

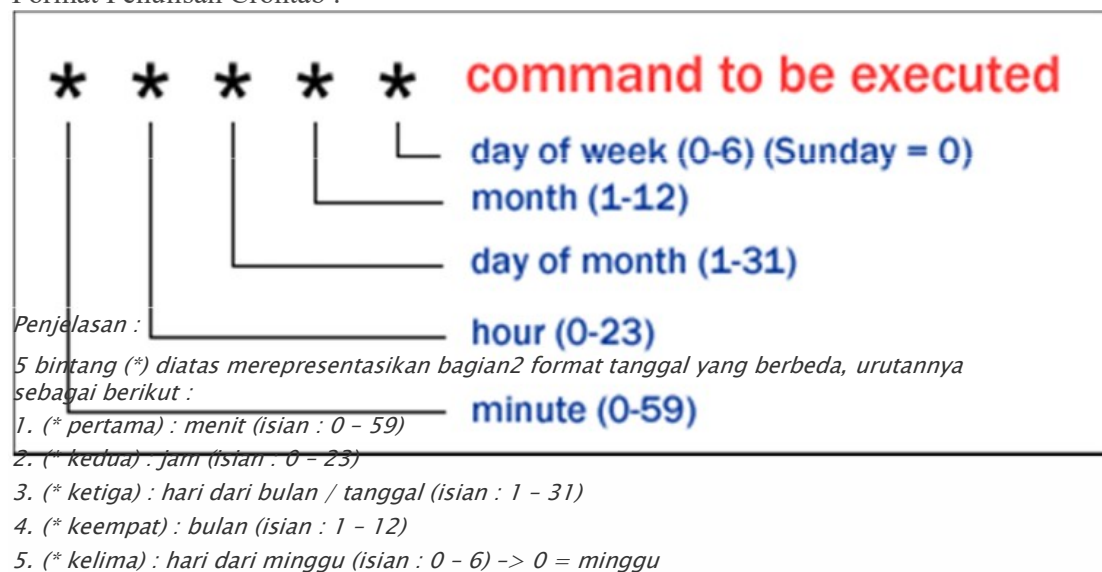
## PENJADWALAN OTOMATIS DI LINUX

Pada sistem operasi berbasis Linux terdapat fasilitas perintah yang dapat melakukan penjadwalan secara otomatis. Di linux dikenal dua perintah penjadwalan yaitu Crontab dan At.

### Crontab

Crontab adalah fasilitas yang tersedia di sistem operasi Unix untuk menjalankan sebuah perintah secara berkala. Frekuensi waktu terkecilnya adalah 1 menit sekali. Frekuensi terbesarnya hingga beberapa tahun sekali. Dapat juga dipilih ingin menjalankan setiap hari tertentu, setiap berapa jam sekali, pada menit beberapa, dst.

Format Penulisan Crontab :



Format penulisan crontab dan artinya :

```
* * * * * <command> #Dijalankan setiap menit
```

Tanda bintang (\*) bisa dikombinasikan atau diganti dengan:

1. tanda (/) : artinya setiap kali sesuai dengan posisi tanda (\*) , kalau di posisi paling kiri \*/30 \* \* \* \* berarti setiap 30 menit.

2. tanda (,) : artinya bisa menjalankan dalam beberapa kali rentang waktu, misalnya \* 1,3 \* \* \* atau setiap pukul 01.00 dan 03.00.
3. tanda (-) : artinya menjalankan selama rentang waktu tersebut, misalnya \* \* \* \* 2-4 atau setiap hari kedua sampai hari keempat dalam satu minggu.

#### **Contoh implementasi crontab:**

##### **> Eksekusi setiap menit**

```
* * * * * /home/sigit/script.sh
```

##### **> Eksekusi 5 menit**

```
* /5 * * * * /home/sigit/script.sh
```

Berikut adalah contoh pemakaian tanda-tanda diatas :

##### **> Eksekusi setiap 10 menit (tiap hari)**

Penulisannya dapat dilakukan dengan :

```
0,10,20,30,40,50 * * * * /home/sachty/run.sh
```

dan perintah diatas bisa di persingkat dengan :

```
*/10 * * * * /home/sachty/run.sh
```

##### **> Eksekusi setiap Sabtu jam 12 siang :**

```
0 12 * * 6 /home/sigit/script.sh
```

(\* pertama) artinya pada menit ke 0, (\* kedua) artinya pada jam ke 12, (\* ketiga) artinya pada tanggal berapapun, (\* keempat) artinya pada setiap bulan, (\* kelima) artinya pada setiap hari sabtu (6 = sabtu).

##### **> Eksekusi pada jam 9 malam setiap hari kerja (Senin – Jumat) :**

```
0 21 * * 1-5 /home/sigit/script.sh
```

Arinya mirip-mirip ama diatas, cuma (\* kelima) diisi rentang, yaitu 1-5 (artinya senin-jumat)

##### **> Eksekusi setiap menit ke 30 (setiap jam) pada setiap tanggal 1 (setiap bulan)**

```
30 * 1 * * /home/sigit/script.sh
```

##### **> Eksekusi setiap 10 menit (tiap hari), penulisannya bisa seperti ini :**

```
0,10,20,30,40,50 * * * * /home/sigit/script.sh
```

Tapi ada trik penulisan yang lebih gampang, caranya seperti ini :

##### **> Atau menggunakan cari sbb : (eksekusi setiap 10 menit**

```
*/10 * * * * /home/sigit/script.sh
```

arti dari (\* /10) adalah setiap 10 menit

##### **> kode-kode special**

crontab menyediakan kode-kode special untuk digunakan, berikut kodenya :

```
@reboot      Run once, at startup
@yearly      Run once a year      "0 0 1 1 *"
@annually    (same as @yearly)
@monthly     Run once a month     "0 0 1 * *"
@weekly      Run once a week      "00**0"
@daily       Run once a day       "0 0 ***"
@midnight    (same as @daily)
@hourly      Run once an hour     "0****"
```

Cara pemakaian :

```
@daily /home/sigit/script.sh
```

### Menyimpan log crontab

Supaya kita bisa memonitor pekerjaan crontab kita, hasil log bisa disimpan dalam file, caranya seperti berikut :

```
@daily /home/sigit/script.sh 2>&1 >> /home/sigit/log/crontab_schedule.log
```

### Mengirim log ke email (harus install mailx dulu), caranya :

```
@daily /home/sigit/script.sh 2>&1 | mail -s "Cronjob Log sgt.wibowo@gmail.com
```

Selain itu bisa terdapat kode special @ yang mempunyai persamaan dengan perintah tertentu sbb :

Entry	Keterangan
@reboot	Run satu kali, saat s
@yearly	Run setahun sekali
@annually	(sama dengan @year
@monthly	Run sebulan sekali
@weekly	Run seminggu sekal

Berikut adalah cara pemakaian perintah @ :

Cara pemakaian :

```
@daily /home/sigit/script.sh
```

sama dengan perintah :

```
0 0 * * * /home/sigit/script.sh
```

Perintah-perintah yang ada pada crontab :

```
# crontab -e - Mengedit file crontab yang sudah ada, atau membuat file baru.
# crontab -l - Menampilkan isi file crontab.
# crontab -r - Menghapus file crontab.
# crontab -u - Mengedit crontab milik user.
```

### Menyimpan log crontab

Supaya kita bisa memonitor pekerjaan crontab kita, hasil log bisa disimpan dalam file, caranya seperti berikut :

```
@daily /home/sigit/script.sh 2>&1 >> /home/sigit/log/crontab_schedule.log
```

## PERCOBAAN CRONTAB

1. Lakukan instalasi crontab  
#apt-get install cron

2. Lakukan edit crontab  

```
# crontab -e
```
3. Tambahkan baris berikut ini :  

```
* * * * * /usr/bin/wall "Latihan Crontab"
```
4. Keluar dan simpan serta restart crontab dengan perintah sbb :  

```
#!/etc/init.d/crontab restart
```
5. Amati hasilnya di terminal apa hasilnya ?
6. Jalan perintah berikut :
  - a. Buat shell script untuk menangkap tanggal jam di linux dimasukkan ke dalam suatu file dan jalankan disetiap 2 menit sekali.
  - b. Jalankan script tersebut dan pastikan ada hasilnya
  - c. Tambahkan dicrontab untuk menjalankan bash script tersebut tiap menit, simpana hasilnya di log. Lihat kembali di dasar teori cara menyimpan log di crontab
  - d. Amati hasil lewat file log yang sudah kamu buat. Catat hasilnya
7. Jalan perintah berikut :
  - a. Buatlah percobaan 6 dijalankan secara otomatis setiap kelipatan 3 menit
  - b. Catat bagaimana script yang terbentuk
8. Latihan 4
  - a. Buatlah percobaan 6 dijalankan secara otomatis setiap jam 11.15
  - b. Catat bagaimana script yang terbentuk
9. Buat script shell untuk melihat user yang login dan masukkan ke dalam suatu file dimana filenya isi file sebelumnya tidak boleh hilang dan jalankan di crontab secript tsb setiap hari Selasa jam 11.20 menit
10. Monitor setiap pekerjaan crontab anda di /home/student/log/crontab\_schedule.log
11. Buat disk baru di vm, jangan terlalu besar misal 100MB. Format dan lakukan mounting disk baru ini. Isi dengan beberapa data dan lakukan backup terhadap disk ini dengan perintah dd. Taruh hasilnya di direktori backup1.  
  
Buatlah backup pada crontab untuk dijalankan secara periodik, lihat hasilnya melalui log maupun direktori hasil backup
12. Catat seluruh percobaan anda