

Penjadualan Tugas

Disusun oleh : Henry Saptono

Diedit oleh : Dudi Fitriahadi

Apa itu penjadualan tugas?

Penjadualan tugas (*task scheduling*) adalah sebuah metode otomatisasi tugas (*task*) agar dilaksanakan oleh sistem komputer, bukan oleh manusia.

Mengapa perlu penjadualan tugas?

- Beberapa tugas (*task*) perlu dijalankan secara berkesinambungan (rutin) di masa yang akan datang
- Manusia tidak dapat konsisten terhadap waktu
- Menyederhanakan tugas manusia menjadi lebih efisien dan efektif
- Otomatisasi tugas

Syarat penjadualan tugas

- Tidak semua tugas (*task*) dapat dijadualkan untuk dijalankan secara otomatis oleh sistem
- Hanya tugas (*task*) yang tidak interaktif yang dapat dijadualkan (diotomatisasi)

Sistem penjadualan tugas

Sistem penjadualan tugas adalah sebuah perangkat lunak yang berperan sebagai sebuah service (daemon), yang berjalan secara terus menerus pada sistem komputer untuk mengeksekusi atau menjalankan tugas-tugas (*tasks*) yang telah dijadualkan oleh manusia untuk dijalankan di masa yang akan datang sesuai dengan waktu yang telah dijadualkan, baik rutin, periodik ataupun hanya sekali.

Aplikasi penjadualan tugas

- At → hanya sekali di waktu yang akan datang
- Cron → rutin (kontinyu)
- Anacron → periodik

at

- Digunakan untuk menjalankan tugas hanya satu kali pada waktu yang telah ditentukan
- Paket at harus sudah terinstal, periksa dengan mengetikkan perintah :

```
$ rpm -q at
```

- daemon/service atd harus sudah jalan, periksa dengan mengetikkan perintah :

```
# service atd status
```

Sintaks at

at waktu

spesifikasi waktu:

- format HH:MM, dimana HH adalah jam dan MM adalah menit
- format MMDDYY, MM/DD/YY, MM.DD.YY, dimana MM adalah bulan, DD adalah tanggal dan YY adalah tahun. Contoh: 041510 berarti tanggal 15 April 2010.
- frase midnight (12:00 a.m.), noon (12:00 p.m.), teatime (4:00 p.m.)

Sintaks at

- frase now, today, tomorrow
- now + waktu, dimana waktu dalam minutes, hours, days atau weeks. Contoh: now + 5 days berarti waktu yang sama lima hari dari sekarang
- Selengkapnya dapat dilihat di /usr/share/doc/at-<versi>/timespec

Contoh Penggunaan

```
$ at now +2 minute
warning: commands will be executed using /bin/sh
at> echo Hallo | wall
at> <EOT>
job 10 at Sat Oct 19 11:00:00 2019
$
```

```
Broadcast message from dudi@dudi-Aspire
(somewhere) (Sat Oct 19 11:12:00 2019):
```

```
hallo
```

Ijin Akses at

- `/etc/at.allow` dan `/etc/at.deny` digunakan untuk membatasi penggunaan perintah `at`.
- Format yang digunakan adalah satu user per baris.
- jika `at.allow` ada, hanya user yang terdaftar pada `at.allow` yang diperbolehkan menggunakan `at`. `at.deny` akan diabaikan.

Ijin Akses at

- Jika `at.allow` tidak ada, maka user yang terdaftar di `at.deny` tidak diperbolehkan menggunakan `at`.
- Jika `at.deny` ada dan kosong, maka semua user diperbolehkan menggunakan `at`.
- Jika `at.allow` dan `at.deny` tidak ada maka semua user kecuali root tidak diperbolehkan menggunakan `at`.

cron

- Digunakan untuk menjalankan tugas atau program secara berulang sesuai dengan jadwal yang telah ditentukan
- Paket cron harus sudah terinstal, periksa dengan mengetikkan perintah:

```
# dpkg -l cron
```

- daemon/service cron harus sudah jalan, periksa dengan mengetikkan perintah :

```
# systemctl status cron
```

File Konfigurasi cron

- File /etc/crontab
- Direktori /etc/cron.d/
- File /var/spool/cron/nama_user

/etc/crontab

- Merupakan file konfigurasi utama cron
- Setiap baris pada file /etc/crontab merupakan tugas yang akan dijalankan

Format file /etc/crontab

minute hour day month dayofweek command

Keterangan:

- minute : angka 0-59, menyatakan menit
- hour : angka 0-23, menyatakan jam
- day : angka 1-31, menyatakan tanggal
- month : angka 1-12 atau singkatan nama bulan, misal: jan, feb, dll.
- dayofweek : angka 0-7 (0 atau 7 berarti Sunday) atau singkatan nama hari, misal: sun, mon, dll.
- command : perintah atau program yg akan dijalankan

Format file /etc/crontab

- Asterisk (*) dapat digunakan untuk menyatakan semua nilai yang valid. Misal, tanda * pada kolom month berarti tugas akan dijalankan setiap bulan
- Hyphen (-) digunakan untuk menyatakan range angka. Misal, 1-4 artinya 1 sampai dengan 4 atau 1, 2, 3 dan 4
- Comma (,) menyatakan dan. Misal, 1,4 pada kolom day artinya tugas akan dijalankan setiap tanggal 1 dan tanggal 4

Format file /etc/crontab

- Slash (/) digunakan untuk melompati nilai tertentu. Misal, */5 pada kolom minute artinya tugas akan dijalankan setiap 5 menit
- Hash mark (#) menyatakan komentar dan tidak akan diproses

Contoh Format Waktu

Format	Keterangan
0 0 1 1 *	Run once a year
0 0 1 * *	Run once a month
0 0 * * 0	Run once a week
0 0 * * *	Run once a day
0 * * * *	Run once an hour

Mengelola File Crontab User

- Mengedit file crontab

```
$ crontab -e
```

- Melihat isi file crontab

```
$ crontab -l
```

- Menghapus isi file crontab

```
$ crontab -r
```

Ijin Akses cron

- `/etc/cron.allow` dan `/etc/cron.deny` digunakan untuk membatasi penggunaan cron.
- Format yang digunakan adalah satu user per baris
- Jika `cron.allow` ada, hanya user yang terdaftar pada `cron.allow` yang diperbolehkan menggunakan cron. `cron.deny` akan diabaikan.

Ijin Akses cron

- Jika cron.allow tidak ada, maka user yang terdaftar di cron.deny tidak diperbolehkan menggunakan cron.
- Jika cron.deny ada dan kosong, maka semua user diperbolehkan menggunakan cron.
- Jika cron.allow dan cron.deny tidak ada maka semua user kecuali root tidak diperbolehkan menggunakan cron.

anacron

- anacron digunakan untuk menjalankan tugas pada komputer yang tidak selalu nyala
- anacron menggunakan skala waktu relatif, misalnya sekali seminggu atau sekali sebulan
- Paket anacron harus sudah terinstal, periksa dengan mengetikkan perintah:

```
$ dpkg -l anacron
```

- daemon/service anacron harus sudah jalan, periksa dengan mengetikkan perintah :

```
# systemctl status anacron
```

File Konfigurasi anacron

/etc/anacrontab

Format file:

period delay job-identifier command

Keterangan:

- period : dinyatakan dalam hari
- delay : dinyatakan dalam menit
- job-identifier : merupakan identifikasi yang akan digunakan dalam time-stamp dan pesan
- command : perintah atau program yang akan dijalankan

Contoh Penggunaan

```
$ 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
1      65      cron.daily run-parts /etc/cron.daily
7      70      cron.weekly run-parts /etc/cron.weekly
30     75      cron.monthly run-parts /etc/cron.monthly
1      5       dudi echo hallo | wall
```