

Correction de l'examen « Unix/Linux »
Polytech – IG3 – octobre 2008
Pierre Pompidor

Q1 : Qu'est-ce une distribution Linux ?

Un système d'exploitation basé sur un noyau UNIX pour un "PC". Il en existe plusieurs versions (Ubuntu...).

Q2 : Qu'est-ce un système de fichiers ?

"Ensemble" des tables systèmes qui organisent le stockage et l'accès aux informations contenues dans une mémoire de masse (disque...).

Q3 : Que faut-il faire pour avoir une partition commune à Windows et à Linux ?

Le plus facilement, la créer sous un système de fichiers compatible pour les deux systèmes installés en Dual Boot (notamment en FATxx).

Q4 : Qu'apporte ext3 par rapport à FAT ?

Principalement la sécurisation grâce aux mécanismes de journalisation.

Q5 : Quel est le rôle de la mémoire virtuelle ?

Pallier la saturation de la mémoire vive en permettant d'utiliser le disque dur pour stocker des informations peu utilisées par les processus (cette mémoire est appelée SWAP sous Linux).

Q6 : Qu'est-ce un serveur ?

Un démon répondant à des requêtes réseau et donc offrant un service, et plus généralement une machine gérant un plusieurs de ces services.

(Vous m'avez presque tous que parlé de l'aspect "machine").

Q7 et Q8 : Comment configurer bash pour qu'il :

explore en premier le répertoire d'accueil lors de l'appel d'un exécutable ;

export PATH="~:\$PATH" ou export PATH="\$HOME:\$PATH" dans le .bashrc

passer automatiquement la main à tcsh.

tcsh dans le .bashrc (bon j'ai donné 0,5pt à ceux qui m'ont parlé de \$SHELL...)

Q9 : Que signifie le méta-caractère & pour bash ?

Le détachement : cela permet au processus père de continuer à être actif sans attendre la mort de son fils (plus ou moins 0,5pt à ceux qui m'ont parlé de tâche de fond...)

Q10 : Que réalise la commande : `chmod 751 ~/.*.pl`

`rwxr-x--x` (ou explications détaillées ;-)) à tous les fichiers de suffixe .pl du répertoire d'accueil

Q11 : Que réalise la ligne de commandes : `ls -l | egrep '^[^0-9]+$'`

Rien (ou pas grand chose) ! : car elle ne cherche à filtrer que les lignes d'informations détaillées sur le répertoire courant qui ne comprennent aucun chiffre (ce qui est difficile (date, taille, etc)).

Q12 : Que cherche à savoir l'utilisateur par la commande suivante : `ps -edf | grep -c firefox`

Le nombre de processus firefox qui tournent sur la machine.

Q13 : Pourquoi le résultat de la commande précédente n'est-il pas vraiment exact ?

Car firefox va apparaître dans le processus lui-même : "ps -edf | grep -c firefox"

Beaucoup de réponses ont tourné sur les désirs véritables/cachés de l'utilisateur, mais là on passe dans le subjectif (j'ai plus ou moins donné des points)...

Q14 : Que cherche à savoir l'utilisateur par la commande : find /etc -name bash* -mtime 1

Trouver l'emplacement de tous les fichiers dont le nom commence par bash, (a priori donc les fichiers de config. de bash), se trouvant dans le répertoire /etc et qui ont été modifiés avant hier (en deux mots (pas avant-hier)).

Q15 : Comment connaître le nombre de démons HTTP qui tournent sur votre machine ?

Et bien comme en Q13 : ps -edf | grep -c httpd

J'ai accepté toutes les tentatives raisonnables pour nommer le démon HTTP (demon_http, http*, apache...)

Q16 : Quelle est la différence d'exécution entre les deux commandes suivantes:

ls -l > fichier; grep *pl fichier et ls -l > fichier; grep ".*pl" fichier

Première question perverse :

Il suffisait de pointer que dans le premier cas * va être consommé par bash (il n'est pas protégé, le filtre portant donc sur le nom des fichiers Perl), ce qui n'est pas le cas dans la deuxième version...

Q17 : Quel problème prévient une connexion ssh avec l'option -X ?

Un problème devenu rare : le non-repositionnement de la variable DISPLAY, c'est à dire le fait de ne pouvoir afficher les applications interfacées lancées de la machine distante sur la machine locale.

Q18 : Quels sont les avantages respectifs des langage de scripts Perl et Python sur les langages système tels que VBS (ou les fichiers batch) sous DOS, ou bash/tcsh sous Linux ?

C'est qu'ils sont portables sous tous les systèmes d'exploitation évolués (Windows, Linux...) alors que cela n'est pas le cas avec VBS, batch ou bash/tcsh qui sont dévolus à leur système de prédilection. Bon j'ai accordé 0,5 sur l'exposé de leurs mérites intrinsèques (typage conciliant, regexp...)

Q19 : Avantages/inconvénients des tableaux associatifs par rapport aux tableaux classiques ?

Avantages : accès rapide (par fonction de hashage) et sémantique du nommage des éléments

Inconvénients : plus grande place en mémoire et surtout listing des éléments dans un ordre aléatoire

Q20 : Que se passe-t-il lors de l'exécution du script (felicitation.pl) (deux prochaines lignes) :

```
#!/usr/bin/perl
```

```
print "Et bien voilà, vous avez fini ! \n";
```

LA question perverse :

le #!/usr/bin/perl est décalé de deux caractères ce qui le désactive (le # doit être le premier caractère de la ligne).

Donc un appel du script directement par son nom ou par "./felicitation.pl" provoque son interprétation par bash, et le plantage du script.

Mais un appel par "perl felicitation.pl" réussit.

Je sais, je sais.

Je ne le referai plus.