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

Práctica 3: Seguridad en redes Ethernet: *sniffers*

1- Introducción

En el mundo de la seguridad informática se presta mucha atención a la protección de los sistemas frente a ataques provenientes del exterior. Sin embargo, la seguridad en el interior de la organización suele estar más descuidada.

En esta práctica vamos a tratar aspectos básicos de la seguridad en redes locales. Conoceremos algunos equipos básicos de comunicaciones como los *hubs* y *switches*; trataremos de comprender qué peligros entraña el que varios usuarios compartan los mismos equipos de comunicación; y qué herramientas y técnicas puede utilizar el administrador para contrarrestarlos.

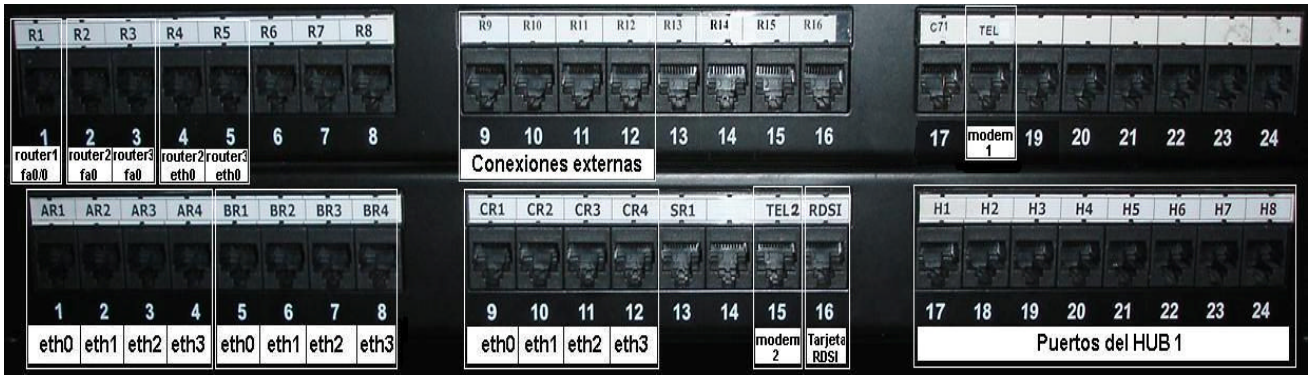
Para ello emplearemos los equipos PC-A, PC-B y PC-C. Cada uno de ellos tiene cuatro interfaces (tarjetas) *ethernet* (*eth0*, *eth1, eth2 y eth3*) que podremos utilizar para esta práctica, aunque nos bastará con configurar uno de cada equipo.

2- Estableciendo una red Ethernet

Para poder interconectar varios ordenadores, lo primero que tendremos que hacer será conectarlos a un concentrador (*hub*) o a un conmutador (*switch*). Un concentrador retransmite la señal que le llega de cada cable a todos los demás, es decir, es como si uniese todos los cables formando un único cable virtual. Por contra, un conmutador es un dispositivo inteligente; lee las tramas Ethernet que le llegan y las reenvía, tan sólo, por el puerto en el que se encuentra conectada la estación de destino. (Referencias: http://es.wikipedia.org/wiki/Switch y http://es.wikipedia.org/wiki/Concentrador).

Como podrá suponer, el rendimiento de un switch es superior al de un hub. Pero aparte de esto, también hay diferencias en cuanto a la seguridad: con un concentrador, cualquier estación puede escuchar todo el tráfico, mientras que, en una red conmutada, una estación sólo escucharía el tráfico que va dirigido a ella (al menos a priori).

Para conectar los PCs a los equipos de red, emplearemos el panel de parcheo (*patch-panel*) del armario que sigue la distribución de puertos de la figura(utilice la documentación sobre los armarios):



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

Ahora, configuraremos las estaciones a nivel IP. Para ello, debemos utilizar el comando *ifconfig*. Como siempre, puede comenzar echando un vistazo a la página del manual. Por ejemplo, para configurar el primer interfaz de red debemos hacer algo del estilo:

```
$ sudo ifconfig eth0 up 192.168.0.1 netmask 255.255.255.0
```

Si lo que queremos hacer es desactivarlo:

```
$ sudo ifconfig eth0 down
```

Las opciones up/down tras el dispositivo, sirven para activarlo/desactivarlo, aunque existen comandos específicos para esto mismo: *ifup* e *ifdown*. La instrucción para desactivar la tarjeta de red podríamos reescribirla como:

```
$ sudo ifdown eth0
```

El empleo de *sudo* precediendo a un comando es necesario para que un usuario normal pueda ejecutar programas que requieren privilegios de *root*. Como en esta práctica vamos a trabajar principalmente con herramientas de red que, en su mayoría, requieren dichos privilegios, habremos de preceder todas las órdenes con el comando *sudo*.

Configure los PCs A, B y C en la misma red (10.3.**armario**.0/24). Recuerde que debe utilizar el comando ifconfig, tal y como se ha explicado anteriormente. Configure solamente el interfaz eth0 de cada equipo.

Un buen comienzo para preparar un ataque en *ethernet*, sería saber si nuestra tarjeta de red está conectada a un *hub* o a un *switch*. Para ello podemos utilizar el comando *mii-diag*, que nos informa de la velocidad de enlace y la configuración de dúplex de las tarjetas *ethernet*.

Checkpoint 3.1: ¿Cuál es la salida del comando *mii-diag* cuando la tarjeta está conectada a un hub? ¿Y a un switch? ¿Cómo se puede distinguir cada uno de los dos casos?

3- Herramientas de sniffing en una red compartida

Veamos a continuación qué ocurre cuando varias máquinas comparten el mismo concentrador (*hub*). Para ello, es necesario que tenga configurado a nivel IP al menos 1 interfaz de cada máquina.

Una vez hecho esto, pruebe a transmitir tráfico y capturarlo. Para generar o recibir tráfico a nivel TCP y UDP, puede utilizar la herramienta *nc,* y para capturarlo *tcpdump* y *ethereal*.

El comando *nc* (vea la página del manual: *man nc*) nos permite realizar varias funciones

- Cliente TCP sencillo que permite interactuar con la conexión desde el teclado
- Cliente UDP sencillo que permite enviar un datagrama UDP con cada línea que escribamos y muestra a su vez el contenido de los datagramas que llegan al puerto indicado
- Servidor TCP que permanece a la escucha y acepta una conexión
- Servidor UDP que permanece a la escucha y permite contestar a quien le escriba

También puede utilizar el comando *telnet* para establecer una conexión TCP a un puerto determinado. La sintaxis será: $ *telnet IP_del_equipo Puerto*.

Practique con los comandos *nc*, *telnet* y *tcpdump*, así como la herramienta *ethereal*. Por ejemplo, ponga un servidor TCP con **`nc`** en uno de los ordenadores y escuchando en un puerto que elijas.

Después lanza contra él una conexión con telnet o nc y comprueba el resultado. Captura el tráfico con las herramientas *tcpdump* y *ethereal*.

Haga que PC B escuche en el puerto TCP 10010 y conéctese desde el PC A. Pruebe a capturar el tráfico desde los tres equipos. ¿Desde cuáles es posible ver el tráfico entre los PCs A y B? Indica qué comandos utilizas en cada PC para escuchar, enviar y capturar el tráfico.

Suponga ahora que hay mucho tráfico entre las estaciones A y B. ¿Qué debe hacer con tcpdump para restringir la captura a los paquetes cuyo puerto TCP de origen o de destino sea el 10010?

Suponga ahora que, a través de la conexión creada, la estación A envía un texto que se supone que es una password al equipo B, en concreto considera que se transmite sin encriptar la *password* de autenticación *claveSSI*. Se pretende averiguarla escuchando desde la estación C.

Checkpoint 3.2: Muestre al profesor de prácticas el paquete con la *password*, capturada con *tcpdump* y con *Wireshark*.

Conéctese ahora desde la máquina A a la B por medio del programa ssh (Secure SHell). Intente averiguar el password de autenticación escuchando desde la estación C. ¿Es posible hacerlo? ¿Por qué?

4- Herramientas de sniffing en una red conmutada

En el apartado anterior ha quedado patente la inseguridad de las redes basadas en un medio compartido. Todo el tráfico es visible desde cualquier estación que comparta el medio. Y cualquier aplicación que no utilice criptografía está comprometida.

En este apartado se pretende estudiar qué sucede si en vez de utilizar un *hub* como elemento de red, empleamos un *switch*. Para ello, utilizaremos el conmutador etiquetado como *switch0* en el armario. Para poder intercomunicar nuestros PCs, usaremos tan sólo los 8 primeros puertos de dicho *switch*, ya que se encuentra preconfigurado con 3 VLANs de forma que se comporta como 3 conmutadores individuales, cada uno de 8 puertos:

Conecte los tres equipos utilizando un switch. Transmita tráfico desde el PC A al PC B. Esta vez, ¿Desde qué equipos es posible escuchar dicho tráfico?

A la vista de los resultados parece que la mejora en cuanto a seguridad es bastante importante. Pero la palabra clave en la anterior frase es parece, ya que existen formas de escuchar el trafico que no va dirigido a nuestro interfaz en una red conmutada. A continuación estudiaremos este tipo de ataques.

La segmentación de redes mediante el uso de *switches* parecía la solución perfecta para evitar el *sniffing,* pero pronto se descubrió que es posible aprovechar una inseguridad en el protocolo ARP para espiar en una red conmutada. El protocolo ARP (A*ddress Resolution Protocol*) es el encargado de traducir las direcciones IP (de 32 bits) a las correspondientes direcciones de hardware o direcciones MAC (de 48 bits en el caso de *ethernet*), y el ataque que aprovecha su vulnerabilidad se conoce con el nombre de envenenamiento ARP

Para poder llevar a cabo un ataque de estas características, necesitaremos familiarizarnos con el programa *ettercap*, una de las herramientas más conocidas para el *sniffing* en redes conmutadas. Como siempre, lo mejor es empezar por consultar la página del *man* de Linux. No debe preocuparnos el hecho de no entenderlo todo a la primera; algunas de las opciones son bastante avanzadas y, de momento, no necesitaremos conocerlas. Lo importante es quedarnos con estas dos ideas:

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

1. Es posible escuchar tráfico en redes conmutadas
2. Una de las técnicas que lo hace posible es el envenenamiento ARP.

Investigue en qué consiste el envenenamiento ARP ¿Cómo le puede permitir a un curioso escuchar el tráfico dirigido a otros usuarios? Capturar tráfico con `Wireshark`, cuando a continuación lance `ettercap`, le resultará de utilidad para resolver esta cuestión.

Suponga que volvemos a transmitir tráfico de la estación A a la B. Escuche desde la máquina C dicho tráfico utilizando `ettercap`. Explique brevemente qué pasos hay que seguir para conseguirlo.

Checkpoint 3.3: Muestre al profesor que es capaz, desde el PC C, de ver los paquetes que intercambian las estaciones A y B. Explique como es eso posible.

Llegados a este punto debería ser capaz de responder a preguntas del tipo:

¿Qué información aporta a un atacante el ARP poisoning? ¿Qué técnica le permite a un atacante utilizar dicha información para esnifar paquetes en una red conmutada? ¿Conoce algún programa que haga uso de esta técnica?

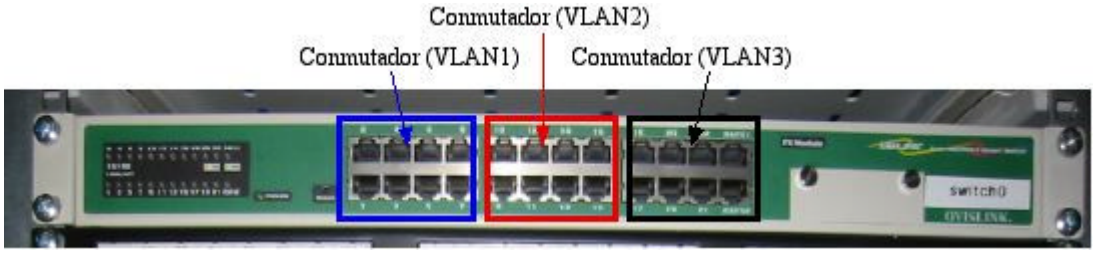
La pregunta que queda formularnos ahora es cómo evitar que suceda lo anterior. La respuesta es que no es posible evitarlo, salvo que empleemos tablas ARP estáticas, lo cual es viable, pero muy poco práctico desde el punto de vista de gestión de una red. La dificultad a la hora de impedirlo radica en que es una característica del propio protocolo ARP, la que posibilita el ataque. Lo que sí podemos hacer es detectar los ataques. De hecho, muchos sistemas de detección de intrusiones (IDS) lo hacen.

5- Un paso más en la seguridad: las VLAN

Las redes locales virtuales (VLAN) surgieron principalmente para dar flexibilidad a las instalaciones tradicionales. Básicamente, lo que hacen es definir redes lógicas por encima de las redes físicas. Existen varias opciones a la hora de crear las redes lógicas: a nivel de puerto *ethernet*, de dirección MAC, de dirección IP..., etc.

El *switch0* que tenemos en los armarios permite la implementación de redes virtuales a nivel de puerto. El mapeo de redes virtuales y puertos es el siguiente:

- Red virtual 1: puertos 1 a 8.
- Red virtual 2: puertos 9 a 16.
- Red virtual 3: puertos 17 a 24.



Checkpoint 3.4: Conecte los PCs A y B a los puertos correspondientes a la primera red virtual. El PC C irá conectado a un puerto de la segunda red virtual. Transmita tráfico entre los PCs A y B e intente escucharlo desde el PC C utilizando Ethercap. ¿Es posible? ¿Por qué?