

Systemd

```
systemctl status
```

show running units

```
systemctl
```

show available unit files

```
systemctl list-unit-files
```

enable unit

```
systemctl enable name.service
```

start unit

```
systemctl start name
```

after change unit file

```
systemctl daemon-reload
```

Systemd as user

enable systemd for specific user

```
loginctl enable-linger <username>
```

- add user to systemd-journal group to permit journalctl operations `usermod -a -G systemd-journal <user> </code>`
- modify `/etc/systemd/journald.conf` `[Journal] Storage=persistent </code>`

[/etc/profile.d/systemd-user.sh](#)

```
export XDG_RUNTIME_DIR="/run/user/$UID"  
export DBUS_SESSION_BUS_ADDRESS="unix:path=${XDG_RUNTIME_DIR}/bus"
```

journalctl -u does not work. Use instead

```
journalctl --user --user-unit
```

Systemd Timer

Una valida e più potente alternativa a cron, integrata in systemd, è rappresentata dai timers, di primo impatto soprattutto nell'aspetto supervisivo tramite journalctl, ma non solo (vedi risposta a questo quesito <http://unix.stackexchange.com/questions/278564/cron-vs-systemd-timers>).

Per attivare un timer servono essenzialmente due cose:

- Il timer: `/etc/systemd/system/nms-restart.timer` [Unit] Description=Perform a nms restart

```
[Timer] OnCalendar=hourly
```

```
[Install] WantedBy=timers.target
```

- il servizio: `/etc/systemd/system/nms-restart.service`

```
[Unit]
Description=Perform a nms restart

[Service]
Type=simple
Nice=19
IOSchedulingClass=2
IOSchedulingPriority=7
ExecStart=/usr/local/sbin/nms-restart
```

Poi si avvia e abilita il timer:

```
sudo systemctl start nms-restart.timer
sudo systemctl enable nms-restart.timer
```

Per la lista dei timer

```
systemctl list-timers --all
```

Per monitorare si usa journalctl in tutte le sue forme, per esempio:

```
journalctl -f -u nms-restart
```

From:
<https://wiki.csgalileo.org/> - **Galileo Labs**

Permanent link:
<https://wiki.csgalileo.org/zibaldone/linux/systemd?rev=1633697638>

Last update: **2021/10/08 14:53**

