

R104 — Introduction aux systèmes d'exploitation et à leur fonctionnement TD 4 : La politique d'accès aux fichiers d'UNIX — 2h

d'après le TD de Jean-François Anne

Objectif(s) :

- Manipuler les droits sous UNIX via `chmod` et `umask`

1 Les messages

Exercice 1.1. L'envoi de message via le terminal

1. Ouvrez deux fenêtres si vous travaillez en mode graphique, ou deux consoles virtuelles (ALT + Fx) sinon ; vous disposez alors de deux terminaux.
2. Avec la commande `tty` récupérez le nom de ces terminaux. Localisez les fichiers correspondant à ces terminaux. Quels sont leurs droits ?
3. Aidez-vous du manuel pour comprendre ce que fait la commande `mesg`. Utilisez `mesg` comme indiqué dans le `man` pour autoriser l'écriture de messages sur votre terminal courant. Regardez à nouveau les droits du fichier spécial correspondant à ce terminal. Que constatez-vous ?
4. Aidez-vous du manuel pour comprendre ce que fait la commande `write` et comment elle fonctionne. Utilisez `write` pour envoyer un message sur un autre terminal.

Remarque

À l'IUT, et selon la config. de votre Linux, cela risque de ne pas marcher^a. Il s'agit d'un problème de droit vis-à-vis de la commande `write`. Si vous en avez les droits, vous pouvez éventuellement utiliser `sudo write` pour outrepasser vos droits. Sur votre système, vous pouvez essayer de corriger le problème. Un `sudo chmod g+s /usr/bin/write` devrait potentiellement faire l'affaire. Plus d'info ici : <https://www.linuxaddict.fr/index.php/2018/06/04/les-droits-speciaux-sous-gnu-linux-setuid-setgid-sticky-bit-et-umask/>

a. <https://bugs.launchpad.net/ubuntu/+source/util-linux/+bug/2064685>

5. Que se passe-t-il si le message que l'on envoie avec `write` correspond à une commande (par exemple `ls`). Proposez une réponse avant même de faire l'expérience sur machine. Après avoir fait cette expérience, êtes-vous surpris par le résultat ? Expliquer le résultat.
6. Utiliser le `man` de la commande `mesg` pour trouver comment un terminal peut interdire l'écriture sur lui-même. Que se passe-t-il pour les tentatives d'écriture depuis le terminal de départ, si le terminal cible a interdit l'écriture sur lui-même ? Proposer une réponse avant même de faire l'expérience sur machine. Après avoir fait cette expérience, êtes-vous surpris par le résultat ? Expliquer le résultat.

2 Les droits

Exercice 2.1. Les droits

1. Dans votre répertoire courant : créez un répertoire courant `essai_droit`.
2. Quels sont les droits par défaut de ce répertoire nouvellement crée ?
3. Pour chacun des cas suivant, donnez les commandes (en notation symbolique et en base 8) pour lui donner les droits suivant (on suppose qu'après chaque commande on remet le répertoire à 755 :

	propriétaire			groupe			les autres		
	lecture	écriture	accès	lecture	écriture	accès	lecture	écriture	accès
commande 1	oui	oui	oui	oui	non	oui	non	non	oui
commande 2	oui	non	oui	non	oui	non	non	non	oui
commande 3	non	oui	non	non	non	oui	oui	non	non
commande 4	non	non	oui	oui	non	oui	non	non	non

Exercice 2.2. Les droits

1. Créez un fichier `droit` dans le répertoire `essai_droit`. Quels sont ses droits par défaut ?
2. En partant du répertoire courant, pour chaque commande de l'exercice précédent, essayez d'accéder au répertoire `essai_droit`, de faire un `ls` dans `essai_droit` et de modifier le fichier `droit` avec un éditeur quelconque (`vi` par exemple... :))).

Exercice 2.3. Permission par défaut

1. Utiliser le man ou le cours pour déterminer à quoi sert la commande `umask`
2. Changer le masque de manière à ce qu'à leur création, les fichiers aient par défaut les droit 640 (`rw-r-----`), et les répertoires 750 (`rwxr-x---`).

Exercice 2.4.

1. Créer un petit fichier texte (de contenu quelconque), qui soit lisible par tout le monde, mais non modifiable (même pas par vous).
2. Créer un répertoire nommé `Secret`, dont le contenu est visible uniquement par vous même. Les fichiers placés dans ce répertoire sont-ils lisibles par d'autres membres de votre groupe ?
3. Créer un répertoire nommé `Connaisseurs` tel que les autres utilisateurs ne puissent pas lister son contenu mais puissent lire les fichiers qui y sont placés.

Exercice 2.5. Soit le fichier suivant : `-rw-rw-r-- 1 c1 cours884 Jul 7 Fich1`

1. Quelles commandes permettent à l'utilisateur `c1` de modifier les droits d'accès sur le fichier `Fich1` pour avoir : `-rwxr-xr-x 1 c1 cours884 Jul 7 Fich1`.
2. Donnez une solution numérique et une solution avec les actions (cf. le cours).

Exercice 2.6.

Les utilisateurs `c1` et `c2` appartiennent au groupe `cours`. L'utilisateur `guest` appartient au groupe `extern`. Le fichier `/tmp/x` existe et est accessible en lecture pour tout le monde. On se trouve dans le répertoire `d1`.

1. Pour cette question, utilisez les raisons évoquées dans la partie « rappel »(ci-après) pour justifier vos réponses. Dans chacune des trois configurations suivantes :

- (a)

```
1 drwxr-xr-- 2 c1 cours 1152 Jul 7 14:49 d1
2 -rw-rw-r-- 1 c1 cours 884 Jul 7 14:52 d1/f1
```
- (b)

```
1 drwxrwxr-x 2 c1 cours 1152 Jul 7 14:49 d1
2 -rw-r--r-- 1 c1 cours 884 Jul 7 14:52 d1/f1
```
- (c)

```
1 drwxrwxr-x 2 c1 cours 1152 Jul 7 14:49 d1
2 ----- 1 c1 cours 884 Jul 7 14:52 d1/f1
```

les utilisateurs `c1`, `c2` et `guest` peuvent-ils exécuter les commandes suivantes :

- ```
1 $ cp /tmp/x f1
2 $ cp /tmp/x f2
3 $ rm f1
```

Justifier vos réponses en utilisant les info de la section « Rappel »ci-après.

### Rappels

- La commande `cp file1 file2` crée le fichier `file2` si celui-ci n'existe pas, sinon écrase les données existantes de `file2` par celles de `file1`. Dans le premier cas, il s'agit d'une création de fichier, dans le deuxième cas, il s'agit d'une écriture.
- (a) Pour accéder à un fichier, il faut avoir le droit de passage dans chacun des répertoires qui constituent son chemin (le droit de passage dans un répertoire est le droit d'exécution `x`).
- (b) Pour écrire dans un fichier, il faut avoir l'autorisation d'écriture (`w`) sur ce fichier.
- (c) Pour modifier les droits d'un fichier, il faut en être le propriétaire.
- (d) Pour créer ou détruire un fichier, il faut avoir le droit d'écriture dans le répertoire contenant ce fichier, puisqu'il s'agit d'ajouter ou de supprimer un lien dans un répertoire.

## Exercice 2.7.

1. Quelle est la commande pour créer un dossier ?
2. A partir du répertoire courant, créez un sous-répertoire `rep1`. Après avoir lancé la commande `umask` et observé les droits du répertoire `rep1`, puis lancé la commande `umask 0` et créé un nouveau répertoire, expliquez à quoi sert la commande `umask`. Comment faire pour que ce répertoire ait automatiquement (dès sa création) les droits `rw-rw-r--` ?
3. Créer un fichier `fich1` et regarder la valeur de ses droits. Comparer cette valeur avec celle du répertoire `rep1`. Que peut-on en conclure sur les valeurs par défaut des droits d'un fichier et d'un répertoire ? Sont-ils égaux ?

## Exercice 2.8.

1. Quelle commande permet de tester l'existence d'un fichier ou d'un répertoire ?
2. Comment récupérer le résultat (échec ou succès) ?

## Exercice 2.9.

Aller faire les exos qui se trouvent ici : <http://iamjmm.ovh/NSI/permissions/site/exercice.html>