MATHÉMATIQUES ET INFORMATIQUE
Sciences
Université Paris Cité

Chapitre 4: Administration des utilisateurs

Chérifa Boucetta

A.U: 2022-2023

Plan du cours

1. Notions d'utilisateur et groupe système
2. Mécanismes de gestion des utilisateurs et groupes système
3. Les fichiers de connexion BASH
4. Politique de gestion des mots de passe

Notion d'utilisateur système

- Les comptes utilisateur permettent de distinguer les différents utilisateurs qui ont accès au système, pour des raisons de sécurité.
  - Chacun d'eux possède un compte personnel, auquel il accède par un identifiant et un mot de passe secret.
  - Ces utilisateurs peuvent définir des permissions d'accès à leurs données, afin d'en autoriser ou d'en interdire l'exploitation par les autres.
- En dehors des **comptes personnels**, il existe des utilisateurs qui ne sont pas forcément des **personnes physiques**.
  - Ces utilisateurs remplissent des fonctions administratives.
  - Exemples:
    - le compte **root** utilisé par l'administrateur pour effectuer la maintenance
    - Les comptes destinés à **des démons** qui doivent avoir accès à certains fichiers sous une identification spécifique (apache, postfix,etc).

Notion d'utilisateur système

- Chaque processus sur le système s'exécute avec le nom d'un utilisateur particulier.
- Chaque fichier est la propriété d'un utilisateur particulier.
- L'accès aux fichiers et aux répertoires est restreint par l'utilisateur auquel un processus en cours d'exécution est associé détermine à quels fichiers et répertoires ce processus peut accéder.
- Pour afficher l'utilisateur associé à un processus, on ajoute l'option **u** à la commande **ps**.

```
#ps aux
```

```
USER   PID  %CPU  %MEM  TTY  STAT  START   TIME    COMMAND
root    508   2.4   1.6    ?     S    02:02  0:03  /usr/sbin/firewalld
user  42266  0.0  0.0  8508  3240 pts/0    R+   11:35   0:00 ps aux
systemd+  863  0.0  0.2  27952  8188 ?      Ss   Sep22   0:03 /lib/systemd/systemd-networkd
systemd+  865  0.0  0.3  24916 12316 ?      Ss   Sep22   0:01 /lib/systemd/systemd-resolved
```

Notion d'utilisateur système

- Pour afficher l'utilisateur associé à un fichier ou répertoire, on utilise la commande **ls -l**. La troisième colonne indique le nom de l'utilisateur :

```
user@ubnt:~$ ls -l
total 24
-rw-r--r-- 1 user user  220 Jun 18  2020 bash_logout
-rw-r--r-- 1 user user 3771 Jun 18  2020 bashrc
drwxr-xr-x 3 root root 4096 Apr  8 14:42 builds
drwx------ 3 root root 4096 Apr  8 14:42 cache
-rw-r--r-- 1 user user  807 Jun 18  2020 profile
drwxr-xr-x 3 user user 4096 Sep  1 12:55 snap
```

- Chaque utilisateur possède un **login** et un **mot de passe.**
- Par défaut, UNIX utilise le fichier **/etc/passwd** pour stocker les informations concernant les utilisateurs locaux.

Format du fichier /etc/passwd

- Le format du fichier **/etc/passwd** est le suivant:

login:x:uid:gid:commentaire:répertoire de connexion:application de connexion

```
user@ubnt:~$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin
sync:x:4:65534:sync:/bin:/bin/sync
games:x:5:60:games:/usr/games:/usr/sbin/nologin
man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
user:x:1000:1000:user:/home/user:/bin/bash
```

- La commande **id** permet d'afficher les informations relatives à l'utilisateur connecté notamment son UID et les groupes auxquels il appartient.

Format du fichier /etc/passwd

- Le rôle de chacun des sept champs sont séparés par le caractère "**:**" est :
  - Le **nom du compte** de l'utilisateur
  - Le **mot de passe** de l'utilisateur.
    - les mots de passe sont stockés dans un fichier distinct appelé /etc/shadow.
  - L'**UID** qui identifie **l'utilisateur** pour le système d'exploitation (UID=User ID, identifiant utilisateur)
    - uid=0 **----->** root
    - 1<uid<500 **---->** user spécial
    - uid >499 **---->** user physique
  - Le **GID** qui identifie le **groupe** de l'utilisateur (GID=Group ID, identifiant de groupe)
  - Le **commentaire** dans lequel on peut retrouver des informations sur l'utilisateur ou simplement son nom réel
  - Le **répertoire de connexion** qui est celui dans lequel il se trouve après s'être connecté au système (Le HOME directory)
  - L'**application de connexion** est celle exécutée **après connexion** au système (c'est fréquemment un interpréteur de commandes /bin/bash, /bin/sh)

Gestion des utilisateurs locaux

- Pour gérer les comptes des utilisateurs et des groupes locaux, il faut être **root**.
- **Ajout d'un nouvel utilisateur**
  - La commande de base permettant l'ajout d'un utilisateur système est la suivante :

    #useradd username
  - Elle définit des valeurs par défaut pour tous les champs du fichier /etc/passwd (Si pas d'option).

    #passwd username
  - Affecte un mot de passe à l'utilisateur qui sera enregistré chiffré dans le fichier /etc/shadow

Gestion des utilisateurs locaux

- Les options les plus utilisées de cette commande sont les suivantes :

Option	Résumé
-c commentaires	Nom complet de l'utilisateur et des commentaires divers
-d rep_personnel	Par défaut dans le répertoire /home
-g groupe_initial	Groupe d'affectation du compte. Doit exister avant la création du compte.
-G liste	Fixe l'appartenance de l'utilisateur à une liste de groupes secondaires (séparateur, sans espace)
-m	Le répertoire personnel sera créé s'il n'existe pas.
-k squelette_rep	Recopie le contenu du répertoire squelette_rep dans le rép. Personnel ; par défaut /etc/skel
-s shell	Par défaut, attribution du shell par défaut bash
-u uid	Pour fixer l'identifiant uid à l'utilisateur

Gestion des utilisateurs locaux

- **Suppression d'un utilisateur**
  - Pour la suppression d'un utilisateur système sous Linux, on utilise la commande suivante :

    #userdel [-r] username
  - L'option « -r » supprime le répertoire personnel de l'utilisateur.
- Remarque :
  - La commande **usermod** [options] login modifie les fichiers d'administration des comptes du système selon les modifications qui ont été indiquées sur la ligne de commande.
  - Parmi ces options :
    - -l newlogin : change le login de cet utilisateur
    - -d newhomedirectory: change le répertoire personnel de cet utilisateur
    - -G nom groupe : affecte l'utilisateur à ce groupe
    - -s SHELL: change l'application de connexion de cet utilisateur

Gestion des groupes

- Le fichier **/etc/group** contient la liste des utilisateurs appartenant aux différents groupes.
  - Lorsque de nombreux utilisateurs peuvent avoir accès au système, ceux-ci sont fréquemment rassemblés en différents groupes ayant chacun leurs propres droits d'accès aux fichiers et aux répertoires.
  - Il se compose de différents champs séparés par "**:**" :

    nom_de_groupe : champ_special : numero_de_groupe : membre1, membre2

  - Le champ spécial est fréquemment vide.
  - Le numéro de groupe est le numéro qui fait le lien entre les fichiers /etc/group et /etc/passwd

Gestion des groupes

```
user@ubnt:~$ cat /etc/group
root:x:0:
daemon:x:1:
bin:x:2:
sys:x:3:
adm:x:4:syslog,user
tty:x:5:
systemd-coredump:x:999:
user:x:1000:
```

- **Ajout d'un nouveau groupe système**
  - La commande qui permet l'ajout d'un nouveau groupe système est la suivante :

    # **groupadd** nouveaugroupe
- **Suppression d'un groupe système**
  - La commande qui permet la suppression d'un groupe système est la suivante :

    # **groupdel** nomgroupe

Gestion des mots de passe

- Pour des raisons de sécurité les mots de passe ne sont plus sauvegardés dans le fichier /etc/passwd (lisible par tous les utilisateurs) mais plutôt dans le fichier **/etc/shadow**.
- Le fichier /etc/shadow contient ainsi les mots de passe et l'information sur l'expiration des comptes pour les utilisateurs et ressemble à cela :

user:Ep6mckrOLChF.:10063:0:99999:7:::

- **Nom d'utilisateur**, jusqu'à 8 caractères. Exactement la même entrée que dans le fichier /etc/passwd.
- **Mot de passe**, 13 caractères codés.
  - Une entrée nulle (exemple. ::) indique qu'un mot de passe n'est pas demandé pour entrer dans le système (une mauvaise idée en général),
  - Une entrée ``*'' (exemple. :*:) indique que le compte a été désactivé.

Gestion des mots de passe

user:Ep6mckrOLChF.:10063:0:99999:7:::

- Le nombre de jours (depuis le 1er Janvier 1970) depuis le dernier changement du mot de passe.
- Age minimum du mot de passe (un 0 indique qu'il peut être changé à n'importe quel moment).
- Age maximum du mot de passe (99999 indique que l'utilisateur peut garder son mot de passe inchangé pendant beaucoup, beaucoup d'années)
- Le nombre de jours pour avertir l'utilisateur qu'un mot de passe ne va plus être valable (7 pour une semaine entière)
- Le nombre de jours avant de désactiver le compte après expiration du mot de passe
- Le nombre de jours depuis le 1er Janvier 1970 pendant lesquels un compte a été désactivé

Gestion de mots de passe

- La commande qui permet de changer les informations sur le mot de passe est la commande **chage** :

$chage [option] login

- Pour afficher les informations sur le mot de passe d'un utilisateur :

$chage -l user

iut@iut-virtual-machine:~/Documents$ chage -l iut

Dernier changement de mot de passe : sept. 01, 2022

Fin de validité du mot de passe : jamais

Mot de passe désactivé : jamais

Fin de validité du compte : jamais

Nombre minimum de jours entre les changements de mot de passe : 0

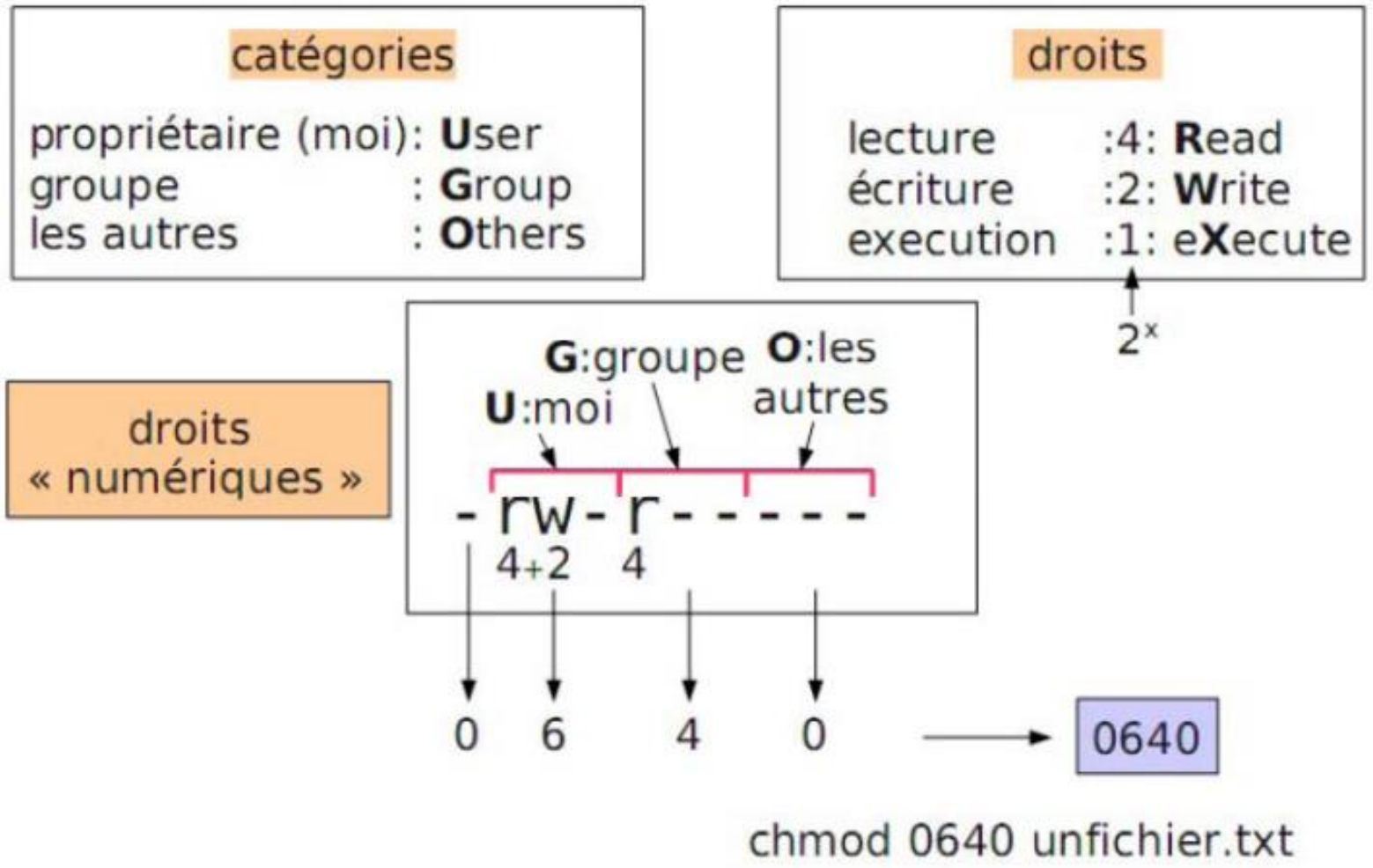
Nombre maximum de jours entre les changements de mot de passe : 99999

Nombre de jours d'avertissement avant la fin de validité du mot de passe : 7

Quelques commandes utiles

- **id** : permet d'afficher les informations concernant l'utilisateur connecté.
- **who** : affiche la liste des utilisateurs en cours de connexion.
- **users** : affiche la liste des utilisateurs connectés.
- **w**: affiche la liste des utilisateurs connectés.
- **last** : affiche la liste des connexions précédentes et en cours.
- **groups** : affiche la liste des groupes auxquels appartient l'utilisateur.

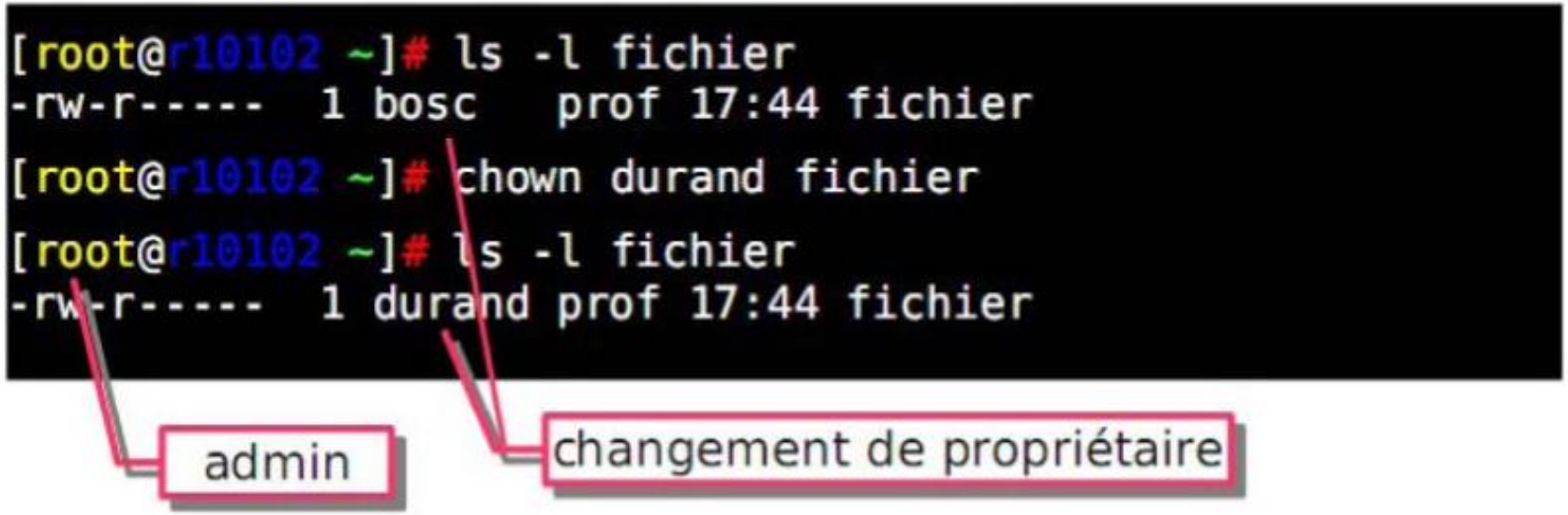
Droits d'accès



Changer de propriétaire

utilisateurs

chown: changer le propriétaire d'un fichier



chgrp: changer le groupe d'un fichier

Initialisation des droits d'accès

- La commande **umask**: permet de définir un masque de protection des fichiers lors de leur création
- La protection d'un fichier ainsi que les noms de son propriétaire et de son groupe sont établis à sa création et ne peuvent être modifiés que par son propriétaire ou par le super utilisateur (root).
- Le masque se comporte comme un filtre et utilise la notation numérique.
  - il ne contient pas la série des 3 chiffres octaux correspondants aux droits à allouer aux fichiers, mais celle correspondant aux droits **à ne pas allouer**.
- **Exemple** : si le masque de protection vaut 037 alors 740 (=777-037) seront les droits alloués à tout nouveau fichier.

  umask 037
  - 777 = rwx rwx rwx = 111 111 111
  - 037 = --- -wx rwx = 000 011 111
  - 740 = rwx r-- --- = 111 100 000

Initialisation des droits d'accès

```
User1@localhost > umask
     177
Æ la valeur du masque définie dans le fichier .bashrc
User1@localhost > touch f1
User1@localhost > ls -l f1
-rw------- 1 user1 etudiant 50  Sep 21 21:30 f1

Æ Le masque définie dans le fichier .bashrc est appliqué

User1@localhost > umask 022
User1@localhost > touch f2
User1@localhost > ls -l f2
-rwxr-xr-x 1 user1 etudiant 50  Sep 21 21:30 f2

Æ Le masque définie par la commande umask est appliqué

Déconnexion puis connexion...

User1@localhost > touch f3
User1@localhost > ls -l f3
-rw------- 1 user1 etudiant 50  Sep 21 21:30 f3

Æ Le masque défini dans le fichier .bashrc est appliqué
```

Droits Spéciaux :Set User ID (SUID)

- Le SUID est un droit qui va permettre de faire exécuter un script avec les droits de l'utilisateur propriétaire du script.
- Un `s` remplace le `x` dans le listage des droits du script :

```
-r-Sr-xr-x 1 root wheel 24876 27 May 18:18 un_script
```

- Appliquer le SUID :

```
# chmod u+s un_script
```


Droits spéciaux :Set Group ID (SGID)

- Le SGID est un droit qui va permettre de faire exécuter un script avec les droits du groupe propriétaire du script.
- Un `s` remplace le `x` dans le listage des droits du script :

```
-r-xr-Sr-x 1 root wheel 24876 27 May 18:18 un_script
```

- Appliquer le SGID :

```
# chmod g+s un_script
```


Gestion des ACLs

- **ACL : Access Control List**

- Absence de gestion atomique des droits Unix.
  - **etudiant1** est un membre du groupe **iris**
  - **etudiant2** et **etudiant3** le sont aussi.
  - **etudiant1** veut partager un document avec **etudiant2**, mais ne veut pas que **etudiant3** soit capable de le lire, ni modifie.
  - On ne peut le faire avec les Permissions Unix POSIX ?
  - Ils sont du même groupe

C'est possible avec les ACLs.

Ajout d'ACL's

- Ajouter une ACL à un utilisateur

```
# setfacl -m u:etudiant:rw /var/www/index.php
```

➔ Ajouter une ACL à un groupe

```
# setfacl -m g:cesi:rw /var/www/index.php
```


Effacer une ACL

- **Effacer une ACL sur un fichier:**

```
# setfacl -b  /var/www/index.php
```

➔ **Rétirer les droits à un utilisateur sur un fichier:**

```
# setfacl -x u:etudiant /var/www/index.php
```


Afficher les ACL's

- La commande getfacl permet d'afficher les ACL's

```
$ getfacl monfichier.txt
# file: monfichier.txt
# owner: etudiant
# group: iris
user::rw-
user:prof:rwx
group::r--
mask::rwx
other::r--
```


Fin du Chapitre 4