ADMINISTRATION DE L'INFRASTRUCTURE INFORMATIQUE

Pr. Najat TISSIR

tissir.najat@gmail.com

Filière: Génie Informatique

CHAPITRE 5: GESTION DES DROITS D'ACCÈS

1. Droits d’accès
2. Modification des droits d'accès
3. Modification du propriétaire d'un fichier/répertoire
4. Modification du groupe d'un fichier/répertoire
5. Enlever des permissions par défaut pour les nouveaux fichiers

DROITS D'ACCÈS

- Linux est un système multi-utilisateur qui permet de gérer les permissions d'accès aux fichiers.
- Chaque utilisateur a un identifiant (UID), un nombre unique qui l'identifie.
- Les utilisateurs appartiennent également à un ou plusieurs groupes.
- Les groupes peuvent être employés pour limiter l'accès à un certain nombre de personnes.
- Pour vérifier votre identification d'utilisateur et voir le groupe(s) auquel vous appartenez, tapez la commande **id**:

```
najat@najat-VirtualBox:~$ id
uid=1000(najat) gid=1000(najat) groups=1000(najat),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),108(kvm),120(lpadmin),131(lxd),132(sambashare),1001(libvirt)
najat@najat-VirtualBox:~$
```

DROITS D'ACCÈS

Propriété et permissions des fichiers:

- Les opérations qui peuvent être effectuées sur un fichier sont: la lecture, l'écriture et l'exécution.
- Il existe un mécanisme permettant de protéger les fichiers des utilisateurs contre tout accès non autorisé ou toute tentative malveillante d'autrui.
  - Il repose sur les permissions des fichiers, qui sont un des concepts de base de la gestion d'utilisateurs multiples sous Unix.
  - Il va permettre d'interdire/autoriser la lecture, l'écriture ou l'exécution de certains fichiers par certains utilisateurs.

DROITS D'ACCÈS

Principe:

- À chaque fois qu'un utilisateur veut effectuer une opération sur un fichier, le système va vérifier que cette opération lui est permise.
- Cette vérification repose sur les 5 informations suivantes :
  - l'utilisateur qui tente d'effectuer l'opération ;
  - ses groupes ;
  - le propriétaire du fichier ;
  - le groupe du fichier ;
  - les permissions (ou droits d'accès) du fichier.
- Les trois dernières informations, relatives au fichier, peuvent être visualisées avec la commande **ls** (option **-l**).

DROITS D'ACCÈS

Principe:

- Les permissions du fichier sont composées de trois parties, chacune s'adressant à une catégorie d'utilisateur :
  - les droits du propriétaire ;
  - les droits des membres du groupe du fichier ;
  - les droits des autres utilisateurs.
- Les droits d'un utilisateur sur un fichier sont uniquement ceux de sa catégorie la plus spécifique (propriétaire, sinon membre du groupe, sinon autre).
- L'opération demandée par l'utilisateur ne sera autorisée que s'il possède les droits qu'elle nécessite.
- Nous avons vu qu'il existe 3 types de droits : r, w et x
- Ces droits n'ont pas la même signification pour un fichier que pour un répertoire

DROITS D'ACCÈS

Les droits pour un fichier:

- Lecture (R) pour lire le contenu ;
- Écriture (W) pour modifier le contenu ;
- Exécution (X) pour l'exécuter s'il s'agit d'un fichier binaire contenant du code exécutable.
  - S'il s'agit d'un fichier texte(contenant ce qu'on appelle un script), il faut aussi le droit de lecture pour l'exécuter

DROITS D'ACCÈS

Les droits pour un répertoire:

- Lecture: Lire le contenu, lister les fichiers (avec ls par exemple)
- Écriture: Modifier le contenu, créer et supprimer des fichiers (avec les commandes « cp », « mv », « rm »)
  - Le droit d'écriture ne sert à rien si on n'a pas aussi le droit d'exécution ;
- Exécution: autoriser l'accès au répertoire. Son absence permet donc d'interdire l'accès à une partie de l'arborescence du système de fichiers.

DROITS D'ACCÈS

Les droits pour un répertoire:

- En effet, sans le droit d'exécution sur un répertoire, un utilisateur ne peut pas le "traverser" et ne peut alors pas :
  - faire de ce répertoire son répertoire de travail, c'est à dire aller dans ce répertoire (en utilisant **cd**) ;
  - référencer un fichier qui se trouve dans l'arborescence de ce répertoire ;
  - obtenir des détails sur les fichiers contenus dans ce répertoire (notamment avec l'option **-l** de **ls**).

DROITS D'ACCÈS

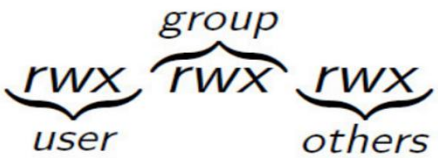
Connaître ses droits sur un fichier:

- Les permissions d'accès pour un fichier peuvent être positionnés par propriétaire, groupe et pour les autres sur la base de permissions en lecture (r), écriture (w) et exécution (x).
- Les droits d'accès sont attribués aux fichiers ou aux répertoires.
- Il y a trois catégories de droits :
  - Lecture (read).
  - Écriture (write).
  - Exécution (execute).

DROITS D'ACCÈS

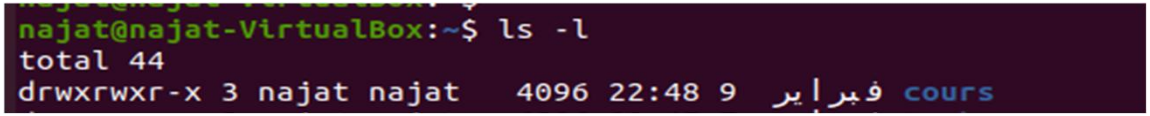
Représentation:

- Les droits sont définis pour trois types d'utilisateurs:
  - le propriétaire du fichier (**u=user**)
  - le groupe auquel appartient le fichier (**g=group**)
  - tous les autres utilisateurs (**o=other**)
- **a=all** signifie **user+group+other**



DROITS D'ACCÈS

Exemple:



- Le premier **d** signifie que c'est un répertoire (tiret pour un fichier).
- Les trois caractères suivants (**rwx**) montrent les droits de l'utilisateur propriétaire du fichier.
- Les trois caractères suivants (**rwx**) montrent les droits du groupe auquel appartient le fichier.
- Les trois derniers caractères (**r-x**) montre les droits des autres utilisateurs.
- **najat** est le propriétaire du fichier.
- **najat** est le groupe auquel ce fichier appartient

DROITS D'ACCÈS

- Quel(s) utilisateur(s) pourra(ont) se déplacer dans le répertoire suivant ?

```
drwxr-x--- 26 hiba Informatique 4096 2008-08-28 16:11 hiba
```

- **Réponse** : « hiba» et les membres du groupe « Informatique »
- Qui pourra créer de nouveaux fichiers dans ce répertoire ?

```
drwxr-xrwx 26 Salim Informatique 4096 2008-08-28 16:11 bilans
```

- **Réponse** : Tout le monde sauf les membres du groupe «Informatique»
- Soit le fichier suivant :

```
-rwxr--r-- 26 Souad ensa 25140 2008-08-28 16:11 rapport2006.odt
```

- Situé dans le répertoire suivant :

```
drwxrwxrwx 26 Salim ensa 4096 2008-08-28 16:11 rapports
```

- Qui pourra effacer ce fichier ?
- **Réponse** : Tout le monde malheureusement !!!

➢ **Le droit «w» accordé à un répertoire permet d'y effacer des fichiers quels que soient le propriétaire et les droits qui s'appliquent à ces fichiers**

1. Droits d’accès
2. Modification des droits d'accès
3. Modification du propriétaire d'un fichier/répertoire
4. Modification du groupe d'un fichier/répertoire
5. Enlever des permissions par défaut pour les nouveaux fichiers

MODIFICATION DES DROITS D'ACCÈS

- Les droits d'accès ne peuvent être modifiés que par le propriétaire ou l'administrateur.
- La commande **chmod** permet de modifier les droits d'accès.
- Elle peut être utilisée de deux façons différentes (symbolique et octal).
- Pour changer les droits, on doit spécifier:
  - Les droits (**r=read, w=write, x=execute**).
  - A qui s'appliquent ces droits (**u=user, g=group, o=other, a=all**).
  - Le ou les fichiers/répertoires dont on veut changer les droits.
- Pour ajouter des droits l'opérateur **+** est utilisé.
- Pour enlever des droits l'opérateur **–** est utilisé.

MODIFICATION DES DROITS D'ACCÈS

Mode symbolique:

- Dans ce mode, les permissions auront la forme suivante :

  chmod qui operateur quoi document

- Où **qui** indique à qui on veut fixer des permissions.
  - C'est une combinaison des lettres u, g, o et a, représentant :
    - **u (user)** le propriétaire ;
    - **g (group)** les membres du groupe ;
    - **o (others)** les autres ;
    - **a (all)** à la fois le propriétaire, les membres du groupe et les autres.
- **quoi** indique les droits à attribuer
- L'**operateur** est l'un des signes **+, =** et **-**, pour indiquer si l'on veut :
  - **+** ajouter des permissions,
  - **=** fixer exactement des permissions,
  - **-** supprimer des permissions.

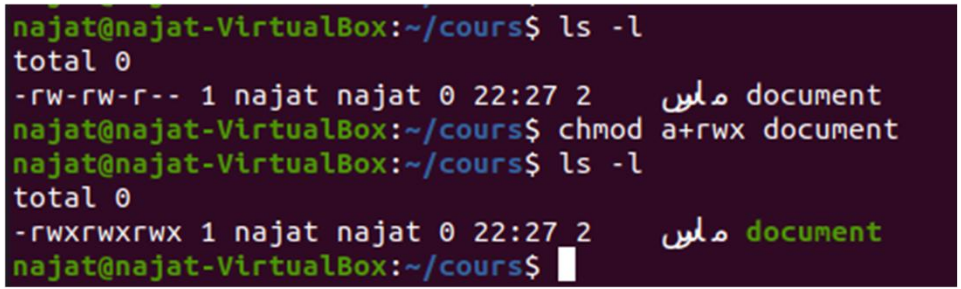
MODIFICATION DES DROITS D'ACCÈS

Mode symbolique:

- Dans l'exemple suivant on donne tous les droits d'accès à tous les utilisateurs :
- $**chmod a+rwx** doucument

Ou

- $**chmod ugo+rwx** doucument



MODIFICATION DES DROITS D'ACCÈS

Mode symbolique:

- Dans l'exemple qui suit, on enlève le droit en écriture et on rajoute le droit en lecture à group et other:
- $**chmod go-w+r** doucument

```
najat@najat-VirtualBox:~/cours$ chmod go-w+r document
najat@najat-VirtualBox:~/cours$ ls -l
total 0
-rwxr-xr-x 1 najat najat 0 22:27 2    ملس document
najat@najat-VirtualBox:~/cours$
```

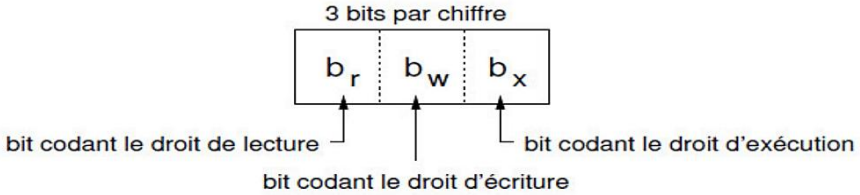
- Dans l'exemple qui suit on enlève le droit read, write et execute à group et other pour les fichiers et les sous répertoires du répertoire courant :
- $**chmod –R go-rwx** *

```
najat@najat-VirtualBox:~/cours$ chmod -R go-rwx *
najat@najat-VirtualBox:~/cours$ ls -l
total 0
-rwx------ 1 najat najat 0 22:27 2    ملس document
najat@najat-VirtualBox:~/cours$
```

MODIFICATION DES DROITS D'ACCÈS

Mode octal:

- Dans ce mode, permissions est un nombre (en octal) de la forme **Cu Cg Co** où Cu, Cg et Co sont 3 chiffres compris entre 0 et 7 :
  - **Cu**: indique les permissions du propriétaire ;
  - **Cg**: indique les permissions du groupe ;
  - **Co**: indique les permissions des autres.
- Chaque chiffre est obtenu en sommant des valeurs correspondant aux différents droits :
- **4 pour la lecture ; 2 pour l'écriture ; 1 pour l'exécution**.
- Plus exactement, chaque chiffre est un champ de 3 bits :



MODIFICATION DES DROITS D'ACCÈS

Mode octal:

- Dans l'exemple suivant, on supprime tous les droits à tout le monde :
- $**chmod 000** document

```
najat@najat-VirtualBox:~/cours$ chmod 000 document
najat@najat-VirtualBox:~/cours$ ls -l
total 0
---------- 1 najat najat 0 22:27 2    ملس document
najat@najat-VirtualBox:~/cours$
```

- Dans l'exemple suivant on donne le droit de lecture et d'écriture au propriétaire et au groupe et uniquement le droit de lecture aux autres utilisateurs :
- $**chmod 664** document

```
najat@najat-VirtualBox:~/cours$ chmod 664 document
najat@najat-VirtualBox:~/cours$ ls -l
total 0
-rw-rw-r-- 1 najat najat 0 22:27 2    ملس document
najat@najat-VirtualBox:~/cours$
```

MODIFICATION DES DROITS D'ACCÈS

Mode octal:

- L'option **–v** permet d'afficher les modifications réalisées :

```
najat@najat-VirtualBox:~/cours$ chmod -v 664 document
mode of 'document' changed from 0000 (---------) to 0664 (rw-rw-r--)
```

1. Droits d’accès
2. Modification des droits d'accès
3. Modification du propriétaire d'un fichier/répertoire
4. Modification du groupe d'un fichier/répertoire
5. Enlever des permissions par défaut pour les nouveaux fichiers

MODIFICATION DU PROPRIÉTAIRE D'UN FICHIER/RÉPERTOIRE

- Seulement **root** peut changer le propriétaire (owner) d'un fichier ou répertoire.
- Pour modifier le propriétaire, vous pouvez utiliser les commandes **chown** (change owner):
- $**chown user1** document
- Ou $**chown user1** /home/najat/cours/document
- Dans cet exemple, **user1** devient le nouveau propriétaire du fichier document:

```
najat@najat-VirtualBox:~/cours$ ls -l
total 0
-rw-rw-r-- 1 najat najat 0 22:27 2    ملس document
najat@najat-VirtualBox:~/cours$ chown user1 document
chown: changing ownership of 'document': Operation not permitted
najat@najat-VirtualBox:~/cours$ sudo chown user1 document
najat@najat-VirtualBox:~/cours$ ls -l
total 0
-rw-rw-r-- 1 user1 najat 0 22:27 2    ملس document
```

MODIFICATION DU PROPRIÉTAIRE D'UN FICHIER/RÉPERTOIRE

- L'option **–v** permet d'afficher les modifications réalisées :

```
najat@najat-VirtualBox:~/cours$ sudo chown -v root document
changed ownership of 'document' from user1 to root
najat@najat-VirtualBox:~/cours$
```

MODIFICATION DU PROPRIÉTAIRE D'UN FICHIER/RÉPERTOIRE

- Pour changer le propriétaire (owner) et le groupe :

```
najat@najat-VirtualBox:~/cours$ sudo chown -v root.users document
changed ownership of 'document' from root:najat to root:users
najat@najat-VirtualBox:~/cours$ ls -l
total 0
-rw-rw-r-- 1 root users 0 22:27 2    ملس document
najat@najat-VirtualBox:~/cours$
```

- Pour savoir quels sont les groupes où vous êtes membre, tapez la commande **groups** :

```
najat@najat-VirtualBox:~/cours$ groups
najat adm cdrom sudo dip plugdev kvm lpadmin lxd sambashare libvirt
najat@najat-VirtualBox:~/cours$ sudo su
root@najat-VirtualBox:/home/najat/cours# groups
root
root@najat-VirtualBox:/home/najat/cours#
```

1. Droits d’accès
2. Modification des droits d'accès
3. Modification du propriétaire d'un fichier/répertoire
4. Modification du groupe d'un fichier/répertoire
5. Enlever des permissions par défaut pour les nouveaux fichiers

MODIFICATION DU GROUPE D'UN FICHIER/RÉPERTOIRE

- Pour modifier le groupe d'un fichier, vous pouvez utiliser la commande **chgrp** (change group).
- Seulement **root** et le propriétaire (owner) s'il appartient au nouveau groupe ont le droit de changer le groupe.
- Dans l'exemple suivant **libvirt** devient le nouveau groupe du fichier document:

```
najat@najat-VirtualBox:~/cours$ ls -l
total 0
-rw-rw-r-- 1 najat najat 0 22:27 2    ملس document
najat@najat-VirtualBox:~/cours$ groups
najat adm cdrom sudo dip plugdev kvm lpadmin lxd sambashare libvirt
najat@najat-VirtualBox:~/cours$ chgrp libvirt document
najat@najat-VirtualBox:~/cours$ ls -l
total 0
-rw-rw-r-- 1 najat libvirt 0 22:27 2    ملس document
najat@najat-VirtualBox:~/cours$
```

MODIFICATION DU GROUPE D'UN FICHIER/RÉPERTOIRE

- On peut aussi utiliser la commande **chown**.
- Dans l'exemple suivant **users** devient le nouveau groupe du fichier document :

```
najat@najat-VirtualBox:~/cours$ sudo chown -v root.users document
changed ownership of 'document' from root:najat to root:users
najat@najat-VirtualBox:~/cours$ ls -l
total 0
-rw-rw-r-- 1 root users 0 22:27 2    ملس document
najat@najat-VirtualBox:~/cours$
```

1. Droits d’accès
2. Modification des droits d'accès
3. Modification du propriétaire d'un fichier/répertoire
4. Modification du groupe d'un fichier/répertoire
5. Enlever des permissions par défaut pour les nouveaux fichiers

ENLEVER DES PERMISSIONS PAR DÉFAUT POUR LES NOUVEAUX FICHIERS

- Tous les fichiers ont des permissions fixées dès leur création.
- Elles dépendent de l'utilitaire employé et du type de fichier créé, ainsi que du masque de création de fichiers.
- Le masque indique au système les permissions **que ne doivent pas avoir** les fichiers lors de leur création (uniquement).
- Sous Bash, il est consultable/modifiable par la commande interne **umask.**

ENLEVER DES PERMISSIONS PAR DÉFAUT POUR LES NOUVEAUX FICHIERS

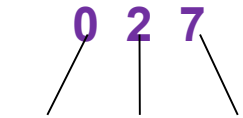
- Syntaxe:

umask [**masque**]

- Sans argument, **umask** indique la valeur actuelle du masque de création de fichiers, sinon **masque** est sa nouvelle valeur.
- L'argument **masque** est un nombre (en octal) de la forme **Cu Cg Co** où Cu, Cg et Co sont 3 chiffres compris entre 0 et 7 :
  - **Cu** indique les permissions à ne pas accorder au propriétaire ;
  - **Cg** indique les permissions à ne pas accorder au groupe ;
  - **Co** indique les permissions à ne pas accorder aux autre

ENLEVER DES PERMISSIONS PAR DÉFAUT POUR LES NOUVEAUX FICHIERS

- Les bits du masque à 1 empêchent le fichier d'obtenir le droit correspondant
- Exemple avec un masque de protection de 027:



000 010 111
rwx rwx rwx Permissions maximum
rwx r-x --- Permissions effectives après application du masque

- Les fichiers et répertoires nouvellement crées seront protégés
- La valeur par défaut du masque est **022**

ENLEVER DES PERMISSIONS PAR DÉFAUT POUR LES NOUVEAUX FICHIERS

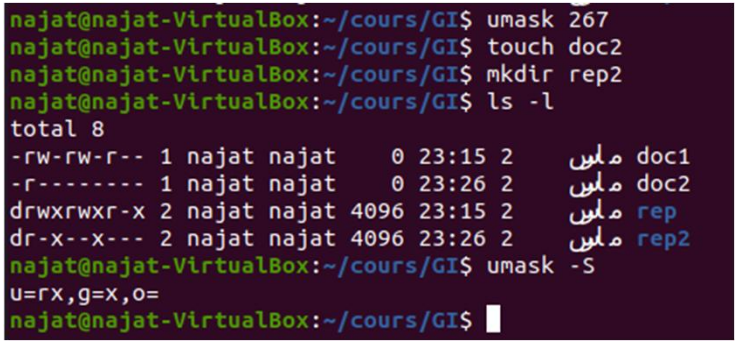
- Exemple:

```
najat@najat-VirtualBox:~/cours$ umask
0002
najat@najat-VirtualBox:~/cours$ mkdir GI
najat@najat-VirtualBox:~/cours$ cd GI
najat@najat-VirtualBox:~/cours/GI$ touch doc1
najat@najat-VirtualBox:~/cours/GI$ ls -l
total 0
-rw-rw-r-- 1 najat najat 0 23:15 2    ملس doc1
najat@najat-VirtualBox:~/cours/GI$
najat@najat-VirtualBox:~/cours/GI$ mkdir rep
najat@najat-VirtualBox:~/cours/GI$ ls -l
total 4
-rw-rw-r-- 1 najat najat    0 23:15 2    ملس doc1
drwxrwxr-x 2 najat najat 4096 23:15 2    ملس rep
najat@najat-VirtualBox:~/cours/GI$
```

  - Le premier chiffre (0) représente le bit « **special** »: Il est généralement 0 car il concerne les bits spéciaux (SetUID, SetGID, Sticky bit).
    - Dans la plupart des cas, il est ignoré.
  - Conformément au masque (002), le droit d'**écriture** ne sera pas accordé aux « autres ».

ENLEVER DES PERMISSIONS PAR DÉFAUT POUR LES NOUVEAUX FICHIERS

- Exemple:



- On ne veut pas que soient accordés le **droit d'écriture** pour le **propriétaire**, les **droits de lecture et d'écriture** pour le **groupe** et **l'ensemble des droits** pour les **autres**.
- **Rq** : L'affichage du masque 000 sous forme symbolique en utilisant l'option **-S** de **umask**