

Compte-rendu TP2 - Systèmes de fichiers - commandes

L3 - Systèmes et Réseaux - 3 octobre 2021

Exercice 1: Manipulation d'une arborescence de fichiers

1. Déterminer votre numéro d'utilisateur (uid) sur le système ainsi que le nom et numéro du groupe auquel vous appartenez (gid).

Réponse : `>echo $UID $GID`

2. Déplacer vous dans votre répertoire d'accueil avec la commande `cd`. Utiliser la commande `pwd`.

Réponse : la commande `pwd` affiche à chaque fois le répertoire courant.

3. Quels sont les droits d'accès de votre répertoire d'accueil ? Cela vous paraît-il logique ?

Réponse : `>ls -l`

Commentaires : On remarque tous les répertoires portent le nom d'un des utilisateurs. Chaque utilisateur possède les droits de lecture, d'écriture et d'exécution sur son dossier, les droits des autres utilisateurs varient. Certains possèdent le droit de lecture ou d'exécution ou encore les deux mais pas le droit d'écriture ce qui est logique pour ne pas avoir ses dossiers et fichiers modifiés ou supprimés par un autre utilisateur.

4. Télécharger le fichier `Cyrano.txt`

- (a) Afficher ce fichier par les commandes `cat` et `more` et tester dessus la commande `wc`.

Réponse : `>cat Cyrano.txt`
`>more Cyrano.txt`
`>wc Cyrano.txt`

Commentaires : la commande `cat` permet d'afficher le contenu du fichier `Cyrano.txt`. La commande `more` permet aussi de visualiser le contenu du fichier mais avec une lecture progressive. La commande `wc` affiche le nombre de lignes, le nombre de mots et le nombre d'octets (taille) du fichier `Cyrano.txt`

- (b) Chercher les lignes contenant le mot `monsieur` dans ce fichier avec la commande `grep`. Tester les différentes options.

Réponse : `>grep 'monsieur' Cyrano.txt`

Commentaires : l'option

-p : permet d'interpréter le motif comme une expression Perl ;

-i : permet d'ignorer la distinction de casse (majuscule / minuscule) ;

-n : indique le numéro des lignes ;

-v : affiche toutes les lignes ne contenant pas le motif ;

-c : compte et affiche le nombre de lignes correspondant au motif ;

-r : lit tous les fichiers avec récursivité dans le dossier ;

5. Tester sur le contenu d'un répertoire entier la commande `file` (pour avoir plusieurs résultats différents).

Réponse : `>file TP`

Commentaires : TP : directory

La commande `file` détermine le type d'un fichier.

- Créer un dossier **MonDossier**. Copier puis déplacer des fichiers d'un autre répertoire dans ce dossier. Changer les droits de ce dossier. Supprimer un des fichiers contenus dans ce dossier. Supprimer ce dossier.
- Retrouver un fichier sur votre session à partir de la commande **find** et du nom du fichier.

Réponse : `>find -name "crTP2.tex"`

Commentaires : la commande indique, si le fichier existe, le chemin pour y accéder à partir du repertoire courant.

- Combien y a t-il sur votre session de fichiers avec l'extension **.pdf**, de plus de 20Mo ?

Réponse : `>find -name "*.pdf" -size +20M`

Résultat : Sur ma session il y a un fichier pdf de taille 23M, la commande affiche le chemin pour accéder à ce fichier

- Sur un fichier volumineux, comparer la taille du fichier et le volume utilisé sur le disque.

Réponse : `>...`

Commentaires : ...

Exercice 2: Redirection

- Créer à partir du shell un fichier texte qui contiendra le nom de la machine, la date et le répertoire courant, puis une ligne vide et le contenu du fichier **/etc/services**.

Réponse :

```
>logname > fich.txt
>date > fich.txt
>pwd > fich.txt
>echo > fich.txt
>cat /etc/services > fich.txt
```

- Lancer la commande `cat fichierInconnu 2> fichRed` . Puis afficher le contenu de **fichRed**.

Réponse : Résultats : Erreur comme il n'y a aucun fichier "fichierInconnu" ni de fichier "2"

Commentaires : un fichier **fichRed** vide est créé.

Exercice 3: Les filtres

Reprenez le fichier **Cyrano.txt**. À partir des commande **tr**, **head**, **tail** et **sort**, donner les commandes pour ...

- Afficher les lignes dans l'ordre croissant, dans l'ordre décroissant, dans un ordre aléatoire.

Réponse :

```
>cat Cyrano.txt | sort -d
>cat Cyrano.txt | sort -d -r
>cat Cyrano.txt | sort -R
```

- Créer un fichier avec les 5 premières lignes, les 5 dernières, les lignes 5 à 10.

Réponse :

```
>cat Cyrano.txt | head -n 5 >
fichier.txt
>cat Cyrano.txt | tail -n 5 >
fichier.txt
>cat Cyrano.txt | head -n 10 |
tail -n 6 > fichier.txt
```

- Remplacer les **a** par des **i**.

Réponse : `>tr 'a' 'i' < fichier.txt`

Exercice 4: Les pipes

À l'aide d'un pipe, écrire des enchaînements de commandes pour

1. Faire afficher le résultat d'un `ls` avec les noms de fichier en majuscules.

Réponse : `>ls | tr '[a-z]' '[A-Z]'`

2. Faire afficher les chemins de la variable `PATH` séparés par des `-` et non par des `:`

Réponse : `>echo $PATH | tr ':' '-'`

3. Recopier dans un fichier nommé `listeVar` la liste des variable d'environnement dont le nom contient au moins une fois le caractère `_`.

Réponse : `>env | grep '_'`

4. afficher l'ensemble des fichiers textes codés en UTF-8

Réponse : `>file -i * | grep -i "utf-8$"`

5. afficher l'ensemble des fichiers dans le répertoire `REP` que le propriétaire peut lire, modifier mais pas exécuter mais qu'un membre du groupe propriétaire peut juste lire.

Réponse : `>ls -l /REP | grep '^rw-r-'`