

## Fiche de TP3

### Exercice 1 : Allocation/Libération mémoire

Les fichiers `/proc/meminfo` et `/proc/vmstat` contiennent des données importantes permettant de connaître l'état de la mémoire et des variables associées.

1. Analysez les valeurs dans ces deux fichiers pour en déduire le nombre de demandes de pages et le nombre de défaut de page.
2. En croisant une donnée entre ces deux fichiers, essayez d'en déduire la taille d'une page.

### Exercice 2 : Système de fichiers

#### Rappels sur quelques commandes

Quelles sont les commandes pour :

- renommer un fichier,
- changer de répertoire,
- créer un répertoire,
- créer un fichier vide,
- effacer un fichier,
- effacer un répertoire,
- effacer un répertoire et son contenu,
- compter le nombre de ligne du fichier `/etc/passwd`,
- compter le nombre de fichiers (et répertoires) dans un répertoire (attention à l'utilisation du `*`),
- compter le nombre de fichiers mais sans inclure les répertoires.

### Exercice 3 : les commandes `df` (disk free), `du` (disk use) et `mount`

1. En utilisant la commande `mount`, déterminez combien y a-t-il de systèmes de fichiers montés sur l'ordinateur ?
2. Quel est le système de fichiers utilisé par votre répertoire de connexion ? Combien reste-t-il d'espace disque libre ?
3. Quelle option de `du` et `df` permet d'afficher les valeurs en multiples d'octets (Kio, Gio ...) ?
4. Quel espace occupe votre répertoire personnel ?

5. À l'aide des commandes `du` et `sort` (voir les pages `man`), affichez la liste des fichiers et répertoires situés dans `/etc`, classée suivant l'espace disque qu'ils occupent (en octet, du plus petit au plus grand).

#### Exercice 4 : les i-nodes

Sous Unix/Linux, pour la gestion du système de fichiers, chaque fichier est identifié par un numéro unique : l'*i-node*. À cet identifiant est associée plusieurs informations concernant le fichier qui permettent au système d'exploitation de bien gérer les actions liées au système de fichiers. Parmi ces informations on retrouve le type, la taille et le contenu du fichier.

1. Avec quelles options les commandes `ls` et `stat` permettent de voir ces informations (taille, *i-node*, droits etc) ?
2. Comment afficher à l'aide de la commande `ls` la liste des fichiers triée du plus ancien au plus récent, en affichant cette date, les droits, la taille etc ?

Sous Unix/Linux il existe la possibilité de créer des liens entre fichiers afin de permettre par exemple de créer plusieurs noms faisant référence au même *i-node*, et donc au même fichier. Pour cela il faut utiliser la commande `ln`. Il existe deux types de lien : les liens physiques (*hard link*) et les liens symboliques (*soft link*).

#### Exercice 4 : les liens physiques

1. Créez deux fichiers: le fichier *one* contenant la valeur 1 et le fichier *two* contenant la valeur 2.
2. Créez ensuite à l'aide de la commande `ln` le fichier *un* et le fichier *deux* en tant que lien physique vers respectivement *one* et *two*.
3. Comment apparaissent ces deux fichiers avec la commande "`ls -l`" ?
4. Modifiez le fichier *one* pour y mettre la valeur 3. Que contiennent les fichiers *one*, *two*, *un* et *deux* ? Que remarquez-vous ?
5. Quels sont les *i-nodes* de ces 4 fichiers ?
6. Que fait donc la commande `ln` en création de lien physique ?
7. Comparez l'*i-node* du dossier courant et l'*i-node* du dossier `"."` situé dans le dossier courant ? À quel répertoire correspond l'*i-node* de `".."` ? Expliquez.

#### Exercice 4 : les liens physiques

Les fichiers de lien symbolique contiennent une référence à un autre fichier et leur contenu peut être vu comme "simplement" un chemin (absolu ou relatif) vers l'autre fichier. Comme pour les liens physiques, toutes les commandes effectuées sur un

fichier lien sont en fait effectuées sur le fichier référencé. Mais contrairement aux liens physiques, il y a bien deux fichiers différents qui sont créés car ils ont deux *i-nodes* différents. On dit simplement que le fichier lien pointe vers un autre fichier.

Pour créer un lien symbolique il suffit d'ajouter l'option `-s` à la commande `ln`. En effet, par défaut la commande `ln` crée un lien physique. La commande "`ln -s -l`" permet d'identifier les liens non seulement par leur type mais indique aussi le fichier référencé. De plus il est possible de créer des liens vers des répertoires car tout répertoire est en fait un fichier.

- Créez un répertoire *FR* et un répertoire *EN*.
- Notez les *i-nodes* des fichiers *un*, *deux*, *one* et *two*
- Placez dans *FR* les fichiers *un* et *deux* précédents et dans le répertoire *EN* les fichiers *one* et *two* précédents.

1. Leurs *i-nodes* ont-ils changé ?

- Créez dans le répertoire *FR* le fichier *trois* contenant la valeur 3.
- Créez dans le répertoire *EN* le fichier *three* en tant que **lien symbolique** vers le fichier *trois* du répertoire *FR*.

1. Quels sont les *i-nodes* de ces deux fichiers ? Qu'affiche la commande `ls -l` dans le répertoire *EN* ?
2. Créez au même niveau que les répertoires *EN* et *FR* un nouveau répertoire *US* comme étant un lien symbolique de *EN*. Quel est l'*i-node* de *US* ? Le répertoire *US* contient-il des fichiers et si oui à quoi correspondent leur *i-nodes* ?
3. Modifiez la valeur du contenu de *US/three* à la valeur 3.0. Quel est le contenu des fichiers *FR/trois* et *EN/three* ?
4. Déplacez le fichier *US/three* dans le répertoire `/tmp/`. Quel est son *i-node* et quel est son contenu ?
5. Remettez le fichier `/tmp/three` à son emplacement original (répertoire *EN*).
6. Ouvrez dans un éditeur le fichier *US/three* et **laissez le ouvert** (par exemple *vim*, *emacs*). **En parallèle**, après avoir fait un `ls`, supprimez le fichier *FR/trois* et faites de nouveau `ls`. Modifiez dans l'éditeur la valeur par 3 et sauvez le fichier (en gardant l'éditeur ouvert). Refaites `ls`. Que remarquez-vous ?
7. Comment peut-on connaître le nombre de noms/liens que possède un fichier ?