

Administration systèmes
TP 5B - shell et manipulations des fichiers

Contents

1 Objectifs du TP	1
1.1 Objectif général	1
1.2 Objectifs spécifiques	1
2 Le shell	1
2.1 Métacaractères et substitution de noms de fichiers	1
2.2 Redirections et tubes	2
3 Manipulation des fichiers	2
3.1 Commandes d'affichage	2
3.2 Décompte et tri	3
3.3 Sélection de lignes et de colonnes	3
3.4 Modification de texte	3
4 Bonus : l'éditeur de texte vim	3

1 Objectifs du TP

1.1 Objectif général

A la fin de cette activité l'apprenant devra être capable d'exploiter adéquatement le shell et de manipuler les fichiers sur linux.

1.2 Objectifs spécifiques

A la fin de cette activité l'apprenant devra être capable de :

1. exploiter les fonctionnalités du shell : métacaractères, substitution de noms de fichiers, redirection et tubes;
2. manipuler les fichiers sur linux : affichage , tri, numérotation, filtre, recherche.

2 Le shell

2.1 Métacaractères et substitution de noms de fichiers

Activité 1

Le répertoire /dev (devices) contient les fichiers spéciaux du système . Placez vous dans ce répertoire, Que font les commandes suivantes :

- | | | | |
|-------------------|-------------------|--------------------|-----------------------|
| 1. \$ ls -l tt* | 4. \$ ls -l tty?? | 7. \$ ls -l [ab]* | 10. \$ ls -l [^ b-z]* |
| 2. \$ ls -l "tt*" | 5. \$ ls -l *0 | 8. \$ ls -ld [ab]* | 11. \$ ls -l *[0-9]* |
| 3. \$ ls -l tty? | 6. \$ ls -l *y* | 9. \$ ls -l pts | 12. ls -ld ????? |

Placez vous dans le répertoire /etc qui contient les fichiers de configuration. Que font les commandes suivantes :

1. `$ cp systemd/*.conf ~/TP4`
2. `$ cp -r cron* ~/TP4`

Activité 2

2.2 Redirections et tubes

Le répertoire /usr/include contient les fichiers d'entête standards en langage C (stdlib.h, ...). placez vous dans ce répertoire et exécutez les commandes suivantes, que font elles?

1. `ls -ld std*`
2. `echo std*`
3. `echo std* >~/TP4/stdfiles`
4. `ls -ld [ab]*`
5. `echo [ab]*`
6. `echo [ab]* | tr " " "\n"`
7. `echo [ab]* | tr " " "\n" | wc -l`
8. `echo [ab]* | tr " " "\n" >~/TP4/abcfiles`
9. `echo c* | tr " " "\n" >>~/TP4/abcfiles`
10. `wc -l <abcfiles`
11. `echo nous avons 'echo std* |wc -w' fichiers std`

3 Manipulation des fichiers

Quatre fichiers liste1,..., liste4 vous ont été remis contenant chacun une liste de 50 personnes (nom, prénoms,sexe,age). Copiez les dans votre répertoire de travail ...

3.1 Commandes d'affichage

1. **cat** concatène des fichiers et les affiche, l'option -n permet de numéroté les lignes :
`$ cat liste1 # affiche le contenu de liste1`
`$ cat liste1 liste2`
`$ cat -n liste1 liste2`
`$ cat -n # prend ses données de l'entrée standard,[Ctrl]+[d] pour arrêter le flux1.`
2. **tac** concatène des fichiers et les affiche à l'envers :
`$ tac liste1`
3. **less** lit un fichier et permet de se déplacer par ligne :
`$ less liste1`
les flèches de direction permettent de se déplacer et q permet de sortir.
4. **head** affiche le début d'un fichier, par défaut 10 lignes :
`$ head liste1`
`$ head liste1 liste2`
`$ head -n 5 liste1 liste2 # affichera 5 lignes`
5. **tail** affiche la fin d'un fichier, elle fonctionne comme head :
`$ tail -n 5 liste1`

¹Toutes les commandes qui suivent fonctionnent dans le même principe

3.2 Décompte et tri

1. **décompte : wc** (word count) affiche le décompte du nombre de lignes, de mots et d'octets d'un fichier
\$ wc liste1 liste2
-l pour le nombre de lignes, -w pour le nombre de mots, -c pour le nombre de caractères, ...
2. **tri : sort** trie les lignes de fichiers texte :
\$ sort liste1 liste2
\$ sort /etc/passwd -t":" -k3 -n # que fait cette commande?
-r pour un tri décroissant, -n pour une comparaison numérique, -t pour un séparateur, -k pour indiquer la colonne, -f ignorer la casse, -u sans doublon. ...

3.3 Sélection de lignes et de colonnes

1. **sélection de ligne : grep** (General Regular Expression Parser) affiche les lignes correspondant à un motif donné:
\$ grep ndiaye liste1 liste2
-c nombre de lignes seulement, -n numéros de lignes seulement, -l nom des fichiers seulement, -i ignore la casse, -v les lignes ne correspondant pas au critère, ...
Remarque : le **motif** est une **expression rationnelle**, reportez vous à la page de manuel de grep ...
2. **sélection de colonnes : cut** affiche des parties de chaque ligne d'un fichier
\$ cut -c5-10 liste1 # affiche du cinquième au dixième caractère de chaque ligne
\$ cut -f2 liste1 liste2 #affiche la deuxième colonne
\$ cut -f1,3 liste1 liste2 #affiche la première et troisième colonne
\$ cut -d: -f3 liste3 #affiche la troisième colonne, le séparateur de colonnes est ":"

3.4 Modification de texte

1. **remplacement : tr** convertit ou remplace des caractères :
\$ tr "ab" "Ax" < liste1 # remplace tous les a par A et tous les b par des X dans le fichier liste1
\$ tr "\t" ":" < liste1 # remplace toutes les tabulations par ":" dans le fichier liste1
\$ tr -d "s" < liste1 # supprime tous les s dans le fichier liste1
2. Pour aller plus loin consulter la commande **sed** (**s**tream **e**ditor).

4 Bonus : l'éditeur de texte vim

vim² (ça se lit vi-ai-m) est un éditeur de texte libre qui assure une compatibilité ascendante avec l'éditeur vi. C'est un outil assez bien apprécié par les administrateurs systèmes car étant présent dans la plupart des systèmes et à haute disponibilité.

Pour cette activité vous exécuterez tout simplement la commande `vimtutor` et suivrez le tutoriel. C'est un tutoriel d'une demi heure sur vim.

²Verifier que vim est installé avec la commande `"vim --version"` sinon l'installer avec la commande `"sudo apt-get install vim"`