

# **Cours UNIX**

## **Création de comptes et montages locaux**

Jean-Claude Bajard

IUT - université Montpellier 2

# Les utilisateurs

# Ouverture d'un compte

Lors de la création d'un compte l'administrateur doit

- inscrire l'utilisateur dans le fichier `"/etc/passwd"` en donnant (séparés par `":"`)
  1. un nom de login
  2. un mot de passe crypté ( un x si `"/etc/shadow"` existe)
  3. un numéro d'identification personnel Numerical user ID
  4. un numéro de groupe (inscrit dans `"/etc/group"`)
  5. des commentaires
  6. un répertoire home
  7. un interpréteur de commande
- créer un répertoire `"home"` en donnant les droits de propriétés ( `"shown"` )

- vérifier que le groupe existe dans “/etc/group” sinon le créer.

groupname : password(x) : GID : userlist(séparés par une virgule)

## Ouverture d'un compte (suite)

- créer une entrée dans “/etc/shadow” et donner un mot de passe avec la commande “passwd”
  1. Login name
  2. Encrypted password
  3. Days since Jan 1, 1970 that password was last changed
  4. Days before password may be changed
  5. Days after which password must be changed
  6. Days before password is to expire that user is warned
  7. Days after password expires that account is disabled
  8. Days since Jan 1, 1970 that account is disabled
  9. A reserved field

## Les principales commandes

useradd, usermod, userdel adduser, deluser	gestion des comptes utilisateur
groupadd, groupmod, groupdel addgroup, delgroup	gestion des groupes
pwck, grpck	vérification des fichiers
passwd	changer le mot de passe d'un utilisateur
chfn, id, groups, finger	utilitaires divers

## Exemple de gestion

- Pour créer l'utilisateur stagex, root passe la commande : `useradd stagex`

Ceci crée : le répertoire personnel `/home/stagex`, portant par défaut le nom du compte une nouvelle entrée dans les 2 fichiers fondamentaux `/etc/passwd` et `/etc/group`. Pour connaître les options de `useradd` (indispensable pour gérer les comptes à l'aide de scripts)

- Pour lui attribuer le mot de passe : `passwd stagex`

saisir 2 fois `stgx`

- Supprimer le compte d'un utilisateur (non connecté), au hasard .. `totox` : `userdel [-r] totox`

L'option -r supprime aussi le rép. personnel et les fichiers de l'utilisateur

La commande supprime toute trace de l'utilisateur dans le fichier de configuration :  
/etc/passwd y compris dans les groupes d'utilisateurs.

- Modifier le compte de l'utilisateur toto : `usermod [options] totox`

Les options sont les mêmes que `useradd` `usermod -G stagiaire,prof stagex` ajoute `stagex` dans les 2 groupes `stagiaire` et `profs` (qui doivent exister)



# Les groupes

- Un groupe est, aussi pour Linux, un ensemble d'utilisateurs qui partagent fichiers et répertoires
- Chaque utilisateur doit faire partie au moins d'un groupe, son groupe primaire.
- Ainsi, dans `/etc/passwd` chaque utilisateur possède un groupe par défaut, précisé par son identifiant gid dans ce fichier.
- L'appartenance au groupe primaire n'étant pas exclusive, tout utilisateur peut faire partie de plusieurs autres groupes, appelés ses groupes secondaires.
- Pour lister tous les groupes (primaire et secondaires) d'un utilisateur : `groups stagex`

- Pour créer un nouveau groupe : `groupadd stagiaire`
- Supprimer un groupe, au hasard .. encore `totox` : `groupdel totox` Le groupe est supprimé du fichier `/etc/group`.
- Pour ajouter un utilisateur à un groupe Le plus simple est d'éditer le fichier `/etc/group` et d'ajouter une liste d'utilisateurs (séparés par des virgules) sur la ligne du groupe (ou utiliser `Linuxconf`).

# Rappel sur les droits: chmod

## Les droits élémentaires

- “u” user, “g” groupe, “o” les autres, “a” tout le monde
- “chmod u=rwx,g=rx,o= nomdufichier” (ou répertoire) pour initialiser les droits
- directement en octal avec r=4, w=2 et x = 1  
“chmod 750 toto”
- “chmod u+w” ou “chmod u-w” pour ajouter ou enlever un droit
- “umask” donne les droits par défaut ex: umask 22 =<sub>i</sub> chmod 755

Les autres droits (si droit d'exécution x)

- le SETUID, donne les droits d'utilisateur à l'exécution u+s (4)
- le SETGID donne les droits du groupe à l'exécution g+s (2)
- le STICKY BIT o+t (1), donne à un répertoire la possibilité de restreindre les droits, seul le propriétaire peut détruire un fichier de ce répertoire.
- chmod 7777 tous les droits.

## retour aux groupes

- Un utilisateur peut appartenir à plusieurs groupes.
- Par défaut lors de la création d'un compte sous linux il y a création d'un groupe au même nom.
- La commande "chgrp" permet de changer le groupe d'appartenance d'un fichier ou d'un répertoire.
- En positionnant le SETGID sur un répertoire tous les fichiers créés dans ce répertoire appartiendront au groupe du répertoire.
- le SETUID n'est pas utile pour un répertoire

# le système de fichier UNIX

- Une seule arborescence
- tout est visible en terme de fichier, ex: les processus dans /proc
- il en est de même de tous les périphériques dans /dev
- pour accéder au contenu d'une unité de stockage il faut effectuer un montage sur l'arborescence.
  - \* déterminer un point de montage (un répertoire de l'arborescence)
  - \* donner le type de montage : cdrom, disquette, partition dos,....

## la commande mount

- la commande standard est: `mount [-fnrvw] [-t type] [-o options] périph rép`
- le périphérique peut être indiqué sous forme de noms de fichiers (spécial bloc), comme `/dev/sda1`, `/dev/hda2...`

- le répertoire de montage doit exister

Attention: lors du montage dans un répertoire son contenu d'origine peut être masqué.

- options générales :
  - v Mode volubile.
  - f Simule le montage du système de fichiers. .

- n Montage sans écriture dans /etc/mtab.
- r Montage du système de fichiers en lecture seule = -o ro.
- w Montage du système de fichiers en lecture/écriture (par défaut) = -o rw.
- -t type : liste dans linux/fs/filesystems.c: ext2, hpfs, fat, msdos, vfat, proc, nfs, iso9660, smb, ...



## la commande mount (suite)

- -o liste d'options séparées par des virgules.

ro : Montage du système en lecture seule.

rw : Montage du système en lecture/écriture.

async : Toutes les entrées/sorties sur le système de fichiers asynchrones.

sync : Toutes les entrées/sorties sur le système de fichiers seront synchrones.

dev : Interpréter les fichiers spéciaux de périphériques présents sur le système de fichiers.

nodev : Ne pas interpréter les fichiers spéciaux de périphériques présents

exec : Permettre l'exécution de fichiers binaires.

noexec: Ne pas permettre l'exécution de fichiers binaires

suid : Prendre en compte les bits Set-UID ou Set-GID des fichiers sur le système monté.

nosuid: Ne pas tenir compte des bits Set-UID ou Set-GID.

user : Autoriser les utilisateurs ordinaires (non root) à monter le système

nouser: Ne pas autoriser d'utilisateur (non root) à monter le système de fichiers (par défaut).

remount: pour changer les attributs de montage

defaults : rw, suid, dev, exec, auto, nouser, et async.

## le fichier /etc/fstab

- Il contient des informations sur les montages et les différents systèmes de fichiers
- Chaque système de fichier est décrit sur une ligne indépendante.
- L'ordre des enregistrements au sein de fstab est important
- Il est lu au démarrage multi-utilisateurs
- la commande “mount -a” effectue les montages indiqués dans ce fichier
- la commande “mount dev” ou “mount rep” effectue le montage décrit dans /etc/fstab

## le fichier /etc/fstab (suite)

Chaque ligne comporte:

1. le périphérique bloc ou le système de fichier distant a monter.
2. le point de montage du système de fichier (“none” pour le swap)
3. le type de système de fichiers : auto, ext2, vfat, iso9660, nfs, swap(voir /proc/filesystems)
4. les options de montage : “noauto” et “user” (man mount)
5. niveau de la commande dump(8) pour déterminer quels sont les systèmes de fichiers à sauvegarder: 0 = pas de sauvegarde

6. ordre de vérification des systèmes de fichiers au démarrage par fsck(8)

## la commande : `umount`

- permet le démontage d'un système de fichiers
- syntaxe usuelle : `umount [-nrv] device — dir`
  - r In case unmounting fails, try to remount read-only.
  - n Unmount without writing in `/etc/mtab`.
  - v Verbose mode.
- `umount -a` pour démonter tous les systèmes non actif de “`/etc/mtab`”

## les fichiers `/etc/mtab`

Il contient la liste des montages actifs (sauf si l'option `-n` de `mount` a été utilisée) maintenu par `mount` et `umount`.

Par contre `/proc/mounts` est complet.

## la commande : free

- Afficher les quantités de mémoires libres et utilisées.
- “/proc/meminfo” – informations sur la mémoire



## la commande : df

- report filesystem disk space usage