Version : **2024.01**

Dernière mise-à-jour : 2024/10/21 15:02

RH13402 - Gestion des Tâches

Contenu du Module

- **RH13402 - Gestion des Tâches**
  - Contenu du Module
  - LAB #1 - cron
    - 1.1 - Le Fichier /etc/crontab
    - 1.2 - Les Champs Temporels
    - 1.3 - Les crontabs Utilisateurs
  - LAB #2 - anacron
    - 2.1 - Le Fichier /etc/anacrontab
    - 2.2 - Les Champs Temporels
    - 2.3 - Le Fichier /var/spool/anacron/cron.daily
  - LAB #3 - at
    - 3.1 - Le Répertoire /var/spool/at/
  - LAB #4 - Timers systemd
    - 4.1 - systemd-tmpfiles-clean.timer
    - 4.2 - Configurer le systemd-tmpfiles-clean.service

cron

Important : Le service crond présume que la machine est allumée en permanence.

Le service crond est normalement lancé au démarrage de la machine. Ce service est chargé de faire exécuter des tâches (commandes et scripts) à des moments précis. Le service crond présume que la machine est allumée en permanence.

Le service crond lit toutes les minutes le fichier **/etc/crontab**, les crontabs se trouvant dans **/etc/cron.d** ainsi que les fichiers propres à chaque utilisateur.

Si une commande produit une sortie, celle-ci est dirigée vers la messagerie.

L'utilisation de crond est réservé à root. Cependant, vous pouvez établir une liste d'utilisateurs qui ont la permission d'utiliser crond en créant un fichier nommé **cron.allow** dans **/etc**. A l'inverse, un fichier **cron.deny** peut contenir une liste d'utilisateurs qui n'ont pas la permission d'utiliser crond.

Sous Red Hat 9, les crontabs des utilisateurs ont comme nom, le nom de l'utilisateur qui les crée et se trouvent dans le répertoire **/var/spool/cron/**. Par exemple, le crontab propre à l'utilisateur trainee est le fichier **/var/spool/cron/trainee**. Le service crond exécute des tâches en rajoutant une ligne dans son fichier de journalisation **/var/log/cron**.

1.1 - Le Fichier /etc/crontab

Sous Red Hat 9 le fichier /etc/crontab prend la forme suivante :

```
[trainee@redhat9 training]$ cat /etc/crontab
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root

# For details see man 4 crontabs

# Example of job definition:
# .---------------- minute (0 - 59)
# |  .------------- hour (0 - 23)
# |  |  .---------- day of month (1 - 31)
# |  |  |  .------- month (1 - 12) OR jan,feb,mar,apr ...
# |  |  |  |  .---- day of week (0 - 6) (Sunday=0 or 7) OR sun,mon,tue,wed,thu,fri,sat
# |  |  |  |  |
```

```
# *  *  *  *  * user-name  command to be executed
```

Important - Dans le fichier /etc/crontab, on peut constater la définition de variables dont SHELL, MAILTO et PATH. Ces variables sont différentes des variables système. Notez que sous Red Hat 9, le fichier **/etc/crontab** ne contient aucune tâche à exécuter.

1.2 - Les Champs Temporels

Chaque ligne dans un fichier /etc/crontab contient 5 champs temporels qui décrivent le périodicité de l'exécution de la tâche concernée.

Les 5 champs sont :

Minutes	Heures	Jour du mois	Mois de l'année	Jour de la sem.
(0-59)	(0-23)	(1-31)	(1-12)	(0-6)*

* le 0 correspond à dimanche.

Les champs temporels peuvent contenir des valeurs différentes :

Exemple	Description
Une valeur absolue telle 10	Dans le champs minutes = 10 minutes après l'heure
Une série de valeurs telle 2,6,8	Dans le champs mois = février, juin et août
Une intervalle telle 1-5	Dans le champs Jour de la Semaine = du lundi au vendredi
Le joker *	Dans le champs minutes = toutes les minutes
Une périodicité telle 0-23/2	Dans le champs heures = toutes les deux heures

Par exemple, voici le fichier **/etc/cron.d/0hourly** :

```
[trainee@redhat9 training]$ cat /etc/cron.d/0hourly
# Run the hourly jobs
```

```
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root
01 * * * * root run-parts /etc/cron.hourly
```

1.3 - Les crontabs Utilisateurs

Chaque utilisateur peut avoir sa propre version d'un fichier crontab. Pour vérifier s'il existe une version de crontab pour lui, l'utilisateur lancera la commande suivante :

```
[trainee@redhat9 training]$ crontab -l
no crontab for trainee
```

Afin de créer votre fichier crontab, lancez la commande crontab avec l'option -e en tant que votre nom d'utilisateur et non en tant que root :

```
[trainee@centos8 ~]$ crontab -e
```

Cette commande lance l'interface de l'éditeur **vi**. Tapez ensuite le texte suivant :

```
* * * * * /bin/pwd > pwd.txt
```

Sauvegardez et quittez vi.

Vous obtiendrez un résultat similaire à celui-ci :

```
[trainee@redhat9 training]$ crontab -e
no crontab for trainee - using an empty one
crontab: installing new crontab
```

Important : Ce fichier crontab va être exécuté toutes les minutes. La commande va inscrire le répertoire courant dans le fichier pwd.txt.

Le fichier crontab que vous venez de créer, au nom de votre utilisateur, a été écrit sur le disque dans le répertoire **/var/spool/cron/**.

Malgré le fait que vous venez de l'éditer, vous ne pouvez pas visualiser votre fichier en tant que simple utilisateur avec la commande cat. Seule la commande crontab -e vous permettra de consulter ou modifier votre fichier crontab.

Passez donc en mode super-utilisateur et visualisez le fichier en utilisant la commande cat :

```
[trainee@redhat9 training]$ su -
Password: fenestros
[root@redhat9 ~]# cat /var/spool/cron/trainee
* * * * * /bin/pwd > pwd.txt
```

Afin d'accorder ou non le droit d'éditer son propre fichier crontab, root a la possibilité d'inscrire les noms d'utilisateurs dans un des deux fichiers :

- **/etc/cron.allow** pour accorder le droit d'éditer son propre fichier crontab
- **/etc/cron.deny** pour interdire le droit d'éditer son propre fichier crontab

Par contre, si root inscrit le nom d'un utilisateur dans /etc/cron.deny et cet utilisateur a déjà planifié des tâches avec son propre fichier crontab, les tâches seront exécutées. Pour détruire les tâches planifiées de l'utilisateur, il faut détruire le fichier crontab de l'utilisateur.

LAB #2 - anacron

L'inconvénient avec crond est qu'afin que celui-ci puisse exécuter des tâches automatiquement, la machine doit rester allumée en permanence.

Il existe un alternatif très efficace pour des machines qui ne sont pas constamment allumée. Cette solution est le logiciel Anacron. Sous Red Hat 9, l'utilisation d'anacron **remplace** l'utilisation de cron pour les tâches système.

2.1 - Le Fichier /etc/anacrontab

Ouvrez donc le fichier /etc/anacrontab :

```
[root@redhat9 ~]# cat /etc/anacrontab
# /etc/anacrontab: configuration file for anacron

# See anacron(8) and anacrontab(5) for details.

SHELL=/bin/sh
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root
# the maximal random delay added to the base delay of the jobs
RANDOM_DELAY=45
# the jobs will be started during the following hours only
START_HOURS_RANGE=3-22

#period in days   delay in minutes   job-identifier   command
1       5       cron.daily              nice run-parts /etc/cron.daily
7       25      cron.weekly             nice run-parts /etc/cron.weekly
@monthly 45     cron.monthly            nice run-parts /etc/cron.monthly
```

Vous noterez que ce fichier fait référence aux répertoires **/etc/cron.daily**, **/etc/cron.weekly** et **/etc/cron.monthly**.

2.2 - Les Champs Temporels

Les deux premiers champs sont des champs temporels. Dans le cas d'anacron ceux-ci correspondent à:

Période	**Délai**
La périodicité en jours	Le délai en minutes, après le **démarrage d'anacron** et avant l'exécution des tâches dans le répertoire concerné

Anacron mémorise dans les fichiers qui se trouvent dans **/var/spool/anacron/** la date à laquelle il a exécuté les commandes qui se trouvent dans anacrontab pour la dernière fois. Ces fichiers sont appelés cron.daily, cron.weekly et cron.monthly.

```
[root@redhat9 ~]# ls /var/spool/anacron/
cron.daily  cron.monthly  cron.weekly
```

```
[root@redhat9 ~]# cat /var/spool/anacron/cron.daily
20241021
[root@redhat9 ~]# cat /var/spool/anacron/cron.weekly
20241021
[root@redhat9 ~]# cat /var/spool/anacron/cron.monthly
20240925
```

2.3 - Le Fichier /var/spool/anacron/cron.daily

Sous Red Hat 9, anacron consulte le fichier **/var/spool/anacron/cron.daily** chaque heure grâce à l'exécution du script **/etc/cron.hourly/0anacron** et, en fonction de la date inscrite dans le fichier et la date du jour, décide si oui ou non s'il doit exécuter la ligne de commande.

Voici le script **/etc/cron.hourly/0anacron** :

```
[root@redhat9 ~]# cat /etc/cron.hourly/0anacron
#!/usr/bin/sh
# Check whether 0anacron was run today already
if test -r /var/spool/anacron/cron.daily; then
    day=`cat /var/spool/anacron/cron.daily`
fi
if [ `date +%Y%m%d` = "$day" ]; then
    exit 0
fi

# Do not run jobs when on battery power
online=1
for psupply in /sys/class/power_supply/* ; do
    if [ `cat "$psupply/type" 2>/dev/null`x = Mainsx ] && [ -f "$psupply/online" ]; then
        if [ `cat "$psupply/online" 2>/dev/null`x = 1x ]; then
            online=1
            break
        else
            online=0
```

```
        fi
    fi
done
if [ $online = 0 ]; then
    exit 0
fi
/usr/sbin/anacron -s
```

Ce script a pour but d'exécuter la commande **/usr/sbin/anacron -s**. L'option **-s** indique à anacron d'attendre la fin d'exécution d'un job avant d'exécuter le suivant.

LAB #3 - at

Tout comme avec la commande cron, root a la possibilité d'accorder ou d'interdire aux utilisateurs le droit d'exécuter des tâches avec at en utilisant les deux fichiers suivants :

- **/etc/at.allow**
- **/etc/at.deny**

Si le fichier at.allow existe, seuls les utilisateurs dans ce fichier pourront exécuter at.

En tant que root, mettez en place maintenant deux tâches pour le 31/12/2042 à 13h00 et 14h00 respectivement :

```
[root@redhat9 ~]# at 13:00 12/31/2042
warning: commands will be executed using /bin/sh
at> pwd > /tmp/test13.atd
at> [^D]
at> <EOT>
job 1 at Wed Dec 31 13:00:00 2042

[root@redhat9 ~]# at 14:00 12/31/2042
warning: commands will be executed using /bin/sh
at> free > /tmp/test14.atd
```

```
at> [^D]
at> <EOT>
job 2 at Wed Dec 31 14:00:00 2042
```

3.1 - Le Répertoire /var/spool/at/

Sous Red Hat 9, les fichiers concernant ces deux tâches sont stockés dans le répertoire **/var/spool/at/** :

```
[root@redhat9 ~]# ls /var/spool/at
a000010249d890  a000020249d8cc  spool
```

A l'examen du deuxième fichier, vous constaterez un résultat similaire à celui-ci :

```
[root@redhat9 ~]# at -c 2
#!/bin/sh
# atrun uid=0 gid=0
# mail trainee 0
umask 22
SHELL=/bin/bash; export SHELL
HISTCONTROL=ignoredups; export HISTCONTROL
HISTSIZE=1000; export HISTSIZE
HOSTNAME=redhat9.ittraining.loc; export HOSTNAME
PWD=/root; export PWD
LOGNAME=root; export LOGNAME
HOME=/root; export HOME
LANG=en_US.UTF-8; export LANG
LS_COLORS=rs=0:di=01\;34:ln=01\;36:mh=00:pi=40\;33:so=01\;35:do=01\;35:bd=40\;33\;01:cd=40\;33\;01:or=40\;31\;01:
mi=01\;37\;41:su=37\;41:sg=30\;43:ca=30\;41:tw=30\;42:ow=34\;42:st=37\;44:ex=01\;32:\*.tar=01\;31:\*.tgz=01\;31:\
*.arc=01\;31:\*.arj=01\;31:\*.taz=01\;31:\*.lha=01\;31:\*.lz4=01\;31:\*.lzh=01\;31:\*.lzma=01\;31:\*.tlz=01\;31:\
*.txz=01\;31:\*.tzo=01\;31:\*.t7z=01\;31:\*.zip=01\;31:\*.z=01\;31:\*.dz=01\;31:\*.gz=01\;31:\*.lrz=01\;31:\*.lz=
01\;31:\*.lzo=01\;31:\*.xz=01\;31:\*.zst=01\;31:\*.tzst=01\;31:\*.bz2=01\;31:\*.bz=01\;31:\*.tbz=01\;31:\*.tbz2=0
1\;31:\*.tz=01\;31:\*.deb=01\;31:\*.rpm=01\;31:\*.jar=01\;31:\*.war=01\;31:\*.ear=01\;31:\*.sar=01\;31:\*.rar=01\
;31:\*.alz=01\;31:\*.ace=01\;31:\*.zoo=01\;31:\*.cpio=01\;31:\*.7z=01\;31:\*.rz=01\;31:\*.cab=01\;31:\*.wim=01\;3
```

```
1:\*.swm=01\;31:\*.dwm=01\;31:\*.esd=01\;31:\*.jpg=01\;35:\*.jpeg=01\;35:\*.mjpg=01\;35:\*.mjpeg=01\;35:\*.gif=01
\;35:\*.bmp=01\;35:\*.pbm=01\;35:\*.pgm=01\;35:\*.ppm=01\;35:\*.tga=01\;35:\*.xbm=01\;35:\*.xpm=01\;35:\*.tif=01\
;35:\*.tiff=01\;35:\*.png=01\;35:\*.svg=01\;35:\*.svgz=01\;35:\*.mng=01\;35:\*.pcx=01\;35:\*.mov=01\;35:\*.mpg=01
\;35:\*.mpeg=01\;35:\*.m2v=01\;35:\*.mkv=01\;35:\*.webm=01\;35:\*.webp=01\;35:\*.ogm=01\;35:\*.mp4=01\;35:\*.m4v=
01\;35:\*.mp4v=01\;35:\*.vob=01\;35:\*.qt=01\;35:\*.nuv=01\;35:\*.wmv=01\;35:\*.asf=01\;35:\*.rm=01\;35:\*.rmvb=0
1\;35:\*.flc=01\;35:\*.avi=01\;35:\*.fli=01\;35:\*.flv=01\;35:\*.gl=01\;35:\*.dl=01\;35:\*.xcf=01\;35:\*.xwd=01\;
35:\*.yuv=01\;35:\*.cgm=01\;35:\*.emf=01\;35:\*.ogv=01\;35:\*.ogx=01\;35:\*.aac=01\;36:\*.au=01\;36:\*.flac=01\;3
6:\*.m4a=01\;36:\*.mid=01\;36:\*.midi=01\;36:\*.mka=01\;36:\*.mp3=01\;36:\*.mpc=01\;36:\*.ogg=01\;36:\*.ra=01\;36
:\*.wav=01\;36:\*.oga=01\;36:\*.opus=01\;36:\*.spx=01\;36:\*.xspf=01\;36:; export LS_COLORS
LESSOPEN=\|\|/usr/bin/lesspipe.sh\ %s; export LESSOPEN
USER=root; export USER
SHLVL=1; export SHLVL
which_declare=declare\ -f; export which_declare
XDG_DATA_DIRS=/root/.local/share/flatpak/exports/share:/var/lib/flatpak/exports/share:/usr/local/share:/usr/share
; export XDG_DATA_DIRS
PATH=/root/.local/bin:/root/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin; export PATH
MAIL=/var/spool/mail/root; export MAIL
cd /root || {
         echo 'Execution directory inaccessible' >&2
         exit 1
}
${SHELL:-/bin/sh} << 'marcinDELIMITER7db8d310'
free > /tmp/test14.atd

marcinDELIMITER7db8d310
```

Pour supprimer cette tâche il convient d'utiliser la commande **atrm** ou **at -d** :

```
[root@redhat9 ~]# at -l
1       Wed Dec 31 13:00:00 2042 a root
2       Wed Dec 31 14:00:00 2042 a root

[root@redhat9 ~]# atq
1       Wed Dec 31 13:00:00 2042 a root
```

```
2       Wed Dec 31 14:00:00 2042 a root

[root@redhat9 ~]# at -d 2

[root@redhat9 ~]# atq
1       Wed Dec 31 13:00:00 2042 a root
```

Finalement, pour exécuter plusieurs commandes à la même heure d'une manière séquentielle, vous pouvez les insérer dans un fichier :

```
[root@redhat9 ~]# touch todo.list

[root@redhat9 ~]# echo pwd > todo.list

[root@redhat9 ~]# echo free >> todo.list

[root@redhat9 ~]# echo who >> todo.list

[root@redhat9 ~]# cat todo.list
pwd
free
who

[root@redhat9 ~]# at 14:30 12/31/2042 < todo.list
warning: commands will be executed using /bin/sh
job 3 at Wed Dec 31 14:30:00 2042
```

Important - La commande **batch** ou son alias **at -b** est utilisée pour exécuter des commandes au moment où la charge sur le système est en dessous un certain niveau. Par défaut la valeur est fixée à **1.5**. Cette valeur peut être modifiée par l'option **-l** da la commande **atd**.

LAB #4 - Timers systemd

Depuis l'introduction de systemd, certaines tâches sont gérées directement par ce dernier au lieu de par les outils traditionnels comme cron, anacron et at.

4.1 - systemd-tmpfiles-clean.timer

Par exemple, afin d'éviter à ce que le disque se remplit inutilement avec des fichiers temporaires, systemd lance le timer **systemd-tmpfiles-clean.timer**. Celui-ci déclenche le service **systemd-tmpfiles-clean.service** selon une fréquence régulière qui, à son tour, lance la commande **systemd-tmpfiles –clean** :

```
[root@redhat9 ~]# systemctl cat systemd-tmpfiles-clean.timer
# /usr/lib/systemd/system/systemd-tmpfiles-clean.timer
#  SPDX-License-Identifier: LGPL-2.1-or-later
#
#  This file is part of systemd.
#
#  systemd is free software; you can redistribute it and/or modify it
#  under the terms of the GNU Lesser General Public License as published by
#  the Free Software Foundation; either version 2.1 of the License, or
#  (at your option) any later version.

[Unit]
Description=Daily Cleanup of Temporary Directories
Documentation=man:tmpfiles.d(5) man:systemd-tmpfiles(8)
ConditionPathExists=!/etc/initrd-release

[Timer]
OnBootSec=15min
OnUnitActiveSec=1d
```

Dans ce fichier on peut constater la section **[Timer]** qui continet deux directives :

- **OnBootSec=15min** - indique que le **systemd-tmpfiles-clean.service** est déclenché 15 minutes après le démarrage du système,
- **OnUnitActiveSec=1d** - indique que par la suite, le service est déclenché 24 heures après son dernier déclenchement.

Constatez donc que le **systemd-tmpfiles-clean.timer** soit bien activé et démarré :

```
[root@redhat9 ~]# systemctl status systemd-tmpfiles-clean.timer
● systemd-tmpfiles-clean.timer - Daily Cleanup of Temporary Directories
     Loaded: loaded (/usr/lib/systemd/system/systemd-tmpfiles-clean.timer; static)
     Active: active (waiting) since Mon 2024-10-21 11:56:07 CEST; 4h 1min ago
      Until: Mon 2024-10-21 11:56:07 CEST; 4h 1min ago
    Trigger: Tue 2024-10-22 12:11:34 CEST; 20h left
   Triggers: ● systemd-tmpfiles-clean.service
       Docs: man:tmpfiles.d(5)
             man:systemd-tmpfiles(8)

Oct 21 11:56:07 redhat9.ittraining.loc systemd[1]: Started Daily Cleanup of Temporary>
```

Par contre le **status systemd-tmpfiles-clean.service** n'est ni activé, ni démarré :

```
[root@redhat9 ~]# systemctl status systemd-tmpfiles-clean.service
○ systemd-tmpfiles-clean.service - Cleanup of Temporary Directories
     Loaded: loaded (/usr/lib/systemd/system/systemd-tmpfiles-clean.service; static)
     Active: inactive (dead) since Mon 2024-10-21 12:11:34 CEST; 3h 53min ago
TriggeredBy: ● systemd-tmpfiles-clean.timer
       Docs: man:tmpfiles.d(5)
             man:systemd-tmpfiles(8)
    Process: 2174 ExecStart=systemd-tmpfiles --clean (code=exited, status=0/SUCCESS)
   Main PID: 2174 (code=exited, status=0/SUCCESS)
        CPU: 30ms

Oct 21 12:11:34 redhat9.ittraining.loc systemd[1]: Starting Cleanup of Temporary Dire>
Oct 21 12:11:34 redhat9.ittraining.loc systemd[1]: systemd-tmpfiles-clean.service: De>
```

```
Oct 21 12:11:34 redhat9.ittraining.loc systemd[1]: Finished Cleanup of Temporary Dire>
<code>

Ceci s'explique par la valeur du STATE du service qui est **static** :

<code>
[root@redhat9 ~]# systemctl list-unit-files | grep systemd-tmpfiles-clean.service
systemd-tmpfiles-clean.service             static          -
```

La valeur **static** implique que l'Unité ne peut ni être démarrée, ni être arrêtée par l'administrateur. Le démarrage et l'arrêt d'une telle Unité est effectué par le système. Ceci est visible par la valeur **TriggeredBy** dans la sortie suivante :

```
[root@redhat9 ~]# systemctl start systemd-tmpfiles-clean.service

[root@redhat9 ~]# systemctl status systemd-tmpfiles-clean.service
○ systemd-tmpfiles-clean.service - Cleanup of Temporary Directories
     Loaded: loaded (/usr/lib/systemd/system/systemd-tmpfiles-clean.service; static)
     Active: inactive (dead) since Mon 2024-10-21 16:19:45 CEST; 6s ago
TriggeredBy: ● systemd-tmpfiles-clean.timer
       Docs: man:tmpfiles.d(5)
             man:systemd-tmpfiles(8)
    Process: 5713 ExecStart=systemd-tmpfiles --clean (code=exited, status=0/SUCCESS)
   Main PID: 5713 (code=exited, status=0/SUCCESS)
        CPU: 25ms

Oct 21 16:19:45 redhat9.ittraining.loc systemd[1]: Starting Cleanup of Temporary Dire>
Oct 21 16:19:45 redhat9.ittraining.loc systemd[1]: systemd-tmpfiles-clean.service: De>
Oct 21 16:19:45 redhat9.ittraining.loc systemd[1]: Finished Cleanup of Temporary Dire>
```

Modifiez la valuer de **OnUnitActiveSec** en utilisant la commande suivante :

```
[root@redhat9 ~]# EDITOR=vim

[root@redhat9 ~]# export EDITOR
```

```
[root@redhat9 ~]# systemctl edit systemd-tmpfiles-clean.timer
### Editing /etc/systemd/system/systemd-tmpfiles-clean.timer.d/override.conf
### Anything between here and the comment below will become the new contents of the file

[Unit]
Description=Daily Cleanup of Temporary Directories
Documentation=man:tmpfiles.d(5) man:systemd-tmpfiles(8)
ConditionPathExists=!/etc/initrd-release

[Timer]
OnBootSec=15min
OnUnitActiveSec=30min



### Lines below this comment will be discarded

### /usr/lib/systemd/system/systemd-tmpfiles-clean.timer
# #  SPDX-License-Identifier: LGPL-2.1-or-later
# #
# #  This file is part of systemd.
# #
# #  systemd is free software; you can redistribute it and/or modify it
# #  under the terms of the GNU Lesser General Public License as published by
# #  the Free Software Foundation; either version 2.1 of the License, or
# #  (at your option) any later version.
#
# [Unit]
# Description=Daily Cleanup of Temporary Directories
# Documentation=man:tmpfiles.d(5) man:systemd-tmpfiles(8)
# ConditionPathExists=!/etc/initrd-release
#
# [Timer]
# OnBootSec=15min
# OnUnitActiveSec=1d
```

Utilisez la commande **systemctl daemon-reload** afin que systemd recharge la configuration :

```
[root@redhat9 ~]# systemctl daemon-reload
```

Visualisez le contenu du fichier

```
[root@redhat9 ~]# cat /etc/systemd/system/systemd-tmpfiles-clean.timer.d/override.conf
[Unit]
Description=Daily Cleanup of Temporary Directories
Documentation=man:tmpfiles.d(5) man:systemd-tmpfiles(8)
ConditionPathExists=!/etc/initrd-release

[Timer]
OnBootSec=15min
OnUnitActiveSec=30min
```

Constatez l'utilisation du fichier .override par le timer :

```
[root@redhat9 ~]# systemctl status systemd-tmpfiles-clean.timer
● systemd-tmpfiles-clean.timer - Daily Cleanup of Temporary Directories
     Loaded: loaded (/usr/lib/systemd/system/systemd-tmpfiles-clean.timer; static)
    Drop-In: /etc/systemd/system/systemd-tmpfiles-clean.timer.d
             └─override.conf
     Active: active (waiting) since Mon 2024-10-21 11:56:07 CEST; 4h 42min ago
      Until: Mon 2024-10-21 11:56:07 CEST; 4h 42min ago
    Trigger: Mon 2024-10-21 16:49:45 CEST; 10min left
   Triggers: ● systemd-tmpfiles-clean.service
       Docs: man:tmpfiles.d(5)
             man:systemd-tmpfiles(8)
             man:tmpfiles.d(5)
             man:systemd-tmpfiles(8)

Oct 21 11:56:07 redhat9.ittraining.loc systemd[1]: Started Daily Cleanup of Temporary Directories.
```

4.2 - Configurer le systemd-tmpfiles-clean.service

Les fichiers de configuration du service sont lus de trois emplacements différents et dans l'ordre suivant :

- **/etc/tmpfiles.d/*.conf**,
  - les fichiers dans ce répertoire configurent le service,
- **/run/tmpfiles.d/*.conf**,
  - les fichiers dans ce répertoire sont volatiles et sont utilisés par des daemons afin de gérer peurs propres fichiers temporaires,
- **/usr/lib/tmpfiles.d/*.conf**,
  - * les fichiers dans ce répertoire sont fournis par des paquets RPM. Il est **déconseillé** de les modifier.

```
[root@redhat9 ~]# ls -l /etc/tmpfiles.d/
total 4
-rw-r--r--. 1 root root 117 Aug 21 13:54 sos.conf

[root@redhat9 ~]# cat /etc/tmpfiles.d/sos.conf
# Ignore the contents of extracted sosreport directories.
#
# Type Path Mode User Group Age Argument
x /var/tmp/sos*
```

La signification de la letter **x** peut être consultée avec la commande suivante :

```
[root@redhat9 ~]# man tmpfiles.d
...
SYNOPSIS
       /etc/tmpfiles.d/*.conf
       /run/tmpfiles.d/*.conf
       /usr/lib/tmpfiles.d/*.conf

       ~/.config/user-tmpfiles.d/*.conf
       $XDG_RUNTIME_DIR/user-tmpfiles.d/*.conf
       ~/.local/share/user-tmpfiles.d/*.conf
```

```
...
/usr/share/user-tmpfiles.d/*.conf

#Type Path                                      Mode User Group Age         Argument
f     /file/to/create                           mode user group -           content
f+    /file/to/create-or-truncate               mode user group -           content
w     /file/to/write-to                         -    -    -     -           content
w+    /file/to/append-to                        -    -    -     -           content
d     /directory/to/create-and-cleanup          mode user group cleanup-age -
D     /directory/to/create-and-remove           mode user group cleanup-age -
e     /directory/to/cleanup                     mode user group cleanup-age -
v     /subvolume-or-directory/to/create         mode user group cleanup-age -
q     /subvolume-or-directory/to/create         mode user group cleanup-age -
Q     /subvolume-or-directory/to/create         mode user group cleanup-age -
p     /fifo/to/create                           mode user group -           -
p+    /fifo/to/[re]create                       mode user group -           -
L     /symlink/to/create                        -    -    -     -           symlink/target/path
L+    /symlink/to/[re]create                    -    -    -     -           symlink/target/path
c     /dev/char-device-to-create                mode user group -           major:minor
c+    /dev/char-device-to-[re]create            mode user group -           major:minor
b     /dev/block-device-to-create               mode user group -           major:minor
b+    /dev/block-device-to-[re]create           mode user group -           major:minor
C     /target/to/create                         -    -    -     cleanup-age /source/to/copy
x     /path-or-glob/to/ignore/recursively       -    -    -     cleanup-age -
X     /path-or-glob/to/ignore                   -    -    -     cleanup-age -
r     /empty/dir/to/remove                      -    -    -     -           -
R     /dir/to/remove/recursively                -    -    -     -           -
z     /path-or-glob/to/adjust/mode              mode user group -           -
Z     /path-or-glob/to/adjust/mode/recursively mode user group -           -
t     /path-or-glob/to/set/xattrs               -    -    -     -           xattrs
T     /path-or-glob/to/set/xattrs/recursively   -    -    -     -           xattrs
h     /path-or-glob/to/set/attrs                -    -    -     -           file attrs
H     /path-or-glob/to/set/attrs/recursively    -    -    -     -           file attrs
a     /path-or-glob/to/set/acls                 -    -    -     -           POSIX ACLs
```

```
       a+    /path-or-glob/to/append/acls             -    -    -     -           POSIX ACLs
       A     /path-or-glob/to/set/acls/recursively    -    -    -     -           POSIX ACLs
       A+    /path-or-glob/to/append/acls/recursively -    -    -     -           POSIX ACLs
...
```

Copyright © 2024 Hugh Norris.