

Droits et permissions

Utilisateurs, groupes, lecture des droits et commande chmod

🎯 Objectifs de ce chapitre

- Expliquer pourquoi un système multi-utilisateur a besoin de droits d'accès.
- Lire les droits d'un fichier ou d'un répertoire affichés par `ls -l`.
- Modifier les droits avec la commande `chmod`.
- Prévoir ce que l'on peut faire ou non sur un fichier, d'après ses droits.

Pourquoi des droits ?

Un système d'exploitation de type Unix est **multi-utilisateur** : plusieurs personnes partagent la même machine, chacune avec ses propres fichiers. Il faut donc pouvoir décider qui a le droit de lire, de modifier ou d'exécuter quoi. Sans cela, un élève pourrait lire ou effacer les fichiers d'un autre, ou casser le système lui-même. Vous avez d'ailleurs déjà rencontré un refus d'accès en tapant `ls /root`.

🐛 Activité 1 — qui a le droit de faire quoi ?

1. Tapez `id`. Repérez votre nom d'utilisateur et la liste de vos groupes.
2. Tapez `ls -l /etc/passwd /etc/shadow /bin/ls`. Pour chaque fichier, notez les dix premiers caractères, le propriétaire et le groupe.
3. Comparez `head -n 2 /etc/passwd` et `head -n 2 /etc/shadow`. Que se passe-t-il ? Reliez ce que vous observez aux droits relevés à l'étape 2.
4. Tapez `ls -ld /root` (l'option `-d` affiche le répertoire lui-même, et non son contenu). Expliquez le message « Permission non accordée » (ou *Permission denied*) obtenu avec `ls /root`.

📖 Ce que tu viens de découvrir

- Chaque fichier a un **propriétaire** et un **groupe**.
- Les droits ne sont pas les mêmes pour le propriétaire, pour son groupe et pour tous les autres.
- Quand tu n'as pas le droit, le système **refuse** : c'est le noyau qui contrôle l'accès à chaque fois.

I. Utilisateurs, groupes et types de droits

Qui ?	Symbole	Signification
propriétaire	<code>u</code> (<i>user</i>)	L'utilisateur à qui appartient le fichier, souvent celui qui l'a créé.
groupe	<code>g</code> (<i>group</i>)	Les utilisateurs du groupe auquel le fichier est rattaché.
autres	<code>o</code> (<i>others</i>)	Tous les autres utilisateurs de la machine.

Pour chacune de ces trois catégories, trois droits peuvent être accordés :

Droit	Sur un fichier	Sur un répertoire
<code>r</code> (<i>read</i>)	lire son contenu (par exemple avec <code>cat</code>)	lister son contenu (par exemple avec <code>ls</code>)

Droit	Sur un fichier	Sur un répertoire
w (<i>write</i>)	modifier son contenu	créer, renommer ou supprimer des fichiers à l'intérieur
x (<i>execute</i>)	l'exécuter comme un programme ou un script	le traverser : y entrer avec <code>cd</code> et accéder à ce qu'il contient

Remarque : l'administrateur du système (**root**) n'est pas soumis à ces restrictions. C'est pour cela qu'on n'utilise ce compte que lorsque c'est nécessaire.

II. Lire les droits avec `ls -l`

La commande `ls -l` affiche les informations détaillées de chaque fichier. Voici un exemple :

```
-rw-r--r-- 1 prof profs 0 févr. 24 16:00 fichier1.txt
```

Partie	Signification
<code>-rw-r--r--</code>	le type de l'élément, puis les droits (détaillés ci-dessous)
<code>1</code>	un nombre technique (nombre de liens), que nous ignorons
<code>prof</code>	le nom du propriétaire
<code>profs</code>	le nom du groupe
<code>0</code>	la taille, en octets
<code>févr. 24 16:00</code>	la date et l'heure de dernière modification
<code>fichier1.txt</code>	le nom du fichier

Les dix premiers caractères se lisent en suivant l'ordre : **1** caractère pour le type, puis **3 + 3 + 3** caractères pour les droits du propriétaire, du groupe et des autres. Un tiret `-` signifie « droit absent ».

Exemple	Type	Propriétaire (u)	Groupe (g)	Autres (o)
<code>-rw-r--r--</code>	<code>-</code> : fichier	<code>rw-</code> : lecture, écriture	<code>r--</code> : lecture	<code>r--</code> : lecture
<code>drwxr-xr-x</code>	<code>d</code> : répertoire	<code>rw-</code> : tous les droits	<code>r-x</code> : lecture, traversée	<code>r-x</code> : lecture, traversée

Pour aller plus loin

Le premier caractère peut prendre d'autres valeurs. Par exemple `c` désigne un fichier spécial de type périphérique : tape `ls -l $(tty)` et regarde le résultat. Pourquoi as-tu pu écrire dans ce fichier avec une redirection dans le TP « tout est fichier » ? Regarde son propriétaire et ses droits.

III. Modifier les droits avec `chmod`

La commande `chmod` (*CHange MODE*) modifie les droits d'un fichier ou d'un répertoire. Seul son **propriétaire** (ou l'administrateur) peut le faire. On écrit :

```
chmod qui opération droits chemin
```

Élément	Valeurs possibles
qui	<code>u</code> propriétaire, <code>g</code> groupe, <code>o</code> autres, <code>a</code> tous
opération	<code>+</code> ajouter un droit, <code>-</code> retirer un droit, <code>=</code> fixer exactement les droits
droits	<code>r</code> , <code>w</code> , <code>x</code> (ou plusieurs à la suite, comme <code>rw</code>)

Exemple	Effet
<code>chmod u+x fichier.txt</code>	ajoute le droit d'exécution pour le propriétaire
<code>chmod g-w fichier.txt</code>	retire le droit d'écriture au groupe
<code>chmod a+r fichier.txt</code>	ajoute le droit de lecture pour tout le monde
<code>chmod go-rwx fichier.txt</code>	retire tous les droits au groupe et aux autres : le fichier devient privé
<code>chmod u=rw fichier.txt</code>	donne au propriétaire exactement la lecture et l'écriture

Remarque : la commande `chown` permet de changer le propriétaire d'un fichier, mais elle est réservée à l'administrateur.



Activité 2 — modifier les droits d'un fichier

Placez-vous dans TP_OS.

1. Créez un fichier avec `echo 'bonjour' > essai.txt`, puis tapez `ls -l essai.txt`. Notez ses droits (ils peuvent varier d'un poste à l'autre).
2. Tapez `chmod u-w essai.txt`, puis `ls -l essai.txt`. Essayez ensuite d'ajouter du texte avec `echo 'autre' >> essai.txt`. Pouvez-vous encore **lire** le fichier avec `cat` ? Rétablissez le droit avec `chmod u+w essai.txt`.
3. Retirez le droit de lecture au propriétaire, puis essayez `cat essai.txt`. Que se passe-t-il ? Rétablissez le droit.
4. Tapez `chmod go-rwx essai.txt`, puis `ls -l essai.txt`. Quels sont maintenant les droits ? Qui peut encore lire le fichier ?



Activité 3 — les droits sur un répertoire

1. Tapez `mkdir test_droits`, puis `echo 'contenu' > test_droits/fichier.txt`.
2. Retirez le droit `x` au propriétaire : `chmod u-x test_droits`. Vérifiez avec `ls -ld test_droits`.
3. Essayez `cd test_droits`, puis `cat test_droits/fichier.txt`. Que se passe-t-il ? Rétablissez ensuite le droit avec `chmod u+x test_droits`.
4. Retirez maintenant le droit `r` : `chmod u-r test_droits`. Essayez `ls test_droits`, puis `cat test_droits/fichier.txt`. Que remarquez-vous ? Quelle est donc la différence entre `r` et `x` sur un répertoire ?
5. Rétablissez tous les droits avec `chmod u+rwx test_droits`, puis supprimez le répertoire et son contenu avec `rm -r test_droits`.

 Activité 4 — le droit d'exécution d'un fichier

1. Tape `echo 'echo Bonjour' > salut.sh`. Ce fichier contient une commande, comme un tout petit programme.
2. Essaie de l'exécuter avec `./salut.sh`. Que se passe-t-il ? Regarde ses droits avec `ls -l salut.sh`.
3. Ajoute le droit d'exécution au propriétaire, puis essaie à nouveau `./salut.sh`. Que se passe-t-il ?

Le `./` signifie « dans le répertoire courant » : c'est un chemin relatif. Nous y reviendrons quand nous écrirons de vrais scripts.

IV. Une autre écriture : la notation octale

 Notation octale (non exigible)

On peut aussi écrire les droits avec des **chiffres**. Chaque droit vaut un nombre : $r = 4$, $w = 2$, $x = 1$. Pour chaque catégorie (propriétaire, groupe, autres), on additionne les droits accordés. On obtient ainsi **trois chiffres**, dans l'ordre u, g, o.

Droits	Calcul	Chiffre
<code>rwX</code>	$4 + 2 + 1$	7
<code>rw-</code>	$4 + 2$	6
<code>r-X</code>	$4 + 1$	5
<code>r--</code>	4	4
<code>---</code>	0	0

Commande	Droits obtenus	Usage courant
<code>chmod 644 fichier</code>	<code>rw-r--r--</code>	un fichier lisible par tous, modifiable par son propriétaire
<code>chmod 600 fichier</code>	<code>rw-----</code>	un fichier privé
<code>chmod 755 script.sh</code>	<code>rxrx-rx-rx</code>	un script ou un répertoire ouvert à tous en lecture
<code>chmod 750 dossier</code>	<code>rxrx-r-x---</code>	un répertoire réservé au propriétaire et à son groupe

Cette notation n'est pas exigible, mais c'est la plus utilisée par les informaticiens : elle est plus courte à taper et tu la rencontreras très souvent.

 Activité 5 — essaie la notation octale

1. Prédisez les droits que donnera `chmod 644 essai.txt`, puis vérifiez avec `ls -l essai.txt`.
2. Faites de même avec `chmod 600 essai.txt`, puis avec `chmod 755 essai.txt`.
3. Quelle commande en notation octale rend `essai.txt` lisible et modifiable par toi seul ?

À retenir

L'essentiel

- Chaque fichier a un **propriétaire**, un **groupe** et des droits distincts pour le propriétaire (u), le groupe (g) et les autres (o).
- Trois droits : lecture r, écriture w, exécution x. Sur un répertoire, x signifie « pouvoir le traverser ».
- `ls -l` affiche les droits : le type, puis trois groupes de trois caractères.
- `chmod` modifie les droits : `chmod u+x fichier`, `chmod go-rwx fichier`. En octal : `chmod 644 fichier`.

Exercices

Exercice 1 — analyse des droits

Pour chaque ligne de droits, indiquez le type (fichier ou répertoire), puis les droits du propriétaire, du groupe et des autres.

- a. `-rwxr-xr--` :
- b. `drwx-----` :
- c. `-rw-rw-r--` :
- d. `-r--r--r--` :

Exercice 2 — écrire des commandes `chmod` (notation symbolique)

Écrivez la commande qui permet de :

- a. donner le droit d'exécution au propriétaire de `script.sh` :
- b. retirer le droit d'écriture au groupe de `notes.txt` :
- c. rendre `secret.txt` illisible pour le groupe et les autres :
- d. donner le droit de lecture à tout le monde sur `annonce.txt` :

Exercice 3 — notation octale (non exigible)

- a. Écrivez en notation octale les droits `rw-r--r--` : et `rwxr-xr-x` :
- b. Écrivez en lettres les droits `600` : et `750` :

Exercice 4 — situation

Vous êtes propriétaire du répertoire `partage`. Vous voulez que les membres de votre groupe puissent **lister son contenu et y entrer, mais pas le modifier**, et que les autres utilisateurs n'aient **aucun accès**.

- a. Quels doivent être les droits (écris les neuf caractères) ?
- b. Quelle commande permet de les obtenir ?