

- ¿Cómo dirige el tráfico el switch?
  - Busca la MAC destino en la tabla CAM.
    - Aparece: se envía la trama por el puerto que indica la CAM
    - No aparece (equipo no envió tráfico o su entrada expiró): se envía la trama por todos los puertos, salvo por el que entró
      - Todos los equipos recibirán la trama. Aquel cuya MAC coincida con la MAC destino de la trama contestará. Esto permitirá al switch añadir registrar el puerto asociado a esa MAC (nueva entrada en la CAM)
  - Gracias a esto, el switch no necesitará inundar (flood) todos los puertos con futuros paquetes dirigidos a ese equipo.

Port flooding (4)

- ¿Qué ocurre si se llena la tabla CAM?
  - En los switches de gama baja, normalmente, las tramas que tengan una dirección MAC destino no almacenada en la tabla CAM se retransmiten por todos los puertos <- El switch se comporta como un HUB.
  - Los switches de gama media/alta, incluyen mecanismos para mitigar el ataque, pero no vienen configurados por defecto!
- Efectos
  - Un atacante puede conectarse a cualquier puerto del switch y capturar tráfico que no recibiría en circunstancias normales.
  - Puede provocar DoS (Denial of Service).
- Técnica útil para capturar tráfico en entorno conmutado, cuando ARP poisoning no es efectivo (p.ej. hay mapeado ARP estático).

Port flooding (5). Herramientas.

- Macof
  - Parte de la suite dsniff (apt-get install dsniff)
  - Ejemplo: `macof -d 192.168.1.1`
- Ettercap
  - Plugin "rand_flood"
  - Necesario modificar "`port_steal_send_delay`" en etter.conf, de 2000 microsegundos a 1 para generar suficiente tráfico como para llenar la tabla CAM

Port flooding (6). Mitigación

- La detección es sencilla, ya que analizando el tráfico de red veríamos gran cantidad de tramas con valores aleatorios
- Los switches de gama media/alta permiten configurar ciertas características para mitigar este tipo de ataques:
  - ***Unicast Flooding Protection***
    - Permite controlar el nivel de inundación (flooding) de paquetes permitido
  - ***Port security***
    - Permite limitar el número de MACs que el switch puede "aprender" por puerto
  - ***Aging time***
    - Tiempo de expiración de las MAC en la tabla CAM

Port stealing (1)

- Similar a port flooding, pero menos "agresivo"
- El atacante envía multitud de tramas ARP
  - Las tramas ARP tienen como MAC origen la MAC de la(s) víctima(s)
  - El objetivo es que el switch "aprenda" que la víctima se encuentra en ese puerto y así dirija el tráfico hacia el.
    - Se le "roba" el puerto a la víctima.
  - Una vez que el atacante recibe paquetes "robados", detiene el proceso de inundación y realiza un ARP request a la víctima (destino real del paquete). Esto provocará que la víctima recupere su puerto.
  - En cuanto el atacante recibe el ARP reply sabe que la víctima ha recuperado su puerto, y le reenvía los paquetes robados. Entonces se puede reiniciar el proceso de inundación esperando nuevos paquetes.
- Técnica útil para capturar tráfico en entorno conmutado, cuando ARP spoofing no es efectivo (p.ej. hay mapeado ARP estático).

Port stealing (2). Implementación con ettercap

- port ([remote],[tree])
  - Inunda la LAN con paquetes ARP (en base al parámetro `port_steal_delay`) con el objetivo de robar el puerto del switch de cada víctima en la lista de hosts
  - La opción remote tiene el mismo significado que en el método "arp" mitm
  - Si no se especifica la opción "tree"
    - La dirección MAC origen será una de las MACs en la lista de hosts
    - La dirección MAC de destino es la misma que la del atacante (otras NICs no verán estos paquetes)
  - Si se especifica "tree"
    - La MAC de destino será una MAC falsa, de modo que estos paquetes serán propagados a otros switches
    - Esto podría permitir robar puertos en otros switches en el árbol (si hay), pero se genera una cantidad ingente de tráfico

Port stealing (3). Implementación con ettercap

- port ([remote],[tree])
  - Cuando se para el ataque, ettercap enviará un ARP request para cada host robado, devolviéndole sus puertos del switch
  - HALF o FULL DUPLEX MITM dependiendo de las máquinas objetivo seleccionadas.
  - Ejemplos:
    - `ettercap -T -M port:remote /10.0.0.1/ /10.0.0.15/`
      - Intercepta y visualiza tráfico entre 10.0.0.1 1 10.0.0.15.
      - También se recibe el tráfico para 10.0.0.1 y 10.0.0.15.
    - `ettercap -T -M port:remote /10.0.0.1/`
      - Intercepta y visualiza todo el tráfico para 10.0.0.1.

Otros ataques

- DNS Spoof
- ICMP redirection
- DHCP Spoof
- …

Session hijacking (secuestro de sesión)

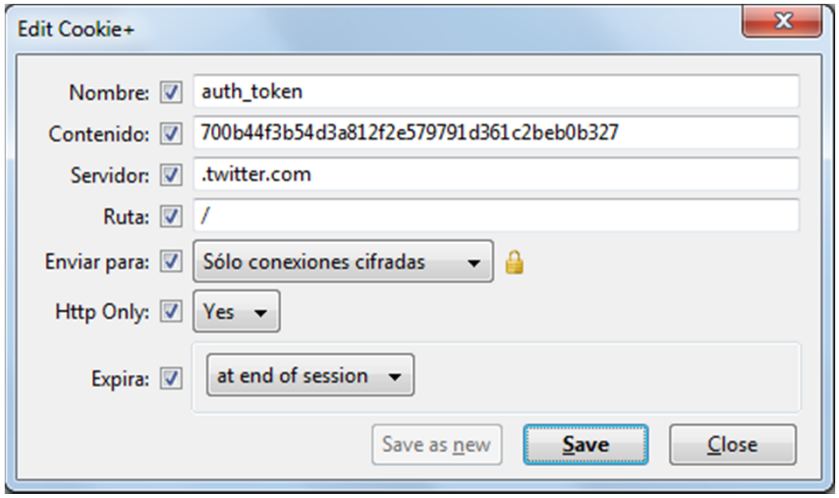
- Objetivo: explotar una sesión válida para obtener acceso no autorizado a información o servicios
- Fundamentos del ataque:
  - Generalmente, una aplicación Web hace uso de sesiones para mantener el estado y suplir así la carencia de HTTP (protocolo sin estado)
  - Conociendo el identificador de la sesión, se puede "generar" en otra máquina (atacante)

Session hijacking (secuestro de sesión)

- ¿Qué es una sesión? (1)
  - Mecanismo utilizado para mantener "estado" entre distintas peticiones HTTP (protocolo sin estado)
  - Son mantenidas por el servidor (las lee y escribe)
  - Tienen un identificador
  - El cliente (navegador) debe enviar ese identificador en cada petición HTTP. De esta forma, el servidor sabe quién está enviando la petición, recupera el estado y crea al usuario la ilusión de sesión
  - Si alguien no autorizado se apropia (hijack) del identificador de la sesión, puede "recrear" la sesión en su máquina. Si el usuario estaba autenticado en la sesión, se consigue la suplantación de su identidad

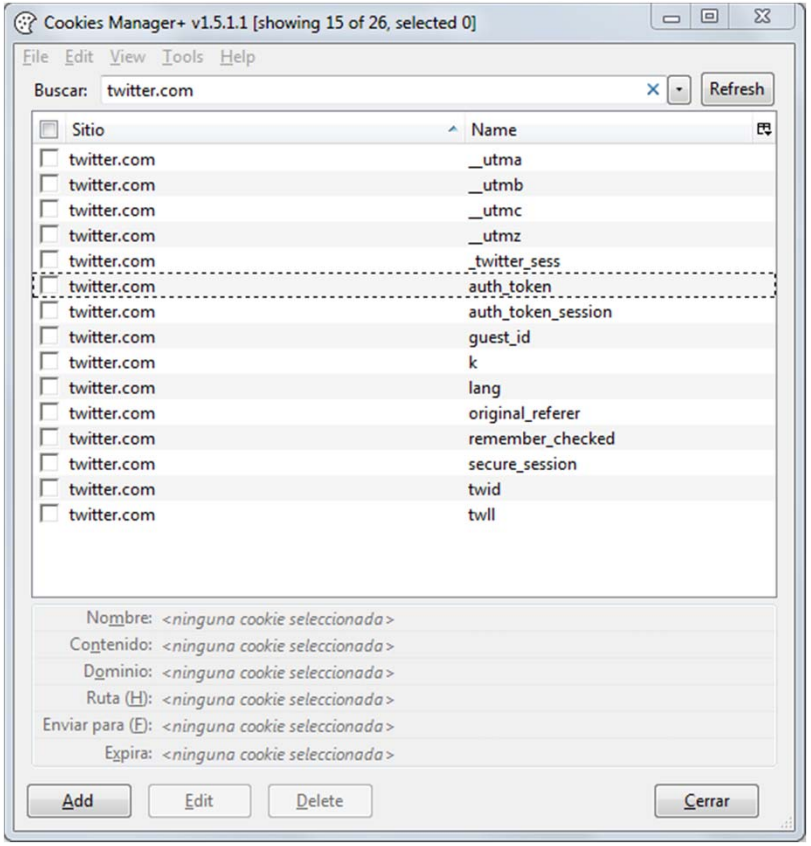
Session hijacking (secuestro de sesión)

- ¿Cómo envía el navegador el identificador de la sesión al servidor?
  - Como parte de la URL
    - http://www.sitio.com/%28X%281%29F%28iSji_itFJzJ9tZglJMvMyFw8v8-R4k-0euN18LvcqPYXNA_ww1O6jVMReOBd4kI-3DM9w9PN3JXgqL2gp3oqjDqb3tk1%29%29/Default.aspx
    - http://www.sitio.com/myservlet;jsessionid=1E6FEC0D14D044541DD84D2D013D29ED
  - En una cookie



Session hijacking a partir de cookies con Wireshark

- Pasos:
  1. Identificar la cookie que necesitamos "secuestrar"
     - Normalmente, los sitios Web utilizan varias cookies. Es necesario averiguar cuál es la necesaria (normalmente, es suficiente con el *token* de autenticación)
     - Herramientas como Cookies Manager (plug-in para Firefox), permiten gestionar las cookies (buscar, editar, crear, ...)

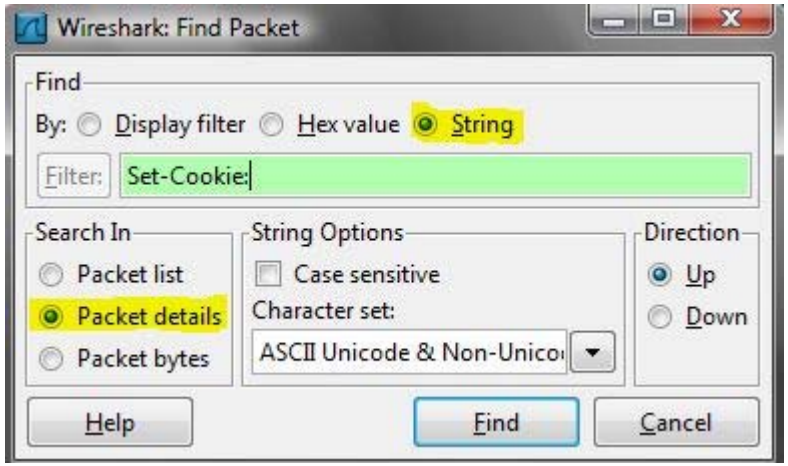


Session hijacking a partir de cookies con Wireshark

- Pasos:
  1. Identificar la cookie que necesitamos "secuestrar"
  2. Capturar tráfico de la víctima

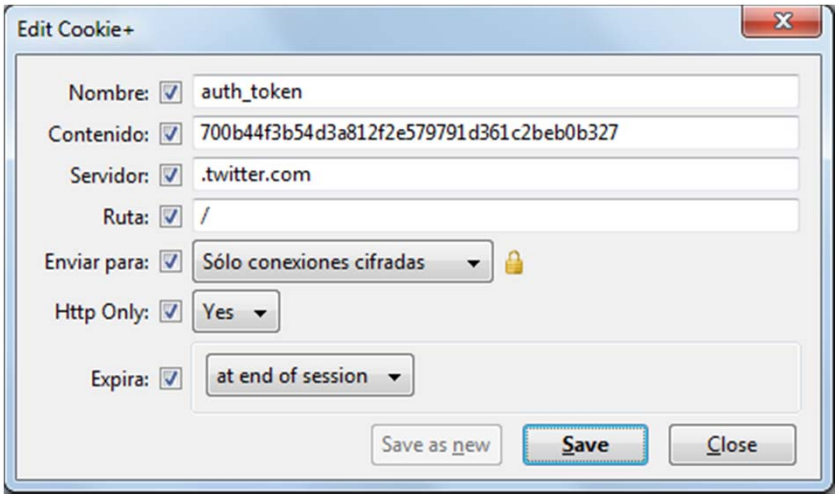
Session hijacking a partir de cookies con Wireshark

- Pasos:
  1. Identificar la cookie que necesitamos "secuestrar"
  2. Capturar tráfico de la víctima
  3. Buscar la cookie
     - Wireshark permite realizar búsquedas en el contenido del tráfico (Ctrl-F o Edit–>Find Packet)
     - Filtrando por "Set-Cookie:" podemos encontrar la cookie que buscamos
     - Encontrado un paquete "candidato", la opción "Follow TCP Stream" nos facilitará su interpretación



Session hijacking a partir de cookies con Wireshark

- Pasos:
  1. Identificar la cookie que necesitamos "secuestrar"
  2. Capturar tráfico de la víctima
  3. Buscar la cookie
  4. Crear una cookie con los datos obtenidos y almacenarla en el navegador
     - Se puede hacer manualmente, pero existen herramientas que facilitan el trabajo (p. ej. Cookies Manager+)



Session hijacking a partir de cookies con Wireshark

- Pasos:
  1. Identificar la cookie que necesitamos "secuestrar"
  2. Capturar tráfico de la víctima
  3. Buscar la cookie
  4. Crear una cookie con los datos obtenidos y almacenarla en el navegador
  5. Dirigirse a la URL del sitio

Session hijacking (secuestro de sesión).

- Hay herramientas que permiten automatizar los pasos vistos previamente
  - Por ejemplo: Firesheep (plugin para firefox)
  - Usuarios sin conocimientos pueden realizar el ataque
- Mitigación
  - Cierre de la sesión <- esto no da la garantía absoluta (algunas aplicaciones no responden correctamente)
  - Prevenir la captura del tráfico de nuestra máquina
    - Uso de protocolos seguros (a distintos niveles)
      - Muchos sitios Web en la actualidad permiten configurar el uso de https siempre (p. ej. Facebook o Twitter)
      - Hay herramientas que nos ayudan a utilizar https siempre que sea posible (p. ej. HTTPS Everywhere)
    - Aplicar las recomendaciones vistas para prevenir ataques MitM

Referencias

- Cisco. Layer 2 Attacks and Mitigation Techniques for the Cisco Catalyst 6500 Series Switches Running Cisco IOS Software. http://www.cisco.com/en/US/prod/collateral/switches/ps5718/ps708/white_paper_c11_603836.html

- Inteco. Análisis de tráfico con wireshark. Disponible en: http://cert.inteco.es/extfrontinteco/img/File/intecocert/EstudiosInformes/cert_inf_seguridad_analisis_trafico_wireshark.pdf

- [*] Santos del Riego, A (2012). ***Legislación [Protección] y Seguridad de la Información***. Disponible en: http://psi-udc.blogspot.com.

- SANS Institute. An Ettercap Primer. http://www.sans.org/reading_room/whitepapers/tools/ettercap-primer_1406