TP3-IUP1

Le SHELL et les FILTRES

(Unix externe)

January 12, 2009

1 Le shell

Les commandes importantes :

- `set`
- `echo $nom_de_variable`
- `export $nom_de_variable`
- `setenv`
- `csh`
- `sh`

Le shell est un interpréteur de commandes. Celui-ci possède des commandes dites *internes*: ce sont des commandes directement implémentées dans le shell. Par exemple, les commandes `cd`, `export ...` sont des commandes internes (pour vérifier qu'une commande est interne on peut utiliser la commande `whereis` qui ne peut indiquer la place de ces commandes dans l'arborescence du système, ou regarder dans la page `man` du shell). Les commandes du système qui ne sont pas internes peuvent être repérées par la commande `whereis`, ce sont des programmes indépendants du shell.

Il existe plusieurs types de shells. Principalement on distingue deux grande famille de shell :

- le bourne shell (commande `csh`)
- le C-shell (commande `sh` ou `bash`)

Le `bash` (**B**ourne **a**gain **s**hell) de Linux est un bourne-shell. (Les pages `info bash` vous donnerons une explication exhaustive du `bash`).

Le langage shell permet de définir des *variables* (comme la variable d'environnement `DISPLAY` vu au premier TP). Pour créer une nouvelle variable, on utilise l'opérateur `=`. Par exemple :

```
> a=toto
```

permet de créer une nouvelle variable de nom `a`, et portant la valeur `toto`. Pour connaître la valeur de cette variable, on utilise la commande `echo` et l'opérateur `$` :

```
> echo $a
toto
```

Pour connaître la liste des variables définis dans le shell on utilise la commande `set`. Les *variables d'environnement* sont des variables particulières gérées par le shell. Ce sont des variables dites *exportables*. Pour connaître la liste des variables d'environnement ¹ on utilise la commande `env`. Pour rendre une variable exportable, on utilise la commande `export`.

Exercice 1.1 *Exécutez la liste des commandes ci-dessous, et expliquez leur interprétation par le shell.*

```
>a=truc
>echo $a
>ssh
>echo $a
>exit
>echo $a
>export a
>ssh
>echo $a
>exit
>setenv b machin
>echo $b
>ssh
>echo $b
```

Exercice 1.2 *Regardez le contenu de la variable d'environnement `PATH`. Expliquez son rôle.*

2 Les filtres

Les filtres principaux :

- `cat`
- `wc [-lwc]`
- `head [-n]`
- `tail [±n]`
- `sort [-rnt] [+POS] [-POS]`
- `grep`
- `sed`

¹ie : uniquement la liste des variables exportables

Exercice 2.1

- A l'aide des pages `man` expliquez ce que fait la commande `wc`.
- Puis donnez une commande qui crée un fichier commençant par la date, suivie du nombre d'utilisateurs logés² sur la machine sur laquelle vous travaillez.

Exercice 2.2 *

- Expliquez ce que fait la commande `head`.
- A l'aide de la commande `find` donnez une commande qui permet d'afficher l'entête des fichiers dont l'extension est `“.c”`.

Exercice 2.3

- Expliquez ce que fait la commande `tail`.
- Puis donnez une commande qui extrait le contenu de la ligne 5 à 15 d'un fichier.

Exercice 2.4

- Expliquez ce que fait la commande `grep`.
- Puis donnez une commande qui affiche l'ensemble des shells qui sont en exécution sur la machine sur laquelle vous travaillez.
- Enfin comptez le nombre de ces shells.

Exercice 2.5

- Expliquez ce que fait la commande `sort`.
- Donnez la liste triée par ordre alphabétique des utilisateurs logés sur la machine sur laquelle vous travaillez.
- Donnez une commande qui permet d'afficher les fichiers d'un répertoire triés par n° d'i-noeud.
- Donnez la liste des processus (format long) triés par n° de processus père (le PPID).

Exercice 2.6 **

- Allez faire un tour sur la page `man` de la commande `sed` puis dans les pages `info`
- Donnez une commande qui remplace toute les occurrences de `“titi”` par `“toto”` dans un fichier.

²On rappelle que pour obtenir la liste des utilisateurs logés on utilise les commandes `who`, ou `users`, ou on compte le nombre de fichiers du répertoire `/dev/pts`