

# La recherche de fichiers sur Linux

## Which et whereis

La commande **which** permet de **localiser une commande** de votre système. **whereis** permet de localiser les exécutables, sources et pages de manuel de la commande.

Exemple :

- localiser les commandes ls et useradd  
*\$ which ls useradd*
- localiser les executables, sources et manuels de bash  
*\$ whereis bash*

### Exercice d'application

Commentez cette suite de commande en trouvant d'abord la signification des options utilisées pour whereis dans les pages de manuel puis le résultat obtenu.

```
$ cd /usr/bin
```

```
$ whereis -u -M /usr/man/man1 -S /usr/src -f *
```

## Locate

La commande **locate** permet de localiser très rapidement n'importe quel fichier présent sur votre système de fichier car elle s'appuie sur une base de données des fichiers mise à jour régulièrement par le système (commande updatedb à travers une tâche cron).

Syntaxe : **locate** nom\_fichier

Exemple :

- chercher le fichier test.pas  
*locate test.pas*

## find

La commande **find** est l'outil par excellence qui permet de chercher des fichiers au sein de l'arborescence (est donc plus lente que locate), et éventuellement d'exécuter une action dessus.

Syntaxe : **find** répertoire -name nom\_du\_fichier

Quelques exemples utiles :

- Cherchez dans toute l'arborescence les fichiers
  - Dont les noms ne contiennent pas de chiffre.  
*find / -name "[!0-9]" -print*

- dont le nom se termine par **.c**, redirigez les erreurs vers le fichier poubelle **/dev/null**  
`$ find / -name "*.c" -print 2>/dev/null`
- Commençant par **X** ou **x**.  
`find / -name "[Xx]*" -print 2>/dev/null`

Option	Signification		Option	Signification
<b>-name</b>	Recherche par <b>nom</b> de fichier.		<b>-atime</b>	Recherche par date de <b>dernier accès</b> .
<b>-type</b>	Recherche par <b>type</b> de fichier.		<b>-mtime</b>	Recherche par date de <b>dernière modification</b> .
<b>-user</b>	Recherche par <b>propriétaire</b> .		<b>-ctime</b>	Recherche par date de <b>création</b> .
<b>-group</b>	Recherche par appartenance à un <b>groupe</b> .		<b>-perm</b>	Recherche par <b>autorisations d'accès</b> .
<b>-size</b>	Recherche par <b>taille</b> de fichier.		<b>-links</b>	Recherche par <b>nombre de liens</b> au fichier.

Il est possible d'appliquer une action au résultat de la recherche :

**-delete** pour supprimer les fichiers trouvés

**-exec** pour appliquer une commande, Exemple :

Modifions l'extension de tous les fichiers .php en extension .php3 :

```
find -name "*.php" -exec mv {} {}3 \;
```

La commande n'affiche rien s'il n'y a pas eu d'erreur.

Pour chaque fichier .php trouvé, on exécute la commande qui suit **-exec** :

- cette commande ne doit PAS être entre guillemets ;
- les accolades {} seront remplacées par le nom du fichier ;
- la commande doit finir par un \ ; obligatoirement.

### **Exercice d'application**

Commentez les commandes suivantes :

```
$ find /dev -type b
```

```
$ find /dev -type c | wc -l
```

```
$ find /usr/include/ -name [abc]*
```

```
$ find -size +1000 -user toure -print 2>/dev/null
```

```
$ find /usr/include/ \( -name 'a*' -a -name '*b*' \) -o -name 'c*' > $HOME/abc.list
```