

Plánování úloh v cronu

RADIM VACULÍK

Co je to cron

- Jedná se o systémového démona, který se v Linuxových a Unixových systémech stará o automatické spouštění naplánovaných úloh
- Běží neustále na pozadí a každou minutu kontroluje, jestli má něco spustit podle naplánovaných pravidel
- Ideální pro opakované úlohy, které se mají spouštět ve specifický čas nebo interval

K čemu se používá

- Zálohování souborů
- Čištění logů nebo cache
- Automatické aktualizace
- Odesílání e-mailových reportů
- Spouštění skriptů nebo serverových úloh
- Restartování služeb



Kde se konfiguruji úlohy a jejich zobrazení

- Konfigurace:
 - Pro běžného uživatele: `crontab -e` (uživatelský plán)
 - Systémové úlohy: `/etc/crontab` nebo `/etc/cron.d/`
 - Speciální složky jako: `/etc/cron.hourly/`, `/etc/cron.daily/`, atd.
- Zobrazení:
 - Osobního seznamu: `crontab -l`
 - Systémového seznamu: `cat /etc/crontab`

Zápis úloh

- Po otevření seznamu s příkazem `crontab -e` se zobrazí následující upravitelný soubor.
- Každý řádek má tento formát:
`* * * * * příkaz/skript`
- Každá hvězdička/asterisk představuje čas:
(minuty 0-59, hodiny 0-23, den v měsíci 1-31, měsíc 1-12, den v týdnu 0-7)
- Pokud není zadáný určitý čas tak se opakuje každý interval (hodinu, den, ...)

```
GNU nano 7.2
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h  dom mon dow  command
* * * * * /home/radim/hello.sh
```

Příklad 1. (jednoduchý výpis)

- Vytvoříme jednoduchý skriptový soubor např. "Hello.sh" a do něj vložíme následující příkaz:

```
echo "Skript byl aktivován $(date)" > /home/uzivatel/hello.txt
```

- Nesmíme zapomenout na to aby byl soubor spustitelný:

```
chmod +x hello.sh
```

- Následně přejdeme do konfigurace a nastavíme aby se skript spouštěl každou minutu

```
* * * * * /home/uzivatel/hello.sh
```

- Nebo každý týden v poledne

```
0 12 * * 1 /home/uzivatel/hello.sh
```

Příklad 2.

(zálohování)

- Vytvoříme soubor `zaloha.sh` která bude obsahovat následující skript (viz. screenshot)
- Proměnné `ZDROJ` a `CIL` odkazují na složky
- V proměnné `DATUM` bude datum ve formátu: (rok, měsíc, den, hodina a minuta)
- Pomocí `tar -czf` vytvoříme a zazipujeme soubor.
- Pokud by jsme měli zálohy starší jak 7 dní tak je smažeme
- Následně zajistíme aby byl soubor spustitelný: `chmod +x zaloha.sh` a přidáme do cronu
- `* */1 * * * /home/uzivatel/zaloha.sh` - spouštění každou hodinu

```
# Zdrojová a cílová složka
ZDROJ="/home/radim/dokumenty"
CIL="/home/radim/zalohy"

# Vytvoř název souboru podle aktuálního data
DATUM=$(date +%Y-%m-%d_%H-%M)
ZALOHA="$CIL/zaloha_${DATUM}.tar.gz"

#provede zálohu
tar -czf "$ZALOHA" "$ZDROJ"

# smaže zálohy starší než 7 dní
find "$CIL" -name "zaloha_*.tar.gz" -mtime +7 -delete
```

Děkuji za pozornost



zdroje

- ♦ https://mis.e-mis.cz/index.php/Plánování_úloh#Zobrazen.C3.AD_seznamu_pl.C3.A1novan.C3.BDch_.C3.BAloh
- ♦ <https://chatgpt.com>
- ♦ <https://www.webglobe.cz/poradna/co-je-cron>