

L1 HLIN102 - Du binaire au Web - TD/TP 2

Commandes système (suite)

Objectifs

L'objectif des exercices suivants est d'aller plus loin dans l'utilisation de la ligne de commandes dans l'environnement Unix/Linux. Pour répondre aux questions, appuyez vous sur le cours, vos acquis et des recherches "efficaces" sur internet.

1 Les alias

Un alias vous permet de créer des substitutions abrégées de commandes, soit longues à taper, soit souvent utilisées, etc.

La syntaxe pour créer un alias est la suivante (il s'agit aussi d'une commande) :

`alias nom_alias='ligne de commande'`

Avec `nom_alias` la substitution souhaitée pour la ligne de commande saisie à droite du signe `'='`.

Dans cet exercice, vous êtes initié à l'utilisation d'un alias pour un cas simple mais très utile. Vous pouvez par la suite vous amuser à créer des alias pour des cas complexes.

A l'exécution de la commande qui vous permet de supprimer un fichier ou un répertoire, on ne vous demande pas une confirmation avant la suppression effective. Il serait utile de créer un alias qui le permette. En effet, il est fréquent que des erreurs de suppression se produisent, en particulier, en utilisant l'étoile dans le nom de fichiers à supprimer ou lorsqu'on est dans le mauvais répertoire. Ces erreurs peuvent être graves, par exemple, lorsqu'il s'agit d'un travail important ou à rendre dans le cadre d'un contrôle de connaissances. Pour avoir une demande de notification automatique, il est possible d'utiliser un alias. Pour ce faire, réalisez, dans l'ordre, les actions suivantes :

1. Trouver la syntaxe de la commande de suppression qui vous permet de demander une confirmation avant suppression effective. Créer un alias avec le même nom que la commande initiale.
2. Tester l'alias pour supprimer une musique de votre répertoire "music" (cf. premier TP). Vérifier que le fichier a bien été supprimé.
3. Ouvrir un autre terminal sans fermer la fenêtre en cours. Dans la nouvelle fenêtre, supprimer une autre musique. Que se passe-t-il ?
4. Pour résoudre ce problème, ouvrir le fichier ".bashrc" se trouvant dans le répertoire d'accueil et insérer l'alias à la fin du fichier. Sauvegarder les modifications. Refaire l'opération demandée en 3.
5. Donner une explication aux points 3 et 4.
6. Supprimer le répertoire "music" en utilisant votre alias.

2 Votre premier script

Rappel Un script shell est une suite de commandes shell écrites dans un fichier texte. Il s'agit ainsi d'un petit programme qui, lors de son exécution, exécute cette suite de commandes. Il peut être utile, par exemple, pour automatiser des tâches répétitives.

Dans cet exercice, vous allez créer un script pour le shell *bash*. Les trois étapes suivantes sont indispensables :

1. créer un fichier avec l'extension `.sh` (en utilisant un éditeur de texte, comme *emacs*). Ce fichier doit contenir comme première ligne :

```
#!/bin/bash
```

2. écrire la suite des commandes souhaitées,
3. changer les droits du fichier pour le rendre exécutable,
4. et enfin, exécuter le script en utilisant la syntaxe suivante :
`./nom_du_fichier_script`

A présent, vous allez créer un message de bienvenu dans un terminal (votre nom utilisateur, date, message/commentaire/memo de quelques commandes/rappel de votre choix, etc.) à travers un script.

- utiliser en premier la commande "echo" pour afficher le texte "Hello!".
- Ajouter une commande qui permet d'afficher votre nom utilisateur.
- Ajouter la date.
- Compléter avec le texte que vous souhaitez voir s'afficher à chaque lancement d'un terminal (éventuellement en utilisant plusieurs commandes)
- Exécuter le script.
- Après avoir fini le développement de votre script, ouvrir le fichier ".bashrc" se trouvant dans votre répertoire d'accueil et ajouter, à la fin, le chemin complet de votre script. Ouvrir un nouveau terminal. Vous devez visualiser le résultat de l'exécution de votre script.

3 Concours de commandes

Suivre les instructions de votre chargé(e) de TD/TP.

4 Gestion de processus

A travers un exemple, cet exercice vous permet d'utiliser les principales commandes à connaître pour gérer vos processus.

- lancez l'application gimp à partir d'un terminal. Pouvez vous exécuter d'autres commandes ?
- quitter gimp et relancer le à nouveau en ajoutant un `&` à la fin de la ligne de commande. Qu'observez vous ?
- en laissant gimp ouvert, exécuter la commande "ps -edf". Que fait cette commande ? Tester avec les options "-aux".
- comment faire pour n'avoir en résultat du ps que ce qui concerne le processus gimp ?
- au lieu de quitter gimp en utilisant son interface graphique, tuez son processus en utilisant la commande :

```
kill -9 numero_processus_gimp
```

Où `numero_processus_gimp` est l'identifiant du processus à retrouver à partir du résultat de la commande ps.

- tester les commandes top et pstree.

4.1 Etude d'un script

Télécharger le script "compter.sh" depuis Moodle, puis :

- analyser son contenu pour en déduire ce qu'il fait,
- exécuter le script en prenant comme exemple le fichier Mac.txt et divers motifs,

- exécuter le script en utilisant d'autres fichiers,
- en déduire l'intérêt d'un script.

4.2 Variables d'environnement

Lorsque vous êtes connecté à votre environnement (Linux ou autre), les données sur votre compte utilisateur et quelques informations sur la configuration/état de votre environnement de travail sont sauvegardées dans des "variables d'environnement". Ces dernières sont mises à votre disposition ainsi qu'aux applications. En d'autres termes, une variable d'environnement est un conteneur mémoire dans lequel des données propres à chaque utilisateur et au système sont stockées. Elle est désignée par un nom/identifiant et son contenu peut être affiché et éventuellement modifié par l'utilisateur. Il existe plusieurs variables d'environnement, les principales à connaître sont :

HOME, USER, GROUPS, UID, PWD, SHELL, PATH et HOSTNAME

Remarque : le nom des variables d'environnement sont par convention en majuscules, il est nécessaire de respecter la casse.

- Trouver la signification de chaque variable.
- Utiliser la commande "echo" pour afficher la valeur ou contenu de chaque variable.
- Etudiez en particulier le contenu de la variable PATH, trouver la syntaxe permettant de modifier ce contenu, y ajouter le répertoire courant, s'assurer que votre réponse est correcte auprès de votre enseignant et déduire l'intérêt de cet ajout.
- Est-ce nécessaire d'inclure cette commande dans le fichier .bashrc ?