# 4 - Utilisateurs, Groupes, Permissions

Laurent Tichit

5 avril 2011

# Plan

# Utilisateurs et Groupes

# Les utilisateurs et les groupes

Un usager possède :

- Un login
- Un numéro d'identification (UID, *User IDentification*)
- Un nom et prénom
- Un mot de passe
- Un nom de groupe principal
- Un numéro de groupe d'appartenance (GID, *Group IDentification*)
- Un répertoire d'accueil
- Un shell par défaut

De plus :

- Unix permet de regrouper des utilisateurs en groupes.
- Un groupe possède certains privilèges.
- Un utilisateur peut appartenir à plusieurs groupes.

# Root

Il existe un compte spécial :

- login = root
- group = root
- HOME = /root
- UID = 0
- GID = 0

Ce compte à tous les droits. C'est le *super-utilisateur*. Les administrateurs de la machine ont accès à ce compte.

# Les utilisateurs et les groupes

- Les informations relatives aux utilisateurs sont stockées dans le fichier **/etc/passwd**.

  `tichit@iml230:~$` **cat /etc/passwd | grep tichit**

  `tichit:x:1000:1000:Tichit Laurent,,,:/home/tichit:/bin/bash`

- La commande **id** permet de connaître les informations relatives à un utilisateur :

  `tichit@iml230:~$` **id**

  `uid=1000(tichit) gid=1000(tichit) groupes=1000(tichit),`
  `4(adm),20(dialout),21(fax),24(cdrom),25(floppy),`
  `26(tape),29(audio),30(dip),44(video),46(plugdev),`
  `104(scanner),112(netdev),113(lpadmin),115(powerdev),`
  `117(admin),119(fuse)`

# Les utilisateurs et les groupes

- Les informations relatives aux groupes sont stockées dans le fichier **/etc/group** :

  `tichit@iml230:~$` **cat /etc/group | grep tichit**

```
adm:x:4:tichit
dialout:x:20:cupsys,tichit
fax:x:21:tichit
cdrom:x:24:haldaemon,tichit
floppy:x:25:haldaemon,tichit
tape:x:26:tichit
audio:x:29:tichit
dip:x:30:tichit
video:x:44:tichit
plugdev:x:46:haldaemon,tichit
scanner:x:104:cupsys,tichit,hplip
netdev:x:112:tichit
lpadmin:x:113:tichit
powerdev:x:115:haldaemon,tichit
admin:x:117:tichit
fuse:x:119:tichit
tichit:x:1000:
```

# Root

- La commande **su [login]** permet de devenir *root* (ou l'utilisateur *login*).
- Sur les distribution Linux récentes, le compte *root* est parfois désactivé.
  Dans ce cas, on se servira de la commande **sudo [login] commande**, au cas pas cas, qui permet de devenir *temporairement root* (ou l'utilisateur *login*) pour exécuter la commande.

# Exercice

- Connectez-vous sur le terminal d'un de vos camarades à l'aide de **su**.
- Vérifiez que vous êtes bien connecté à votre compte : **whoami**.
- Déconnectez-vous (**CTRL-D** ou **logout**).

# Droits d'accès

# Les protections et les privilèges

3 manières d'accéder à un fichier

- Lire son contenu (Read)
- Modifier son contenu (Write)
- S'il s'agit d'un programme, l'exécuter (eXecute)

3 catégories d'accès à un fichier

- Le propriétaire du fichier (User)
- Les membres du groupe auquel appartient le fichier (Group)
- Tous les autres usagers (Other)

$\Rightarrow$ 9 protections pour un fichier donné :

- lecture pour son propriétaire
- écriture pour son propriétaire
- exécution pour son propriétaire
- lecture pour les membres du groupe du fichier
- écriture pour les membres du groupe du fichier
- exécution pour les membres du groupe du fichier
- lecture pour les autres usagers
- écriture pour les autres usagers
- exécution pour les autres usagers

# Visualisation des protections

La commande **ls -l** permet de connaître les droits d'un fichier :

```
-rwxrwxrwx  1 tichit enseig 826 2009-04-21 12:13 Unix.pl
-rwxr-x--x  1 tichit enseig 826 2009-04-21 12:13 Unix.sh
```

Les 9 lettres `rwxrwxrwx` et `rwxr-x--x` symbolisent dans l'ordre :

- Les 3 droits associés au propriétaire
- Les 3 droits associés aux membres du groupe auquel appartient le fichier
- Les 3 droits associés aux autres usagers

- **r** indique une autorisation de lire
- **w** indique une autorisation d'écrire
- **x** indique une autorisation d'exécuter (ou accéder)
- - indique une interdiction

- *Unix.pl* est donc lisible, modifiable et exécutable par tous.
- *Unix.sh* est lisible par l'utilisateur *tichit* et les membres du groupe *enseig*, modifiable par l'utilisateur *tichit* uniquement, exécutable par tout le monde.

# Visualisation du type de fichier

Le premier caractère fourni par **ls -l** indique le type de fichier

- - Fichier standard
- **d** Répertoire
- **p** Pipe (tube nommé)
- **l** Lien symbolique
- **s** Socket
- **c** Périphérique en mode caractère (clavier)
- **b** Périphérique en mode bloc (disque dur)

```
-rw-r--r-- 1 tichit tichit 2912 2009-03-03 11:10 Pub.doc
drwxr-xr-x 2 tichit tichit 4096 2009-04-22 14:55 unix
```

*Pub.doc* est un fichier standard, *unix* un répertoire.

# Mise en place des permissions

Seul le propriétaire du fichier peut modifier les droits d'accès à un fichier par la commande
**chmod mode nom_de_fichier**.
On peut modifier les permissions de 2 manières :

- de manière absolue (sans considérer les permissions actuelles)
- relatives (en se modifiant les permissions actuelles)

# Permissions absolues

Elles sont exprimées en octal. Si l'on désire uniquement que :

- Le fichier soit lisible
  - Par le propriétaire $\rightarrow$ 400
  - Par les membres du groupe $\rightarrow$ 040
  - Par les autres $\rightarrow$ 004
- Le fichier soit modifiable
  - Par le propriétaire $\rightarrow$ 200
  - Par les membres du groupe $\rightarrow$ 020
  - Par les autres $\rightarrow$ 002
- Le fichier soit exécutable
  - Par le propriétaire $\rightarrow$ 100
  - Par les membres du groupe $\rightarrow$ 010
  - Par les autres $\rightarrow$ 001

# Permissions absolues

Bien entendu, on peut composer les permissions. Il suffit d'additionner les valeurs.

- **chmod 644 fichier.txt** : propriétaire peut lire et modifier, groupe et autre peuvent uniquement lire.
- **chmod 755 fichier.pl** : idem plus tous peuvent exécuter.
- **chmod 710 fichier.sh** : le propriétaire a tous les droits, les membres du groupe peuvent uniquement exécuter, les autres n'ont aucun accès.

# Exercice

- A quoi correspondent les permissions **541** ?
- Donnez la commande pour que personne n'ait accès au fichier **f1.txt** en écriture, que vous-même et les membres de votre groupe puissent le lire, et que tout le monde puisse l'exécuter ?

# Permissions relatives (Symbolique)

En symbolique le mode est de la forme :

**chmod “qui” “quelle action” “quelle permission” fichier**

*“Qui”*

- **u** : user
- **g** : group
- **o** : other
- **a** : all

*“Quelle action”*

- **+** : ajoute des permissions
- **-** : retire des permissions

Ex :

- **chmod o-rw fichier.txt** :

# Permissions relatives (Symbolique)

En symbolique le mode est de la forme :

**chmod “qui” “quelle action” “quelle permission” fichier**

*“Qui”*

- **u** : user
- **g** : group
- **o** : other
- **a** : all

*“Quelle action”*

- **+** : ajoute des permissions
- **-** : retire des permissions

Ex :

- **chmod o-rw fichier.txt** : enlève les permissions en lecture et écriture pour les autres.
- **chmod a+x fichier.pl** :

# Permissions relatives (Symbolique)

En symbolique le mode est de la forme :

**chmod “qui” “quelle action” “quelle permission” fichier**

*“Qui”*

- **u** : user
- **g** : group
- **o** : other
- **a** : all

*“Quelle action”*

- **+** : ajoute des permissions
- **-** : retire des permissions

Ex :

- **chmod o-rw fichier.txt** : enlève les permissions en lecture et écriture pour les autres.
- **chmod a+x fichier.pl** : donne à tous le droit d’exécuter.
- **chmod go-w fichier.sh** :

# Permissions relatives (Symbolique)

En symbolique le mode est de la forme :

**chmod “qui” “quelle action” “quelle permission” fichier**

*“Qui”*

- **u** : user
- **g** : group
- **o** : other
- **a** : all

*“Quelle action”*

- **+** : ajoute des permissions
- **-** : retire des permissions

Ex :

- **chmod o-rw fichier.txt** : enlève les permissions en lecture et écriture pour les autres.
- **chmod a+x fichier.pl** : donne à tous le droit d’exécuter.
- **chmod go-w fichier.sh** : enlève aux membres du groupe et aux autres le droit de modifier.

# Permissions

Pour changer le propriétaire d'un fichier (ou le groupe propriétaire) :

- **chown nom fichier**
- **chgrp groupe fichier**

Permet de changer le nom (uid) ou le groupe (gid) d'appartenance du fichier.
Exécuté principalement par le super-utilisateur.