

Les indispensables de l'Admin système sous Debian #Rappels

by tontonfred - mardi, janvier 19, 2016

<http://www.tontonfred.net/blog/?p=1464>



Ceci n'est pas un cours mais un petit "TIPS" pour vous aider à vous remettre dans le bain..

Créer un fichier vide

touch le-nom-du-fichier

touch -t le-nom-du-fichier (-t utilise la date indiquée plutôt que la date actuelle)

Modifier la permission d'un fichier ou dossier

chmod 755 le-nom-du-dossier-ou-fichier

chmod -R 755 le-nom-du-dossier (Modifie récursivement les autorisations des dossiers et leurs contenus)

Modifier le propriétaire d'un fichier ou Dossier

chown propriétaire le-nom-du-dossier-ou-fichier

chown -R propriétaire le-nom-du-dossier (Modifie récursivement l'appartenance des répertoires et leurs contenus)

Gestion des comptes utilisateurs

- *su*
sudo
- *adduser*
- *useradd*

Redémarrer un service sous Debian

/etc/init.d/lenomduservice start , stop

Editer un fichier sous debian

Sous Unix, et en particulier sous Linux, la configuration du système et des programmes se fait très souvent en éditant des fichiers textes qui contiennent des paramètres de configuration. Ces paramètres de configuration suivent une certaine syntaxe, différente pour chaque programme, et que l'utilisateur doit connaître. Généralement, il y a une instruction de configuration par ligne de texte. Le système ou le programme va alors lire son ou ses fichier(s) de configuration et s'adapter à la configuration demandée.

Presque tous les programmes et systèmes Unix sont conçus avec une règle qui dit qu'il ne tient pas compte des lignes du fichier de configuration qui commencent par un certain caractère (souvent #). L'utilisateur peut alors mettre des lignes de commentaires dans le fichier de configuration en

commençant ces lignes par le caractère particulier. Il peut aussi facilement activer ou désactiver une ligne du fichier de configuration en enlevant ou en ajoutant le caractère particulier au début de la ligne. Le fait de désactiver ainsi une ligne de configuration se dit « *commenter une ligne* » et le fait d'activer ainsi une ligne de configuration se dit « *décommenter une ligne* ». Ces expressions seront régulièrement utilisées dans la suite de cette formation.

vi /etc/nom-du-fichier-a-editer (Editeur incontournable sur un système unix/linux)

gedit /etc/nom-du-fichier-a-editer (Editeur sous Gnome)

kedit /etc/nom-du-fichier-a-editer (Editeur sous KDE)

vim /etc/nom-du-fichier-a-editer (Version améliorée de vi mais à installer)

emacs /etc/nom-du-fichier-a-editer (Pour les puristes)

Changer son ip sous Debian

gedit /etc/network/interfaces

En statique :

auto eth0

iface eth0 inet static

address 192.168.0.7

netmask 255.255.255.0

gateway 192.168.0.254

En client DHCP :

auto eth0

allow-hotplug eth0

iface eth0 inet dhcp

Redémarrer les services réseau :

/etc/init.d/networking restart

Vider le cache DNS (Client)

Il suffit simplement de relancer le daemon NSCD :

/etc/init.d/nscd restart

Si cela ne marche pas, vous devez installer le paquet « nscd »

apt-get install nscd

Changer le nom d'hôte de votre Debian

Editez le fichier /etc/hostname

gedit /etc/hostname

Redémarrer le service hostname

/etc/init.d/hostname.sh

Testez votre modification en utilisant les commandes

hostname

hostname -f

Affiche l'état de tout les processus en cours

ps -e

Affiche les informations mise à jour périodiquement sur les processus, l'utilisation de la mémoire, du CPU.

top

Plus complet regardez ceci : <http://www.tontonfred.net/blog/?p=865>

Déport d'affichage par SSH

SSH possède une fonction de déport d'affichage. Il faut que le serveur SSH distant ait autorisé cette fonction.

ssh -X login@serveur.exemple.org

La programmation de tâches

Exécuter un commande périodiquement

La cron est un programme (installé par défaut) qui est chargé de lancer d'autres programmes de manière périodique et automatique. Chaque utilisateur peut définir avec sa *crontab* les programmes qu'il veut lancer périodiquement. Il lui suffit d'éditer sa *cron* et de définir la commande et sa périodicité d'exécution.

Par exemple, je veux que mon ordinateur me réveille tous les matins à 7h12 en jouant un mp3. J'édite ma crontab :

% crontab -e

Je me retrouve alors dans *vim* avec un fichier vide. J'ajoute la ligne :

```
12 7 * * * music123 ~/music/fichier.mp3
```

J'enregistre et je quitte ; les changements sont alors automatiquement pris en compte par le système.

Explications :

1. Les 5 premiers ensembles de caractères séparés par des espaces (ici, *12 7 * * **) définissent la fréquence. Dans l'ordre, on trouve :

- les minutes,
- les heures,
- le jour du mois,
- le mois,
- le jour de la semaine (sachant que Lundi vaut 1, Mardi vaut 2, etc.).

Pour exécuter une commande chaque jour à 7h12, on fixe donc le champ minute à 12, le champ heure à 7, puis on met des étoiles dans les autres champs pour dire qu'il faut que ce soit exécuté tous les jours dans le mois, tous les mois et tous les jours de la semaine.

2. Enfin, on trouve la commande à exécuter : (ici, *music123 ~/music/fichier.mp3*).

Quand la cron lance un programme, elle envoie par mail à l'utilisateur le texte que ce programme écrirait sur la console s'il était lancé à la main ; sauf si le programme n'a rien écrit. Pour éviter de recevoir un mail tous les matins qui me dit qu'il a bien joué mon fichier mp3, je modifie l'entrée dans la cron en renvoyant la sortie texte du programme vers */dev/null* :

```
12 7 * * * music123 ~/music/fichier.mp3 1>/dev/null 2>&1
```

Autre exemple : j'ai une connexion permanente à Internet et je veux aussi que fetchmail aille chercher mes mails tous les quarts d'heure. J'édite ma cron et je rajoute la ligne suivante :

```
*/15 * * * * fetchmail -silent
```

Explications :

1. Les 5 premiers ensembles de caractères `*/15 * * * *` définissent la fréquence « toutes les quinze minutes ».
2. J'exécute fetchmail avec l'option `-silent` qui n'écrit dans sa sortie que les messages d'erreur ; comme ça, je ne recevrais un mail que quand le rappatriement des mails se passe mal.

Exécuter une commande à une date donnée

Par exemple, pour exécuter les commandes `date` puis `df -h` le 28 septembre 2008 à midi 42, tapez :

```
% at 12:42 28.09.2008
```

Vous voyez alors apparaître le prompt du programme `at` pour taper les commandes :

```
at> date
```

```
at> df -h
```

Une fois que vous avez fini de taper les commandes, utilisez la combinaison de touches `Ctrl+d`. Le résultat de la série de commandes, appelée `job`, vous sera envoyé par mail juste après leur exécution.

Pour voir la liste des jobs en attente, utilisez la commande `atq`. Pour annuler un job, tapez `atrm numéro_du_job`.

Surveiller les connexions réseau du serveur "Humour"

- `netstat -laeptu`

- [netstat -taupe](#)
- [netstat -nao](#)



Envoyer l'article en PDF

PDF generated by Kalin's PDF Creation Station