

1 - Recherchez dans /etc/passwd la ligne correspondant à votre utilisateur, et seulement celui-ci.

`grep "^login:" /etc/passwd`. Chaque ligne commence par le login.

2 - Recherchez dans /etc/passwd les utilisateurs n'ayant pas bash comme shell par défaut. Utilisez un tube (pipe).

`cat /etc/passwd | grep -v "bash$"`. Le shell est la dernière colonne, éliminez tout ce qui finit par bash.

3 - À l'aide d'une simple expression régulière, supprimez les lignes vides et les commentaires (commençant par #) de /etc/hosts.

`grep -v -E "^(#|$)" /etc/hosts`. Une ligne vide commence par ^ et finit par \$.

4 - Affichez uniquement la liste des logins et UID associés (champ 3) de /etc/passwd.

`cut -d":" -f1,3 /etc/passwd`.

5 - Quelle suite de commande permet de connaître le nombre de lignes réel (non vides et pas les commentaires) de /etc/hosts ? Utilisez deux fois les tubes et numérotez les lignes.

`cat -n /etc/hosts | grep -v -E "^(#|$)" | wc -l`

6 - Triez le fichier /etc/group par ordre de GID (troisième colonne) décroissant.

`sort -n -r -t: -k 3 /etc/group`

7 - Donnez la liste des différents shells utilisés par les utilisateurs déclarés sur votre machine. Il ne doit pas y avoir de doublons.

`cut -d: -f7 /etc/passwd | sort | uniq`

8 - Dans un fichier texte de votre choix (appelé ici fic) remplacez tous les é, è, ê, ë par un e (attention cette question ne fonctionne pas en unicode).

`cat fic | tr "éèêë" "e"`

9 - Comment récupérer la 27e ligne d'un fichier fic (partant du principe que fic en contient plus de 27) ?

- A - `cat -n fic | grep ^27 | cut -d" " -f2-`
- B - `grep -n "" fic | grep ^27 | cut -d: -f2-`
- C - `tail -27 fic | head -1`
- D - `head -27 fic | tail -1`

D. Les réponses A et B en l'état actuel ne fonctionnent plus si on arrive à 270 ou 2700, etc.

10 - faites en sorte que toute la liste soit placée dans un fichier ET affichée.

En utilisant tee : `2>&1 | tee liste`