

DEV1 – ENVL – Laboratoires d'environnement**TD 1 – Les permissions****Table des matières**

1	Se documenter	2
1.1	Les options	2
1.2	Recherche d'informations	2
2	Les permissions	3
2.1	Les groupes d'utilisateurs	3
2.2	Propriétaire d'un fichier	4
2.3	Groupes d'un fichier	4
2.4	Les permissions	5
2.5	Conclusion	7

1 Se documenter

1.1 Les options

La plupart des commandes possèdent des **options**.

Les options

Une **option** modifie le **sens** d'une commande ;

- ▷ Elle commence par le signe - suivi d'une seule lettre ;
- ▷ ou encore par le double tiret -- suivi d'un nom d'option.
- ▷ Elle est placée n'importe où après le nom de la commande.

Experience 1

Expérimenter les options

- ✍ Tapez la commande `ls -l`. Vous constatez que le résultat obtenu est beaucoup plus verbeux que celui obtenu sans l'option.
- ✍ Placez vous dans le dossier `td1` et tapez la commande `cat -n test`. L'option demande de numéroté les lignes
- ✍ Essayez `cat --number test`, la version *longue* équivalente à la précédente.

1.2 Recherche d'informations

Non seulement, il y a beaucoup de commandes à connaître mais, en plus, chacune dispose d'une multitude d'options. Impossible de tout retenir ! Comment faire ?

Information

- ▷ `nomCommande --help` affiche une aide succincte sur la commande.
- ▷ `man nomCommande` affiche une documentation plus complète. (*q* pour quitter).

Si vous ne connaissez pas le nom de la commande, consultez les documents que l'on met à votre disposition, notamment le *Linux Cheat Sheet*.

Exercice 1

Trouver la bonne option

La commande `ls -l` affiche le contenu du dossier en format *long*. La 5^e colonne donne la taille du fichier (en octets). Lorsque les nombres sont grands, ce n'est pas très lisible. Trouvez l'option qui permet d'afficher cette taille sous un format plus lisible.

De grâce, **cherchez** la réponse, ne la **demandez pas** à votre voisin. Le but de cet exercice n'est pas de connaître l'option (elle n'est pas si utile que ça) mais d'apprendre à trouver soi-même l'information.

Exercice 2

Trouver la bonne commande

Quelle commande permet de

- ▷ *nettoyer* l'écran ?
- ▷ afficher la date et l'heure ?

2 Les permissions

Comme Linux est un système partagé, il est important de parler de sécurité. Ici, on va se concentrer sur la sécurité au niveau des fichiers et répondre aux questions suivantes :

- ▷ À qui **appartient** un fichier ?
- ▷ **Qui peut faire** quoi avec un fichier ?

Mais pour commencer il faut d'abord comprendre la notion de *propriétaire* et celle de *groupe*.

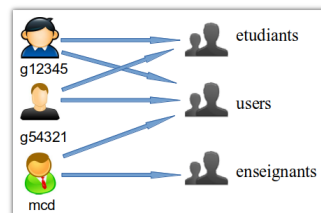
2.1 Les groupes d'utilisateurs

Les groupes

Les utilisateurs d'un système Linux sont placés dans des **groupes**. Un groupe contient un ou plusieurs utilisateur(s) et un utilisateur appartient à un ou plusieurs groupe(s).

Exemple : Dans l'exemple ci-contre on peut constater que :

- ▷ l'utilisateur `g12345` appartient aux groupes `users` et `etudiants` ;
- ▷ l'utilisateur `g54321` appartient aux mêmes groupes ;
- ▷ l'utilisateur `mcd` appartient aux groupes `users` et `enseignants`.



À quoi servent les groupes ? À gérer les permissions. On va pouvoir dire un truc du genre : « Seuls les professeurs peuvent voir le contenu de ce fichier ».

Visualiser les groupes

La commande **groups** renseigne sur les groupes.

- ▷ `groups` : affiche les groupes de l'utilisateur.
- ▷ `groups loginUtilisateur` : affiche les groupes de l'utilisateur donné.

Le premier groupe de la liste est le **groupe principal**. C'est celui qui sera utilisé par défaut (par exemple, lorsqu'on crée un fichier).

Exercice 3

Visualiser les groupes

- ▷ Visualiser les groupes auxquels vous appartenez.
- ▷ Quel est votre groupe principal ?
- ▷ Quels sont les groupes auxquels appartient votre professeur ?
- ▷ Avez-vous un groupe en commun avec lui ?
- ▷ Quel(s) groupe(s) Linux avez-vous en commun avec les autres étudiants de votre groupe ÉSI ?

Les groupes qui existent concrètement sur une machine sont définis par l'administrateur ^a de la machine. Sur **linux1**, il y a 3 groupes pour les utilisateurs.

- ▷ **users** : tous les utilisateurs sont dans ce groupe
- ▷ **etudiants** : tous les étudiants sont dans ce groupe
- ▷ **enseignants** : tous les professeurs sont dans ce groupe

^a. L'administrateur est la personne qui installe un système d'exploitation et le gère au quotidien : installation de logiciels, gestion des comptes utilisateurs et des groupes, sauvegardes, gestion des panne... Sur Linux, on dit aussi le « super user », le « compte root » ou le « root ».

Attention, on remarque que vous confondez souvent la notion de **groupe d'étudiants** à l'école et les **groupes sur Linux**. C'est un même mot qui recouvre 2 concepts complètement différents.

2.2 Propriétaire d'un fichier

« La propriété est un piège : ce que nous croyons posséder nous possède. » – Alphonse Karr

« La propriété c'est le vol. » – Pierre Joseph Proudhon

Propriétaire

Chaque fichier/dossier **appartient** à une personne, son **propriétaire**. On peut le visualiser via la commande `ls -l`.

```
~> ls -l
total 8
drwxrwxr-x 2 mcd users 4096 mai  6 23:46 dev1
-rw-rw-r-- 1 mcd users  42 mai  6 23:46 welcome
```

Le propriétaire

Exercice 4 Propriétaire

Visualisez le propriétaire des fichiers de votre dossier personnel.

2.3 Groupes d'un fichier

Tutoriel 1 Groupe

1. Visualisez vos fichiers et déterminez à quel groupe ils appartiennent.
2. Créez un fichier de test et modifiez le groupe auquel il appartient.

FAQ Les fichiers dans mon dossier personnel ne sont pas automatiquement à moi ?

Non. En pratique c'est généralement le cas, mais on peut très bien trouver dans un dossier personnel un fichier qui appartient à quelqu'un d'autre.

2.4 Les permissions

à présent que vous savez qu'un fichier a un propriétaire et appartient à un groupe, on peut étudier la notion de permission.

Restez concentré! Cette partie est plus longue et un peu plus difficile que ce que vous avez déjà appris mais c'est absolument nécessaire pour la suite.

Un retour vers les points 17 à 19 du guide visuel sera peut-être encore nécessaire...

Exercice 5

Déterminez les bonnes permissions

Remplissez les blancs avec la permission correcte (r, w, x ou -). Il s'agit de trouver la permission minimale à mettre pour répondre à la demande.

- ▷ Pour un fichier Java, la permission la plus adéquate est _ _ _
 - ▷ Le fichier qui contient (l'exécutable de) `ls` a probablement comme permission _ _ _
- Vous pouvez le vérifier. Comment ?

Exercice 6

Exercice

Soit le fichier "Max.java" de la capture d'écran ci-dessous.

```
-> ls -l
total 20
drwxr-xr-x 2 g32671 users 1024 mai 29 14:27 bin
drwxr-xr-x 2 g32671 users 1024 oct 21 2008 cours
drwx----- 2 g32671 users 1024 mai 29 14:27 Documents
drwxr-xr-x 3 g32671 users 1024 nov 20 2008 evaluation
-rw-r--r-- 1 g32671 users 417 oct 21 2008 Max.java
-rw-r--r-- 1 g32671 users 418 oct 21 2008 Max.java~
-rw-r--r-- 1 g32671 users 0 oct 21 2008 Maxnombre.java
```

FIGURE 1 – Contenu détaillé d'un dossier

Est-ce qu'un professeur peut l'éditer ?

Exercice 7

Déterminez les bonnes permissions

Soit le fichier "Max.java" de la capture d'écran ci-dessus.

On voudrait que l'étudiant g32671 puisse travailler normalement, que les autres étudiants ne puissent pas tricher sur lui mais que les professeurs puissent lire son travail.

- ▷ Quel groupe faut-il donner au fichier ?

- ▷ Quelle commande permet de donner ce groupe au fichier ?

- ▷ Quelles permissions minimales donner au fichier ?
_ _ _ _ _
- ▷ Quelle commande permet de donner ces permissions au fichier ?

```

-> ls -l
total 20
drwxr-xr-x 2 g32671 users 1024 mai 29 14:27 bin
drwxr-xr-x 2 g32671 users 1024 oct 21 2008 cours
drwx----- 2 g32671 users 1024 mai 29 14:27 Documents
drwxr-xr-x 3 g32671 users 1024 nov 20 2008 evaluation
-rw-r--r-- 1 g32671 users 417 oct 21 2008 Max.java
-rw-r--r-- 1 g32671 users 418 oct 21 2008 Max.java~
-rw-r--r-- 1 g32671 users 0 oct 21 2008 Maxnombre.java

```

FIGURE 2 – Contenu détaillé d'un dossier

Exercice 8

Reprenez les permissions affichées dans la capture d'écran ci-dessous et exprimez-les avec un nombre de 3 chiffres.

Tutoriel 2

Permissions par défaut

1. Si ce n'est pas encore fait, créez un dossier "td2".
2. Créez-y un fichier vide.
3. Demandez les détails du fichier (propriétaire, groupe, permission)

On constate qu'un nouveau fichier appartient à celui qui l'a créé (on s'en doute) et au groupe principal du créateur. Il y a aussi des permissions par défaut (plutôt permissives dans notre cas).

Modifier les permissions

Vous savez que la commande qui permet de modifier les permissions d'un fichier est `chmod`.

Prenez le temps de **lire** la page de **manuel** de cette commande.

Tutoriel 3

Modifier les permissions

1. Créez un fichier "brol" dans le dossier "td2" avec quelques mots.
2. Faites en sorte que personne d'autre ne puisse en voir le contenu.
3. Faites en sorte que tout le monde puisse voir son contenu mais pas le modifier.
4. Faites en sorte que les autres étudiants ne puissent pas voir son contenu mais les professeurs bien. Attention, pour ce faire, il faut pouvoir distinguer les étudiants des enseignants; et donc, distinguer les groupes.

Tutoriel 4

Modifier encore les permissions

Modifiez les droits de votre dossier "td2" et, si nécessaire, des fichiers qui s'y trouvent pour que tout le monde puisse

1. voir quels fichiers s'y trouvent mais sans pouvoir lire le contenu de ces fichiers;
2. modifier le contenu d'un des fichiers mais pas supprimer ce fichier;
3. supprimer un fichier mais pas modifier son contenu.

FAQ

Pourquoi avoir choisi 4, 2 et 1 pour désigner les permissions ?

C'est lié au code binaire. 'rwx' donne, pour 'r', la valeur '100' soit 4 en décimal.

Vous dites que dans un affichage en format long, le premier caractère indique si c'est un fichier simple ('-') ou un dossier ('d'). Pourtant j'ai déjà vu d'autres symboles. C'était quoi ?

Il existe d'autres types de fichiers que les deux que nous avons vus. Ils se rencontrent moins souvent et sont surtout utilisés par le système. Par exemple, certains définissent des *pilotes* vers le matériel. Si vous voulez en savoir plus, vous pouvez lire ceci (en.wikipedia.org/wiki/Unix_file_types).

Vous avez mentionné les permissions 'r', 'w' et 'x'. Pourtant j'ai déjà vu d'autres lettres dans la zone réservée aux permissions. C'était quoi ?

Il y a 3 permissions dont nous n'avons pas parlé parce qu'elles sont moins courantes : le *suid* (set user id), le *sgid* (set group id) et le *sticky*. Si vous voulez en savoir plus, vous pouvez lire ceci (fr.wikipedia.org/wiki/Permissions_UNIX).

Vous n'avez pas expliqué le sens de la 2^e colonne fournie par la commande ls (juste avant le propriétaire) ?

C'est vrai mais c'est moins utile et plus lié à la structure interne du système de fichier. Je veux bien vous dire qu'il s'agit du nombre de liens physiques sur le fichier mais je sens que vous commencez déjà à regretter d'avoir posé la question ;)

Nous avons vu qu'un fichier est créé avec des permissions par défaut. C'est configurable ?

Oui. Voyez la commande `umask`.

2.5 Conclusion

FAQ

Euh ! Je n'ai pas fini. C'est normal ? Non, c'est que vous avez mal préparé votre TD avant de venir au labo. Terminez-le en remédiation et/ou à la maison car la semaine prochaine, nous en attaquons un nouveau et, cette fois, essayez de mieux vous y préparer !

Je n'ai pas Linux à la maison. Je peux quand même terminer mon TD ?

Non. Vous devrez rapidement disposer d'un Linux fonctionnel à la maison. Ce qui devrait d'ailleurs être le cas si vous avez fait les exercices demandés au cours d'introduction à l'OS.

J'ai un Linux à la maison et les groupes ne sont pas les mêmes. C'est normal ?

Oui. Les groupes dépendent à la fois de la distribution particulière utilisée et de la façon dont l'administrateur (le root) a configuré le système.