

ISS - Initiation aux Systèmes d'exploitation et au Shell

LU2IN020

TP 06 – Les expressions régulières

Julien Sopena

octobre 2023

Le but de cette cinquième semaine est d'étudier les expressions régulières, tant d'un point de vue syntaxique, que pratique. Ainsi, nous verrons leur mise-en-oeuvre dans le cadre de la commande **grep**.

Exercice 1 : Connaître son système

Il est souvent utile de récupérer dans une variable une information ou une liste d'informations sur le système. Pour les trouver, il faut parser les fichiers de configuration ou les résultats de commandes.

Dans cet exercice, nous allons utiliser l'ensemble des connaissances acquises en matière de composition de commande. Toutes les solutions se passent de structure de contrôle et ne sont composées que de commande simple reliée par des tubes. Vos réponses se limiteront donc à utiliser : **cat**, **tail**, **head**, **wc**, **cut**, **sort**, **uniq**, **grep** et **tr**.

Certaines, de ces questions sont difficiles. Il vous faudra travailler progressivement en vous approchant petit à petit du résultat. Testez ainsi régulièrement votre solution et ne développez pas directement de longue composition.

Question 1

Le fichier **/etc/hosts** contient un certain nombre d'adresses fixées en dure. Parmi elles, celle dédiée à *localhost*. Proposez une commande qui affiche l'IPv4 (*w.x.y.z* où *w*, *x*, *y* et *z* sont des nombres compris en 0 et 255) de *localhost*.

Question 2

La commande **lpoptions -l** affiche l'ensemble des options disponibles pour l'imprimante en indiquant avec une ***** celles qui sont activées. Proposez une commande qui affiche la liste des options actives.

Question 3

La commande **ip addr** affiche des informations sur l'état des différentes interfaces réseau. Proposez une commande qui affiche l'adresse IPv4 de toutes les interfaces *ethX* où *X* est un chiffre.

Question 4

Le fichier **/etc/passwd** est le fichier qui définit le nom des utilisateurs, *i.e.*, qui associe à chaque **uid** (entier) utilisé par l'OS un nom (chaîne de caractère) plus simple à utiliser au quotidien.

Ainsi on trouve sur chaque ligne :

- le nom de l'utilisateur (le *login*)
- une marque ('**x**') indiquant s'il y a un mot de passe associé
- un **uid**, un entier identifiant l'utilisateur
- un **gid**, un entier identifiant le groupe principal de l'utilisateur
- une chaîne de caractère décrivant l'utilisateur
- le chemin vers la home de l'utilisateur
- le shell utilisé par défaut par l'utilisateur à la connexion ou **nologin** pour les utilisateurs qui ne peuvent pas exécuter de commandes.

Compter le nombre d'utilisateurs présents dans votre machine, puis les utilisateurs qui ont **nologin** comme shell par défaut.

Question 5

Comme on a pu l'observer à la question précédente, la majorité des utilisateurs déclarés ne correspondent pas à des utilisateurs réels, mais sont utilisés pour lancer des services (impression, audio, ...). Dans la majorité des distributions, les vrais utilisateurs ont des **uid** qui commencent à 1000.

Implémentez un script `whoInstalled.sh` qui affiche le login correspondant à l'**uid** 1000, en général celui de la personne qui a installé la machine.

Question 6

Le fichier `/etc/group` est le fichier qui définit les groupes d'utilisateurs. Ainsi on trouve sur chaque ligne :

- le nom du groupe
- une marque ('**x**') indiquant s'il y a un mot de passe associé
- le **gid**, un entier identifiant le groupe
- la liste de logins des utilisateurs appartenant à ce groupe

En appelant le script `whoInstalled.sh` de la question précédente, proposez une commande qui affiche, sur une seule ligne, la liste des groupes de l'utilisateur correspondant à l'**uid** 1000. Votre commande se basera sur le contenu du fichier `/etc/group`.

Vous pourrez utiliser la commande `groups` pour vérifier votre résultat.

Question 7

La commande `df` affiche des statistiques sur les différents systèmes de fichiers. Proposez une commande qui affiche le point de montage (dernière colonne) de la partition ayant le plus de place disponible.