

QCM : Gestion des processus sous Linux

Section 1 : Questions à choix multiples (18 points)

Q1 : Un processus est :

- a) Un programme stocké sur le disque.
- b) Un programme en exécution + son contexte.
- c) Une instruction exécutée par le processeur.
- d) Un fichier d'instructions en attente d'exécution.

Réponse correcte : b

Q2 : Quelle commande permet d'afficher l'arbre des processus sous Linux ?

- a) `ps`
- b) `top`
- c) `pstree`
- d) `jobs`

Réponse correcte : c

Q3 : Quel état d'un processus signifie qu'il attend un événement particulier sans consommer le CPU ?

- a) Actif
- b) Suspendu
- c) Endormi
- d) Zombie

Réponse correcte : c

Q4 : Que représente le PID d'un processus ?

- a) Le numéro unique de son processus parent.
- b) Un identifiant unique attribué par le noyau.
- c) Le chemin absolu de son programme associé.
- d) Une priorité assignée par l'ordonnanceur.

Réponse correcte : b

Q5 : Quelles sont les priorités possibles pour un processus standard modifié par un utilisateur ordinaire ?

- a) De -20 à +20
- b) De 0 à +19
- c) De -10 à +10
- d) Fixées par le noyau uniquement

Réponse correcte : b

Q6 : Comment mettre un processus en pause et le réactiver ensuite ?

- a) `kill -STOP` puis `kill -CONT`
- b) `fg` puis `bg`
- c) `jobs` puis `ps`
- d) `ctrl+c` puis `ctrl+z`

Réponse correcte : a

Q7 : Quelle commande permet d'envoyer un signal pour tuer un processus en forçant l'arrêt immédiat ?

- a) `kill -15`
- b) `kill -1`
- c) `kill -9`
- d) `kill --force`

Réponse correcte : c

Q8 : Le statut "Zombie" d'un processus signifie :

- a) Il utilise activement le CPU.
- b) Il est suspendu par l'utilisateur.
- c) Il est terminé mais conserve des ressources mémoire.
- d) Il attend l'assignation d'une priorité.

Réponse correcte : c

Q9 : Quelle commande permet de modifier dynamiquement la priorité d'un processus en cours d'exécution ?

- a) `nice`

- b) renice
- c) top
- d) ps

Réponse correcte : b

Q10 : Quelle commande liste les processus lancés par un utilisateur précis ?

- a) ps -u [nom_utilisateur]
- b) pstree
- c) pidof
- d) top

Réponse correcte : a

Q11 : Quel signal est utilisé pour suspendre un processus sans possibilité d'interruption ?

- a) SIGKILL
- b) SIGSTOP
- c) SIGHUP
- d) SIGTERM

Réponse correcte : b

Q12 : Si vous voulez exécuter un programme en arrière-plan directement, quelle syntaxe utilisez-vous ?

- a) programme
- b) programme --bg
- c) programme &
- d) programme [background]

Réponse correcte : c

Q13 : Quelle commande permet d'afficher uniquement les processus ayant un certain nom ?

- a) jobs
- b) pstree
- c) ps -C [nom]
- d) pidof

Réponse correcte : c

Q14 : Quel est le rôle de la commande `fg` ?

- a) Lister les processus actifs.
- b) Réactiver un processus en avant-plan.
- c) Suspendre un processus.
- d) Afficher les tâches en arrière-plan.

Réponse correcte : b

Q15 : Quelle commande permet de trouver le PID d'un processus spécifique ?

- a) `pidof`
- b) `jobs`
- c) `pstree`
- d) `ps -u`

Réponse correcte : a

Q16 : Quel signal est envoyé par défaut avec la commande `kill` ?

- a) `SIGTERM`
- b) `SIGKILL`
- c) `SIGHUP`
- d) `SIGSTOP`

Réponse correcte : a

Q17 : Quel paramètre permet à `top` de trier les processus par utilisation mémoire ?

- a) `m`
- b) `p`
- c) `u`
- d) `t`

Réponse correcte : a

Q18 : Qu'est-ce que le "nice" d'un processus ?

- a) Sa priorité initiale assignée par l'utilisateur.
- b) Son pourcentage d'utilisation du CPU.
- c) Le signal qu'il attend.
- d) Sa capacité à interagir avec d'autres processus.

Réponse correcte : a

Section 2 : Questions ouvertes (10 points)

Q19 : Expliquez les états possibles d'un processus sous Linux et donnez un exemple de commande permettant de consulter son état.

Correction : Les états possibles sont : actif (running), prêt (ready), endormi (sleeping), suspendu (stopped), zombie. La commande `ps` permet de consulter ces états en affichant la colonne "STAT".

Q20 : Quelle est la différence entre un processus d'avant-plan et un processus d'arrière-plan ?

Correction : Un processus d'avant-plan (foreground) interagit directement avec l'utilisateur via le terminal. Un processus d'arrière-plan (background) s'exécute sans interaction directe. Exemple : `nano` (avant-plan), `xeyes` & (arrière-plan).

Q21 : Donnez un exemple de situation où un processus peut devenir un zombie. Comment résoudre ce problème ?

Correction : Un processus devient zombie lorsqu'il termine son exécution mais que son parent n'a pas récupéré son statut via un appel `wait`. Pour résoudre ce problème, il faut s'assurer que le processus parent gère correctement ses enfants ou utiliser `kill` pour terminer le parent.