

## **cp : copier un fichier**

Parmi les opérations de base que l'on veut pouvoir faire avec les fichiers, il y a la copie et le déplacement de fichier. C'est un peu le genre de chose que l'on fait tous les jours, il est donc important de savoir s'en servir.

La commande `cp` (abréviation de *CoPy*, « copier » en anglais) vous permet comme son nom l'indique de copier un fichier... mais aussi de copier plusieurs fichiers à la fois, et même de copier des dossiers !

Si on essayait de copier le fichier `fichierbidon` qu'on a créé tout à l'heure ?

Ça fonctionne comme ceci :

```
cp fichierbidon fichiercopie
```

Le premier paramètre est le nom du fichier à copier, le second le nom de la copie du fichier à créer.

En faisant cela, on aura donc deux fichiers identiques dans le même répertoire : `fichierbidon` et `fichiercopie`.

N'oubliez pas d'utiliser l'autocomplétion avec la touche `Tab` ! Lorsque vous avez écrit `cp fic`, tapez `Tab`, et `fichierbidon` devrait se compléter tout seul !

### **Copier dans un autre dossier**

On n'est pas obligé de copier le fichier dans le même dossier, bien sûr. On peut très bien utiliser le système de répertoires relatifs et absolus qu'on a vu dans le chapitre précédent.

Par exemple, si je veux copier `fichierbidon` dans le sous-dossier `mondossier` que j'ai créé tout à l'heure :

```
cp fichierbidon mondossier/
```

Le fichier `fichierbidon` sera copié dans `mondossier` sous le même nom.

Notez que mettre le `/` à la fin n'est pas obligatoire. Si vous le voyez là, c'est parce que l'autocomplétion me l'a automatiquement ajouté lorsque j'ai appuyé sur `Tab`. Ehhh oui, je suis tellement flemmard que je n'écris même pas `mondossier` en entier, j'écris juste `mon` suivi de `Tab`, et hop c'est écrit en entier ! Ça va beaucoup plus vite lorsqu'on prend ce réflexe.

Si vous voulez copier `fichierbidon` dans `mondossier` sous un autre nom, faites comme ceci :

```
cp fichierbidon mondossier/fichiercopie
```

Avec cette commande, on aura créé une copie de `fichierbidon` dans `mondossier` sous le nom `fichiercopie` !

Enfin là, j'utilise des répertoires relatifs, mais je peux aussi écrire un répertoire en absolu :

```
cp fichierbidon /var/log/
```

... copiera `fichierbidon` dans le dossier `/var/log`.

## Copier des dossiers

Avec l'option `-R` (un « R » majuscule !), vous pouvez copier un dossier, ainsi que tous les sous-dossiers et fichiers qu'il contient !

Tout à l'heure, on a créé un dossier `animaux` qui contenait un autre dossier `vertebres`, qui lui-même contenait le dossier `chat`. Si vous tapez cette commande :

```
cp -R animaux autresanimaux
```

... cela aura pour effet de copier `animaux` ainsi que tous ses sous-dossiers sous le nom `autresanimaux`.

Faites des `ls` après pour vérifier que les sous-dossiers sont bien là et que je ne vous mène pas en bateau !

## Utiliser le joker \*

Le symbole `*` est appelé *joker*, ou encore *wildcard* en anglais sous Linux.

Il vous permet de copier par exemple tous les fichiers image `.jpg` dans un sous-dossier :

```
cp *.jpg mondossier/
```

Vous pouvez aussi vous en servir pour copier tous les fichiers dont le nom commence par « `so` » :

```
cp so* mondossier/
```

Le joker est un atout très puissant, n'hésitez pas à l'utiliser !

C'est avec des outils comme le joker que la console deviendra pour vous progressivement plus puissante que l'explorateur de fichiers que vous manipulez à la souris.

## **mv : déplacer un fichier**

Très proche de `cp`, la commande `mv` (*MoVe*, « déplacer » en anglais) a en fait deux utilités :

- déplacer un fichier (ou un dossier) ;
- renommer un fichier (ou un dossier).

Vous allez comprendre pourquoi.

### **Déplacer un fichier**

La commande `mv` s'utilise pratiquement comme `cp` :

```
mv fichierbidon mondossier/
```

Au lieu de copier `fichierbidon` dans `mondossier` comme on l'a fait tout à l'heure, ici on a juste déplacé le fichier. Il n'existe plus dans son dossier d'origine.

Vous pouvez déplacer des dossiers aussi simplement :

```
mv animaux/ mondossier/
```

... déplacera le dossier `animaux` (et tous ses sous-dossiers) dans `mondossier`.

Vous pouvez aussi utiliser les jokers :

```
mv *.jpg mondossier/
```

### **Renommer un fichier**

La commande `mv` permet de faire quelque chose d'assez étonnant : renommer un fichier. En effet, il n'existe pas de commande spéciale pour renommer un fichier en console sous Linux, c'est la commande `mv` qui est utilisée pour ça.

Par exemple :

```
mv fichierbidon superfichier
```

... renommera fichierbidon en superfichier. Après cette commande, fichierbidon n'existe plus, il a été renommé.

## Déplacer et renommer un fichier à la fois

Vous pouvez aussi déplacer fichierbidon dans mondossier tout en lui affectant un nouveau nom :

```
mv fichierbidon mondossier/superfichier
```

Et voilà le travail !

Je vous conseille **fortement** de vous entraîner à utiliser cp et mv dans tous les sens : avec ou sans joker, en déplaçant, renommant des dossiers, en déplaçant / renommant à la fois, en utilisant des chemins relatifs et absolus, etc.

C'est assez intuitif normalement, mais il faut pratiquer et pas seulement se contenter de lire ce que j'écris pour que ça rentre.

N'oubliez pas d'utiliser l'autocomplétion de fichiers et dossiers avec la touche Tab ; si vous ne le faites pas dès maintenant, vous perdrez du temps et vous trouverez la console nulle alors que vous devriez la trouver géniale.

Autre chose : le symbole .. signifie « dossier précédent », et . signifie « dossier dans lequel je me trouve ». Vous pourriez en avoir besoin lorsque vous copiez ou déplacez un fichier.

Si vous avez la tête qui tourne à force de copier et déplacer des fichiers dans des dossiers, c'est normal. Ça commence à devenir un beau bazar dans vos dossiers d'ailleurs, non ?

Il est temps de faire un peu de ménage avec la commande permettant de **supprimer** : rm !