**systemd/cron**

systemd est capable de réaliser une partie des fonctionnalités de cron et d'anacron à l'aide des options Persistent et OnCalendar.

# Unités Minuterie (timer)

Les fichiers de minuterie sont des unités systemd qui se terminent avec le suffixe .timer.   
Ce type de fichier suit les mêmes règles que les fichiers de configuration pour les unités (units).

La configuration spécifique d'une minuterie se trouve dans la partie [timer].

Il y a deux manières de définir quand le service sera activé ...

* le **temps est relatif à un point de départ** en utilisant **OnUnitActiveSec** ...  
  Les points de départ les plus courants sont le démarrage du système, en utilisant l'option OnBootSec ainsi que le moment ou la minuterie a été activé  
  Dans ce cas il faut utiliser;
* le **temps relatif au temps réel**, en utilisant l'option **OnCalendar** …  
  Il faut donc choisir l'option qui correspond à ses besoins.  
  Cela dépend donc si on souhaite démarrer le service après chaque démarrage ou non.

Pour chaque fichier .timer, un service doit exister pour décrire l'unité à activer lors de l’activation de la minuterie (*timer*). Par défaut, un service portant le même nom que la minuterie (à l'exception du suffixe) sera actif.

# Mise en place

Ce guide travaille la création d’une unité temps.timer pour activer une autre unité temps.service.

Comme le service est démarré par l'unité Minuterie (*timer*), il n'est pas nécessaire d'écrire une section [Install] dans le service correspondant.   
On ne devra pas activer pas activer le service … C'est la minuterie qui va l’activer.

Toutefois, les unités minuterie ont besoin d'une section [Install] et doivent être activées par un timers.target.

## Affichage de la liste des minuteries

Pour consulter toutes les minuteries actives ainsi que leurs détails ...  
**>> sudo systemctl list-timers  
NEXT LEFT LAST PASSED UNIT ACTIVATES  
Tue 2022-12-13 07:46:41 EST 6min left n/a n/a systemd-tmpfiles-clean.timer systemd-tmpfiles-clean.service  
Tue 2022-12-13 07:54:53 EST 15min left Tue 2022-10-25 09:18:14 EDT 1 months 18 days ago fstrim.timer fstrim.service  
…  
7 timers listed.  
Pass --all to see loaded but inactive timers, too.**

## Création d'une minuterie relative à un point de départ

Si on souhaite démarrer un script d'archivage archivage.service une fois par semaine et que ce service doit être démarré après chaque démarrage.

On crée un script nommé archivage.sh …  
**>> sudo nano|vim /var/tmp/archivage.sh  
#!/bin/bash  
sudo apt update  
sudo apt upgrade --yes  
sudo apt autoremove --yes**

On crée le fichier **/etc/systemd/system/archivage.service** …  
**>> sudo nano|vim etc/systemd/system/archivage.service  
[Unit]  
Description=Sauvegarde du système  
  
[Service]  
Nice=19  
IOSchedulingClass=2  
IOSchedulingPriority=7   
ExecStart=/var/tmp/archivage.sh**

Il faut par la suite créer l'unité correspondante ...  
**>> sudo nano|vim /etc/systemd/system/archivage.timer  
[Unit]  
Description=Sauvegarde hebdomadaire ainsi qu'après le démarrage du système  
  
[Timer]  
OnBootSec=15min # le service démarrera 15 minutes après le démarrage de l’hôte  
OnUnitActiveSec=1w # le service démarre une fois/semaine, une semaine après la dernière activation de la minuterie  
  
[Install]  
WantedBy=timers.target**

Il faut ensuite activer et démarrer la minuterie.   
On doit créer un lien symbolique vers **/etc/systemd/system/timers.target.wants/archivage.timer**.

Pour s'assurer que les nouvelles minuteries et services sont activés ...  
**>> sudo systemctl start archivage.service  
>> sudo systemctl status archivage.timer**

**Remarque** …  
Ce service est marqué inactif étant donné qu'il ne doit pas rester actif une fois que la sauvegarde a été effectuée.   
Ainsi il n'est pas nécessaire d'utiliser l'option **RemainAfterExit=yes** dans l'unité correspondante.

La minuterie est indiquée en attente car il attend la prochaine activation.

Dans le cas où le service est gourmand en ressource, il est conseillé de démarrer la minuterie suffisamment de temps après le démarrage du système.   
Sinon le démarrage de X peut être sérieusement ralentis.

# Minuterie (timer) avec anacron

Pour planifier une tâche anacron, il faut pour cela utiliser les directives OnCalendar et Persistent.

Pour la création d’une minuterie hebdomadaire ...  
**>> sudo nano|vim /etc/systemd/system/archivage-anacron.service  
[Unit]  
Description=Archivage - une fois par semaine  
  
[Timer]  
OnCalendar=weekly # Autoriser la persistence entre les redémarrages  
Persistent=yes  
Unit=<le nom de l'unité à lancer>  
  
[Install]  
WantedBy=<selon le service>**

La notion de Persistent permet de déclencher la minuterie même si la date initialement prévue pour que la minuterie ne se lance a été dépassée (système éteint ou en mode veille).   
Cela permet également aux minuteries lentes (annuelles, mensueles, …) de se lancer sans avoir besoin d'une disponibilité (*uptime*) d'un an ou un mois.

# Evénements horaires, journaliers, mensuels, ...

La méthode à utiliser pour obtenir la fonctionnalité de cron est à l'aide de minuteries qui déclenchent des cibles (targets). Tous les services qui ont besoin d'être activés une fois par heure, par exemple, peuvent être déclarés comme dépendance de ces cibles

Comme première étape, il faut effectuer la création de certains répertoires ...

**>> sudo** **mkdir /etc/systemd/system/timer-{hourly,daily,weekly,monthly,yearly}.target.wants**

Il faut adapter selon ses besoins. Ensuite il faut créer les fichiers nécessaires.

## Evénements horaires

**Fichier /etc/systemd/system/timer-hourly.timer  
[Unit]  
Description=Événement horaire  
  
[Timer]  
OnBootSec=5min  
OnCalendar=hourly  
Unit=timer-hourly.target  
  
[Install]  
WantedBy=basic.target**

**Fichier /etc/systemd/system/timer-hourly.target  
[Unit]  
Description=Cible Événement horaire  
StopWhenUnneeded=yes**

## Evénements journaliers

**Fichier: /etc/systemd/system/timer-daily.timer  
[Unit]  
Description=Événement journalier  
  
[Timer]  
OnCalendar=daily  
Persistent=true  
Unit=timer-daily.target  
  
[Install]  
WantedBy=basic.target**

**Fichier /etc/systemd/system/timer-daily.target  
[Unit]  
Description=Cicle Événement journalier  
StopWhenUnneeded=yes**

## Evénements hebdomadaires

**Fichier: /etc/systemd/system/timer-weekly.timer  
[Unit]  
Description=Événements hebdomadaires  
  
[Timer]  
OnCalendar=weekly  
Persistent=true  
Unit=timer-weekly.target  
  
[Install]  
WantedBy=basic.target**

**Fichier: /etc/systemd/system/timer-weekly.target  
[Unit]  
Description=Cible Événement hebdomadaire  
StopWhenUnneeded=yes**

## Evénements mensuels

**Fichier: /etc/systemd/system/timer-monthly.timer>  
[Unit]  
Description=Événement mensuel  
  
[Timer]  
OnCalendar=monthly  
Persistent=true  
Unit=timer-monthly.target  
  
[Install]  
WantedBy=basic.target**

**Fichier: /etc/systemd/system/timer-monthly.target  
[Unit]  
Description=Cible Événement mensuel  
StopWhenUnneeded=yes**

## Evénements annuels

**Fichier: /etc/systemd/system/timer-yearly.timer  
[Unit]  
Description=Événement annuel  
  
[Timer]  
OnCalendar=yearly  
Persistent=true  
Unit=timer-yearly.target  
  
[Install]  
WantedBy=basic.target**

**Fichier: /etc/systemd/system/timer-yearly.target  
[Unit]  
Description=Cible Événement annuel  
StopWhenUnneeded=yes**

# Ajout d'un événement

Ajouter un service à ces cibles (targets) consiste à le placer dans le bon répertoire.   
Si on souhaite par exemple créer une tâche journalière, il faut créer un service pour systemd.   
Si on veut que le service tux.bar soit activé une fois par jour, il faut créer le fichier suivant ...

**Fichier: /etc/systemd/system/foo.service  
[Unit]  
Description=Démarrage du programme bar  
Wants=timer-daily.timer  
  
[Service]  
Nice=19  
IOSchedulingClass=2  
IOSchedulingPriority=7  
ExecStart=/usr/bin/bar --option1 --option2 # Plus d'un ExecStart peuvent être utilisés si nécessaire  
  
[Install]  
WantedBy=timer-daily.target**

Il faut alors démarrer et activer les services.

## Activation et démarrage une minuterie (*timer*)

La commande systemctl enable ... active une minuterie …  
**>> sudo systemctl enable timer-{hourly,daily,weekly,monthly,yearly}.timer   
>> sudo** **systemctl start timer-{hourly,daily,weekly,monthly,yearly}.timer**

## Activation d'événements par rapport au calendrier

Si on souhait activer un service par rapport à une date du calendrier, il faudra créer une nouvelle minuterie et lui lier le service.

**Fichier: /etc/systemd/system/tux.timer  
[Unit]  
Description=tux timer  
  
[Timer]  
# Pour choisir le bon format, se référer aux pages de man systemd.time   
OnCalendar=Mon-Thu \*-9-28 \*:30:00  
Persistent=true  
Unit=tux.service  
  
[Install]  
WantedBy=basic.target**

Le service devra ensuite être créé de la même manière que pour les services à intervalle.   
Il faut les déposer dans le répertoire /etc/systemd/system.

Référence …  
<https://wiki.archlinux.fr/Systemd/cron>