

Linux - systemd

Radim Kliment

systemd - 😈

- Výchozí init systém a správce služeb pro většinu moderních linuxových distribucí
 - Příliš rozsáhlý systém
- Spravuje spouštění systému, služby a procesy
- Alternativy
 - runit, OpenRC, sysvinit



Základní rozdělení

- Jednotka (Unit): Konfigurační soubor (např. `.service`, `.timer`), který definuje, jak `systemd` spravuje určitý zdroj
- Služba (Service): Pozadí procesu (např. webový server, databáze), který je spravován systémem `systemd`

Vybrané unit typy

- .service - long running process
- .socket - start na vyžádání (sshd)
- .path - monitorování změn filesystemu
- .timer - cron
- ...

systemctl

- `sudo systemctl start nginx`
- `sudo systemctl stop nginx`
- `sudo systemctl restart nginx`
- `sudo systemctl reload nginx`
- `sudo systemctl enable nginx`
- `sudo systemctl disable nginx`

Logy - journalctl

- journalctl
 - all logs
- journalctl -f
 - follow tail
- journalctl -u nginx
 - filter by service/unit
- journalctl -p err

Service - /etc/systemd/system/myscript.service

[Unit]

Description=My Custom Script

After=network.target

[Service]

ExecStart=/usr/local/bin/myscript.sh

Restart=always

User=root

[Install]

WantedBy=multi-user.target

[Unit]

- **Requires**
 - Tvrdá závislost. Pokud selže uvedená jednotka, selže i tato jednotka.
- **Wants**
 - Měkká závislost. Pokud selže uvedená jednotka, tato jednotka pokračuje.
- **Before=: Opak k After=.**
 - Tato jednotka se musí spustit před uvedenou jednotkou.
- **Conflicts=:**
 - Pokud je uvedená jednotka aktivní, tato jednotka nemůže běžet (a naopak).
- **BindsTo=:**
 - Podobné jako Requires=, ale navíc zastaví tuto jednotku, pokud se zastaví závislost.

[Service]

- ExecStop=: Command to run when the service stops (e.g., cleanup).
- ExecReload=: Command to run when the service is reloaded (e.g., nginx -s reload).
- WorkingDirectory=: Set the working directory for the service.
- Environment=: Set environment variables (e.g., Environment="VAR=value").

Timer - cron like

/etc/systemd/system/**backup**.service

/etc/systemd/system/**backup**.timer

[Unit]

Description=Denní zálohování v 14:00

[Timer]

OnCalendar=*-*-* 14:00:00

Persistent=true

[Install]

WantedBy=timers.target

OnCalendar

*

*_*_*

..*
.*.*

#Day Of the week

Year-Month-Date

Hour:Minute:Second

Every Minute: *_*_* *.*:00

Every 2 minute: *_*_* *.* /2:00

Every Day at 1am: *_*_* 01:00:00

monday to friday: Mon...Fri *_*_* 00:00:00

Every weekend: Sat,Sun *_*_* 00:00:00

Start timer

```
sudo systemctl daemon-reload
```

```
sudo systemctl enable backup.timer
```

```
sudo systemctl start backup.timer
```

```
systemctl list-timers
```

Úkol 1: Řešení problémů se službou

- Zkuste zjistit stav služby *vymysleny-service*
 - `sudo systemctl status vymysleny-service`
- Vytvořte falešnou službu `/etc/systemd/system/vymysleny-service.service` s neexistujícím skriptem
 - `/home/student/Documents/run.sh`
- Zkuste službu spustit a zjistěte chybu
 - `sudo systemctl daemon-reload`
 - `sudo systemctl start vymysleny-service`
 - `sudo systemctl status vymysleny-service`
 - `sudo journalctl -u vymysleny-service -n 10`
- Opravte chybu
 - vytvořte skript `/home/student/Documents/run.sh`

Úkol 2: Vytvoření vlastní služby

Vytvořte jednoduchou službu, která spustí skript, který vypíše aktuální datum a čas do souboru `/tmp/muj_log.txt` každých 10 sekund.

Úkol 3: Vytvoření vlastního timeru

Vytvořte service a k ni timer, který každou minutu otestuje pomocí **curl**, **grep** zda se na webové stránce objevil text “kočka”, výsledek zaloguje do stdout.

otestujte logy pomocí **journalctl**

<https://gist.albertov.cloud/radimkliment/88c3d4501ec84a5eb44e8cace575c16d/raw/HEAD/cron.txt>