

Présentation du système d'exploitation UNIX

1 Introduction

Un système d'exploitation est un ensemble de programmes servant d'interface entre le matériel et l'utilisateur.

UNIX est un système d'exploitation multi-tâche, multi-utilisateur.(fenêtrage...)

UNIX permet l'utilisation en réseau de station de travail.

Le **shell** est un interpréteur de commandes de UNIX.

NFS Network Files System : installation de stations "serveurs" et de stations "clientes", visualisation de répertoires situés sur différents disques durs de différentes stations en une seule arborescence.

NIS Network Information Service: données communes entre différents serveurs du réseau.

2 Session de travail

login: début d'une session de travail

ex: **canard login : nom-de-login**

L'accès à une station donne la possibilité de travailler sur un réseau international, ceci impose un verrou qui est le mot de passe **non trivial**. password (NIS)

ex: **password: (entrée en aveugle du mot de passe)**

Changement du mot de passe avec la commande **passwd**. Un mot de passe est composé de 8 caractères (par exemple un mélange de 4 minuscules 2 majuscules et 2 chiffres)

Vu le danger que représente une station en libre accès **ne pas oublier de sortir** soit avec la commande **exit** pour une connexion via un terminal soit en sélectionnant l'item correspondant à la fin de votre session dans le menu principal (en général en bas à gauche de l'écran) ,

3 Le système de fichiers

l'arborescence (NFS) fait que vous pouvez vous installer devant n'importe quelle machine sans voir de différences. Vos fichiers sont toujours stockés au même endroit dans le réseau.

- **pwd** affiche le chemin du répertoire courant.
- **cd** permet de changer de repertoire soit par chemin relatif (à partir du répertoire courant) soit par chemin absolu (à partir de la racine).

- **ls** permet de lire le contenu d'un répertoire.
- Quelques raccourcis: `~` (pour le home-directory), `~user` (pour le home-directory de user), `.` (le répertoire courant) , `..` (le répertoire père), `/` (la racine).
- les métacaractères : `?` (remplace un caractère), `*` (remplace 0 ou plusieurs caractères), `[-]` (un caractère de la liste), `{ }` (remplace une chaîne de caractères).

Exercice:

- a) Lire le répertoire courant suite au "login". Vérifier que c'est votre home-directory.
- b) Déplacez vous à la racine. Vérifiez. Lire le contenu de ce répertoire. Retournez dans votre home-directory. Lire le contenu du répertoire `/usr` sans vous déplacer.
- c)Afficher les fichiers et répertoires de `/usr/bin`
 - 1) qui se terminant par "t".
 - 2) contenant "st".

Autres commandes: **history**, **alias**.

4 Visualisation d'un fichier dans une fenêtre "terminal"

- **cat** permet de lister un fichier texte.
- **more** affiche page à page un fichier texte. Le mode d'emploi est donné en appuyant sur la touche **h**
- **man** donne le mode d'emploi de nombreuses commandes unix ainsi que de fonctions du langage C (le manuel est aussi disponible sous forme hypertexte sous l'utilitaire d'aide de la barre de menu)
- **whatis**, **whereis**, **which** ces commandes donnent des informations sur les utilitaires : fonction, lieu, laquelle...

Exercice:

- a) Afficher le contenu du fichier `/usr/include/math.h` avec la commande `cat`. (les combinaisons `CTR-S` et `CTR-Q` peuvent arrêter ou reprendre le défilement). Faire `"cat"` sans nom de fichier. Que remarquez-vous? Sortir avec `CTR-D`. Faire `"cat /etc/passwd /etc/group"`. Que remarquez-vous?
- b) Afficher le même fichier avec la commande `more`. Appuyer sur la touche "h" pour avoir la liste des commandes.
- c)Faire `"man ls"`. Appuyer sur la touche "h" pour avoir la liste des commandes.
- d) Faire `"whatis ls"`. Que remarquez-vous?

Autre commande: **grep**

5 le système de fichiers: création, suppression...

- **mkdir** permet de créer un répertoire
- **rmdir** permet de supprimer un répertoire vide
- **cp** pour copier un fichier
- **mv** pour déplacer un fichier
- **rm** pour détruire un fichier (voire des répertoires)
- **chmod** pour changer les droits d'accès et d'exécution

Toutes ces actions peuvent aussi être faites à partir d'un utilitaire comme `konqueror` sous `kde`.

Exercice:

- a) Retourner dans votre home-directory. Créer un répertoire "ESSAI".
- b) Copier les fichiers `/etc/passwd` et `/etc/group` dans le répertoire `ESSAI` sous des noms différent.
- c) Retourner dans votre home-directory. Créer un répertoire "COPIES".
- d) Déplacer un des fichiers de "ESSAI" dans "COPIES"
- e) Chercher à l'aide de "man" l'option de la commande "ls" visualisant les droits d'accès des fichier.
- f) Modifier les droits d'un fichier avec la commande "chmod" (utiliser "man").
- g) Détruire le contenu du répertoire "COPIES" avec la commande "rm". Détruire le répertoire "COPIES" avec la commande "rmdir".
- h) Essayer de détruire le répertoire "ESSAI" avec la commande "rmdir". Que remarquez-vous? Chercher à l'aide de "man" l'option de la commande "rm" permettant de détruire le répertoire "COPIES".

6 Editeurs de texte

`vi` est un editeur rudimentaire livré avec UNIX, de même on peut utiliser "sed" pour faire des modifications dans de gros fichiers. (faire "man")

Mais les éditeurs utilisés sont principalement **kedit**, **gedit** ou **emacs**.

Exercice :

- a) Retourner dans votre home-directory. Créer un répertoire "ESSAI".
- b) Editer un fichier à l'aide de l'éditeur de votre choix. Le sauver. Vérifier avec `ls` et `more` l'existence et le contenu.

7 Entrée et sorties standards

les commandes de redirection

- De l'entrée standard (le clavier): <
- De la sortie standard (le moniteur): > nouveau-fichier ou >> suite-ancien-fichier.
- De la sortie standard d'une commande vers l'entrée standard d'une autre : commande1 | commande2

Exercice:

- a) Rediriger la commande "ls /" dans un nouveau fichier.
- b) Rediriger la commande "cat /etc/passwd" dans le fichier précédent.
- c) Effectuer la commande "ls /usr/bin" puis "ls /usr/bin | more".
Que remarquez-vous?

8 Jobs et processus

Processus sont les tâches élémentaires gérées par le systèmes d'exploitation.

- Une tâche peut être lancée à partir d'une fenêtre "terminal". Dans ce cas la fenêtre est occupée, et la tâche peut être suspendue avec **CTR-Z** ou détruite avec **CTR-C**.
- Une tâche peut être lancée en tâche de fond (background) **tâche &** . Dans ce cas la fenêtre reste libre. De même une tâche suspendue peut être mise en fond avec la commande **bg**, ou en avant avec **fg**.
- La commande **jobs** permet de voir les tâches lancées à partir de cette fenêtre.
- La commande **ps** permet de voir tous les processus du système.
- La commande **kill** permet de détruire un processus.

Exercice:

- a) Lancer à partir d'une fenêtre "terminal" la commande "gedit nom-de-fichier &". Que remarquez-vous?
- b) A l'aide de la commande "ps" trouver quel est le PID de la commande que vous venez de lancer.
- c) Effectuer "kill -9 ce-PID". Que remarquez-vous?
- d) Lancer à partir d'une fenêtre "terminal" la commande "gedit nom-de-fichier". Positionnez le curseur de la souris dans la fenêtre "terminal" précédente. Faites CTR-Z. Faites les commandes "bg" et "fg" en alternance. Que remarquez-vous?