Systemd Timers > Cron

By: Sean Twiehaus

Crontabs

```
# /etc/crontab: system-wide crontab
# Unlike any other crontab you don't have to run the `crontab'
# command to install the new version when you edit this file
# and files in /etc/cron.d. These files also have username fields,
# that none of the other crontabs do.

SHELL=/bin/sh
# You can also override PATH, but by default, newer versions inherit it from the environment
#PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# 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
17 *    * * *   root    cd / && run-parts --report /etc/cron.hourly
25 6    * * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47 6    * * 7   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52 6    1 * *   root    test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly )
```

Systemd Timer

```
# /lib/systemd/system/apt-daily.timer
[Unit]
Description=Daily apt download activities
ConditionACPower=true

[Timer]
Unit=apt-daily.service
OnCalendar=*-*-* 6,18:00
Persistent=true

[Install]
WantedBy=timers.target
```

Easily See ALL Jobs On A System

Easily See ALL Cron Jobs On A System

- /etc/crontab
- /etc/cron.d/*
- /etc/cron.hourly
- /etc/cron.daily
- /etc/cron.weekly
- /etc/cron.monthly
- /var/spool/cron/crontabs/*

Easily See ALL Cron Jobs On A System

```
user@hostname:~$ cat /etc/cron.d/*
30 3 * * 0 root test -e /run/systemd/system || SERVICE_MODE=1
/usr/lib/x86_64-linux-gnu/e2fsprogs/e2scrub_all_cron

10 3 * * * root test -e /run/systemd/system || SERVICE_MODE=1 /sbin/e2scrub_all -A -r
PATH=/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin

# TRIM the first Sunday of every month.
24 0 1-7 * * root if [ $(date +\%w) -eq 0 ] && [ -x /usr/lib/zfs-linux/trim ]; then
/usr/lib/zfs-linux/trim; fi

# Scrub the second Sunday of every month.
24 0 8-14 * * root if [ $(date +\%w) -eq 0 ] && [ -x /usr/lib/zfs-linux/scrub ]; then
/usr/lib/zfs-linux/scrub; fi
```

Easily See ALL Systemd Timers On A System

- systemctl list-timers --all

Easily See ALL Systemd Timers On A System

```
NEXT                         LEFT              LAST PASSED UNIT                         ACTIVATES
Wed 2024-01-10 05:31:50 CST 1min 28s left      n/a  n/a    update-notifier-download.timer update-notifier-download.service
Wed 2024-01-10 05:41:47 CST 11min left         n/a  n/a    systemd-tmpfiles-clean.timer   systemd-tmpfiles-clean.service
Wed 2024-01-10 06:42:29 CST 1h 12min left      n/a  n/a    apt-daily-upgrade.timer        apt-daily-upgrade.service
Wed 2024-01-10 07:57:01 CST 2h 26min left      n/a  n/a    motd-news.timer                motd-news.service
Wed 2024-01-10 08:32:34 CST 3h 2min left       n/a  n/a    apt-daily.timer                apt-daily.service
Thu 2024-01-11 00:00:00 CST 18h left           n/a  n/a    dpkg-db-backup.timer           dpkg-db-backup.service
Thu 2024-01-11 00:00:00 CST 18h left           n/a  n/a    logrotate.timer                logrotate.service
Thu 2024-01-11 05:12:08 CST 23h left           n/a  n/a    man-db.timer                   man-db.service
Sun 2024-01-14 03:10:47 CST 3 days left        n/a  n/a    e2scrub_all.timer              e2scrub_all.service
Mon 2024-01-15 00:16:10 CST 4 days left        n/a  n/a    fstrim.timer                   fstrim.service
Fri 2024-01-19 19:31:04 CST 1 week 2 days left n/a  n/a    update-notifier-motd.timer     update-notifier-motd.service
```

Maintain Less Shell Code

Randomized Startup Delay

Cron - Randomized Startup Delay

```
# sleep for a random interval of time (default 30min)
# (some code taken from cron-apt, thanks)
random_sleep()
{
   RandomSleep=1800
   eval $(apt-config shell RandomSleep APT::Periodic::RandomSleep)
   if [ $RandomSleep -eq 0 ]; then
       return
   fi
   if [ -z "$RANDOM" ] ; then
       # A fix for shells that do not have this bash feature.
       RANDOM=$(( $(dd if=/dev/urandom bs=2 count=1 2> /dev/null | cksum | cut -d' ' -f1) %
32767 ))
   fi
   TIME=$(($RANDOM % $RandomSleep))
   sleep $TIME
}
```

Systemd Timer - Randomized Startup Delay

```
# /lib/systemd/system/apt-daily.timer
[Unit]
Description=Daily apt download activities
ConditionACPower=true

[Timer]
OnCalendar=*-*-* 6:00
OnCalendar=*-*-* 18:00
RandomizedDelaySec=30m
Persistent=true

[Install]
WantedBy=timers.target
```

Maintain Less Shell Code

Conditionals

Cron - Only run when on AC Power

```
check_power()
{
    # laptop check, on_ac_power returns:
    #       0 (true)    System is on main power
    #       1 (false)   System is not on main power
    #       255 (false) Power status could not be determined
    # Desktop systems always return 255 it seems
    if command -v on_ac_power >/dev/null; then
        if on_ac_power; then
            :
        elif [ $? -eq 1 ]; then
            return 1
        fi
    fi
    return 0
}
```

Systemd Timer - Only run on AC Power

```
# /lib/systemd/system/apt-daily.timer
[Unit]
Description=Daily apt download activities
ConditionACPower=true

[Timer]
OnCalendar=*-*-* 6:00
OnCalendar=*-*-* 18:00
RandomizedDelaySec=30m
Persistent=true

[Install]
WantedBy=timers.target
```

Systemd Conditions

- ConditionArchitecture
- ConditionFirstBoot
- ConditionVirtualization
- ConditionHost
- ConditionNeedsUpdate
- ConditionPathExists
- …

Override Package Defaults

Cron - Override Package Defaults

Systemd Timers - Override Package Defaults

- Edit the timer
    - sudo systemctl edit apt-daily-upgrade.timer
- This will create: /etc/systemd/system/apt-daily-upgrade.timer.d/override.conf
- Supply Preferred Values

journalctl

Systemd Timers - Override Package Defaults

- systemctl list-timers
- journalctl -eu name.service