IUT Charlemagne
Département Informatique

LP ASRALL
21 janvier 2015

# Sécurité et système Linux

Ce premier TD a pour but de vous familiariser avec l'environnement UNIX/Linux et de manipuler les quelques commandes de bases qui vous permettent de faire rapidement un premier bilan de sécurité sur une machine.

Le but n'est pas d'apprendre ces commandes par cœur mais de savoir quel type d'information on cherche lorsque l'on cherche à sécuriser une machine (et la liste n'est pas exhaustive).

## 1. Le système de fichier

Les droits des fichiers sont le cœur de la sécurité d'un système unix.

### *1.1. Chercher les fichiers en écriture pour tous (dans votre home puis sur tout le système).*

find ~ -type f -perm -o+w

find / -type f -perm -o+w

### *1.2. Chercher les fichiers en écriture pour le groupe (dans votre home puis sur tout le système).*

find ~ -type f -perm -g+w

find / -type f -perm -g+w

### *1.3. Chercher les répertoires en écriture pour tous (dans votre home puis sur tout le système).*

find ~ -type d -perm -o+w

find / -type d -perm -o+w

### *1.4. Chercher les répertoires en écriture pour le groupe (dans votre home puis sur tout le système).*

find ~ -type d -perm -g+w

find / -type d -perm -g+w

### *1.5. Quels sont les droits attachés au répertoire /tmp ? Pourquoi ?*

ls -ald /tmp

drwxrwxrwt 7 root root 200 2011-02-17 13:05 /tmp

Le bit 't' ou sticky bit est un droit particulier qui permet à n'importe quels utilisateur d'écrire

dans le répertoire avec la restriction que seul le propriétaire du fichier peut le modifier et le

supprimer.

### *1.6. Cherchez les fichier possédant des droits suid et sgid dans le dans votre répertoire home et dans le répertoire /bin ?*

find ~ -type f -perm -4000

find /bin -type f -perm -2000

### *1.7. Quel est l'intérêt de ces droits ? Quelle est l'alternative ?*

Ces droits particuliers (droits s) concernent uniquement le bit d'exécution et définissent l'identité sous laquelle le programme sera réalisé. Une alternative intéressante est le programme sudo (qui possède le bit 's') et qui permet de configurer de façon plus fine qui peut endosser les droits root par exemple (fichier /etc/sudoers) à l'aide de la commande visudo.

### *1.8. Quels sont les fichiers modifiés ces 60 dernières minutes ?*

find / -not -path '/sys*' -not -path '/dev*' -not -path '/proc*' -mmin -60

Ici, on exclut les répertoires /sys, /dev et /proc.

### *1.9. Quelles sont les partitions montées en ce moment sur votre machine ? Quels sont les droits déclarés sur ces partitions ?*

df -hT (h pour « human-readable », T pour avoir le type)

more /etc/fstab ou mount

Une option de montage intéressante au niveau de la sécurité est par exemple l'option nosuid qui interdit l'utilisation du bit 's' sur une partition. D'autres existent (noexec, etc.)

## 2. Augmentation de privilège par les fichiers suid

En préparations, effectuez les commandes suivantes.

```
cd ~
cat >suidtest.c <<FIN
#include<stdio.h>
#include<stdlib.h>
#include<unistd.h>
main() {
printf("on est : ");
execlp("whoami", "whoami", NULL);
}
FIN
cc suidtest.c -o  suidtest
```

sudo chown root suidtest ; sudo chmod 4755  suidtest

### *2.1. Que font-elles ? Lancez le. Que constatez vous ?*

Nous sommes durant l'execution du programme sous le compte de root.

### *2.2. Détournez ce script.*

La commande whoami est dans le PATH, dans /usr/bin. En modifiant son PATH et en créant un fichier exécutable, essayer de faire exécuter à ce script en root autre chose que cette commande.

Editer un fichier ~/whoami qui est un shell script avec par exemple :

#!/bin/sh

echo "je suis le maitre du monde et le fais ce que je veux car je suis"

Puis les commandes suivantes :

PATH=.:$PATH

chmod 755 ~/whoami

Notez bien que vous n'etes pas passé root pour faire cela. En relançant ~/.suitest, il exécutera votre commande à la place de /usr/bin/whoami.

Il suffit de placer recopier /bin/sh dans le programme ~/whoami pour obtenir un shell root.

# 3. Les comptes

Sur les systèmes informatiques modernes, il existe des comptes privilégiés permettant d'administrer la machine et d'exécuter différentes tâches. Ces comptes sont de fait des cibles privilégiés des pirates informatiques puisqu'ils permettent de contrôler complètement une machine avec la possibilité d'effacer toutes traces d'intrusions.

Sur le architectures de type linux/unix, le compte root est le compte administrateur par défaut et doit donc faire l'objet de la plus grande attention.

### *3.1. Vérifiez les droits de votre home directory. Comment faire que tout nouveau fichier créé ait des droits spécifiques (lecture owner uniquement par exemple) ?*

Les droits doivent être très restrictif, c'est à dire ni en écriture (bien sûr) et ni en lecture. Le masque de création de fichiers (umask) doit être positionné de façon à ce que tout fichier crée reste non lisible pour les autres (umask 077 par exemple).

### *3.2. Vérifier les comptes existants dans le fichier /etc/passwd*

Il existe un certain nombre de comptes spéciaux qui n'ont pas vocation à être interactifs et ne doivent pas l'être. Pour cela, on déclare un shell spécial attaché au compte (celui qui permettrait de démarrer une session), par exemple /bin/false ou bien /bin/nologin. On regarde aussi les compte d'uid 0 et faisant partie des groupes à pouvoir (root, adm, sys, sudo, etc.)

### *3.3. Création d'utilisateurs et de groupes.*

Vous allez créer un nouveau groupe unix, et deux utilisateurs (avec une home mais pas de password ni de shell) dont ce groupe sera leur groupe principal. Ainsi ces deux utilisateurs pourront partager des données uniquement entre eux en mettant les fichiers concernés dans ce groupe avec les droits ad-hoc.

addgroup testgroup

adduser --home /home/barbara --shell /bin/false --ingroup testgroup --disabled-password barbara

adduser --home /home/natacha --shell /bin/false --ingroup testgroup --disabled-password natacha

### *3.4. A quoi peuvent servir des comptes utilisateurs sans possibilité de s'y connecter ?*

Ils servent pour faire tourner un service réseau (un serveur WEB par exemple, une base de donnée, …) qui ne nécessite pas de fonctionner sous root.

### *3.5. Créez un fichier et un répertoire lisible uniquement par ce groupe et vérifiez que vous n'y accédez pas.*

Tout de même... mkdir, chgrp, chown, touch...

### *3.6. Rajoutez vous dans le groupe que vous avez créé. Vérifiez que vous accédez désormais aux fichiers et répertoires précédents.*

addgroup testgroup <votre login>

### *3.7. Si vous créez un fichier, sera-t-il dans ce groupe ? Comment faire pour que ce soit le cas ?*

Newgrp testgroup

### *3.8. Comment partager un répertoire en rwx pour deux utilisateurs et eux seuls, sans leur dédier un group ?*

Il faut utiliser les commandes setacl (ACL de fichiers POSIX).

Mkdir /tmp/dossier_a_partager

chown -R <user1> /tmp/dossier_a_partager

chmod 700 /tmp/dossier_a_partager

setfacl -Rm u:<user2>:rwx /tmp/dossier_a_partager

# 4. Le système de log (/var/log)

Pour trouver les commandes : apropos log.

Regarder le répertoire /var/log/.

### 4.1. Qui est loggé en ce moment sur votre machine ?

who

### 4.2. Qui s'est loggé sur les 10 derniers jours sur votre machine ?

lastlog -t 10

### 4.3. Quelles sont les 3 dernières tentative de login qui ont échoué ?

less /var/log/auth_messages

less /var/log/auth.log

### 4.4. Que fait la commande dmesg

Elle permet d'afficher les informations du noyau, notamment celles liées au boot de la machine.

### 4.5. Quel est le processus qui occupe le plus de mémoire ?

top

(taper 'm' )

### 4.6. Effacer ses traces

Dans cette partie vous allez prendre la place du pirate et essayer d'effacer toute trace de votre passage, afin d'en mesurer la difficulté.

Pour cela, connectez vous en ssh localhost en temps qu'utilisateur, puis passez root. Effectuez une commande quelconque comme envoyer un mail ou changer la configuration d'un service.

Lorsque cela est fait, essayez d'effacer toutes trace de cette connexion, du passage root et de l'action faite.

Quelles actions faites vous ?

Il faut repérer la date d'accès au système, puis éffacer dans les fichiers de logs (/var/logs/wtmp, var/log/auth.log, /var/log/syslog eventuellement ) et logs applicatifs suivant leur configurations.

Ce n'est pas simple pour utmp et wtmp, car ils sont dans un format particulier, qui nécessite un programme dédié (clear + utmpdump + ...)

Supprimer l'historique de l'utilisateur (~/.bash_history si on est en bash et celui de root).

Changer les dates de modification et de lecture des fichiers édités pour leur remettre l'ancienne date (touch -m -a -t <date>). Néanmoins, une trace reste visible avec la commande stat

Enfin, à la déconnexion, de nouvelles logs seront posés.

...Bref pas simple d'être invisible...

Comment se prémunir « simplement » contre cela ?

Il suffit de configurer syslog(ng) et les services qui ne l'utilisent pas pour dupliquer toutes les logs vers une autre machine. Ainsi, les logs seront préservées (hormis wtmp et utmp qui restent locaux). Man rsyslogd pour plus de détails.

# 5. Les services et les ports réseaux

### *5.1. Listez les ports tcp/udp utilisés en ce moment sur votre machine (commande netstat).*

netstat -atup

### *5.2. Identifiez la configuration des services sur votre machine : lesquels doivent être démarrés ?*

Pour Debian, lister les répertoires et utiliser

runlevel=`/sbin/runlevel | cut -f 2 -d\ `

cd /etc/rc${runlevel}.d \

&& ls -l S* |awk '{ print $10; }' \

|xargs -n 1 basename

...

# 6. Bibliographie

man chmod find ls grep last netstat who ...