Detección de intrusiones

Sistemas Confiables
Máster Universitario en Investigación en Ciberseguridad

Noviembre 2022

Para la realización de los siguientes ejercicios utilizaremos la infraestructura creada en la práctica anterior en la que se implementó una DMZ. Todas las instalaciones se realizarán sobre Docker, y aunque existen imágenes Docker con el software pre-instalado, no las usaremos. Llevaremos a cabo la instalación de las diferentes herramientas como si se tratara de una instalación nueva en un equipo.

Entregables

Con respecto a la entrega, han de tenerse en cuenta las siguientes pautas generales:

- Es **obligatorio** grabar uno o varios vídeos, de corta duración, que permita visualizar el funcionamiento global del sistema en los términos descritos en cada ejercicio. Las pruebas a realizar están descritas en el guión y etiquetas desde T_1 a T_8.
- La entrega debe contener todos los scripts, ficheros Dockerfile y docker-compose.yml, otros ficheros de configuración, etc., editados o creados en cada ejercicio debidamente etiquetados.
- También se incluirá un fichero de manifiesto (README) indicando los ficheros que conforman la entrega y su contenido.

1. Network IDS - Snort

En este ejercicio instalaremos y configuraremos una sonda Snort version 3.1.0.0 (disponible para Ubuntu18 y Ubuntu20). Para ello usaremos la máquina *fw* como sonda de snort. Para instalar Snort, sigue los pasos indicados en la sección de documentación de la página oficial de Snort[1].

1.1. Preparación del entorno

Se utilizará snort como NIDS. Para ello, durante la práctica se usarán las máquinas *ext1*, *dmz1* y *fw*, cada una con el rol que se describe a continuación.

- La máquina *fw* se usará como sonda de Snort en todas las actividades. Permitirá detectar todo lo que ocurra, ya que es la maquina encargada de encaminar todo el tráfico.
- Los ataques se llevarán a cabo desde la máquina *ext1*, que simulará una máquina maliciosa ubicada en cualquier parte de Internet.
- Como máquina víctima se usará *int1*, de la red interna.

1.2. Preparar el fichero de configuración

El fichero de configuración de snort (`/usr/local/etc/snort/snort.lua`) contiene ocho secciones (defaults, inspection, bindings, performance, detection, filters, outputs, tweaks) que permiten localizar cada una de las configuraciones y ajustarlas en función de las necesidades.

En nuestro caso únicamente modificaremos la primera sección que contiene los ajustes por defecto. Debemos establecer las redes interna y externa: la red interna debe estar formada por las redes INT y DMZ, y todas las demás deben considerarse como externas.

[1] https://www.snort.org/documents

```
HOME_NET = [[ 10.5.2.0/24 10.5.1.0/24 ]]
EXTERNAL_NET = 'any'
```

Además vamos a utilizar las reglas de comunidad "snort3-community-rules"disponibles en la propia página de Snort. Una vez descargadas será necesario ubicarlas en el directorio /usr/local/etc/rules/. Y es necesario descomentar todas las reglas que contiene el archivo, para ello:

```
sed -i 's/# alert/alert/g' /usr/local/etc/rules/snort3-community.rules
```

Además eliminaremos 3 reglas que contienen el valor `$AIM_SERVERS` ya que para esta práctica no lo configuraremos y la sintaxis del archivo de reglas no sería correcta si las mantenemos, para ello:

```
sed -i '/$AIM_SERVERS/d' /usr/local/etc/rules/snort3-community.rules
```

1.3. Generar y analizar alertas

El primer paso tras la configuración inicial consiste en poner en marcha snort y comprobar que efectivamente es capaz de detectar intentos de intrusión y que las alertas aparecen en el registro del sistema.

Para comprobar si la sintaxis de las reglas y la configuración es correcta se puede probar mediante el siguiente comando:

```
snort -c /usr/local/etc/snort/snort.lua -R /usr/local/etc/rules/snort3-community.rules
```

Para poner en marcha snort usando la configuración preparada, basta con ejecutar el siguiente comando:

```
snort -c /usr/local/etc/snort/snort.lua -R /usr/local/etc/rules/snort3-community.rules -i eth1 -A full
```

Tened en cuenta que la interfaz en la que Snort debe escuchar, en este caso, es la interfaz que tiene conexion con el exterior. Una vez puesto en marcha snort, podremos ver en la salida por terminal aquellas conexiones que cumplan algunas de las reglas. Para generar las primeras alertas es suficiente con realizar un barrido de puertos sobre *int1*. Para ello, en *ext1*, basta con ejecutar la orden un nmap sobre 10.5.2.20.

1.4. Crear una regla de snort para detectar ataques TCP SYN flood

Una vez realizado todo el preprocesamiento, snort analiza el tráfico aplicando las reglas indicadas en el fichero de configuración. Un aspecto importante del trabajo con snort consiste en comprender el lenguaje de descripción de reglas. Para ello vamos realizar un ataque DOS sencillo a la máquina *int1*. El ataque TCP SYN flood, se trata de un tipo de ataque basado en protocolo que tiene como objetivo inundar el servidor y no permitir que responda las peticiones de clientes legítimos. Un atacante puede usar cualquier herramienta para este ataque, vamos a usar en este caso la herramienta Hping3 para atacar y generar una inundación. El objetivo es crear una regla que identifique este tipo de ataques. Para ello:

- En la máquina *fw* lanzaremos snort.
- Desde la máquina *ext1* lanzaremos el ataque a la maquina *int1* mediante el comando:

  ```
  hping3 -F --flood -p 80 10.5.2.20
  ```

- Podremos ver que Snort no detecta este tipo de ataque.

T_1 Es el momento de crear un regla para detectar el ataque, esta debe alertar si existe una conexión con origen la red externa y destino la red de la DMZ y cualquier puerto. Además, ese paquete debe tener la flag correspondiente activada. Y por último se debe mostrar un mensaje adecuado que identifique el ataque.

Para probarlo, de las misma forma que antes:

- Una vez definida la nueva regla, lanzaremos nuevamente snort.
- Desde la máquina *ext1* lanzaremos el ataque a la maquina *int1* mediante el comando:

  ```
  hping3 -F --flood -p 80 10.5.2.20
  ```

Si la configuración se ha realizado correctamente, podremos ver que en este caso Snort si detecta este tipo de ataque gracias a la regla que hemos creado.

2. Honeypot - Cowrie

En este ejercicio instalaremos y configuraremos la honeypot Cowrie. La instalación se realizará en la maquina dmz1 de la práctica anterior. Ya que simula un servicio que una organización podría tener expuesto.

Cowrie se trata de un honeypot de media interacción que simula el servicio SSH. Para instalar Cowrie, sigue los pasos indicados en la documentación oficial [2]. Cabe destacar que Cowrie necesita ser ejecutado por un usuario sin privilegios.

Aviso: es posible que a la hora de ejecutar cowrie, de un error sobre la codificación ASCII, esto se soluciona con la siguiente serie de comandos:

```
# apt-get -y install locales
# echo "LC_ALL=en_US.UTF-8" >> /etc/environment
# echo "en_US.UTF-8 UTF-8" >> /etc/locale.gen
# echo "LANG=en_US.UTF-8" > /etc/locale.conf
# locale-gen en_US.UTF-8
```

Cowrie, por defecto, dispone de un sistema de ficheros debian 5.0 simulado, con el que el atacante puede interactuar. Es posible crear nuestro propio sistema de ficheros, configurando los usuarios y los archivos que nos interesen. Para ello sería necesario crear un nuevo sistema de ficheros base en el que crear el sistema que realizar la copia que se va a utilizar en el honeypot. En nuestro caso trabajeremos con la configuración por defecto de Cowrie, pero llevaremos a cabo algunas modificaciones que hagan el sistema más real. En el archivo de configuración de Cowrie será necesario:

T_3 Modificar el hostname de la máquina con vuestro nombre.

T_4 Modificar la zona horaria de la máquina.

T_5 Modificar el archivo userdb.txt de cowrie, en el introducimos un usuario con vuestro nombre y una password con la que podréis entrar posteriormente a la honeypot.

La ejecución de Cowrie realmente no expone el puerto 22, si no que expone el puerto 2222 que es un puerto que no está reservado. Por ello, es necesario modificar la configuración de las máquinas *fw* y *dmz1*, para que permitan:

- Máquina *fw*: Acceso al puerto 22 de la maquina *dmz1* desde la red externa.
- Máquina *dmz1*: Las conexiones al puerto 22 se deben encaminar al puerto 2222 en el que realmente escucha la honeypot.

Para probar que las configuraciones se han realizado correctamente, realiza la siguientes pruebas:

T_6 Comprueba que puertos tiene expuestos la maquina *dmz1* con el comando nmap, desde la maquina *ext1*.

T_7 Prueba a conectarte a la honeypot desde la maquina de la red externa (ext1) usando el servicio que hay expuesto en el puerto 22 con el usuario y contraseña que has introducido en el archivo userdb.txt.

T_8 Comprueba que se generan los logs correspondientes a la conexión que has hecho en la maquina *dmz1*. (cowrie/var/log/cowrie.json)

[2] https://cowrie.readthedocs.io/en/latest/README.html