



Maîtriser le temps...

Le 21 octobre 2008 à 16:14.

Garder son PC à l'heure ou synchroniser plusieurs machines sur la même base de temps n'est pour diverses raisons pas toujours évident. Et pourtant c'est primordiale lorsque l'on est plusieurs à accéder à une même ressource, ou simplement pour éviter de louper un rendez-vous...

Les distributions règlent généralement ce problème en collant un serveur NTP en mémoire mais c'est un peu prendre un marteau pour écraser une mouche et surtout cela manque de souplesse. Voyons donc comment fonctionne le temps sur un PC, et sur GNU/Linux en particulier.

Horloge Système

Depuis que le PC est PC, il y a toujours eu deux horloges, l'une logicielle et l'autre matérielle.

L'horloge logicielle des premiers PC et PC/XT, était prise en charge par une puce [Intel 8253](#) [1] relevée avec les PC/AT par la [8254](#) [2]. Ces puces "timer-counter" étaient programmées par le BIOS pour générer une interruption toutes les 54.936 secondes, soit environ 18.206 fois par secondes. A l'époque le manque de précision impliquant que l'horloge se désynchronisait avec la réalité en perdant en gros une minute par jours. Aujourd'hui les sources sont beaucoup plus précises avec des puces comme les [PIT](#) [3] (Programmable Interrupt Timer) utilisé notamment par Linux (clock=pit au démarrage du kernel).

La commande pour lire et écrire dans cette horloge est donc :

```
# réglage de l'horloge, petite remontée dans le temps...
root# date --set="9/22/96 16:45:05"
dim. sept. 22 16:45:05 CEST 1996

# lecture du l'horloge système
root# date
dim. sept. 22 16:45:17 CEST 1996
root#
```

Mais même si elle sont plus précises, les horloges logicielles gardent une lacune de taille, elle ne savent pas compter le temps lorsque le PC est éteint.

Horloge Matérielle



Cette horloge là est dite RTC (Real Time CLock) et prise en charge par un composant spécifique sur la carte mère (dans le temps, la fameuse "Dallas") même si aujourd'hui elle est de plus en plus intégrée au southbridge. Cette horloge est généralement basée sur un quartz qui oscille, comme celui des montres, à une fréquence de 32768Hz, soit 2^{15} fois par secondes, ce qui est pratique pour le comptage binaire. Ce circuit a sa propre alimentation, par une simple pile au lithium ou un [supercondensateur](#) [4]. Le résultat en est une horloge relativement précise qui tourne tout le temps, même PC éteint. C'est aussi grâce à ce composant que l'on peut, via le BIOS, allumer un PC à une heure précise.

Pour lire et écrire dans l'horloge matérielle la commande est

```
# lecture de l'horloge (on remarque que c'est bien une heure différente de celle de
"date")
root# hwclock --show
mar. 21 oct. 2008 15:24:57 CEST -0.408169 secondes

# Réglage de l'heure hardware
root# hwclock --set --date="9/22/98 16:45:05"

# vérification
root# hwclock --show
dim. 22 sept. 1998 16:45:09 CEST -0.751474 secondes
root#
```

Vous avez constaté en lançant ces commandes que la réaction n'est pas instantanée, en lecture comme en écriture, ce qui explique sûrement (mais je n'en suis pas bien sûr) pourquoi les OS n'utilisent pas uniquement cette horloge.

Maintenant voyons comment synchroniser l'horloge matérielle avec l'horloge logicielle et inversement :

```
# Horloge interne -> système en utilisant le temps universel
root# hwclock --hctosys --utc
```

```
# L'horloge système est maintenant en 1998
# Horloge système -> interne en utilisant le temps universel
root# hwclock --systohc --utc
root#
```

Horloge distante

Maintenant nos deux horloges sont bien sympathique mais qu'en est-il du "vrai" temps qui passe ? L'idéal pour être à l'heure est de la demander à un serveur qui sait de quoi il parle. Ce sont les serveurs NTP (Net Time Protocol). Et il y'en a de nombreux de disponible, alors autant en profiter.

Le seul problème est que la majorité des distributions passent par une usine à gaz pour récupérer le temps sur un serveur distant en passant par l'installation locale d'un service `ntpd` alors que là aussi, il existe une commande faite pour cela.

Cette commande c'est `ntpdate` du paquet `ntp-client` et son rôle est simplement de mettre à jour l'horloge système avec un serveur NTP distant. Ce serveur distant peut être sur [Internet](#) ^[5], mais aussi installé [par vos soins](#) ^[6].

Une fois que vous votre adresse, ici j'utilise `fr.pool.ntp.org`, vous pouvez lancer la commande de synchronisation :

```
# récupération de l'heure du serveur NTP
root# ntpdate -s -b fr.pool.ntp.org

# Vérification de la synchronisation avec l'horloge système
root# date
mar. oct. 21 15:40:27 CEST 2008

# Et l'horloge matérielle est quant à elle encore en 1998
root# hwclock --show
dim. 22 sept. 1998 16:59:15 CEST -0.251837 secondes
root#
```

Mise en musique

Maintenant nous n'avons plus qu'à mettre tout cela dans un script

```
#!/bin/sh

# synchronisation serveur NTP -> horloge logicielle
/usr/sbin/ntpdate -s -b fr.pool.ntp.org

# Quelque chose c'est mal passé ?
if [ $? != 0 ] ; then
    echo "Erreur lors de la synchronisation de l'heure" > /dev/stderr
else
    # synchronisation horloge logicielle -> horloge matérielle
    /sbin/hwclock --systohc --utc
fi
```

`/sbin/sync-clock.sh`

Ce script peut être lancé à la main en cas de besoin ou mieux, être collé dans le CRON pour être exécuté sur une base journalière. Mais Comme le fait très justement remarquer Axone dans un commentaire un peu plus bas ce n'est pas très sympathique pour les serveurs distant su tout le monde les appelle à heure entière (ex. xx:00). Donc nous allons faire cela plus intelligemment en créant un fichier `/etc/cron.d/time-sync` :

```
| 17 23 * * * /sbin/sync-clock.sh sh [7]
```

`/etc/cron.d/time-sync`

Ainsi la mise à jour est faite tous les jours à 23:17, ce qui limite bien les risques.

Conclusion

Voilà, pas de grosse artillerie donc pour simplement s'assurer que deux machines fonctionnent sur la même base de temps ou que votre portable ne vous fait pas louper un rendez-vous...

- [1] [http://fr.wikipedia.org/wiki/Intel 8253](http://fr.wikipedia.org/wiki/Intel_8253)
- [2] <http://ftp.utcluj.ro/pub/users/calceg/PMP/231164.pdf>
- [3] <http://www.netburner.com/downloads/mod5270/APPNOTE-MOD5270-PIT.pdf>
- [4] <http://fr.wikipedia.org/wiki/supercondensateur>
- [5] <http://www.pool.ntp.org/zone/europe>
- [6] <http://blog.nicolargo.com/2008/10/installation-dun-serveur-ntp.html>
- [7] <http://pwet.fr/man/linux/commandes/sh>