

Análisis de Red y Seguridad con Nmap

Tarea 1: Investigación y Preparación

1. Investigación sobre Nmap:

¿Qué es Nmap y cómo se utiliza en la administración de redes y la seguridad informática?

Nmap (Network Mapper) es una utilidad de línea de comandos de código abierto que se utiliza para explorar redes y realizar auditorías de seguridad. En esencia, Nmap envía paquetes a hosts de destino en una red y luego analiza las respuestas para determinar información sobre ellos.

En la administración de redes, Nmap se utiliza para:

- **Descubrimiento de hosts:** Identificar qué dispositivos están activos en una red.
- **Mapeo de la red:** Crear un mapa de la infraestructura de red, mostrando las conexiones entre los dispositivos.
- **Inventario de activos:** Recopilar información sobre los dispositivos en la red, como direcciones IP y MAC.
- **Monitoreo de la disponibilidad de hosts y servicios:** Verificar si los servidores y servicios críticos están en línea y funcionando.

En la seguridad informática, Nmap es una herramienta indispensable para:

- **Auditoría de seguridad (Penetration Testing):** Evaluar la seguridad de una red o sistema identificando puertos abiertos, servicios en ejecución y posibles vulnerabilidades.
- **Detección de puertos abiertos:** Determinar qué puertos están escuchando en un host, lo que indica qué servicios están disponibles.
- **Identificación de servicios y versiones:** Descubrir qué aplicaciones y versiones específicas se están ejecutando en los puertos abiertos. Esta información es crucial para identificar vulnerabilidades conocidas asociadas a esas versiones.
- **Detección de sistemas operativos:** Intentar determinar el sistema operativo que está ejecutando un host remoto.
- **Ejecución de scripts de seguridad (NSE - Nmap Scripting Engine):** Automatizar tareas de seguridad complejas, como la detección de

vulnerabilidades específicas, la realización de pruebas de fuerza bruta o la recopilación de información adicional.

¿Qué tipos de escaneos puedes realizar con Nmap? (ej. escaneo de puertos, detección de versiones, identificación de sistemas operativos).

Nmap ofrece una amplia variedad de técnicas de escaneo, cada una diseñada para un propósito específico y con diferentes características en cuanto a velocidad, sigilo y la información que pueden obtener. Algunos de los tipos de escaneos más comunes incluyen:

- **Escaneo de puertos:** El tipo de escaneo más fundamental. Se utiliza para determinar qué puertos TCP y/o UDP están abiertos, cerrados o filtrados en un host de destino.
- **Detección de versiones de servicios:** Intenta determinar el nombre y la versión de la aplicación que está escuchando en un puerto abierto.
- **Identificación de sistemas operativos:** Intenta adivinar el sistema operativo que está ejecutando el host de destino basándose en las respuestas a ciertos paquetes.
- **Escaneo de scripts (NSE):** Permite ejecutar scripts escritos en Lua para automatizar una amplia gama de tareas, desde la detección de vulnerabilidades hasta la recopilación de información específica.
- **Escaneo de hosts (Ping Scan):** Se utiliza para determinar qué hosts están activos en una red sin necesidad de escanear puertos.
- **Escaneo de topología de red:** Intenta mapear la ruta que toman los paquetes hacia el host de destino.

¿Cuáles son los tipos de escaneos más comunes en Nmap, como TCP Connect Scan, SYN Scan y UDP Scan? Explica sus diferencias y cuándo usarlos.

Estos tres tipos de escaneos son fundamentales para el escaneo de puertos TCP y UDP.

1. TCP Connect Scan (-sT):

- a. **Cómo funciona:** Este es el tipo de escaneo TCP más básico. Nmap intenta establecer una conexión TCP completa (three-way handshake: SYN -> SYN/ACK -> ACK) con cada puerto de destino. Si la conexión se establece, significa que el puerto está abierto. Luego, Nmap cierra la conexión inmediatamente. Si recibe un RST (reset) del host de destino, indica que el puerto está cerrado. Si no recibe respuesta después de varios reintentos, el puerto podría estar filtrado por un firewall.

- b. **Diferencias:** Es el tipo de escaneo TCP más confiable, ya que completa la conexión. Sin embargo, también es el más ruidoso, ya que los sistemas de destino registran la conexión completa.
- c. **Cuándo usarlo:** Se utiliza cuando no tienes privilegios suficientes para realizar un SYN Scan (requiere permisos de root en la mayoría de los sistemas) o cuando necesitas una alta confiabilidad en los resultados, incluso a costa de ser más detectable.

2. SYN Scan (-sS):

- a. **Cómo funciona:** También conocido como "half-open scan" o escaneo sigiloso. Nmap envía un paquete SYN al puerto de destino. Si el puerto está abierto, el host de destino responderá con un paquete SYN/ACK. Nmap luego envía un paquete RST para cerrar la conexión antes de que se complete el handshake. Si el puerto está cerrado, el host de destino responderá con un paquete RST. Si no hay respuesta, el puerto podría estar filtrado.
- b. **Diferencias:** Es más sigiloso que el TCP Connect Scan porque nunca completa la conexión TCP. Esto hace que sea menos probable que los sistemas de destino registren la conexión. Además, suele ser más rápido.
- c. **Cuándo usarlo:** Es el tipo de escaneo TCP más común y preferido cuando tienes privilegios de root, ya que es más rápido y menos detectable que el TCP Connect Scan.

3. UDP Scan (-sU):

- a. **Cómo funciona:** Se utiliza para escanear puertos UDP (User Datagram Protocol). Nmap envía un paquete UDP vacío (o con datos específicos de la aplicación) al puerto de destino. Si el puerto está abierto, es posible que no reciba ninguna respuesta (muchos servicios UDP no responden a paquetes vacíos) o que reciba una respuesta específica de la aplicación. Si el puerto está cerrado, el host de destino debería responder con un mensaje ICMP "Port Unreachable". Si no recibe ninguna respuesta después de varios reintentos, el puerto podría estar filtrado o el servicio podría estar respondiendo sin enviar una respuesta ICMP.
- b. **Diferencias:** El escaneo UDP es generalmente más lento y menos confiable que los escaneos TCP. Esto se debe a la naturaleza sin conexión de UDP y a que muchos firewalls y sistemas operativos limitan las respuestas ICMP de "Port Unreachable".
- c. **Cuándo usarlo:** Se utiliza cuando necesitas identificar qué servicios UDP están escuchando en un host. Es importante tener en cuenta que los resultados pueden ser menos precisos y que puede llevar más tiempo.

2.Instalación de Nmap:

Abrimos la terminal y actualizamos la lista de paquetes:


```
sudo apt update
```

Instalamos Nmap:

```
sudo apt install nmap
```

Verificamos la instalación:

```
nmap --version
```



```
salva@salva-ubuntu: ~  
Leyendo la información de estado... Hecho  
0 actualizados, 0 nuevos se instalarán, 1 reinstalados, 0 para eliminar y 13 no  
actualizados.  
Se necesita descargar 1.731 kB de archivos.  
Se utilizarán 0 B de espacio de disco adicional después de esta operación.  
Des:1 http://es.archive.ubuntu.com/ubuntu jammy-updates/universe amd64 nmap amd6  
4 7.91+dfsg1+really7.80+dfsg1-2ubuntu0.1 [1.731 kB]  
Descargados 1.731 kB en 1s (1.628 kB/s)  
(Leyendo la base de datos ... 241316 ficheros o directorios instalados actualmen  
te.)  
Preparando para desempaquetar .../nmap_7.91+dfsg1+really7.80+dfsg1-2ubuntu0.1_am  
d64.deb ...  
Desempaquetando nmap (7.91+dfsg1+really7.80+dfsg1-2ubuntu0.1) sobre (7.91+dfsg1+  
really7.80+dfsg1-2ubuntu0.1) ...  
Configurando nmap (7.91+dfsg1+really7.80+dfsg1-2ubuntu0.1) ...  
Procesando disparadores para man-db (2.10.2-1) ...  
salva@salva-ubuntu:~$ nmap --version  
Nmap version 7.80 ( https://nmap.org )  
Platform: x86_64-pc-linux-gnu  
Compiled with: liblua-5.3.6 openssl-3.0.2 nmap-libssh2-1.8.2 libz-1.2.11 libpcr  
e-8.39 libpcap-1.10.1 nmap-libdnet-1.12 ipv6  
Compiled without:  
Available nsock engines: epoll poll select  
salva@salva-ubuntu:~$
```

Tarea 2: Escaneos Básicos

1.Escaneo de una Máquina Local:

Obtenemos la dirección IP de nuestra máquina local:

```
ip a | grep "inet "
```

```
salva@salva-ubuntu:~$ ip a | grep "inet "  
    inet 127.0.0.1/8 scope host lo  
    inet 192.168.0.24/24 brd 192.168.0.255 scope global dynamic noprefixroute wl  
p3s0  
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0  
salva@salva-ubuntu:~$
```

Escaneo básico de puertos:

```
nmap 192.168.0.24
```

```
salva@salva-ubuntu:~$ nmap 192.168.0.24  
Starting Nmap 7.80 ( https://nmap.org ) at 2025-04-04 12:02 CEST  
Nmap scan report for 192.168.0.24  
Host is up (0.00017s latency).  
Not shown: 997 closed ports  
PORT      STATE SERVICE  
111/tcp   open  rpcbind  
3000/tcp  open  ppp  
8000/tcp  open  http-alt  
  
Nmap done: 1 IP address (1 host up) scanned in 13.08 seconds  
salva@salva-ubuntu:~$
```

Tres puertos están abiertos:

- **Puerto 111/tcp:** rpcbind (Este servicio es utilizado para Remote Procedure Calls (RPC), que permiten a programas en diferentes máquinas comunicarse entre sí. Es comúnmente asociado con NFS (Network File System).
- **Puerto 3000/tcp:** Grafana (interfaz de monitorización y visualización de datos).
- **Puerto 8000/tcp:** Un servidor HTTP alternativo (http-alt).

1. Escaneo de Puertos Específicos:

Para esta tarea, utilizaré un servidor que está explícitamente configurado para pruebas de seguridad y escaneo de puertos.

scanme.nmap.org

Este servidor está mantenido por los creadores de **Nmap** y está diseñado para que la gente pueda practicar sus habilidades de escaneo sin infringir ninguna regla.

Ejecutamos el comando Nmap con la dirección del servidor `scanme.nmap.org`:

```
nmap -p 22,80,443 scanme.nmap.org
```

```
salva@salva-ubuntu:~$ nmap -p 22,80,443 scanme.nmap.org
Starting Nmap 7.80 ( https://nmap.org ) at 2025-04-04 12:34 CEST
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.22s latency).

PORT      STATE SERVICE
22/tcp    open  ssh
80/tcp    open  http
443/tcp   closed https

Nmap done: 1 IP address (1 host up) scanned in 0.69 seconds
salva@salva-ubuntu:~$
```

- **Puerto 22/tcp:** Está **abierto** y el servicio que se está ejecutando es **ssh** (Secure Shell). SSH se utiliza para establecer conexiones seguras de forma remota a un sistema.
- **Puerto 80/tcp:** Está **abierto** y el servicio que se está ejecutando es **http** (Hypertext Transfer Protocol). Este es el puerto estándar para servidores web no cifrados. Podemos acceder a la página web de `scanme.nmap.org` a través de este puerto.
- **Puerto 443/tcp:** Está **cerrado**. Esto significa que no hay ningún servicio escuchando en este puerto en este momento. El puerto 443 se utiliza comúnmente para HTTPS (HTTP Secure), que es la versión segura y cifrada de HTTP.

Tarea 3: Escaneos Avanzados

1.Detección de Versiones de Servicios:

nmap -sV scanme.nmap.org

```
salva@salva-ubuntu:~$ nmap -sV scanme.nmap.org
Starting Nmap 7.80 ( https://nmap.org ) at 2025-04-04 12:50 CEST
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.18s latency).
Not shown: 993 closed ports
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.13 (Ubuntu Linux
; protocol 2.0)
80/tcp    open  http         Apache httpd 2.4.7 ((Ubuntu))
135/tcp   filtered msrpc
139/tcp   filtered netbios-ssn
445/tcp   filtered microsoft-ds
9929/tcp  open  nping-echo   Nping echo
31337/tcp open  tcpwrapped
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap
.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 26.85 seconds
salva@salva-ubuntu:~$
```

- **Puerto 22/tcp:** Está **abierto** y ejecuta **OpenSSH versión 6.6.1p1** en Ubuntu Linux. La información adicional indica que utiliza el protocolo SSH 2.0.
- **Puerto 80/tcp:** Está **abierto** y ejecuta **Apache httpd versión 2.4.7** en Ubuntu.
- **Puerto 135/tcp:** Está **filtrado** y Nmap sugiere que podría ser **msrpc** (Microsoft Remote Procedure Call). El estado "filtrado" significa que un firewall está probablemente bloqueando las peticiones de Nmap, por lo que no se puede determinar con certeza si el puerto está abierto o cerrado.
- **Puerto 139/tcp:** También está **filtrado** y Nmap sugiere que podría ser **netbios-ssn** (NETBIOS Session Service).
- **Puerto 445/tcp:** También está **filtrado** y Nmap sugiere que podría ser **microsoft-ds** (Microsoft Directory Services).
- **Puerto 9929/tcp:** Está **abierto** y ejecuta **nping-echo**, que es parte de la herramienta Nping del proyecto Nmap.
- **Puerto 31337/tcp:** Está **abierto** y el servicio se identifica como **tcpwrapped**. Esto a menudo indica que un servicio está siendo gestionado por TCP wrappers (tcpd), que pueden controlar el acceso a los servicios de red basados en direcciones IP o nombres de host.

¿Por qué es importante saber qué versiones de servicios están corriendo?

Conocer las versiones exactas de los servicios que se están ejecutando en un sistema es crucial por varias razones de seguridad:

- **Identificación de vulnerabilidades conocidas:** Cada versión de software puede tener vulnerabilidades de seguridad específicas que son descubiertas y documentadas con el tiempo (por ejemplo, en bases de datos como CVE - Common Vulnerabilities and Exposures). Si conoces la versión exacta, puedes verificar si existen vulnerabilidades conocidas para ese software en particular.
- **Gestión de parches:** Saber la versión te permite determinar si el software está actualizado con los últimos parches de seguridad. Las versiones antiguas suelen tener más vulnerabilidades sin corregir.
- **Evaluación de riesgos:** La gravedad de las vulnerabilidades puede variar entre versiones. Conocer la versión ayuda a evaluar el riesgo real que representa un servicio en tu sistema o red.
- **Planificación de la mitigación:** Si se identifica una vulnerabilidad en una versión específica, la información te permite planificar la mejor manera de mitigar el riesgo, ya sea aplicando un parche, actualizando el software o implementando otras medidas de seguridad.

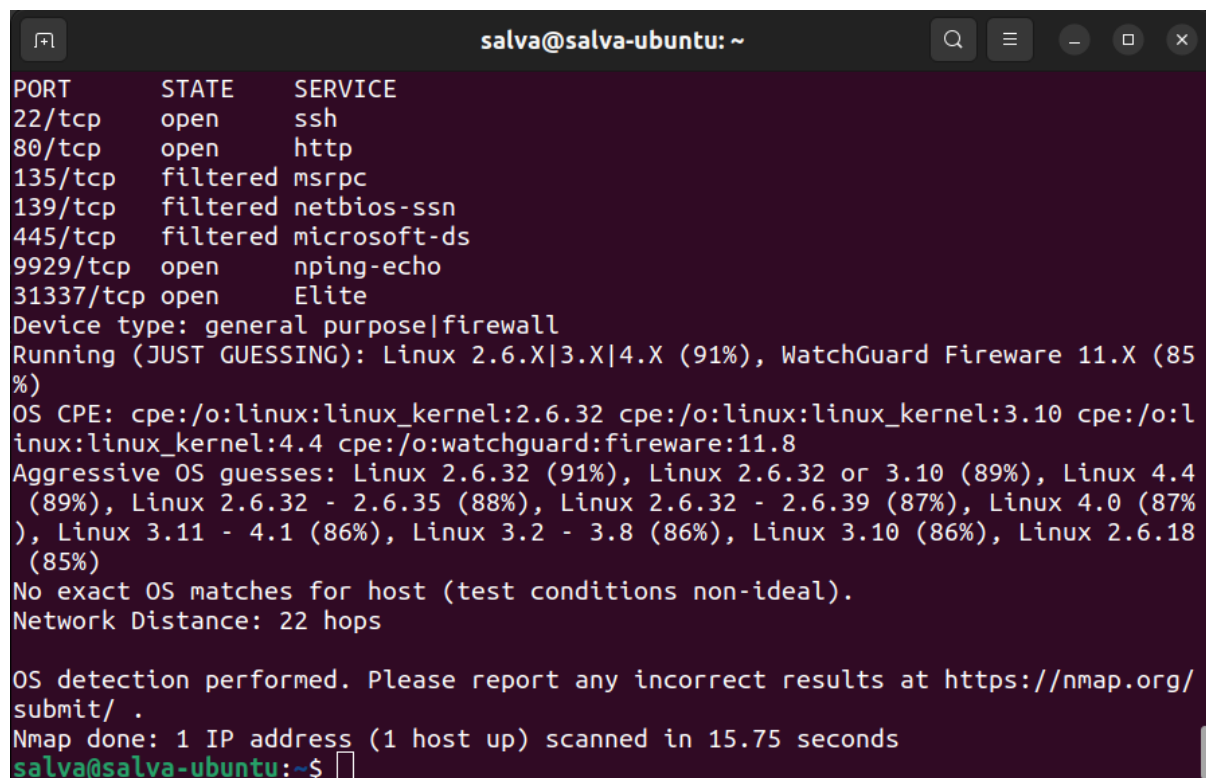
¿Cómo puede esta información ayudar a mejorar la seguridad?

La información de las versiones de los servicios ayuda a mejorar la seguridad de las siguientes maneras:

- **Priorización de actualizaciones:** Permite priorizar la actualización de los servicios que tienen versiones vulnerables o desactualizadas.
- **Implementación de contramedidas específicas:** Para ciertas versiones de software, pueden existir contramedidas o configuraciones de seguridad específicas recomendadas.
- **Auditoría de seguridad más precisa:** Facilita la realización de auditorías de seguridad más precisas al enfocar la búsqueda de vulnerabilidades en las versiones identificadas.
- **Reducción de la superficie de ataque:** Al mantener los servicios actualizados a las últimas versiones seguras, se reduce la cantidad de posibles puntos de entrada para los atacantes.
- **Respuesta a incidentes:** En caso de un incidente de seguridad, conocer las versiones de los servicios comprometidos puede ayudar a comprender cómo se produjo la intrusión y a tomar las medidas correctivas adecuadas.

2.Detección de Sistema Operativo:

```
sudo nmap -O scanme.nmap.org
```



```
salva@salva-ubuntu: ~
PORT      STATE      SERVICE
22/tcp    open       ssh
80/tcp    open       http
135/tcp   filtered  msrpc
139/tcp   filtered  netbios-ssn
445/tcp   filtered  microsoft-ds
9929/tcp  open       nping-echo
31337/tcp open       Elite
Device type: general purpose|firewall
Running (JUST GUESSING): Linux 2.6.X|3.X|4.X (91%), WatchGuard Firewall 11.X (85%)
OS CPE: cpe:/o:linux:linux_kernel:2.6.32 cpe:/o:linux:linux_kernel:3.10 cpe:/o:linux:linux_kernel:4.4 cpe:/o:watchguard:fireware:11.8
Aggressive OS guesses: Linux 2.6.32 (91%), Linux 2.6.32 or 3.10 (89%), Linux 4.4 (89%), Linux 2.6.32 - 2.6.35 (88%), Linux 2.6.32 - 2.6.39 (87%), Linux 4.0 (87%), Linux 3.11 - 4.1 (86%), Linux 3.2 - 3.8 (86%), Linux 3.10 (86%), Linux 2.6.18 (85%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 22 hops

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 15.75 seconds
salva@salva-ubuntu:~$
```


- **Tipo de dispositivo:** Nmap sugiere que el dispositivo es de propósito general o podría ser un firewall.
- **Sistema operativo:** Nmap tiene dos principales conjeturas sobre el sistema operativo:
 - **Linux en la serie de kernels 2.6.X, 3.X o 4.X**, con una **confianza del 91%**.
 - **WatchGuard Fireware versión 11.X**, con una **confianza del 85%**.
- Nmap también proporciona una lista de conjeturas más específicas sobre versiones de Linux, todas con un alto nivel de confianza. Algunos ejemplos incluyen Linux 2.6.32, Linux 2.6.32 o 3.10, Linux 4.4, y otras.
- **Sin coincidencia exacta:** Nmap indica que no encontró una coincidencia exacta en su base de datos para este host, posiblemente debido a condiciones de prueba no ideales.

Nmap está bastante seguro de que el sistema operativo que se ejecuta en `scanme.nmap.org` es alguna versión de **Linux**, aunque también considera la posibilidad de que sea un dispositivo **WatchGuard Fireware**. La falta de una coincidencia exacta sugiere que la configuración del servidor podría ser un poco inusual o que las condiciones de la red no permitieron una identificación precisa al 100%.

¿Cómo determina Nmap el sistema operativo?

Nmap utiliza una técnica llamada **fingerprinting de la pila TCP/IP** para intentar determinar el sistema operativo de una máquina remota. Funciona de la siguiente manera:

1. **Envío de sondas:** Nmap envía una serie de paquetes TCP y UDP especialmente diseñados a diferentes puertos de la máquina objetivo (tanto abiertos como cerrados).
2. **Análisis de las respuestas:** Analiza las respuestas (o la falta de ellas) a estas sondas. Las características de estas respuestas, como los valores de ciertos flags TCP, las opciones de la cabecera IP, el orden y el contenido de las respuestas, son únicos o muy característicos de ciertos sistemas operativos y sus versiones.
3. **Comparación con una base de datos:** Nmap compara estas características con una base de datos interna (llamada `nmap-os-db`) que contiene "huellas dactilares" de miles de sistemas operativos diferentes.
4. **Estimación del sistema operativo:** Si Nmap encuentra una coincidencia lo suficientemente cercana, informa el sistema operativo que probablemente está ejecutando la máquina objetivo, junto con un nivel de confianza (porcentaje).

Es importante tener en cuenta que la detección del sistema operativo no siempre es precisa al 100%, especialmente si hay firewalls u otros dispositivos de red que modifican los paquetes.

¿Por qué esta información es útil?

Conocer el sistema operativo de una máquina objetivo es muy útil por varias razones:

- **Investigación de vulnerabilidades específicas:** Las vulnerabilidades de seguridad a menudo son específicas de un sistema operativo y una versión en particular. Si conoces el sistema operativo, puedes enfocar tu búsqueda de vulnerabilidades en aquellas que son relevantes para ese sistema.
- **Selección de exploits:** Si estás realizando pruebas de penetración, muchos exploits están diseñados para funcionar solo en sistemas operativos específicos.
- **Comprensión del entorno:** Saber el sistema operativo te da una idea del tipo de software y servicios que probablemente estén ejecutándose en la máquina.
- **Mapeo de la red:** En una red, identificar los sistemas operativos de diferentes dispositivos puede ayudar a comprender la arquitectura de la red y el rol de cada máquina.
- **Respuesta a incidentes:** En caso de un incidente de seguridad, conocer el sistema operativo de un sistema comprometido puede ayudar a determinar la causa de la intrusión y las posibles acciones del atacante.

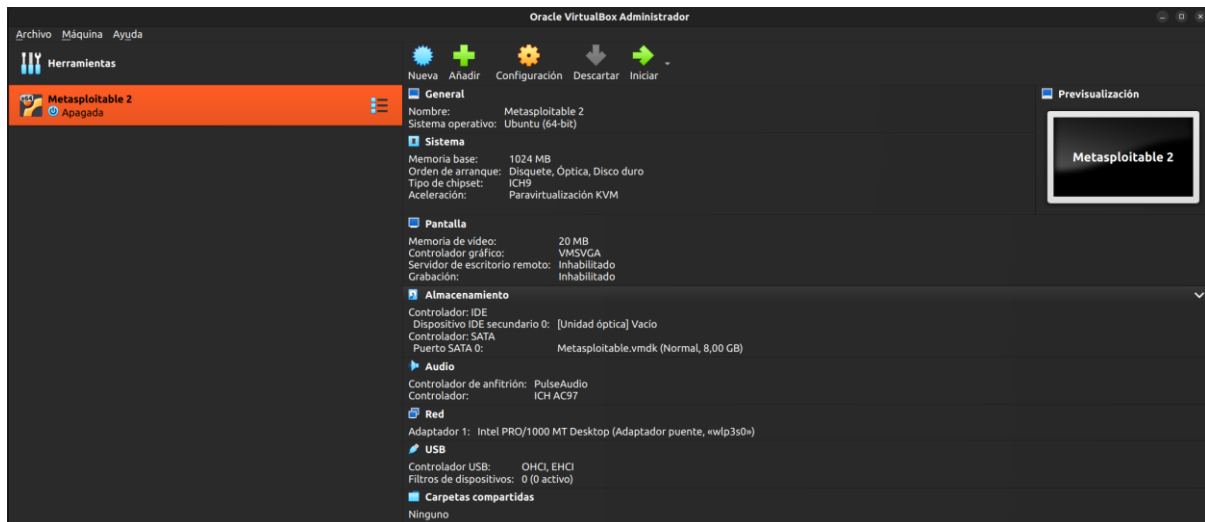
Tarea 4: Pruebas de Seguridad

1. Escaneo de Vulnerabilidades:

Escanear servidores remotos sin permiso es ilegal y poco ético. Así que la mejor opción para esta tarea es utilizar un **servidor de pruebas que hayamos configurado**. Una excelente opción para esto es **Metasploitable 2**.

Metasploitable 2 es una máquina virtual Linux que está **deliberadamente configurada con vulnerabilidades** para que los profesionales de la seguridad y estudiantes puedan practicar sus habilidades de pentesting y escaneo de vulnerabilidades en un entorno seguro y controlado.

Así que descargo **Metasploitable 2** y lo monto en **VirtualBox**.



Iniciamos sesión en **Metasploitable 2**, y ejecutamos el siguiente comando para ver la configuración de red:

Ifconfig

```
Metasploitable 2 [Corriendo] - Oracle VirtualBox
To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:7e:0b:dc
          inet addr:192.168.0.47  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe7e:bdc/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:359 errors:0 dropped:0 overruns:0 frame:0
          TX packets:83 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:26256 (25.6 KB)  TX bytes:8814 (8.6 KB)
          Base address:0xd010 Memory:e2200000-e2220000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:109 errors:0 dropped:0 overruns:0 frame:0
          TX packets:109 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:27661 (27.0 KB)  TX bytes:27661 (27.0 KB)

msfadmin@metasploitable:~$
```

Ejecutamos el comando de escaneo de vulnerabilidades, con la dirección IP de Metasploitable 2.

nmap --script=vuln 192.168.0.47

```
salva@salva-ubuntu: ~  
le Blog / FCKEditor File Upload  
| /admin/jscript/upload.html: Lizard Cart/Remote File upload  
|_ /webdav/: Potentially interesting folder  
  
Host script results:  
|_samba-vuln-cve-2012-1182: ERROR: Script execution failed (use -d to debug  
)  
|_smb-double-pulsar-backdoor: ERROR: Script execution failed (use -d to deb  
ug)  
|_smb-vuln-conficker: ERROR: Script execution failed (use -d to debug)  
|_smb-vuln-cve-2017-7494: ERROR: Script execution failed (use -d to debug)  
|_smb-vuln-ms06-025: ERROR: Script execution failed (use -d to debug)  
|_smb-vuln-ms07-029: ERROR: Script execution failed (use -d to debug)  
|_smb-vuln-ms08-067: ERROR: Script execution failed (use -d to debug)  
|_smb-vuln-ms10-054: false  
|_smb-vuln-ms10-061: ERROR: Script execution failed (use -d to debug)  
|_smb-vuln-ms17-010: ERROR: Script execution failed (use -d to debug)  
|_smb-vuln-regsvcs-dos: ERROR: Script execution failed (use -d to debug)  
  
Nmap done: 1 IP address (1 host up) scanned in 336.60 seconds  
salva@salva-ubuntu:~$
```

El resultado del escaneo muestra varias vulnerabilidades potenciales detectadas por los scripts NSE de Nmap:

- **Puerto 21/tcp (ftp):**
 - **VULNERABLE: vsFTPD version 2.3.4 backdoor (CVE-2011-2523):** Esta es una vulnerabilidad muy grave. Nmap detectó un backdoor en la versión 2.3.4 de vsFTPD que permite la ejecución remota de código. Nmap incluso pudo ejecutar el comando `id` y obtener acceso de root, lo que significa que un atacante podría tomar el control completo del sistema a través de este puerto.
- **Puerto 80/tcp (http):**
 - **http-csrf:** Nmap encontró posibles vulnerabilidades de Cross-Site Request Forgery (CSRF) en varias páginas web. CSRF permite a un atacante obligar a un usuario autenticado a realizar acciones no deseadas.
 - **http-slowloris-check:** El servidor web es vulnerable al ataque de denegación de servicio Slowloris.
 - **http-sql-injection:** Nmap identificó varios puntos de inyección SQL, lo que podría permitir a un atacante acceder, modificar o eliminar datos de la base de datos.
 - **http-trace:** El método HTTP TRACE está habilitado, lo que podría utilizarse en combinación con otros ataques.
- **Puerto 1099/tcp (rmiregistry):**

- **VULNERABLE: RMI registry default configuration remote code execution vulnerability:** Esta vulnerabilidad en el registro RMI también permite la ejecución remota de código, lo que podría permitir a un atacante ejecutar comandos arbitrarios en el sistema.
- **Puerto 5432/tcp (postgresql):**
 - **ssl-ccs-injection (CVE-2014-0224):** Vulnerable a un ataque de hombre en el medio (MITM).
 - **ssl-dh-params:** Utiliza parámetros Diffie-Hellman débiles, lo que permite la interceptación de tráfico.
 - **ssl-poodle (CVE-2014-3566):** Vulnerable al ataque POODLE, otro ataque MITM.
- **Puerto 6667/tcp (irc): irc-unrealircd-backdoor:** El servidor UnrealIRCd parece ser una versión troyanizada, lo que sugiere la presencia de una puerta trasera.
- **Puerto 8180/tcp:**
 - **http-cookie-flags:** Nmap detectó que la cookie JSESSIONID no está configurando el flag HttpOnly. Este flag es importante para evitar que los scripts del lado del cliente (como JavaScript) accedan a la cookie, lo que puede ayudar a mitigar los ataques de Cross-Site Scripting (XSS).

El análisis de vulnerabilidades en Metasploitable 2 demostró la presencia de numerosas vulnerabilidades, algunas de ellas críticas. Estos resultados subrayan la importancia de mantener los sistemas actualizados y parcheados, configurar los servicios de forma segura, realizar evaluaciones de seguridad periódicas, y comprender los riesgos asociados con las vulnerabilidades comunes.