

TP III : UNIX et GNU/Linux : Exercices

TP 1 : créer un script `test-fichier`, qui précisera le type du fichier passé en paramètre, ses permissions d'accès pour l'utilisateur :

ex : résultats

Le fichier `/etc` est un répertoire :

`/etc` est accessible par `root` en lecture écriture exécution

Le fichier `/etc/smb.conf` est un fichier ordinaire qui n'est pas vide :

`/etc/smb.conf` est accessible par `titi` en lecture.

TP 2 : afficher le contenu d'un répertoire

Écrire un script bash `listerc_dir.sh` permettant d'afficher le contenu d'un répertoire en séparant les fichiers et les (sous)répertoires :

\$ `./listerc_dir.sh` affichera :

```
----- Fichiers dans /etc/rc.d -----  
rc  
rc.local  
rc.sysinit  
----- Repertoires dans /etc/rc.d -----  
init.d  
rc0.d  
rc1.d  
rc2.d  
rc3.d  
rc4.d  
rc5.d  
rc6.d
```

TP 3 : lister les utilisateurs

Écrire un script bash affichant la liste des noms de login des utilisateurs définis dans `/etc/passwd` ayant un UID supérieur à 500.

Indication : `for in $(cat /etc/passwd)` permet de parcourir les lignes du dit fichier.

TP 4 : lecture au clavier

La commande bash `read` permet de lire une chaîne au clavier et de l'affecter à une variable.

```
ex: echo -n "Entrer votre nom : "  
    read nom  
    echo "Votre nom est $nom"
```

La commande `file` affiche des informations sur le contenu d'un fichier (elle applique des règles basées sur l'examen rapide du contenu du fichier).

Les fichiers de texte peuvent être affichés page par page avec la commande `more` :

1- Tester ces trois commandes ;

2- Écrire un script qui propose à l'utilisateur de visualiser page par page chaque fichier texte du répertoire spécifié en argument. Le script affichera pour chaque fichier texte (et seulement ceux là) la question "voulez vous visualiser le fichier biduletruc ?". En cas de réponse positive, il lancera `more`, avant de passer à l'examen du fichier suivant.

TP 5 : étude d'un script de service UNIX

Ouvrir le fichier `/etc/init.d/network` dans Emacs (en lecture seule). Ce script est responsable du lancement et de l'arrêt du service "réseau", et est en général lancé par la commande `service network start`.

Vous n'êtes sans doute pas encore en mesure de comprendre tous les détails de ce script, mais devez pouvoir en comprendre l'architecture et répondre aux questions suivantes :

- 1- Quels sont les arguments acceptés par le script sur la ligne de commande ? Que fait-il si on donne un argument invalide ?
- 2- Que fait le script si le fichier `/etc/sysconfig/network` n'existe pas ?
- 3- Où est définie la fonction bash "action" ?
- 4- A quoi sert le fichier `/var/lock/subsys/network` ?