

TP-LINUX: Tâches horaires**Objectif :** Automatiser des procédures de maintenance du système Linux**AUTOMATISATION DE CERTAINES TACHES.**

Linux possède un service en arrière-plan (*daemon*) appelé *cron* qui sert à exécuter des commandes à heure fixe. La commande utilisateur permettant le paramétrage est *crontab*.

**1 Crontab : Tâches des utilisateurs**

La documentation *crontab* vous expliquera que vous devez saisir une ligne de ce type :

```
15 14 1,3 * * /home/monscript.bash
```

Explication :

15	représente les minutes de l'heure (entre 0 et 59)
14	représente les heures (0-24)
1,3	représente le 1 ^{er} et le 3 du mois (entre 1 et 31)
*	représente le numéro du mois (1 à 12) ; * indique TOUS les mois
*	représente le numéro du jour de semaine (1 à 7) ; * indique TOUS les jours

DONC : le script */home/monscript.bash* s'exécutera le 1^{er} et le 3 de chaque mois à 14h15

TRAVAIL : Utilisez la commande *crontab* avec l'option *-e*



Dans *crontab -e*, vous devez choisir l'éditeur (Nano ou VI) Si vous choisissez VI, vous serez obligés d'utiliser les commandes *vi* vues lors d'une séance précédente.

Trouvez la ligne à ajouter pour faire le réglage suivant :

Exécuter votre script *enregistrement.bash* (créé dans les séances précédentes) toutes les minutes.

Vérifiez que *crontab* a bien enregistré votre demande (option 'L' en minuscule) : `crontab -l`

Vérifiez que le script est bien exécuté chaque minute (en consultant la base de données).

2 Crontab : Supprimer un travail programmé

Supprimez ensuite la tâche répétitive (option *R* en minuscule), pour ne pas saturer la base de données :

```
crontab -r
```



3 CRON : Tâches du système

Les tâches automatiques des utilisateurs ne s'exécutent que lorsque l'utilisateur est connecté.

Cron gère également des tâches « système » qui s'exécutent lorsque le système est allumé, même si aucun utilisateur n'est connecté. Il y a 2 possibilités :

Cas 1 : Fichier spécial dans le dossier */etc/cron.d*

Ce fichier à créer (son nom n'a pas d'importance) contiendra des lignes semblables à celles de *crontab* avec en + le nom de l'utilisateur sous lequel le script doit être lancé.

Exemple : 10 23 * * 1 pi /home/backup/backup.cmd

→ Le script */home/backup/backup.cmd* sera exécuté par l'utilisateur *pi* tous les lundis à 23h10

Travail à faire : Démarrer le script *enregistrement.bash* toutes les minutes, par l'utilisateur *root*.

Redémarrer le Raspberry SANS VOUS CONNECTER pour vérifier que l'enregistrement se fait même si vous n'avez pas ouvert de session. Au bout de 3 minutes vérifiez les enregistrements dans la base de données.

Cas 2 : (Dans */etc* :) *cron.hourly*, *cron.daily*, *cron.weekly*, *cron.monthly*

On n'a pas besoin de préciser le nom de l'utilisateur ni l'heure/jour/semaine/mois exacts :

Il suffit de copier le script dans un des dossiers ci-dessus.

TRAVAIL :

- Supprimez le réglage fait précédemment (cas1, dossier *cron.d*)
- Exécuter le script *enregistrement.bash* toutes les heures (dossier *cron.hourly*)

Pour tester, redémarrez votre Raspberry, et laissez-le connecté.

Votre prof consultera la base de données pour voir si votre nom apparaît toutes les heures.