

Table des matières

Utilisation de crontab	2
Syntaxe	2
Note	2
Exemples	2
Redirection Erreur	3



Utilisation de crontab

Syntaxe

Afin d'éditer la table cron, il suffit d'exécuter la commande suivante:

```
crontab -e
```

Qui lancera l'éditeur par défaut (souvent vi) présentant la table actuelle (ou un fichier vide s'il s'agit du premier lancement de crontab). Chaque ligne du fichier correspond à une tâche à exécuter, renseignée de la manière suivante:

```
mm hh jj MMM JJJ tâche > log
```

Où:

- mm représente les minutes (de 0 à 59)
- hh représente l'heure (de 0 à 23)
- jj représente le numéro du jour du mois (de 1 à 31)
- MMM représente le numéro du mois (de 1 à 12) ou l'abréviation du nom du mois (jan, feb, mar, apr, ...)
- JJJ représente l'abréviation du nom du jour ou le chiffre correspondant au jour de la semaine (0 représente le dimanche, 1 représente le lundi, ...)
- tâche représente la commande ou le script shell à exécuter
- log représente le nom d'un fichier dans lequel stocker le journal des opérations. Si la clause > log n'est pas spécifiée, cron enverra automatiquement un mail de confirmation. Pour éviter cela il suffit de spécifier > /dev/null

Note

Pour chaque unité de temps (minute/heure/...) les notations sont possibles:

- * : à chaque unité de temps
- 2-5 : les unités de temps (2,3,4,5)
- */3 : toutes les 3 unités de temps (0,3,6,...)
- 5,8 : les unités de temps 5 et 8

Exemples

Imaginons que l'on souhaite mettre en place un journal automatisé (dans le fichier /logs/log_df.log par

exemple) de l'espace disque libre (commande df) à des intervalles de temps précis:

Tous les jours à 23h30:

```
30 23 * * * df >> /logs/log_df.log
```

Toutes les heures, passées de 5 minutes :

```
5 * * * * df >> /logs/log_df.log
```

Tous les premiers du mois à 23h30:

```
30 23 1 * * df >> /logs/log_df.log
```

Tous les lundis à 22h28:

```
28 22 * * 1 df >> /logs/log_df.log
```

Du 2 au 5 de chaque mois à 10h12:

```
12 10 2-5 * * df >> /logs/log_df.log
```

Tous les jours pairs du mois à 23h59:

```
59 23 */2 * * df >> /logs/log_df.log
```

Toutes les 5 minutes:

- `*/* * * * * df » /logs/log_df.log`

Il est également possible d'exécuter automatiquement des commandes plus complexes à l'aide d'un script shell.

Il suffit pour cela de le déclarer en tant que tâche dans la table cron.

Redirection Erreur

Le Cas de **dead.letter**

c'est quoi >/dev/null?

/dev/null représente le rien.

Ce rien peut être comparé à un trou noir dans l'espace.

Ce qui a pour conséquence, que chaque flux envoyé dans ce trou noir ne pourra jamais être récupéré.

exemple de commande avec >/dev/null dans un crontab:

```
***4* /home/fred/script/test.sh >/dev/null
```



Chaque tâche planifiée dans un crontab envoie un mail à l'administrateur du système(ou root), pour éviter l'envoi systématique on redirige tout le flux STDOUT dans /dev/null.

A savoir, que lorsque /dev/null est utilisé sans précision, seule la sortie du flux STDOUT est concernée.

Donc, si une erreur se produit l'administrateur sera prévenu grâce à la sortie STDERR.

c'est quoi < /dev/null?

Dans un script, il se peut qu'une commande ait besoin d'une intervention humaine pour répondre à une question généralement yes ou no(y/n).

Forcer la réponse à yes

Ajouter l'option -f, exemple:

```
cp -f fichier1 directory/
```

La copie du fichier1 dans le répertoire directory est forcée.

Forcer la réponse à no

< /dev/null sera utilisé, exemple:

```
cp fichier1 /directory </dev/null
```

Si le fichier1 existe déjà dans le répertoire, avec l'option < /dev/null, le script n'écrasera pas le fichier1 déjà présent dans directory.

c'est quoi /dev/null 2>&1

2>&1 redirige la sortie d'erreur(STDERR) vers la sortie standard(STDOUT) et la sortie STDOUT dans /dev/null.

exemple en reprenant celui de la tâche cron:

```
*4*** /home/fred/script/test.sh >/dev/null 2>&1
```

Dans ce cas, l'administrateur ou root ne sera jamais si la commande c'est bien exécutée ou pas. Sauf, si le test est prévu dans le script test.sh.

— [sylvain](#) 2020/02/22 18:18

From:

<https://wiki.mazinger.fr/wiki/> - **My Personal Wiki**

Permanent link:

<https://wiki.mazinger.fr/wiki/doku.php?id=linux:commande:shell:crontab>

Last update: **2024/03/03 12:56**

