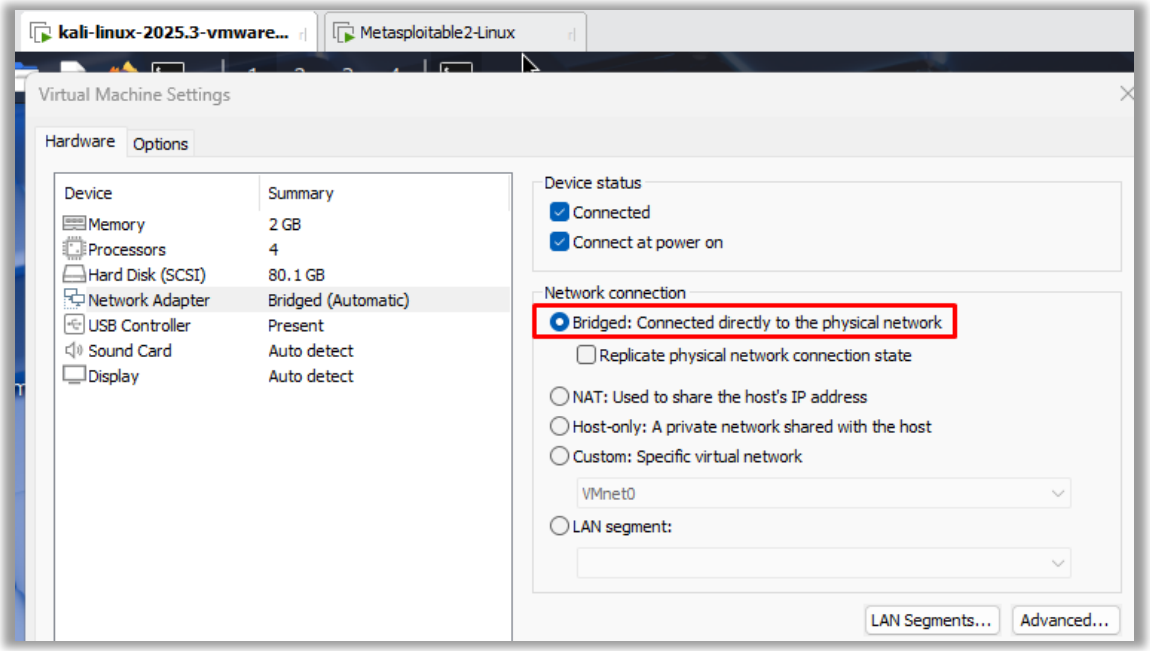
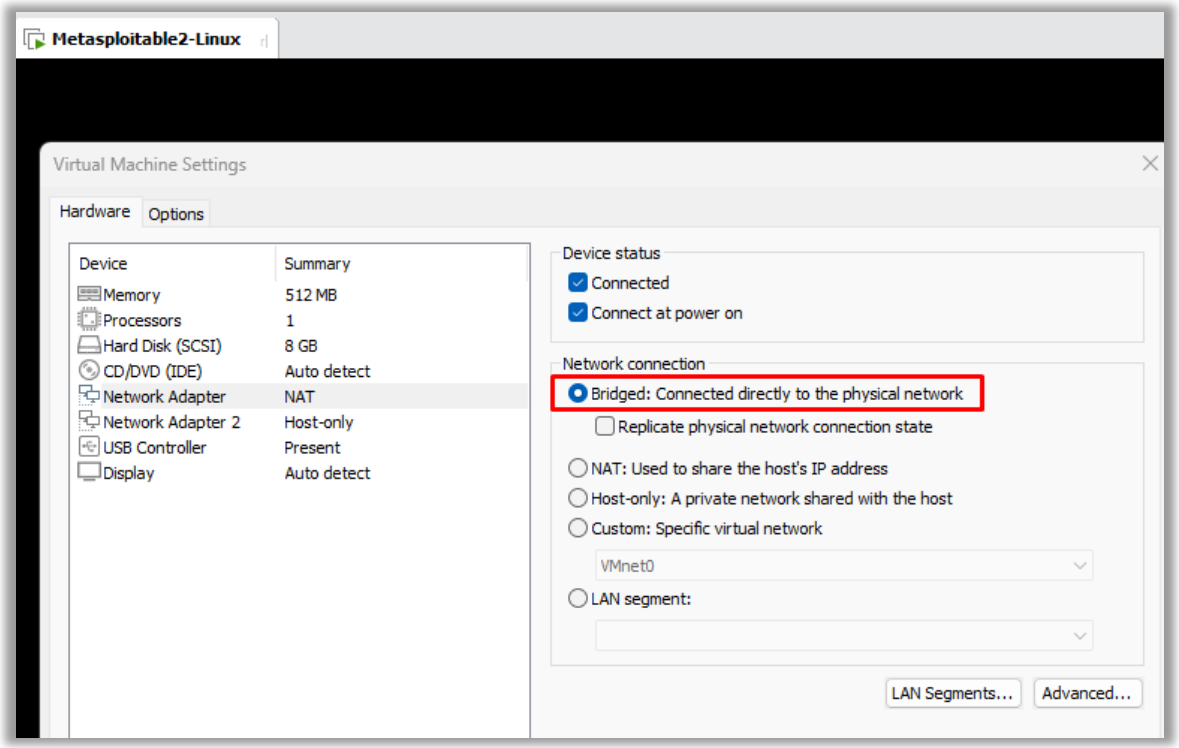
Práctica: Escaneo y Mitigación de Vulnerabilidades en Metasploitable 2

Parte 1: Configuración del Entorno

1.1 Instalación de Metasploitable 2

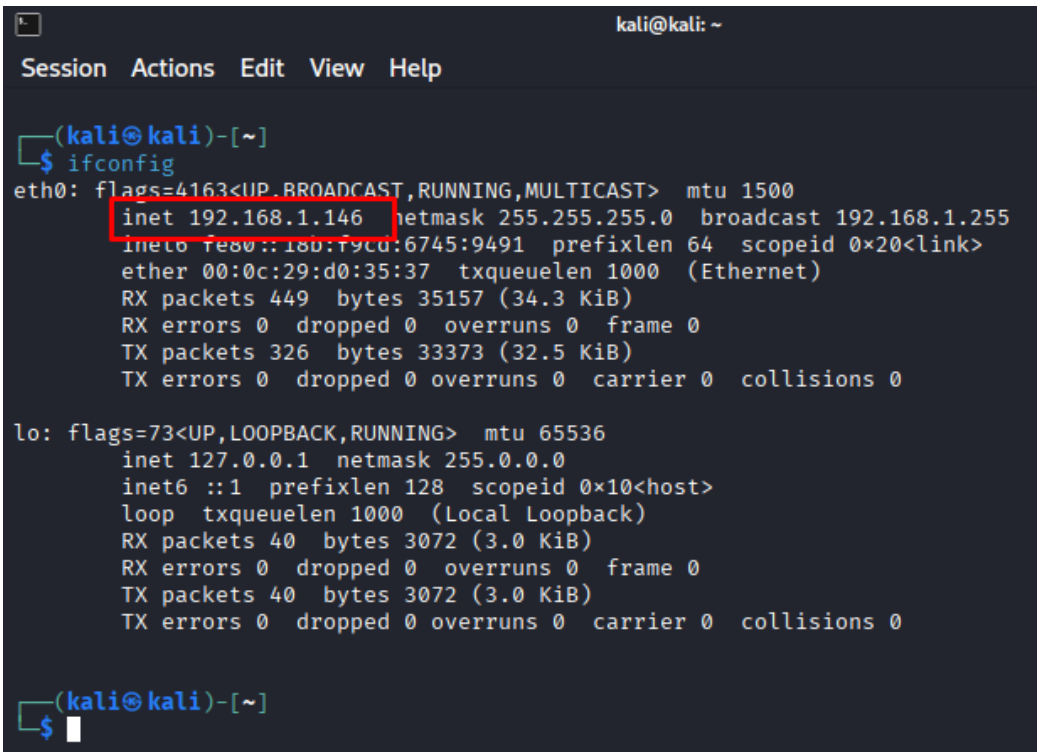
1. Descarga la imagen de Metasploitable 2 desde el sitio oficial de Rapid7.
2. Importa la imagen en VMware y configura la red en modo Puente o Bridge, para que pertenezcan a la misma red.





3. Inicia la máquina virtual y asegúrate de que tienes conectividad con tu máquina anfitriona, Kali y Metasploitable

4. Obten la IP de Metasploitable 2 con: ifconfig



```
msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0c:29:3e:6d:bb
          inet addr:192.168.1.147  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe3e:6dbb/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:39 errors:0 dropped:0 overruns:0 frame:0
          TX packets:67 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:4370 (4.2 KB)  TX bytes:7242 (7.0 KB)
          Interrupt:17 Base address:0x2000

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:96 errors:0 dropped:0 overruns:0 frame:0
          TX packets:96 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:21437 (20.9 KB)  TX bytes:21437 (20.9 KB)

msfadmin@metasploitable:~$ _
```

Confirmación de Alcance de la Red con ping:

```
┌──(kali㉿kali)-[~]
└─$ ping 192.168.1.147
PING 192.168.1.147 (192.168.1.147) 56(84) bytes of data.
64 bytes from 192.168.1.147: icmp_seq=1 ttl=64 time=0.773 ms
64 bytes from 192.168.1.147: icmp_seq=2 ttl=64 time=0.436 ms
64 bytes from 192.168.1.147: icmp_seq=3 ttl=64 time=0.488 ms
64 bytes from 192.168.1.147: icmp_seq=4 ttl=64 time=0.540 ms
64 bytes from 192.168.1.147: icmp_seq=5 ttl=64 time=0.442 ms
64 bytes from 192.168.1.147: icmp_seq=6 ttl=64 time=0.526 ms
^C
── 192.168.1.147 ping statistics ──
6 packets transmitted, 6 received, 0% packet loss, time 5108ms
rtt min/avg/max/mdev = 0.436/0.534/0.773/0.113 ms

┌──(kali㉿kali)-[~]
└─$
```

```
msfadmin@metasploitable:~$ ping 192.168.1.146
PING 192.168.1.146 (192.168.1.146) 56(84) bytes of data.
64 bytes from 192.168.1.146: icmp_seq=1 ttl=64 time=0.401 ms
64 bytes from 192.168.1.146: icmp_seq=2 ttl=64 time=0.683 ms
64 bytes from 192.168.1.146: icmp_seq=3 ttl=64 time=0.317 ms
64 bytes from 192.168.1.146: icmp_seq=4 ttl=64 time=0.555 ms
64 bytes from 192.168.1.146: icmp_seq=5 ttl=64 time=0.316 ms
64 bytes from 192.168.1.146: icmp_seq=6 ttl=64 time=0.477 ms
64 bytes from 192.168.1.146: icmp_seq=7 ttl=64 time=0.487 ms

--- 192.168.1.146 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 5994ms
rtt min/avg/max/mdev = 0.316/0.462/0.683/0.123 ms
msfadmin@metasploitable:~$
```

1.2 Configuración de Herramientas de Escaneo

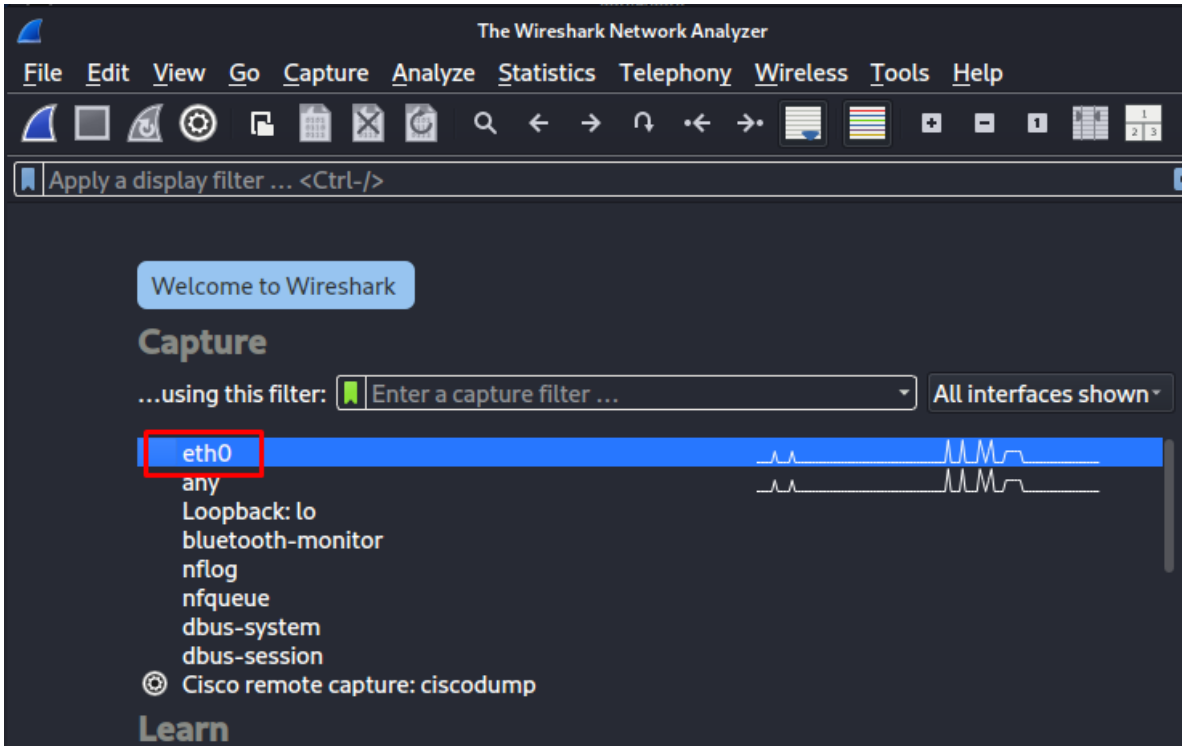
1. Verifica que **Nmap** y **Wireshark** estén correctamente instalados en el sistema anfitrión.

- En Linux, usa:

```
┌──(kali㉿kali)-[~]
└─$ sudo apt update && sudo apt install nmap wireshark -y
[sudo] password for kali:
Get:1 http://archive-4.kali.org/kali kali-rolling InRelease [34.0 kB]
Get:2 http://archive-4.kali.org/kali kali-rolling/main amd64 Packages [20.9 MB]
Get:3 http://archive-4.kali.org/kali kali-rolling/main amd64 Contents (deb) [52.3 MB]
Fetched 73.3 MB in 8s (9,649 kB/s)
3 packages can be upgraded. Run 'apt list --upgradable' to see them.
nmap is already the newest version (7.95+dfsg-3kali1).
nmap set to manually installed.
wireshark is already the newest version (4.4.9-1).
wireshark set to manually installed.
The following packages were automatically installed and are no longer required:
  amass-common        libportmidi0   python3-bluepy                      python3-protobuf
  libbluray2          librav1e0.7    python3-click-plugins               python3-zombie-imp
  libbson-1.0-0t64    libtheoradec1  python3-gpg                         samba-ad-dc
  libjs-jquery-ui     libtheoraenc1  python3-kismetcapturebtgeiger       samba-ad-provision
  libjs-underscore    libudfread0    python3-kismetcapturefreaklabszigbee samba-dsdb-modules
  libmongoc-1.0-0t64  libx264-164    python3-kismetcapturertl433
  libmongocrypt0      libxml2        python3-kismetcapturertladsb
  libplacebo349       libyelp0       python3-kismetcapturertlamr
Use 'sudo apt autoremove' to remove them.

Summary:
  Upgrading: 0, Installing: 0, Removing: 0, Not Upgrading: 3

┌──(kali㉿kali)-[~]
└─$
```

2. Asegúrate de que **Wireshark** esté configurado para capturar tráfico en la interfaz de red adecuada.

3. Asegúrate de tienes configurado y listo **Git y GitHub** para gestionar la documentación

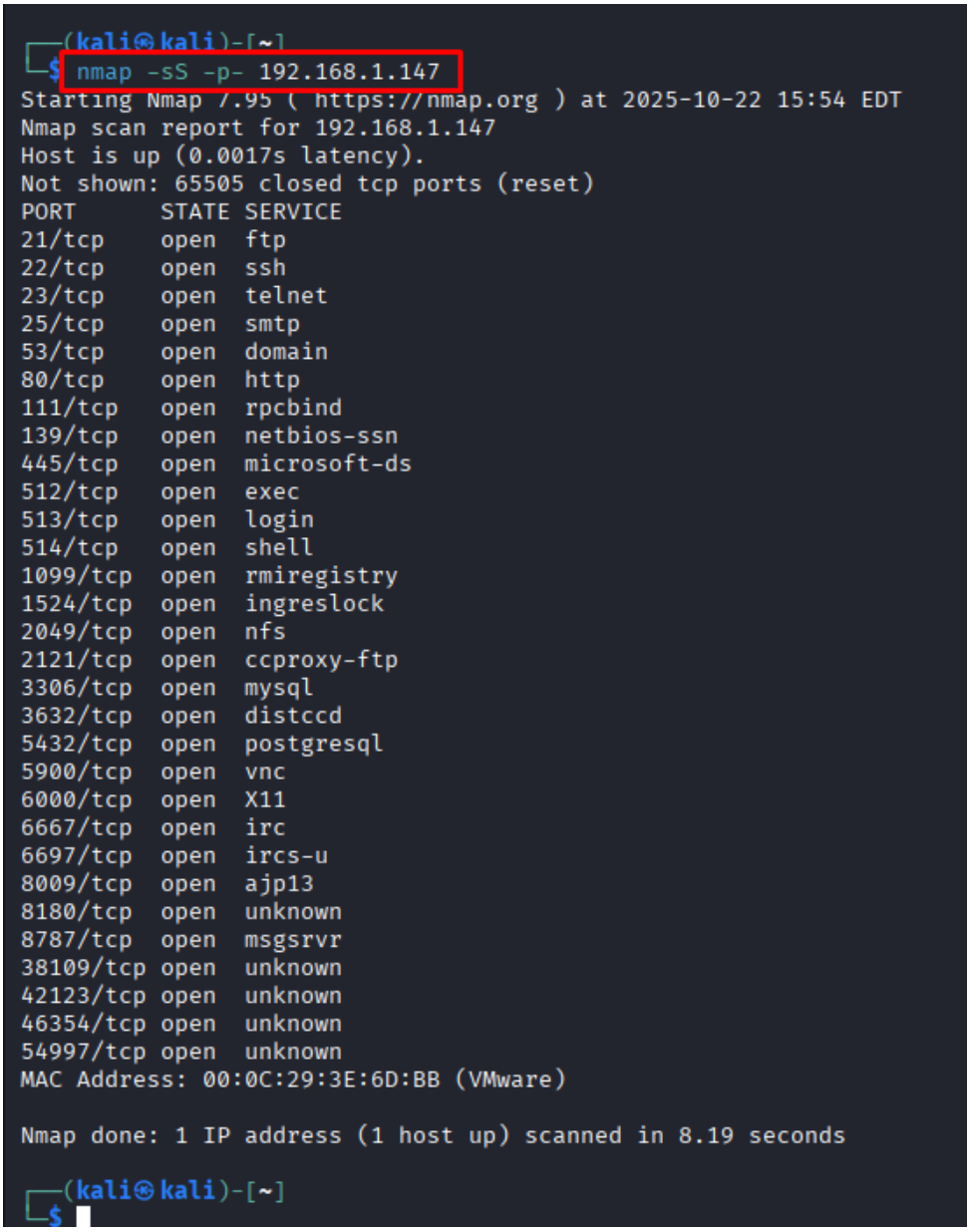


Parte 2: Escaneo de Puertos y Servicios con Nmap

2.1 Escaneo de Puertos Básico (Identificación de Puertos Abiertos)

1. En la máquina anfitriona, abre una terminal y realiza un escaneo de todos los puertos de Metasploitable 2:

Utilizaremos el escaneo SYN (-sS), que es rápido y "sigiloso" (menos invasivo), para escanear todos los 65535 puertos (-p-).



2. **Análisis de Resultados**: Examina los puertos abiertos e identifica los servicios principales, que incluirán:

FTP (21), **SSH (22)**, **Telnet (23)**, **HTTP (80)**, **MySQL (3306)**.

Puertos Abiertos Identificados:

- **21/tcp** (Servicio ftp)
- **22/tcp** (Servicio ssh)
- **23/tcp** (Servicio telnet)
- **80/tcp** (Servicio http)
- **3306/tcp** (Servicio mysql)

- Además, se encontraron otros servicios, como SMTP (25), DNS/Domain (53), y PostgreSQL (5432).

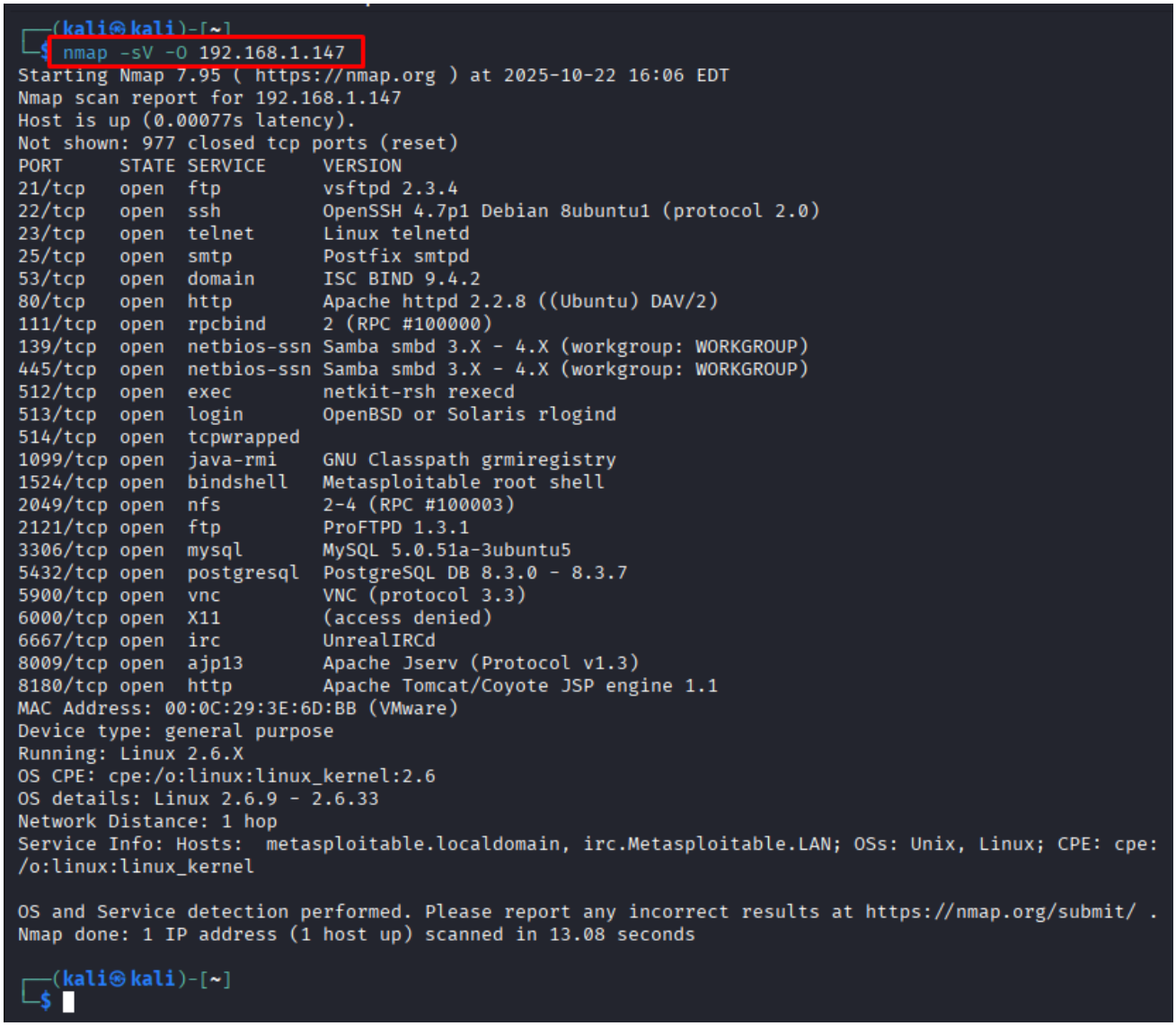
3. Tarea: Documenta los riesgos asociados a cada puerto abierto y servicio encontrado.

Puerto	Servicio	Riesgos Asociados
21/tcp	**FTP**	**Riesgo Crítico.** Se espera que sea la versión **vsftpd 2.3.4**, que contiene una **puerta trasera (backdoor)** que permite la ejecución de comandos remotos sin autenticación.
22/tcp	**SSH**	**Riesgo Alto.** Suele estar configurado para permitir el *login* como *root* con credenciales débiles o por defecto (msfadmin/msfadmin).
23/tcp	**Telnet**	**Riesgo Alto.** Telnet transmite toda la sesión, incluyendo credenciales (usuario y contraseña), **en texto plano (sin cifrar)**. Esto permite a un atacante en la misma red capturar fácilmente las credenciales con una herramienta como Wireshark.
80/tcp	**HTTP**	**Riesgo Alto.** Aloja aplicaciones web conocidas por ser vulnerables (como DVWA) que permiten ataques de **Inyección SQL**, *Cross-Site Scripting* (XSS) y ejecución remota de código.
3306/tcp	**MySQL**	**Riesgo Alto.** El servidor de base de datos puede ser vulnerable a ataques de *fuerza bruta* o permitir el acceso con credenciales débiles o por defecto, lo que comprometería la información almacenada.

2.2 Escaneo de Versiones y Sistema Operativo

1. Realiza un escaneo para identificar la versión del sistema operativo y los servicios específicos en cada puerto abierto:

Utilizaremos la opción de detección de versión (-sV) y la detección del sistema operativo (-O).



2. Análisis de Resultados: Investiga vulnerabilidades asociadas a cada versión detectada.

- Usa bases de datos como **CVE** o **Exploit-DB** para encontrar exploits. (**ANEXO I**)

A. Análisis de vsftpd 2.3.4 (Puerto 21)

- **Vulnerabilidad:** Este es el servicio más crítico. La versión **vsftpd 2.3.4** contiene una puerta trasera (backdoor) intencionalmente colocada.
- **Identificador CVE: CVE-2011-2523**.
- **Impacto:** Permite a un atacante ejecutar comandos arbitrarios de forma remota en el servidor sin necesidad de autenticación.

- **Exploit-DB:** Existe el exploit **Exploit-DB: 38132** que aprovecha esta puerta trasera.

B. Análisis de Apache httpd 2.2.8 (Puerto 80)

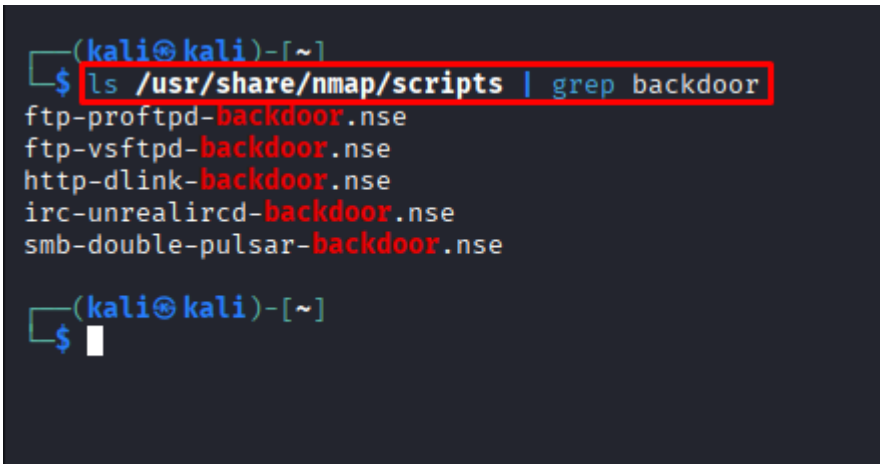
- **Vulnerabilidad:** Esta versión de Apache también es antigua.
- **Riesgo:** Permite acceso a servicios web vulnerables como **DVWA y Mutillidae** (que se encuentran en Metasploitable 2), que son objetivos directos para ataques de **Inyección SQL** y **XSS**.
- **Investigación Adicional:** La versión 2.2.8 tiene vulnerabilidades conocidas como **CVE-2009-1891** (ejecución remota de código en ciertas configuraciones de módulos) o **CVE-2008-2364**.

C. Otros Servicios Vulnerables

- **MySQL 5.0.51a-3ubuntu5:** Versión muy antigua que probablemente tiene credenciales débiles o por defecto.
- **PostgreSQL 8.3.7:** También versión obsoleta, vulnerable a inyecciones SQL en ciertas funciones.

2.3 Escaneo de Puertas Traseras con Scripts NSE de Nmap

1. Los scripts NSE (Nmap Scripting Engine) permiten identificar puertas traseras específicas en servicios vulnerables. Ejecuta el siguiente comando:

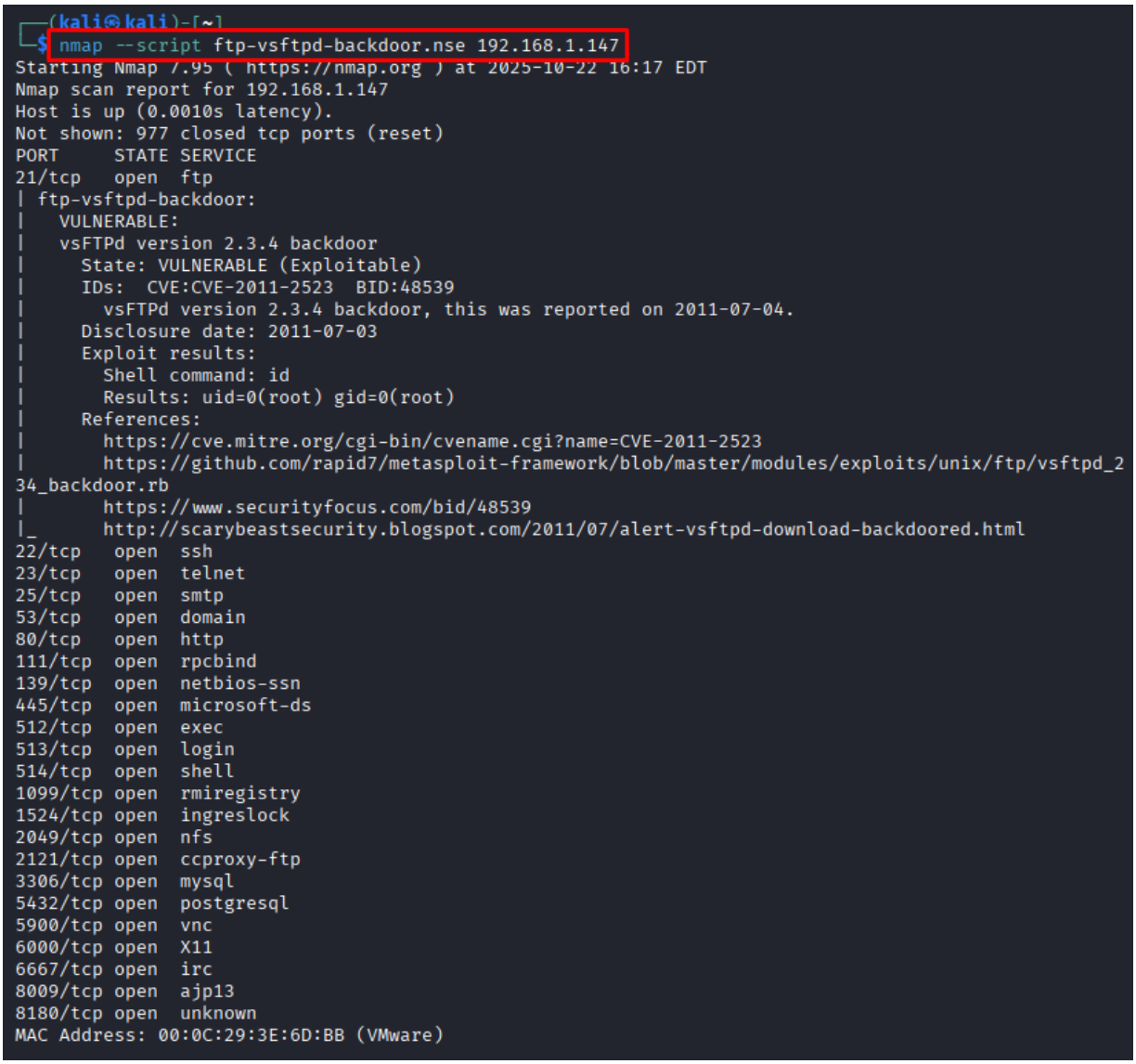


El comando ls /usr/share/nmap/scripts | grep backdoor ha identificado el script crucial:

- **Script a Usar:** ftp-vsftpd-backdoor.nse

Ahora, ejecutaremos este script contra Metasploitable 2 para confirmar que la versión **vsftpd 2.3.4** es vulnerable a la puerta trasera

nmap --script ftp-vsftpd-backdoor.nse 192.168.1.147



2. Análisis de Resultados: Revisa los resultados para detectar puertas traseras como:

- **vsftpd 2.3.4** (puerta trasera de FTP) o **ProFTPd** (ejecución remota de

código).

El escaneo con el script ftp-vsftpd-backdoor.nse confirmó la existencia de la puerta trasera:

- **Vulnerabilidad Detectada:** La versión del servidor FTP, **vsftpd 2.3.4**, es vulnerable a una puerta trasera.
- **Confirmación de Exploit:** El script ejecutó exitosamente el comando id de Linux.

- **Nivel de Acceso:** La respuesta del comando fue uid=0(root), lo que significa que un atacante puede obtener control de root (máximo privilegio) del sistema sin necesidad de autenticación.

3. Documenta las puertas traseras identificadas y los riesgos que representan.

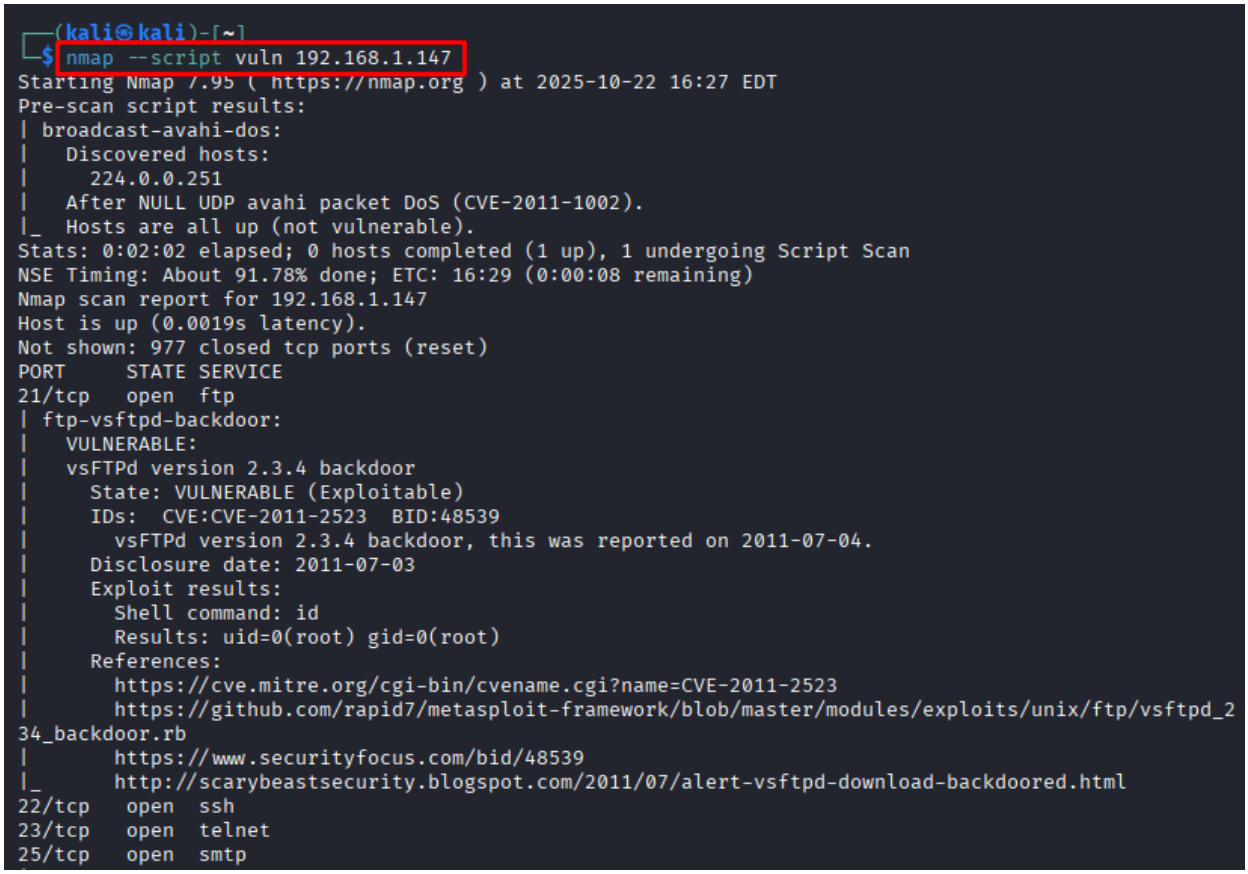
Puerta Trasera Identificada

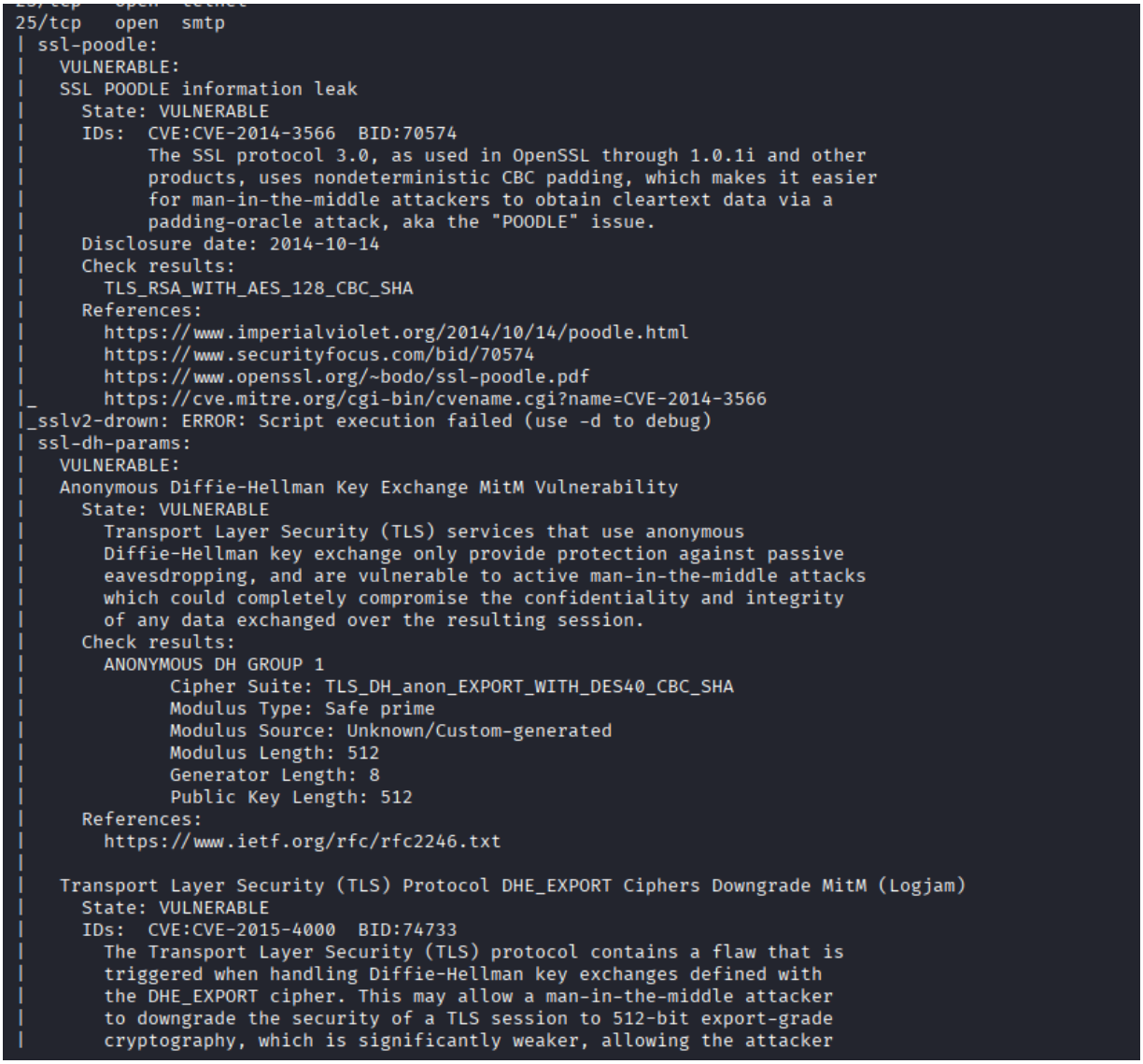
Se ejecutó el script NSE ftp-vsftpd-backdoor.nse contra la IP objetivo (192.168.1.147)

Hallazgo	Valor Confirmado
Servicio Vulnerable	vsftpd versión 2.3.4
Identificador CVE	CVE-2011-2523
Prueba de Exploit	La ejecución de la Shell command: id devolvió uid=0(root)

2.4 Escaneo de Vulnerabilidades Adicionales

1. Usa scripts NSE de vulnerabilidad general en servicios específicos:





2. **Análisis de Resultados**: Este comando ejecuta varios scripts de vulnerabilidad en servicios como FTP, MySQL, Apache y Tomcat.

- Documenta los resultados y prioriza los servicios vulnerables para mitigación.

  - El escaneo con --script vuln no solo reconfirmó la vulnerabilidad del FTP, sino que también identificó otras fallas y debilidades:

Servicio	Puerto	Vulnerabilidad Detectada	Riesgo y Priorización
FTP (vsftpd 2.3.4)	21	**ftp-vsftpd-backdoor** (CVE-2011-2523)	**CRÍTICO (Prioridad 1)**: Acceso remoto de **root**.
HTTP (Apache 2.2.8)	80	**http-enum**: Detección de posibles carpetas administrativas (/admin/, /login/, etc.). **http-slowloris-check**: Posible DoS (CVE-2007-6750).	**ALTO (Prioridad 3)**: Exposición de rutas sensibles y riesgo de Denegación de Servicio.
SMTP	25	**ssl-poodle** (CVE-2014-3566).	**ALTO (Prioridad 2)**: Falla de SSL 3.0 que permite la fuga de información cifrada (*plaintext*).
SMB (Samba)	445	**smb-vuln-ms10-061** y **smb-vuln-regsvc-dos** (fallos de ejecución de *scripts*).	**MEDIO/ALTO**: Indica que el servicio SMB/NetBIOS está activo y expuesto, a menudo con fallas de seguridad conocidas.

Parte 3: Captura y Análisis del Tráfico de Red con Wireshark

3.1 Captura de Tráfico Generado por Escaneos de Nmap

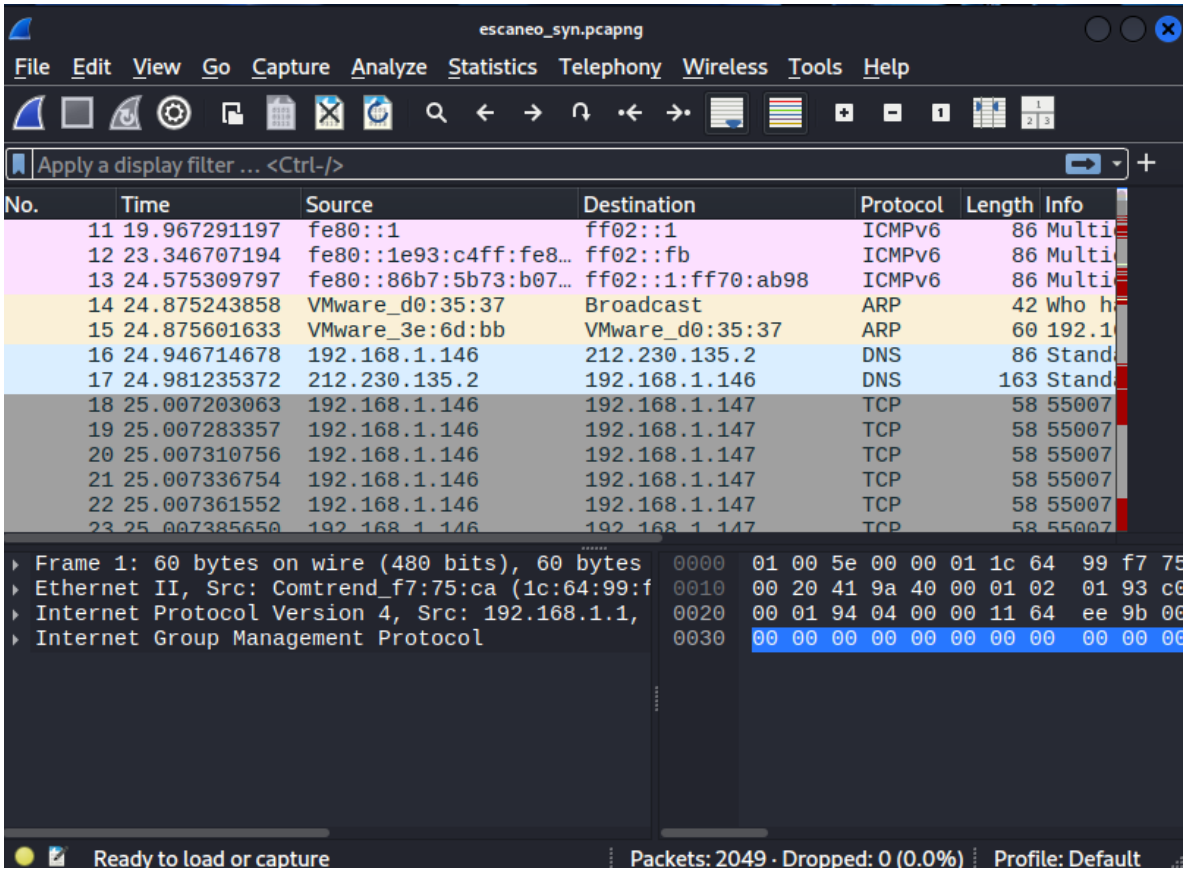
1. Inicia **Wireshark** y selecciona la interfaz de red que conecta con Metasploitable 2.
2. **Inicia la captura de tráfico** antes de lanzar un nuevo escaneo de Nmap.
3. **Detén la captura** cuando el escaneo termine y guarda el archivo .pcap.
4. Documenta los paquetes capturados durante el escaneo y explica el proceso de detección de puertos.

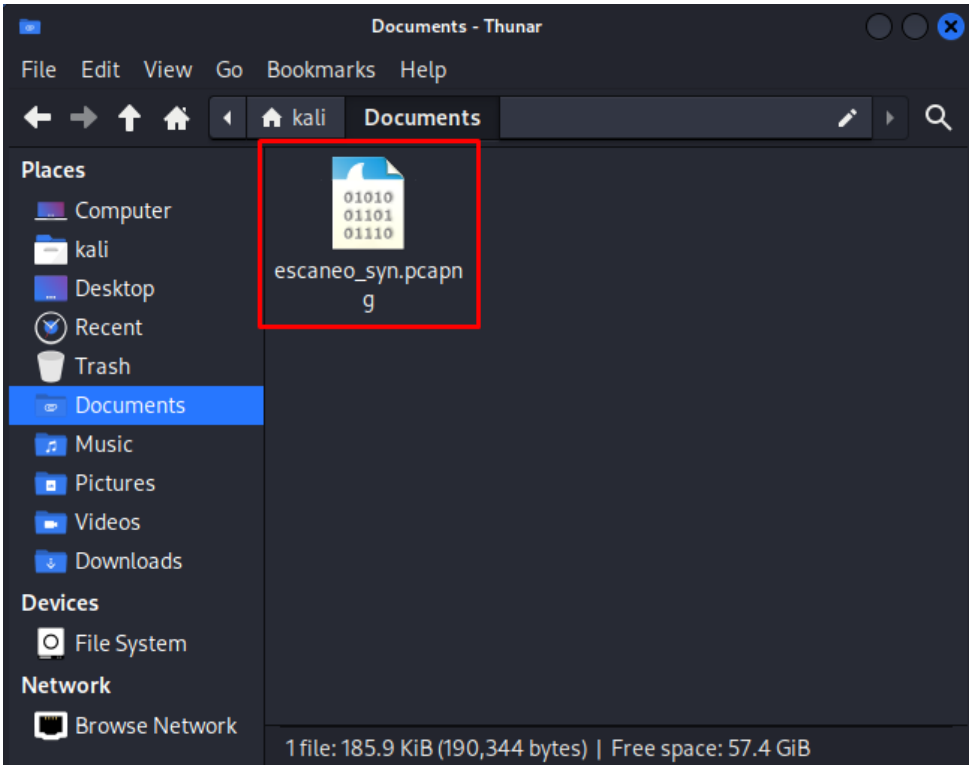
3.2 Filtrado y Análisis del Tráfico Capturado

1. Utiliza filtros para analizar el tráfico específico de cada servicio:

- **Escaneo SYN**: **tcp**.flags.syn == 1 && **tcp**.flags.ack == 0

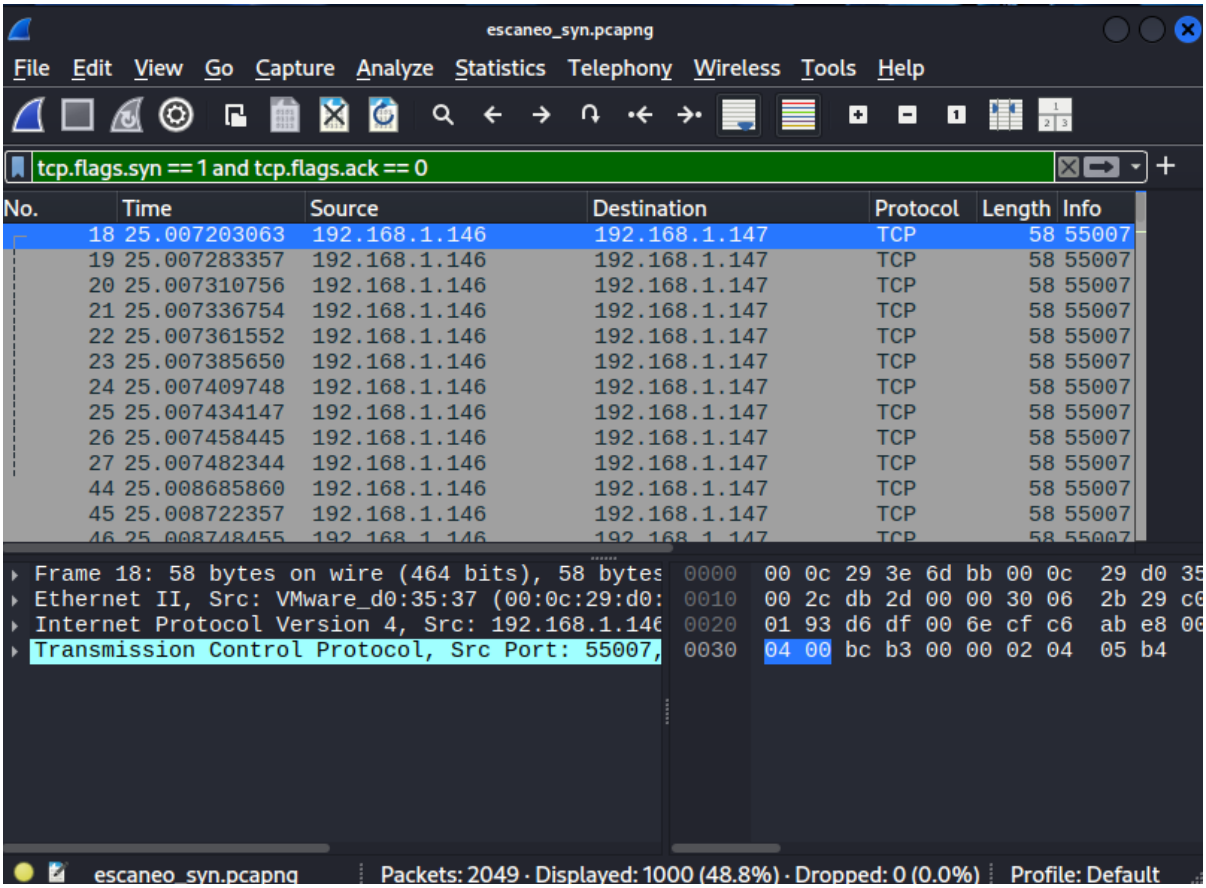
```
┌──(kali㉿kali)-[~]
└─$ nmap -sS 192.168.1.147
Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-22 16:46 EDT
Nmap scan report for 192.168.1.147
Host is up (0.0022s latency).
Not shown: 977 closed tcp ports (reset)
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
23/tcp   open  telnet
25/tcp   open  smtp
53/tcp   open  domain
80/tcp   open  http
111/tcp  open  rpcbind
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
512/tcp  open  exec
513/tcp  open  login
514/tcp  open  shell
1099/tcp open  rmiregistry
1524/tcp open  ingreslock
2049/tcp open  nfs
2121/tcp open  ccproxy-ftp
3306/tcp open  mysql
5432/tcp open  postgresql
5900/tcp open  vnc
6000/tcp open  X11
6667/tcp open  irc
8009/tcp open  ajp13
8180/tcp open  unknown
MAC Address: 00:0C:29:3E:6D:BB (VMware)

Nmap done: 1 IP address (1 host up) scanned in 0.32 seconds

┌──(kali㉿kali)-[~]
└─$
```





Para ver los paquetes **SYN** que inician el escaneo, usamos el siguiente filtro:

tcp.flags.syn == 1 and tcp.flags.ack == 0



Identificación de Direcciones IP

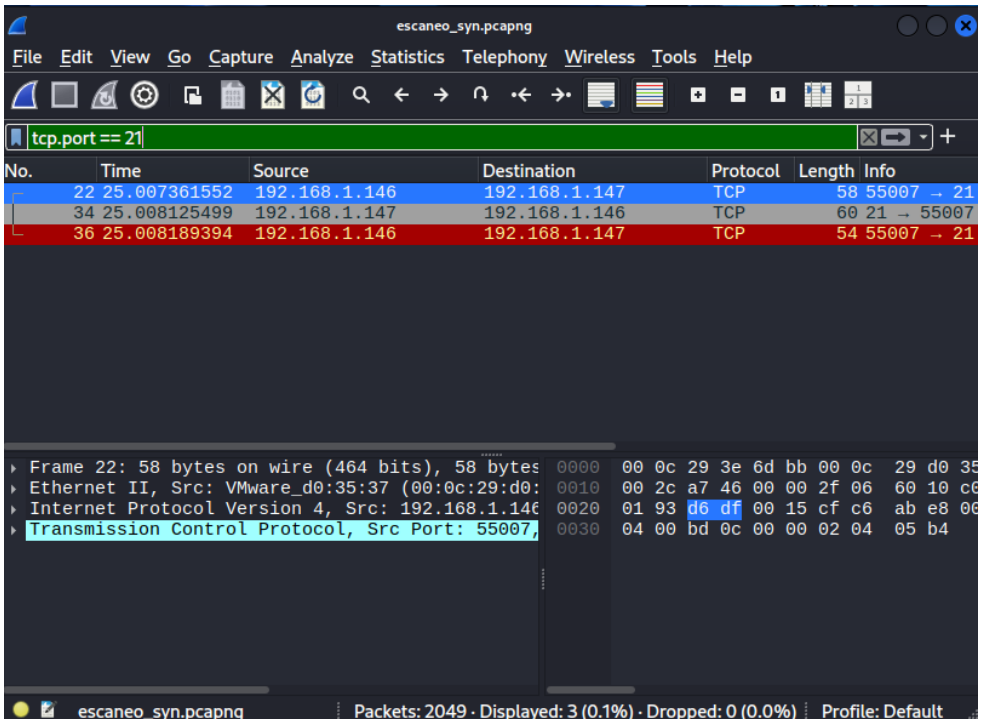
- **IP de Origen (Kali):** 192.168.1.146
- **IP de Destino (Metasploitable 2):** 192.168.1.147

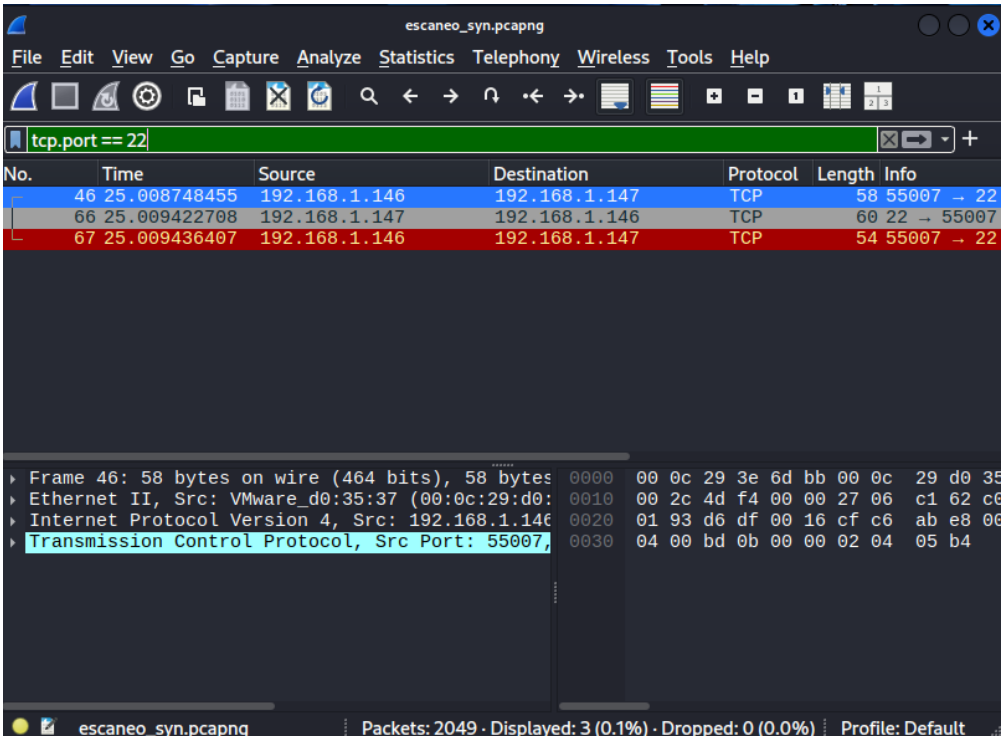
Análisis de Patrones de Escaneo SYN (-sS)

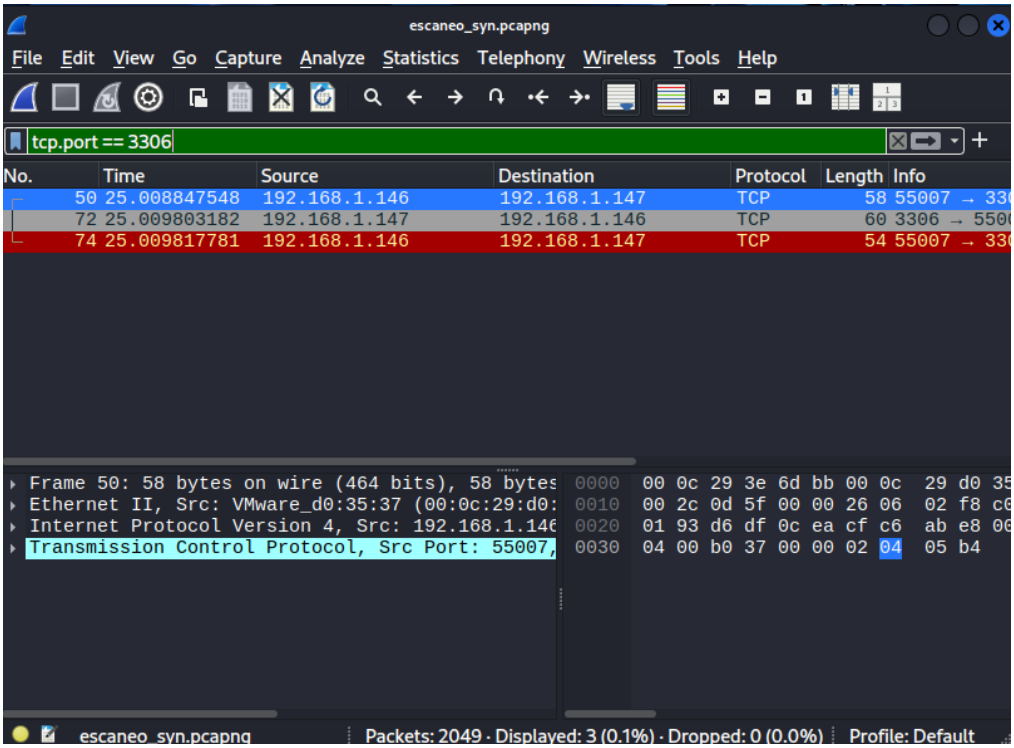
El escaneo SYN es conocido como "semi-abierto" o stealth, ya que no completa la conexión TCP de tres vías. Se detecta de la siguiente manera:

Patrón Detectado	Significado en el Escaneo SYN
192.168.1.146 (SYN) → 192.168.1.147 + 192.168.1.147 (SYN/ACK) → 192.168.1.146	Indica que el puerto está **ABIERTO**. El servidor responde con SYN/ACK, confirmando el servicio. Kali envía un RST inmediatamente después para no registrar la conexión por completo.
192.168.1.146 (SYN) → 192.168.1.147 + 192.168.1.147 (RST/ACK) → 192.168.1.146	Indica que el puerto está **CERRADO**. El servidor rechaza inmediatamente la conexión con un paquete RST/ACK.

Análisis de Patrones de Tráfico




escaneo_syn.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
tcp.port == 22
No. Time Source Destination Protocol Length Info
46 25.008748455 192.168.1.146 192.168.1.147 TCP 58 55007 → 22
66 25.009422708 192.168.1.147 192.168.1.146 TCP 60 22 → 55007
67 25.009436407 192.168.1.146 192.168.1.147 TCP 54 55007 → 22
Frame 46: 58 bytes on wire (464 bits), 58 bytes
Ethernet II, Src: VMware_d0:35:37 (00:0c:29:d0:
Internet Protocol Version 4, Src: 192.168.1.146
Transmission Control Protocol, Src Port: 55007,
0000 00 0c 29 3e 6d bb 00 0c 29 d0 35
0010 00 2c 4d f4 00 00 27 06 c1 62 c0
0020 01 93 d6 df 00 16 cf c6 ab e8 00
0030 04 00 bd 0b 00 00 02 04 05 b4
escaneo_syn.pcapng
Packets: 2049 · Displayed: 3 (0.1%) · Dropped: 0 (0.0%)
Profile: Default


escaneo_syn.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
tcp.port == 3306
No. Time Source Destination Protocol Length Info
50 25.008847548 192.168.1.146 192.168.1.147 TCP 58 55007 → 330
72 25.009803182 192.168.1.147 192.168.1.146 TCP 60 3306 → 550
74 25.009817781 192.168.1.146 192.168.1.147 TCP 54 55007 → 330
Frame 50: 58 bytes on wire (464 bits), 58 bytes
Ethernet II, Src: VMware_d0:35:37 (00:0c:29:d0:
Internet Protocol Version 4, Src: 192.168.1.146
Transmission Control Protocol, Src Port: 55007,
0000 00 0c 29 3e 6d bb 00 0c 29 d0 35
0010 00 2c 0d 5f 00 00 26 06 02 f8 c0
0020 01 93 d6 df 0c ea cf c6 ab e8 00
0030 04 00 b0 37 00 00 02 04 05 b4
escaneo_syn.pcapng
Packets: 2049 · Displayed: 3 (0.1%) · Dropped: 0 (0.0%)
Profile: Default

Puerto Analizado	Evidencia en Wireshark (Captura)	Patrón de Paquetes Confirmado	Estado del Puerto
21 (FTP)	Muestra 3 paquetes en la secuencia	**SYN → SYN/ACK → RST**	**ABIERTO**
22 (SSH)	Muestra 3 paquetes en la secuencia	**SYN → SYN/ACK → RST**	**ABIERTO**
3306 (MySQL)	Muestra 3 paquetes en la secuencia	**SYN → SYN/ACK → RST**	**ABIERTO**

La secuencia de **SYN → SYN/ACK → RST** es la huella digital del escaneo SYN. El hecho de que se replique en los puertos 21, 22 y 3306 demuestra que todos están abiertos y que la prueba de la vulnerabilidad crítica (Puerto 21) se ejecutó sobre un servicio activo y escuchando.

Parte 4: Aplicación de Mitigaciones en Servicios Vulnerables

4.1 Análisis de Vulnerabilidades

Documenta cada vulnerabilidad identificada en los servicios de Metasploitable 2:

- **FTP vulnerable (vsftpd 2.3.4)**: Acceso no autorizado por puerta trasera.
- **ProFTPd con ejecución remota**: Exploits permiten acceso de alto nivel.
- **MySQL con acceso sin autenticación fuerte**.
- **SSH con acceso de root y credenciales débiles**.

4.2 Aplicación de Soluciones

Aplica mitigaciones según los hallazgos.

- El servicio **vsftpd versión 2.3.4** en el Puerto 21 es una vulnerabilidad **CRÍTICA** (CVE-2011-2523), ya que permite el acceso remoto de root sin autenticación.
- La solución inmediata para mitigar este riesgo en Metasploitable 2 es eliminar el paquete de software vsftpd.

Comandos a Ejecutar (En Metasploitable 2):

sudo netstat -ntp

Forzar la terminación del proceso FTP vulnerable:

sudo killall -9 vsftpd

Reiniciar el sistema para obligar al sistema operativo a detener el proceso vsftpd por completo, ya que fallaron los métodos de software:

sudo reboot

```
msfadmin@metasploitable:~$ sudo netstat -ntp
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
PID/Program name
tcp        1      0 192.168.1.147:1099      192.168.1.146:52168     CLOSE_WAIT
5157/rmiregistry
tcp        0      0 192.168.1.147:1099      192.168.1.146:51356     CLOSE_WAIT
5157/rmiregistry
tcp        1      0 192.168.1.147:1099      192.168.1.146:37912     CLOSE_WAIT
5157/rmiregistry
msfadmin@metasploitable:~$
```

Comprobar:

```
msfadmin@metasploitable:~$ sudo netstat -ntp
[sudo] password for msfadmin:
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
PID/Program name
msfadmin@metasploitable:~$ _
```

Después del reinicio, el servicio FTP fue detenido.

```
┌──(kali㉿kali)-[~]
└─$ nmap 192.168.1.147
Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-22 17:49 EDT
Nmap scan report for 192.168.1.147
Host is up (0.00100s latency).
Not shown: 977 closed tcp ports (reset)
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
23/tcp   open  telnet
25/tcp   open  smtp
53/tcp   open  domain
80/tcp   open  http
111/tcp  open  rpcbind
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
512/tcp  open  exec
513/tcp  open  login
514/tcp  open  shell
1099/tcp open  rmiregistry
1524/tcp open  ingreslock
2049/tcp open  nfs
2121/tcp open  ccproxy-ftp
3306/tcp open  mysql
5432/tcp open  postgresql
5900/tcp open  vnc
6000/tcp open  X11
6667/tcp open  irc
8009/tcp open  ajp13
8180/tcp open  unknown
MAC Address: 00:0C:29:3E:6D:BB (VMware)
```

Fallo: El escaneo de **nmap 192.168.1.147** en Kali sigue reportando el Puerto 21 como **open**.

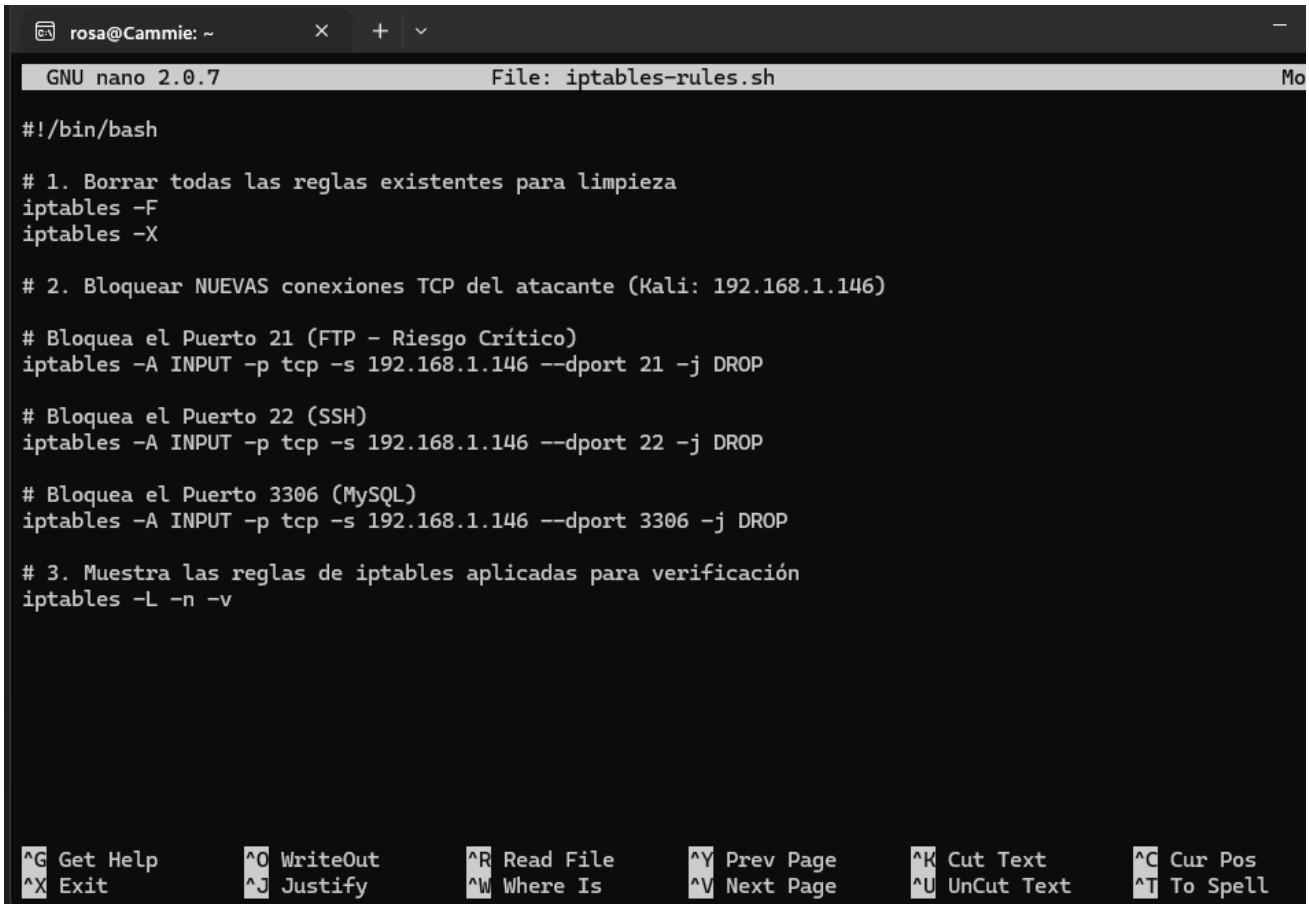
Creamos un script para establecer reglas de firewall en Metasploitable 2 que bloqueen el acceso de la máquina (Kali Linux: 192.168.1.146) a los puertos más vulnerables(21, 22, 3306)

a. Nos conectamos por ssh:

```
rosa@Cammie: ~
rosa@Cammie:~$ ssh -oHostKeyAlgorithms=+ssh-rsa -oPubkeyAcceptedAlgorithms=+ssh-rsa msfadmin@192.168.1.147
The authenticity of host '192.168.1.147 (192.168.1.147)' can't be established.
RSA key fingerprint is SHA256:BQHm5EoHX9GCiOLuVscegPXLQOsuPs+E9d/rrJB84rk.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.1.147' (RSA) to the list of known hosts.
msfadmin@192.168.1.147's password:
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

To access official Ubuntu documentation, please visit:
http://help.ubuntu.com/
No mail.
Last login: Wed Oct 22 18:05:54 2025
msfadmin@metasploitable:~$
```

b. Crear y editar el script con **nano**

```
msfadmin@metasploitable:~$ nano iptables-rules.sh
Error opening terminal: xterm-256color.
msfadmin@metasploitable:~$ export TERM=xterm
msfadmin@metasploitable:~$ nano iptables-rules.sh
msfadmin@metasploitable:~$
```



Hacer el script ejecutable

```
msfadmin@metasploitable:~$ chmod +x iptables-rules.sh
msfadmin@metasploitable:~$ ls -l iptables-rules.sh
-rwxr-xr-x 1 msfadmin msfadmin 544 2025-10-22 18:14 iptables-rules.sh
msfadmin@metasploitable:~$
```

Ejecutar el script (aplicar reglas)

```
msfadmin@metasploitable:~$ sudo ./iptables-rules.sh
[sudo] password for msfadmin:
Chain INPUT (policy ACCEPT 383 packets, 67261 bytes)
 pkts bytes target     prot opt in     out     source               destination
    0     0 DROP       tcp  --  *      *       192.168.1.146        0.0.0.0/0           tcp dpt:21
    0     0 DROP       tcp  --  *      *       192.168.1.146        0.0.0.0/0           tcp dpt:22
    0     0 DROP       tcp  --  *      *       192.168.1.146        0.0.0.0/0           tcp dpt:3306

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination

Chain OUTPUT (policy ACCEPT 369 packets, 70613 bytes)
 pkts bytes target     prot opt in     out     source               destination
msfadmin@metasploitable:~$
```

Verificar las reglas aplicadas

```
msfadmin@metasploitable:~$ sudo iptables -L INPUT -n -v --line-numbers
Chain INPUT (policy ACCEPT 399 packets, 69993 bytes)
num   pkts bytes target     prot opt in     out     source               destination
1        0     0 DROP       tcp  --  *      *       192.168.1.146        0.0.0.0/0           tcp dpt:21
2        0     0 DROP       tcp  --  *      *       192.168.1.146        0.0.0.0/0           tcp dpt:22
3        0     0 DROP       tcp  --  *      *       192.168.1.146        0.0.0.0/0           tcp dpt:3306
msfadmin@metasploitable:~$
```

Subir el script desde la máquina anfitrión

```
msfadmin@metasploitable:~$ scp iptables-rules.sh msfadmin@192.168.1.147:/home/msfadmin/
The authenticity of host '192.168.1.147 (192.168.1.147)' can't be established.
RSA key fingerprint is 56:56:24:0f:21:1d:de:a7:2b:ae:61:b1:24:3d:e8:f3.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.1.147' (RSA) to the list of known hosts.
msfadmin@192.168.1.147's password:
iptables-rules.sh                                             100%  544     0.5KB/s   00:00
msfadmin@metasploitable:~$
```

Luego, en la VM:

```
msfadmin@metasploitable:~$ chmod +x /home/msfadmin/iptables-rules.sh
msfadmin@metasploitable:~$ ls -l /home/msfadmin/iptables-rules.sh
-rwxr-xr-x 1 msfadmin msfadmin 544 2025-10-22 18:19 /home/msfadmin/iptables-rule
s.sh
msfadmin@metasploitable:~$ _
```

Ejecutamos el escaneo nmap

```
Session  Actions  Edit  View  Help

┌──(kali㉿kali)-[~]
└─$ nmap 192.168.1.147
Starting Nmap 7.95 ( https://nmap.org ) at 2025-10-22 18:26 EDT
Nmap scan report for 192.168.1.147
Host is up (0.0038s latency).
Not shown: 977 closed tcp ports (reset)
PORT     STATE    SERVICE
21/tcp   filtered ftp
22/tcp   filtered ssh
23/tcp   open     telnet
25/tcp   open     smtp
53/tcp   open     domain
80/tcp   open     http
111/tcp  open     rpcbind
139/tcp  open     netbios-ssn
445/tcp  open     microsoft-ds
512/tcp  open     exec
513/tcp  open     login
514/tcp  open     shell
1099/tcp open     rmiregistry
1524/tcp open     ingreslock
2049/tcp open     nfs
2121/tcp open     ccproxy-ftp
3306/tcp filtered mysql
5432/tcp open     postgresql
5900/tcp open     vnc
6000/tcp open     X11
6667/tcp open     irc
8009/tcp open     ajp13
8180/tcp open     unknown
MAC Address: 00:0C:29:3E:6D:BB (VMware)

Nmap done: 1 IP address (1 host up) scanned in 1.43 seconds

┌──(kali㉿kali)-[~]
└─$
```

- Puerto 21 (FTP): Estado cambió de open a filtered.
- Puerto 22 (SSH): Estado cambió de open a filtered.
- Puerto 3306 (MySQL): Estado cambió de open a filtered.

El estado **filtered** confirma que el tráfico saliente de la máquina atacante (Kali) está siendo descartado **(DROP)** por las reglas de iptables en Metasploitable 2.

Se completó exitosamente la mitigación de riesgos al bloquear el acceso a los puertos críticos y vulnerables.