

Scripts shell

F. Lassabe

Contents

1	Introduction	2
2	Exercice 1	2
3	Exercice 2	2
4	Exercice 3	2
5	Exercice 4	2
6	Exercice 5	2

1 Introduction

Ce TD se base sur les scripts shell qui sont similaires à des programmes écrits en bash (ou tout autre version de shell). Dans notre cas, nous travaillerons en bash. Les scripts shell sont très utilisés pour les opérations d'administration systèmes afin de réaliser des opérations trop complexes pour être réalisées par une commande seule. Les possibilités de ces scripts sont très larges. Les scripts étant assimilables à des programmes en bash, pour cette session de TP, vous ne répondrez pas directement dans le PDF, mais vous déposerez les fichiers contenant vos scripts dans l'espace de dépôt de Moodle. Vous nommerez les scripts *ExerciceN.sh* où *N* est le numéro de l'exercice dans le document. Lorsqu'un exercice avance par étapes, vous ne déposez sur Moodle que la dernière version du script.

2 Exercice 1

Écrire un script qui affiche le répertoire et l'utilisateur courants.

3 Exercice 2

Écrire un script qui a deux noms (créer un lien physique pour le second) : *TriUsers.sh* et *InvTriUsers.sh*. Ce script affiche la liste des utilisateurs présents dans */etc/passwd* dans l'ordre alphabétique pour *TriUsers* et dans l'ordre inverse pour *InvTriUsers*. L'affichage doit être paginé si la liste d'utilisateurs est plus longue que l'écran.

4 Exercice 3

Écrire un script qui indique, pour le nom donné en paramètre, le type du fichier et, si ce dernier est de type ordinaire, les droits sur ce fichier.

5 Exercice 4

Écrire un script qui permet de faire une sauvegarde d'un répertoire source dans un répertoire cible. Le script prend en argument le chemin vers la source et le chemin vers la cible. Il contrôle qu'il a bien reçu ces deux arguments et qu'ils sont valables (i.e. ce sont des répertoires). Puis il procède comme suit :

- Pour chaque répertoire de la source, il cherche si ce répertoire existe dans la cible. Si oui, il ne fait rien, si non, il crée le répertoire dans la cible.
- Pour chaque fichier de la source, il cherche si celui ci existe dans la cible. Si non, il le copie dans la cible. Si oui, alors le fichier source est copié sur la cible à condition que leurs dates de modification **et** leurs sommes MD5 différent.

6 Exercice 5

Modifier le script de l'exercice 4 pour qu'une mise à jour d'un fichier se fasse en conservant une sauvegarde de la cible (en renommant un *fichier.ext* en *fichier.ext-timestamp*, où *timestamp* est le nombre de secondes depuis le 1er janvier 1970). La date est la dernière date de modification du fichier de la sauvegarde, avant son remplacement. Vous pouvez utiliser la commande **stat**.