Seguridad en Redes de Ordenadores
Práctica 3: IDS y Pentesting
Parte I

GSyC
Departamento de Teoría de la Señal y Comunicaciones y
Sistemas Telemáticos y Computación

Abril de 2017

Resumen

Esta práctica se va a realizar con la herramienta netgui-ng.sh. Es muy importante que arranques esta herramienta y no `netgui.sh` ya que la nueva versión de NetGUI tiene instalados paquetes necesarios para la realización de la práctica.

Configuración previa

En esta práctica usaremos una configuración de máquinas virtuales con 128MB de memoria RAM, para ello deberás usar el fichero `netkit.conf` que te proporcionamos. Copia el fichero `netkit.conf` en el directorio de tu carpeta personal `.netkit`. Comprueba que lo has copiado correctamente, deberías verlo en esa carpeta:

```
miusuario@maquinaLaboratorio ~ $ ls ~/.netkit
```

Si quieres arrancar las máquinas de las prácticas anteriores deberás borrar el fichero `netkit.conf` para que se use el valor por defecto de memoria RAM.

1. Configuración IP del escenario

En la figura 1 se muestra una empresa que tiene una zona DMZ (DeMilitarized Zone) donde se encuentran los servicios que la empresa permite su acceso desde exterior, servidor de correo, DNS, web, etc y una zona interna a la que sólo deberían tener acceso los usuarios internos y que tiene configurado un direccionamiento privado.

Descomprime el escenario de Netgui-NG `lab-analisisRed.tgz`. Modifica los ficheros:

`lab-analisisRed/<nombreDeMáquina>/etc/network/interfaces`

de las máquinas para que las direcciones IP incluyan en el segundo byte más significativo, el identificador que se te asignó con la práctica de `openVPN`. Por ejemplo, dada la dirección IP: `X.Y.Z.W` → `X.<ID>.Z.W`. Recuerda que también deberás modificar las rutas y/o `gateway`.

De esta forma cuando arranques las máquinas tendrán configuradas las direcciones IP y rutas adecuadas.

2. Detección de intrusos

La máquina `ids1` va a ejecutar una herramienta IDS, `snort`, que detecta accesos potencialmente maliciosos y los registra en un fichero de log. Esta herramienta no está instalada en las máquinas, para instalarla, ejecuta lo siguiente:

```
ids1:~/ dpkg -i *.deb
```

Al instalarla, preguntará la red a explorar dentro de `snort`: `100.<ID>.0.0/8` que incluye las máquinas que están conectadas al hub1 y hub2. La configuración por defecto te devolverá un error debido a las restricciones de memoria que tenemos en NetGUI-NG. A continuación realizaremos la configuración manual.

Vamos a utilizar una configuración simplificada de `snort` ya que es una herramienta pesada que necesita más memoria de la que tienen asignadas las máquinas de NetGUI-NG. Por eso, en la instalación por defecto

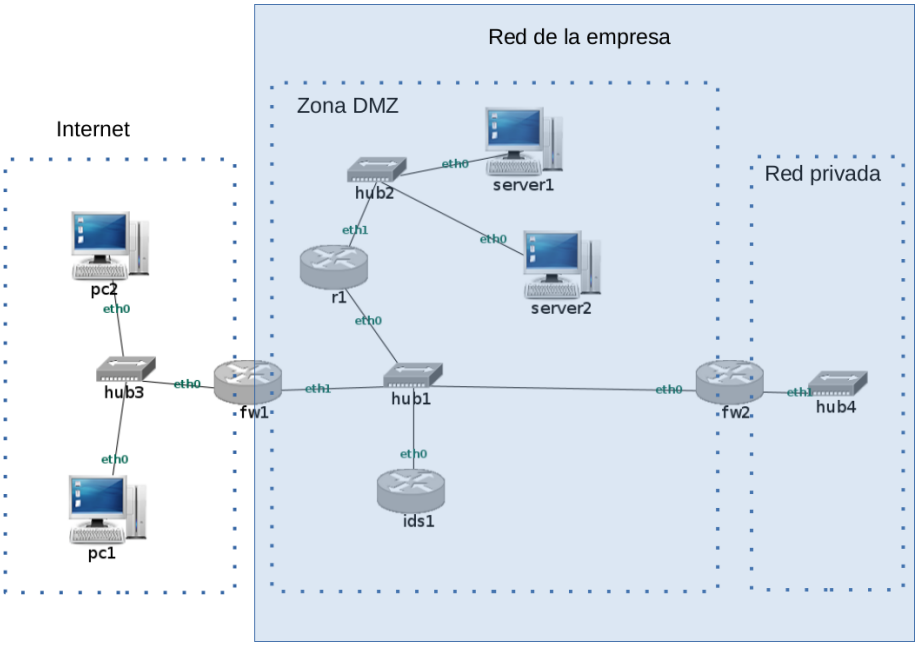

Figura 1: Escenario de red de una empresa conectada a Internet a través de una zona DMZ.

te muestra un error con la configuración que trae por defecto el paquete. Copia el fichero de configuración de snort que te proporcionamos a la carpeta de configuración de `snort`:

```
ids1:~/ cp snort.conf /etc/snort
```

El fichero de configuración es complejo y sólo tiene activadas la detección de cierto tipo de vulnerabilidades. Cuando `snort` descubra tráfico potencialmente malicioso escribirá una alerta en un fichero de logs y almacenará el tráfico malicioso en un fichero de captura. Estos ficheros se encontrarán en la carpeta `/var/log/snort`.

En la carpeta `/etc/snort/rules/` se describen reglas predefinidas en `snort` para la inspección de tráfico. Dependiendo de la configuración del fichero `/etc/snort/snort.conf` se podrán cargar las reglas que se desean aplicar al tráfico que el IDS examine.

A continuación responde a las siguientes preguntas en la memoria de la práctica:

1. Mira el apartado "Step #7: Customize your rule set" en el fichero `/etc/snort/snort.conf` e indica cuáles son los ficheros con reglas que se están cargando.
2. Mira el contenido del fichero `/etc/snort/rules/icmp.rules`. Las líneas que comienzan por # son comentarios, las reglas están escritas cada una en una única línea. Incluye la última regla de ese fichero en la memoria y explica el contenido.

   Ten en cuenta que todas las reglas siguen este formato, por ejemplo:

   ```
   alert tcp any any -> 192.168.1.0/24 111 \
        (content: "|00 01 86 a5|"; msg: "mountd access";)
   ```

   donde el texto antes del primer paréntesis es la cabecera de la regla y el texto contenido en el paréntesis son las opciones.

cabecera de la regla	(opciones de la regla)

   En la cabecera de la regla se encuentran los siguientes campos:

acción	protocolo	direcciónIP1	puerto1	sentido de la comunicación	direcciónIP2	puerto2

   Para la regla previa, los campos de la cabecera de la regla serían: acción a aplicar `alert` (guarda el paquete en un fichero de captura y escribe un mensaje en el fichero log), el protocolo `tcp`, direcciónIP1 `any`, puerto1 `any`, sentido de la comunicación `->` (desde la direcciónIP1 y puerto1 dirigido a la direcciónIP2, puerto2), direcciónIP2 `192.168.1.0/24` y puerto2 `111`.

Las opciones se escriben separadas por ';', donde cada opción queda definida por el formato:

`nombre_opción:valor_opción`.

En el ejemplo, la opción `msg` indica el mensaje que se va a guardar en el fichero de log (`"mountd access"`). La opción `content` busca un patrón en el contenido de un paquete. Otras opciones hacen referencia a características adicionales que tiene que cumplir el paquete, por ejemplo `ttl:100`.

Entre las opciones se encuentra su clasificación y prioridad, con el nombre de opción `classtype` se asocia un valor cuyo significado y el valor de prioridad de la alerta se pueden consultar en el fichero `/etc/snort/classification.config`. Números bajos de prioridad significa, prioridad muy alta.

La opción `reference` da información de sistemas en los que se puede encontrar más información sobre esta alerta. La opción `sid` es el identificador Snort de la regla, se usa para herramientas de búsqueda de reglas en la configuración. La opción `rev` hace referencia al número de versión de esa regla.

3. Busca en el fichero `icmp.rules` la regla que es una alerta que escribe el mensaje `"ICMP PING NMAP"`[1], inclúyela en la memoria y explica todo lo que puedes saber de su contenido.
4. Busca en el fichero `icmp-info.rules` la regla que es una alerta que escribe el mensaje `ICMP PING *NIX"`, inclúyela en la memoria y explica todo lo que puedes saber de su contenido.
5. Explica cuál es la diferencia entre ambas reglas y el nivel de prioridad de cada una de ellas. ¿Por qué una tiene mayor prioridad que otra?
6. Lanza `snort` en la máquina `ids1` (`snort -A console -c /etc/snort/snort.conf`) para que comience a detectar tráfico potencialmente peligroso y déjalo lanzado para que te vaya mostrando las alertas que detecte en los apartados sucesivos.

3. Pentesting (Penetration Testing)

Vamos a realizar algunas pruebas sencillas de penetración para ver el comportamiento de los protocolos y analizar algunas vulnerabilidades. Para este apartado no configuraremos las reglas del firewall `fw1`/`fw2`. En la siguiente parte de la práctica se observará como al configurar correctamente `fw1`/`fw2` se pueden evitar los ataques más simples.

3.1. DoS en TCP

Vamos a realizar una prueba sencilla de DoS contra un servidor en la máquina `server1`, no podemos dejar ejecutándose mucho tiempo esta prueba porque se genera demasiado tráfico y NetGUI-NG no puede procesar tantos paquetes.

1. Con la herramienta `nc` lanza un servidor TCP en la máquina `server1` y puerto 2222, ejecútalo en segundo plano para poder utilizar la consola de `server1`. Con la herramienta `netstat` puedes explorar las conexiones que hay activas en una determinada máquina. Ejecuta `netstat -ant4` en `server1`[2]. Explica toda la información que te muestra la herramienta `netstat`. A continuación, ejecuta periódicamente `netstat`, ayúdate de la herramienta `watch`:

   ```
   watch -n 0.5 netstat -ant
   ```

2. Inicia una captura de tráfico en `r1(eth0)` almacenando el contenido en el fichero `analisis-01.cap`. Desde `pc1` realiza un ataque SYN Flood hacia `server1`, a la aplicación ejecutándose en el puerto 2222, con `hping3` utilizando `--rand_source` y déjalo funcionando hasta que veas que `netstat` te muestra varias conexiones. Después interrumpe `hping3` inmediatamente con `Ctrl+C`. Interrumpe también la captura en`r1(eth0)`. Fíjate como después de haber interrumpido `hping3` todavía `server1` tiene información sobre las conexiones ¿por qué? ¿en qué estado se encuentran? ¿Crees que este ataque le hace daño a `server1`?
3. Fíjate en la cantidad de paquetes capturados en tan poco tiempo y explica el contenido de la captura.
4. Observa en la máquina IDS si se ha generado algún mensaje de alerta en el terminal.
5. Busca en el fichero `/etc/snort/rules/icmp-info.rules` la alerta que ha generado el mensaje y explícala.
6. Comprueba el fichero de captura que ha dejado `snort` con los paquetes que han provocado la alerta. Se encuentra en `/var/log/snort/tcpdump.log.*`[3]. Examina el fichero y explica por qué esos paquetes cumplen la regla que has encontrado en `/etc/snort/rules/icmp-info.rules`.

[1] Nmap es una aplicación que realiza escaneo de redes, aplicaciones y servicios. Se utiliza para Pentesting.

[2] La dirección IP 0.0.0.0 indica que se esperan recibir conexiones en cualquiera de las direcciones IPv4 (0::0 para las direcciones IPv6) que tenga configurada la máquina local (IN_ADDRANY).

[3] Si hay varios ficheros, pertenecerán a varias ejecuciones de snort, utiliza el fichero que tenga la fecha más moderna, que será el correspondiente a la última ejecución.

7. ¿Por qué crees que `snort` no considera que los mensajes TCP SYN enviados desde `pc1` sean una alerta de seguridad?[4]

Antes de comenzar los siguientes apartados, interrumpe `snort` y lánzalo nuevamente en `ids1`, de la misma forma que lo hiciste anteriormente.

3.2. smurf attack

Para realizar este ataque, se va a enviar un mensaje ICMP Echo Request a la dirección destino broadcast de subred, modificando la dirección IP origen de los paquetes con la dirección de la máquina víctima, de esta forma todas la máquinas de la subred enviarán los ICMP Echo Response a la máquina víctima.

Este ataque no se puede realizar en la actual distribución de Linux de NetGUI-NG porque el kernel tiene configurado ignorar los mensajes ICMP dirigidos al broadcast[5]. Para poder probar este ataque se va a modificar esta configuración. En todas las máquinas conectadas al `hub1` cambia la configuración ejecutando:

```
echo 0 > /proc/sys/net/ipv4/icmp_echo_ignore_broadcasts
```

Realiza una captura de tráfico `analisis-02.cap` en `r1(eth0)`. Desde `fw1` realiza las siguientes pruebas una a continuación de la otra:

- Envía un `ping` de un sólo paquete (`-c 1`) a `r1(eth0)`.
- Envía utilizando `hping3` un ataque `icmp smurf` desde la máquina `fw1` haciendo que la víctima sea `r1(eth0)`, enviando un único paquete (`-c 1`). Este ataque sólo se puede realizar desde una máquina que esté directamente conectada a la subred donde se desea realizar el broadcast ya que, los paquetes dirigidos a broadcast de subredes no se encaminan entre diferentes subredes.

Interrumpe la captura.
Responde de forma razona a las siguientes preguntas en la memoria:

1. Explica el contenido de la captura. Los paquete pueden aparecer en un orden extraño, no te preocupes por eso, trata de ver qué paquetes se han generado y sus respuestas.
2. ¿Por qué crees que un ataque de este tipo puede hacer daño a la máquina víctima?
3. Fíjate que `snort` ha detectado unas alertas ICMP, en particular la alerta `ICMP PING NMAP`, la que habías examinado en el apartado 2.3. Fíjate si los campos de esa alerta se corresponden con la captura realizada y explícalos[6].
4. ¿Por qué el primer paquete ICMP Echo Request que envió `fw1` no generó la alerta `ICMP PING NMAP`?

[4] Para la detección de este tipo de ataques se puede utilizar iptables

[5] Ten en cuenta que esta configuración podría variar en función del SO que se tenga instalado o su versión o si no está bien administrado. Los ataques de seguridad explotan sistemas que no están bien actualizados o administrados.

[6] snort cree que este paquete ha sido generado por otra de las herramientas para el escaneo de red y puertos, nmap. hping3 envía los paquetes ICMP Echo Request de forma similar a nmap