**LAB8 : Planification de tâches Linux**

Ce TP aura pour objectif d'étudier les éléments essentiels à la planification de taches Linux.

***Partie 1 : Planification de taches avec AT***

***Rappel :***

Pour avoir les commandes réelles qui s'exécuteront lors de l'exécution d'un job, il faut utiliser la commande at -c JOBNUMBER.

La commande atrm JOBNUMBER supprime un job planifié lorsqu’il n’est plus nécessaire. Par exemple, lorsqu'une configuration de pare-feu distant a réussi, et ne

doit pas être réinitialisée.

**Exercice 1 :**

1. Pour obtenir un aperçu des jobs en attente pour l'utilisateur actuel, utilisez la commande atq ou at –l
2. Planifiez l'exécution d'un job dans trois minutes à partir de maintenant à l'aide de la commande at. Le job doit enregistrer la sortie de la commande date dans /home/student/myjob.txt
3. Utilisez la commande at pour lister tous les jobs planifiés.
4. Utilisez la commande watch atq pour pour surveiller la file d'attente en temps réel. Le job est supprimé de la file d'attente après son exécution.
5. Utilisez la commande cat pour vérifier que le contenu de myjob.txt correspondre à la sortie de la commande date.
6. Utilisez la commande at pour planifier de manière interactive un job avec la file d'attente g qui s'exécute à teatime (16h00). Le job doit exécuter une commande qui affiche le message « Il est temps de teatime » dans le fichier tea.txt.
7. Utilisez la commande at pour planifier de manière interactive un autre job avec la file d'attente b qui s'exécute à 16h05. Le travail doit exécuter une commande qui imprime le message « Les cookies sont bons » dans le fichier cookies.txt.
8. Utilisez la commande atq pour afficher les numéros de jobs en attente.
9. Utiliser la commande at pour voir les commandes associées au job en attente 2.
10. Utiliser la commande at pour voir les commandes associées au job en attente 3.
11. Utilisez la commande atq pour afficher le numéro de job qui s'exécute à l'heure du thé (16:00) et supprimez-le à l'aide de la commande atrm.
12. Utilisez la commande atq pour afficher la liste des jobs en attente et confirmer que le job prévu pour s'exécuter à l'heure du thé (16h00) n'existe plus.

***Partie 2 : Planification de taches avec Cron***

Exercice 2 :

1. Utilisez la commande crontab -e pour ouvrir la crontab à l'aide de l'éditeur de texte par défaut.
2. Insérer la ligne suivante :

\*/2 08-20 \* \* Mon-Fri /usr/bin/date >> my\_first\_cron\_job.txt

Dans l'éditeur de texte, appuyez sur Echap et tapez: wq pour enregistrer les modifications et quitter l’éditeur. Lorsque l'éditeur se ferme, vous devriez voir la sortie suivante:

...output omitted...

crontab: installing new crontab

1. Utilisez la commande crontab -l pour voir les jobs récurrents planifiés.
2. Utilisez la commande while pour que votre invite shell se mette en veille jusqu'à ce que le fichier my\_first\_cron\_job.txt est créé suite à la réussite de l'exécution du

tâche récurrente que vous avez planifiée. Attendez que votre invite shell revienne.

1. Utilisez la commande cat pour vérifier que le contenu de my\_first\_cron\_job.txt correspond à la sortie de la commande date.
2. Utilisez la commande crontab -r pour supprimer tous les jobs récurrents planifiés
3. Utilisez la commande crontab -l pour vérifier qu'aucun job n'existe

Exercice 3

1. Planifier un job qui exécute la commande /usr/local/bin/annual\_backup exactement le 2 février à 9a.m chaque année.
2. Planifier un job qui affiche le mot « tekup » au propriétaire de job, toutes les cinq minutes entre 9 h et 17 h, tous les vendredis de juillet.
3. Planifier un job qui exécute la commande /usr/local bin /daily\_report tous les jours de la semaine à deux minutes avant minuit.
4. Planifier un job qui exécute la commande mutt pour envoyer le message électronique Checking in vers le destinataire boss@example.com tous les jours ouvrables (du lundi au vendredi), à 9 h.

Exercice 4 :

1. Planifier un job qui permet à l’utilisateur root d’exécuter le script « full-backup » chaque 10 juin à 08:30 du matin
2. Planifier une tache qui permet d’écrire la date et l’heure dans un fichier nommé date toutes les 2 h, de 8h à 12h et de 14h à 18h, à la 2ième minute
3. Planifier une tache qui écrit dans un fichier user le nom d’utilisateur connecté toutes les 30 secondes
4. Provoquer un reboot de la machine chaque 1er et 15 du mois à 2h 30 du matin
5. Ecrire hello dans un fichier test tous les quarts d'heure de 15h à 19h du lundi au vendredi seulement en 1ère quinzaine du troisième trimestre
6. Ecrire dans les fichiers logs le texte suivant « bonjour tekup » les matins du lundi au vendredi à 7 h 30
7. Faites une mise à jour des paquets tous les deux mois, le 1er de ce mois à midi et minuit
8. Ecrire le nombre des utilisateurs connectés dans un fichier users toutes les 5 minutes pendant la première demi heure.
9. Planifier une tache pour le 19 Aout 2020 à 15h.00 qui permet de supprimer l’utilisateur student
10. Planifiez l'exécution d'un job dans trois minutes à partir de maintenant qui permet de changer le shell de l’utilisateur student à /sbin/nologin
11. Planifier de manière interactive un job avec la file d'attente b qui s'exécute à 16h00. Ce job permet de verrouiller le compte de l’utilisateur mohamed .