



جامعة محمد الأول بوجدة
UNIVERSITE MOHAMMED PREMIER OUJDA
المدرسة الوطنية للعلوم التطبيقية
École Nationale des Sciences Appliquées d'Oujda

Université Mohammed Premier Oujda
École Nationale des Sciences Appliquées
Département : Électronique, Informatique et Télécom
Module : Administration des systèmes
Filières : Génie Informatique
Niveau : GI4

TP11 Administration des Systèmes : Planification des tâches par un système Linux

Enseignant : Mohammed SABER

Ressources requises

Ressources nécessaires :

1. Un ordinateurs Windows 7 avec un logiciel de virtualisation ;

Consignes pour le TP

1. Suivez les instructions pour chaque énoncé.
2. A la fin de TP, SVP réorganiser votre table :
 - Éteindre toutes les machines.
 - Réorganiser les chaises à ces places avant de sortir.
 - MERCI d'avance.
3. Un rapport de TP individuel est rendu sur la plateforme Moodle à la fin de TP (en format PDF ou DOC).
4. **Chaque étudiant ne respect pas les consignes de TP sera sanctionné.**

Énoncé 1 : Planification des tâches par cron

Rappel : outil de planification des tâches cron

cron est le démon en charge d'exécuter des commandes planifiées et récurrentes (chaque minute, chaque heure, chaque jour, chaque semaine, chaque mois.) ;

Dans un système Unix, de nombreuses tâches sont régulièrement planifiées :

- La rotation des logs ;
- La mise à jour de la base de données du programme locate ;
- Les sauvegardes ;
- Des scripts d'entretien (comme le nettoyage des fichiers temporaires).

Par défaut, tous les utilisateurs peuvent planifier l'exécution de tâches. C'est pourquoi chacun dispose de sa propre **crontab**, où il peut consigner les commandes à planifier. Il peut la modifier en exécutant **crontab -e** (ses informations sont stockées dans le fichier `/var/spool/cron/crontabs/utilisateur`).

Chaque ligne significative d'une **crontab** décrit une commande planifiée grâce aux six (ou sept) champs suivants :

- La condition sur les minutes (nombres compris de 0 à 59) ;
- La condition sur les heures (de 0 à 23) ;
- La condition sur le jour du mois (de 1 à 31) ;
- La condition sur le mois (de 1 à 12) ;
- La condition sur le jour de la semaine (de 0 à 7, le 1 correspondant au lundi — le dimanche est représenté à la fois par 0 et par 7 ; il est également possible d'employer les trois premières lettres du nom du jour en anglais comme Sun, Mon, etc.) ;
- Le nom d'utilisateur sous lequel la commande devra s'exécuter (dans le fichier `/etc/crontab` et dans les fragments déposés dans `/etc/cron.d/`, mais pas les **crontabs** des utilisateurs) ;
- La commande à exécuter (quand les conditions définies par les cinq premières colonnes sont remplies).

Tous les détails sont documentés dans la page de manuel **crontab(5)** ou **cron**.

L'utilisateur **root** dispose de sa **crontab** personnelle, mais peut également employer le fichier `/etc/crontab` ou déposer des **crontab** supplémentaires dans le répertoire `/etc/cron.d/`. Ces deux dernières solutions ont l'avantage de pouvoir préciser l'utilisateur sous l'identité duquel exécuter la commande.

Le paquet **cron** propose par défaut des commandes planifiées qui exécutent :

- Une fois par heure les programmes du répertoire `/etc/cron.hourly/` ;
- Une fois par jour les programmes du répertoire `/etc/cron.daily/` ;
- Une fois par semaine les programmes du répertoire `/etc/cron.weekly/` ;
- Une fois par mois les programmes du répertoire `/etc/cron.monthly/`.

On peut restreindre l'accès à **cron** en créant le fichier d'autorisation explicite `/etc/cron.allow`, où l'on consignera les seuls utilisateurs autorisés à planifier des commandes. Tous les autres seront automatiquement dépourvus de cette fonctionnalité. Inversement, pour n'en priver qu'un ou deux trouble-fête, on écrira leur nom dans le fichier d'interdiction explicite `/etc/cron.deny`.

Pour annuler une tâche planifiée pour **cron**, il suffit, d'utiliser la commande **crontab -r** pour supprimer un job d'un utilisateur, où lors d'un appel à **crontab -e** de supprimer la ligne correspondante dans la **crontab** où la tâche est définie.

Travail à faire

1. Se connecter en tant que «**root**» sur une console texte.
2. Vérifier que le démon **cron** s'exécute actuellement, sinon le lancer. (**Indication** : utiliser les commandes **service cron status** ou **/etc/init.d/cron status**).
3. Déterminer le chemin du répertoire des fichiers **crontab** des utilisateurs.
4. Lancer un terminal la commande **man crontab** pour visualiser le manuel de la commande **crontab**.
5. Se connecter en tant que utilisateur «**utilisateur principal de votre système**» sur une autre console texte. (**Indication** : utiliser compte de l'utilisateur **ensao**).
6. Afficher le contenu de la **table cron** de cet utilisateur. (**Indication** : utiliser la commande **crontab** avec les options adéquates).
7. Déterminer le nom de terminal de connexion de l'utilisateur. (**Indication** : utiliser la commande **tty**).
8. Programmer l'affichage de l'heure sur votre terminal toutes les minutes. (**Indication** : utiliser la commande **crontab** avec les options adéquates).
9. Vérifier que le job (tâche) est bien enregistré. (**Indication** : utiliser la commande **crontab** avec les options adéquates).
10. Vérifier le bon fonctionnement de la tâche une fois programmée.
11. Retourner sur la console texte sous «**root**».
12. Vérifier dans le répertoire des fichiers **crontab** qu'un fichier a bien été créé au nom de votre utilisateur.
13. Trouvez ce répertoire grâce à la commande **crontab**. (**Indication** : utiliser la commande **crontab** avec les options adéquates).
14. Vérifier que le système **CRON** laisse-t-il des traces via **SYSLOG** ? (**Indication** : utiliser la commande **cat /var/log/syslog | grep cron**). Que remarquez-vous ?
15. Retourner sur la console texte sous l'**utilisateur**.
16. Supprimer la tâche programmée dans la question 8. (**Indication** : utiliser la commande **crontab** avec les options adéquates).
17. Créer le fichier **/home/ensao/file.out**.
18. Programmer les tâches suivantes (**Indication** : utiliser la commande **crontab** avec les options adéquates).
 - (a) L'affichage de l'heure sur le fichier **/home/ensao/file.out** toutes les minutes.
 - (b) L'affichage de l'emplacement actuel sur le fichier **/home/ensao/file.out** toutes les trois minutes.
 - (c) L'affichage de l'état actuel de **cron** sur le fichier **/home/ensao/file.out** toutes les deux minutes. (**Indication** : utiliser les commandes **service cron status** ou **/etc/init.d/cron status**).
19. Vérifier que les jobs (tâches) sont bien enregistrés. (**Indication** : utiliser la commande **crontab** avec les options adéquates).
20. Vérifier le bon fonctionnement des tâches une fois programmées. Au bout d'une, deux et trois minutes, afficher le contenu du fichier **/home/ensao/file.out**. (**Indication** : utiliser la commande **cat /home/ensao/file.out**).
21. Retourner sur la console texte sous «**root**».

22. Vérifier que le système **CRON** laisse-t-il des traces via **SYSLOG** ? (**Indication** : utiliser la commande `cat /var/log/syslog | grep cron`). Que remarquez-vous ?
23. Arrêter le démon **cron**. (**Indication** : utiliser les commandes `service cron stop` ou `/etc/init.d/cron stop`).
24. Vérifier que le démon **cron** s'exécute actuellement. (**Indication** : utiliser les commandes `service cron status` ou `/etc/init.d/cron status`).
25. Vérifier que le système **CRON** laisse-t-il des traces via **SYSLOG** ? (**Indication** : utiliser la commande `cat /var/log/syslog | grep cron`). Que remarquez-vous ?
26. Retourner sur la console texte sous l'utilisateur.
27. Vérifier le fonctionnement des tâches déjà programmées. Au bout d'une, deux et trois minutes, afficher le contenu du fichier `/home/ensao/file.out`. (**Indication** : utiliser la commande `cat /home/ensao/file.out`). Que remarquez-vous ?
28. Retourner sur la console texte sous «root».
29. Démarrer le démon **cron**. (**Indication** : utiliser les commandes `service cron start` ou `/etc/init.d/cron start`).
30. Vérifier que le système **CRON** laisse-t-il des traces via **SYSLOG** ? (**Indication** : utiliser la commande `cat /var/log/syslog | grep cron`). Que remarquez-vous ?
31. Retourner sur la console texte sous l'utilisateur.
32. Vérifier le fonctionnement des tâches déjà programmées. Au bout d'une, deux et trois minutes, afficher le contenu du fichier `/home/ensao/file.out`. (**Indication** : utiliser la commande `cat /home/ensao/file.out`). Que remarquez-vous ?
33. Retourner sur la console texte sous «root».
34. Lancer les deux commandes suivantes :
 - (a) `cat /dev/null > /etc/cron.allow`.
 - (b) `cat /dev/null > /etc/cron.deny`.
35. Vérifier que le démon **cron** s'exécute actuellement, sinon le lancer. (**Indication** : utiliser les commandes `service cron status` ou `/etc/init.d/cron status`).
36. Retourner sur la console texte sous l'utilisateur.
37. Éditer de nouveau la crontab de cet utilisateur. Est-ce possible ? (**Indication** : utiliser la commande `crontab` avec les options adéquates). Que remarquez-vous ? Pourquoi ?
38. Vérifier le fonctionnement des tâches déjà programmées. Au bout d'une, deux et trois minutes, afficher le contenu du fichier `/home/ensao/file.out`. (**Indication** : utiliser la commande `cat /home/ensao/file.out`). Que remarquez-vous ? Pourquoi ?
39. Retourner sur la console texte sous «root».
40. Supprimer le fichier `/home/ensao/file.out`. (**Indication** : utiliser la commande `rm`).
41. Retourner sur la console texte sous l'utilisateur.
42. Au bout d'une minute, afficher le contenu du fichier `/home/ensao/file.out`. Qu'en déduire ?
43. Supprimer la tâche programmée dans la question 18. (**Indication** : utiliser la commande `crontab` avec les options adéquates). Que remarquez-vous ? Pourquoi ?
44. Retourner sur la console texte sous «root».
45. Ajouter le nom de l'utilisateur de votre session au fichier `/etc/cron.allow`.
46. Redémarrer le démon **cron**. (**Indication** : utiliser les commandes `service cron restart` ou `/etc/init.d/cron restart`).

47. Retourner sur la console texte sous l'utilisateur.
48. Éditer de nouveau la crontab de cet utilisateur. Est-ce possible? (**Indication** : utiliser la commande `crontab` avec les options adéquates).
49. Au bout d'une minute, afficher le contenu du fichier `/home/ensao/file.out`. Qu'en déduire?
50. Retourner sur la console texte sous «root».
51. Ajouter le nom de l'utilisateur de votre session au fichier `/etc/cron.deny` et supprimer le dans le fichier `/etc/cron.allow`.
52. Redémarrer le démon **cron**. (**Indication** : utiliser les commandes `service cron restart` ou `/etc/init.d/cron restart`).
53. Retourner sur la console texte sous l'utilisateur.
54. Au bout d'une minute, afficher le contenu du fichier `/home/ensao/file.out`. Qu'en déduire?
55. Supprimer la tâche programmée dans la question 18. (**Indication** : utiliser la commande `crontab` avec les options adéquates). Que remarquez-vous? Pourquoi?
56. Retourner sur la console texte sous «root».
57. Supprimer la crontab de l'utilisateur. (**Indication** : utiliser la commande `crontab` avec les options adéquates).
58. Redémarrer le démon **cron**. (**Indication** : utiliser les commandes `service cron restart` ou `/etc/init.d/cron restart`).
59. Visualiser le fichier crontab système `/etc/crontab`. Que signifient les lignes de ce fichier? (**Indication** : utiliser la commande `cat /etc/crontab`).
60. Programmer deux fois par jour, à 7h45 et 13h45, la suppression dans `/tmp` des fichiers dont la date de modification est supérieure à un jour.
61. Faire en sorte que les modifications apportées à la crontab système soient prises en compte.