

Le stockage des fichiers

Bien qu'un peu moins courante, la commande `ln` vous sera certainement utile un jour ou l'autre. Elle permet de créer des liens entre des fichiers, c'est-à-dire (pour employer des mots que vous connaissez) qu'elle permet de **créer des raccourcis**.

Ces « raccourcis », qu'on appelle des **liens** sous Linux, sont un peu plus complexes que ceux que vous avez l'habitude de voir sous Windows. En effet, on peut créer deux types de liens :

- des liens **physiques** ;
- des liens **symboliques**.

Ces deux types ne fonctionnent pas de la même manière. Pour comprendre ce qui les différencie, il faut savoir comment un OS tel que Linux gère les fichiers sur le disque dur.

Allons, allons, ne faites pas cette tête-là, un peu de théorie sur le fonctionnement des OS, c'est toujours très intéressant ! :-)

Sur le disque dur, chaque fichier est grosso-modo séparé en deux parties :

- son nom ;
- son contenu.

Vous avez bien lu : la liste des noms de fichiers est stockée à un autre endroit que leur contenu. Cette séparation aide Linux à s'organiser.

Je simplifie ici volontairement les choses. En pratique, c'est (toujours) un peu plus compliqué. Il y en fait trois parties : le nom, les informations de gestion (droits d'accès) et le contenu. Mais nous allons faire simple car notre but est juste de comprendre l'idée générale du fonctionnement.

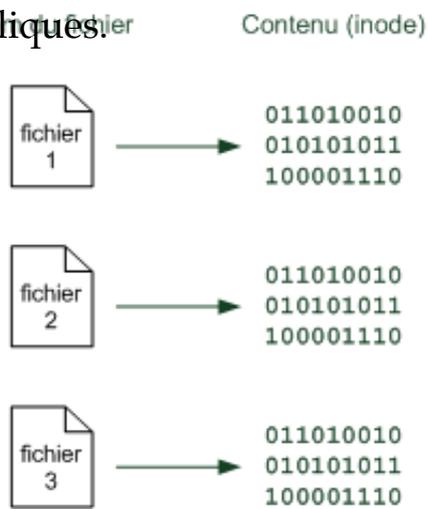
Chaque contenu de fichier se voit attribuer un numéro d'identification appelé *inode* (figure suivante). Chaque nom de fichier est donc associé à un inode (son contenu).

Fichiers associés à des inodes

C'est tout ce que vous avez besoin de savoir pour comprendre la suite.

Nous allons maintenant découvrir comment créer des liens physiques puis des liens

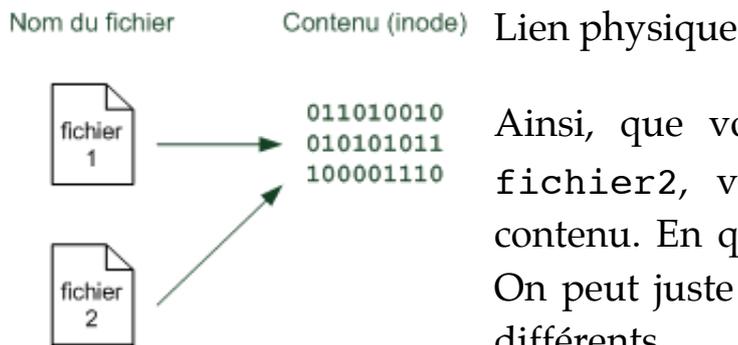
symboliques.



Créer des liens physiques

Ce type de lien est plus rarement utilisé que le lien symbolique, mais il faut tout de même le connaître car il peut se révéler pratique.

Un lien physique permet d'avoir deux noms de fichiers qui partagent exactement le même contenu, c'est-à-dire le même inode (figure suivante).



On ne peut pas créer de liens physiques sur des répertoires. Cela ne fonctionne qu'avec les fichiers.

Il existe des options pour que ça fonctionne avec des répertoires, mais c'est un peu particulier et on n'en parlera pas. Pour faire un « raccourci » vers un répertoire, on préférera utiliser un lien symbolique.

Pour créer un lien physique, nous allons utiliser la commande `ln`. Je vous propose tout d'abord de créer un répertoire pour nos tests :

```
mkdir tests  
cd tests
```

Une fois dans ce dossier, créez un fichier avec la commande `touch` par exemple :

```
touch fichier1
```

Nous voulons maintenant créer un lien physique : nous allons créer un `fichier2` qui partagera le même inode (le même contenu) que `fichier1`. Tapez :

```
ln fichier1 fichier2
```

Si vous listez les fichiers du répertoire, vous avez l'impression d'avoir deux fichiers

différents :

```
mateo21@mateo21-desktop:~/tests$ ls -l
total 0
-rw-r--r-- 2 mateo21 mateo21 0 2008-07-31 13:55 fichier1
-rw-r--r-- 2 mateo21 mateo21 0 2008-07-31 13:55 fichier2
```

A priori, rien ne nous permet ici de deviner que ces fichiers modifient le même contenu. Le lien physique est donc un lien dur, pas évident à détecter au premier coup d'œil.

La seconde colonne de la liste (qui indique « 2 » pour chacun des fichiers) correspond au nombre de fichiers qui partagent le même inode. C'est le seul indice qui vous permet de savoir que quelqu'un a fait un lien physique, mais vous ne pouvez pas savoir lequel. Le seul moyen de vérifier que ces fichiers partagent le même contenu, c'est de faire `ls -li` pour afficher les numéros d'inode correspondants et de vérifier que ces deux fichiers sont associés au même inode.

En temps normal, sur la plupart des fichiers la seconde colonne indique donc « 1 ». Si c'est un dossier, ce nombre indique en revanche le nombre de fichiers à l'intérieur.

Si vous supprimez un des deux fichiers, l'autre fichier reste en place et le contenu sera toujours présent sur le disque. L'inode est supprimé uniquement quand plus aucun nom de fichier ne pointe dessus.

En clair, supprimez `fichier1` pour voir. Vous verrez que `fichier2` existe toujours et qu'il affiche toujours le même contenu. Il faut supprimer `fichier1` ET `fichier2` pour supprimer le contenu.

Créer des liens symboliques

Les liens symboliques ressemblent plus aux « raccourcis » dont vous avez peut-être l'habitude sous Windows. La plupart du temps, on crée des liens symboliques sous Linux pour faire un raccourci, et non des liens physiques qui sont un peu particuliers.

Le principe du lien symbolique est que l'on crée un lien vers un autre nom de fichier. Cette fois, on pointe vers le nom de fichier et non vers l'inode directement (figure suivante).

Lien symbolique

Supprimez le `fichier2` que nous avons créé tout à l'heure (sous forme de lien physique) :

Nom du fichier Contenu (inode) `rm fichier2`



011010010
010101011
100001110



Créons maintenant un nouveau `fichier2`, cette fois sous forme de lien symbolique. On utilise là encore la commande `ln`, mais avec le paramètre `-s` (s comme symbolique) :

```
ln -s fichier1 fichier2
```

Cette fois, la commande détaillée `ls -l` sera

beaucoup plus précise :

```
mateo21@mateo21-desktop:~/tests$ ls -l
total 0
-rw-r--r-- 1 mateo21 mateo21 0 2008-07-31 13:55 fichier1
lrwxrwxrwx 1 mateo21 mateo21 8 2008-07-31 14:15 fichier2 -
> fichier1
```

On note deux choses :

- la toute première lettre de la seconde ligne est un `l` (comme *link*, c'est-à-dire lien) ;
- tout à la fin de la seconde ligne, une flèche montre clairement que `fichier2` pointe vers `fichier1`.

Bref, les liens symboliques sont beaucoup plus faciles à repérer que les liens physiques !

Ok, mais quelles différences à part ça ? Le résultat revient au même, non ? Qu'on ouvre `fichier1` ou `fichier2`, on éditera le même contenu au final !

Tout à fait. Il y a quand même quelques subtilités :

- par exemple, si vous supprimez `fichier2`, il ne se passe rien de mal. Par contre, si vous supprimez `fichier1`, `fichier2` pointera vers un fichier qui n'existe plus. Le lien symbolique sera cassé et ne servira donc plus à rien. On parle de « lien mort » ;
- d'autre part, l'avantage des liens symboliques est qu'ils fonctionnent aussi sur des répertoires, contrairement aux liens physiques.

En résumé

- `cat` permet d'afficher tout le contenu d'un fichier, mais lorsque celui-ci est long, il est préférable d'utiliser `less` qui affiche le fichier page par page.
- On peut obtenir uniquement le début ou la fin d'un fichier avec `head` et `tail`. En utilisant `tail -f` on peut suivre l'évolution d'un fichier en temps réel, ce qui est utile sur les fichiers de log qui enregistrent l'activité du système.
- `mkdir` permet de créer un dossier, `touch` permet de créer un fichier vide.
- `cp` permet de copier un fichier ou un dossier, tandis que `mv` permet de les déplacer ou de les renommer.
- `rm` supprime un fichier. Il n'y a pas de corbeille en console, la suppression est définitive ; il faut donc être prudent.
- On peut créer des liens (raccourcis) vers des fichiers et dossiers à l'aide de la commande `ln`.