

Monitoring Suhu Komputer

Suhu yang terlampau tinggi terkadang menjadi penyebab suatu sistem gagal bekerja. Lebih parah lagi, *hardware* sistem bisa ikut rusak. Di artikel ini, kita akan mengatur langkah antisipatif dengan memeriksa suhu komputer secara berkala dan memberikan aksi yang bersesuaian.

Komputer adalah seperti manusia. Bisa kepanasan dan berhenti bekerja apabila kepanasan. Dan, toleransi untuk berhenti bekerja pada komputer sangatlah rendah. Apalagi untuk sebuah server yang dituntut untuk memiliki *uptime* lebih dari 99,9%. Betul bahwa kita bisa membeli server besar yang secara *hardware* telah sangat hebat. Tapi, tidak semua dari kita mampu membeli server *branded* yang mahal. Dan, sejujurnya, server *branded* yang mahal tidaklah selalu diperlukan. Terutama bagi perusahaan kecil sampai menengah.

Bagi Anda yang menggunakan server rakitan yang secara *hardware* sama seperti komputer desktop, sekaligus berada di dalam ruang kerja tidak berpendingin ruangan, namun harus tetap mampu melayani, maka ada baiknya bagi Anda untuk mampu memelihara komputer tersebut secara *software*.

Kepanasan merupakan salah satu faktor yang sangat umum dalam membuat komputer berhenti bekerja ataupun membuat kerusakan permanen pada *hardware*. Tidak ada satu pun sistem operasi yang mampu menahan kerusakan pada *hardware*. Yang paling bisa kita lakukan adalah membuat sistem berhenti bekerja sebelum kepanasan mencapai batas yang tidak dapat ditoleransi.

Sebenarnya, solusi dengan membuat sistem berhenti bekerja sebelum kepanasan seharusnya disisipkan pada level kernel, sehingga bisa lebih responsif dan memiliki hak yang lebih tinggi. Namun, hal tersebut jauh lebih susah untuk dilakukan.

Sebagai gantinya, kita akan membuat aplikasi user space dengan memanfaatkan shell scripting. Prinsip kerjanya sangat sederhana. Kita akan menjalankan suatu program secara berkala (misal, setiap me-

nit), dimana di dalam program tersebut, kita memeriksa suhu sistem dan mengambil tindakan yang bersesuaian dengan suhu yang didapat tersebut. Untuk menjalankan program secara berkala, kita akan menggunakan cron. Sementara, untuk mengambil suhu sistem, Anda bisa menggunakan salah satu dari program-program berikut:

- **ACPID.** Sangat fleksibel dan ekstensible dalam menangani event ACPI. Daemon ini akan membaca `/proc/acpi/event` dan menjalankan aksi yang bersesuaian. Sayangnya, tidak semua sistem mendukung *acpi*.
- **Xmbmon.** Alternatif ACPID yang sangat menyenangkan. Kita akan menggunakan program yang satu ini dalam tulisan kali ini. Sayangnya, ada beberapa sistem yang tidak dikenal oleh *xmbmon* sehingga aplikasi ini tidak dapat digunakan sebagai alat bantu. Dari namanya, kita bisa menduga bahwa program ini bisa dijalankan dari X. Hal tersebut tidak kita inginkan. Untungnya, pembuatnya juga membuat versi konsolnya (*mbmon*), yang akan kita gunakan.

Untuk tindakan yang akan diambil apabila suhu mencapai batas yang diinginkan, secara default adalah *shutdown*, yang berarti, kita akan memanfaatkan program *shutdown* (`/sbin/shutdown`) yang telah dimiliki oleh semua sistem. Namun, bagi Anda yang ingin pendekatan lain, *software suspend* (`/sbin/swsusp`) bisa dipergunakan apabila tindakan yang akan diambil adalah melakukan *suspend* secara *software*. Untuk melakukan *suspend*, pastikan telah didukung oleh kernel.

Untuk *men-shutdown* sistem, tentu saja program monitoring harus dijalankan oleh root. Bagi Anda yang tidak ingin menjalankan sebagai root, *sudo* bisa digunakan.

Tulisan ini akan mengasumsikan Anda menjalankan program sebagai root.

Cron

Berikut ini, kita akan melihat dasar-dasar penggunaan cron secara sederhana, untuk menjalankan aplikasi kita secara berkala. Tergantung dari distribusi yang Anda gunakan, umumnya, cron daemon sudah dijalankan pada saat booting. Pastikan *crond* sudah dijalankan karena kita akan membutuhkannya.

Setelah itu, login-lah sebagai root, dan berikan perintah berikut ini untuk membuat entri tabel cron yang baru:

```
# crontab -e
```

Apabila belum ada entri sama sekali, maka Anda akan mengedit file kosong. Untuk membuat entri tabel yang baru yang memuat contoh tugas yang akan dijalankan setiap menit, cobalah memberikan baris berikut ini:

```
* * * * * /bin/echo `date` >> /tmp/DATE
```

Contoh entri tersebut akan menulis jam dan tanggal aktif ke `/tmp/DATE` setiap menit. Untuk menambahkan entri tabel *crontab* yang baru, Anda bisa membuatnya di baris baru.

Cron sendiri memiliki lima field yang pada contoh sebelumnya kita isi dengan `*`. Berikut ini adalah penjelasan kelima field yang digunakan:

- Field pertama adalah menit beberapa. Nilai yang diizinkan adalah 0-59. Kita bisa mengisi nilai `*` yang berarti awal sampai akhir.
- Field kedua adalah jam beberapa. Nilai yang diizinkan adalah 0-23.
- Field ketiga adalah hari beberapa dalam satu bulan. Nilai yang diizinkan adalah 1-31.

- Field keempat adalah bulan keberapa dalam satu tahun. Nilai yang diizinkan adalah 1-12.
- Field kelima adalah hari dalam satu minggu. Nilai yang diizinkan adalah 0-7. Nilai 0 atau 7 adalah minggu (sun).

Sebagai contoh, apabila kita ingin menjalankan suatu tugas setiap jam 21.15 setiap hari, maka berikanlah entri tabel berikut:

```
15 21 * * * /path/to/app
```

xmbmon

Berikutnya, kita akan melihat penggunaan program xmbmon. Versi xmbmon yang digunakan adalah 2.0.5. Program ini dibuat oleh Yoshifumi R. Shimizu (<http://www.nt.phys.kyushu-u.ac.jp/shimizu/index.html>). Distribusi xmbmon umumnya berada dalam format source code. Untuk itu, Anda perlu melakukan konfigurasi dan kompilasi sebelum bisa digunakan:

```
$ ./configure
$ make
```

Setelah kompilasi selesai, tergantung konfigurasi sistem Anda, Anda mungkin akan menjumpai program xmbmon dan mbmon di direktori aktif. Yang kita perlukan adalah program mbmon.

Untuk mendapatkan suhu sistem, jalankanlah mbmon tanpa parameter. Sebagai contoh:

```
# ./mbmon
```

```
Temp.= 44.8, 32.1, 23.8;
```

```
Rot.= 0, 0, 0
```

```
Vcore = 1.54, 3.19;
```

```
Volt. = 3.26, 4.98, 12.01,
0.00, 0.00
```

```
Temp.= 44.8, 32.1, 23.8;
```

```
Rot.= 0, 0, 0
```

```
Vcore = 1.54, 3.19;
```

```
Volt. = 3.26, 4.98, 12.01,
0.00, 0.00
```

```
Temp.= 44.8, 32.1, 23.8;
```

```
Rot.= 0, 0, 0
```

```
Vcore = 1.54, 3.19;
Volt. = 3.26, 5.00, 12.01,
0.00, 0.00
```

Secara default, mbmon akan menampilkan informasi suhu sistem dan informasi lainnya setiap lima detik sekali. Dalam aplikasi yang akan kita buat, kita perlu meminta kepada mbmon untuk menjalankan hanya sekali tanpa perulangan. Untuk itu, kita perlu memberikan opsi -c1 pada saat menjalankan:

```
# ./mbmon -c1
```

```
Temp.= 43.7, 30.8, 23.8;
```

```
Rot.= 0, 0, 0
```

```
Vcore = 1.54, 3.20;
```

```
Volt. = 3.26, 4.98, 12.01,
0.00, 0.00
```

Informasi yang kita dapatkan terlalu lengkap. Bagaimana kalau kita hanya ingin mengambil suhu sistem saja? Jalankanlah mbmon dengan tambahan opsi -T1:

Professional 100% Linux Training & Solution

Ingin Menguasai Linux Secara LENGKAP ?!

Paket A-Z Linux

- Linux Concept & Fundamental
- Linux System Administration
- Linux Internet + Intranet Server
- Linux Security

56 hours (14 day @ 4 hours)

Only : Rp.4.850.000,-

Special Offer Crash Programme !

PATIN (Paket Intensif)

- Linux Concept and Fundamental
- Linux System Administration
- Linux Internet + Intranet Server

42 hours (6 days@ 7 hour)

Only : Rp.3.750.000,-

PAKIS (Paket Ekonomis)

- Linux Concept and Fundamental
- Linux System Administration
- Linux Internet + Intranet Server.

44 hours (11 days@ 4 hour)

Only : Rp.3.650.000,-

Ketik: Info PATIN atau Info PAKIS kirim SMS ke 0856 7771030 SMS Server powered by eSMSis (www.eSMSis.com)

SMS Server & Gateway

eSMSis Ver. 1.5

- Web based, Internet Ready
- Broadcast, GroupCast, MemberCast
- Scheduled SMS, Product Informations
- Auto Response, Remote SMS, Alert, etc

visit: www.eSMSis.com

MySMSPass

Start Making Money from your Website!

- SMS Autentication System for Web Content
- Short Number by Telco Operators

Demo Website : www.InfoLINUX.web.id/sections

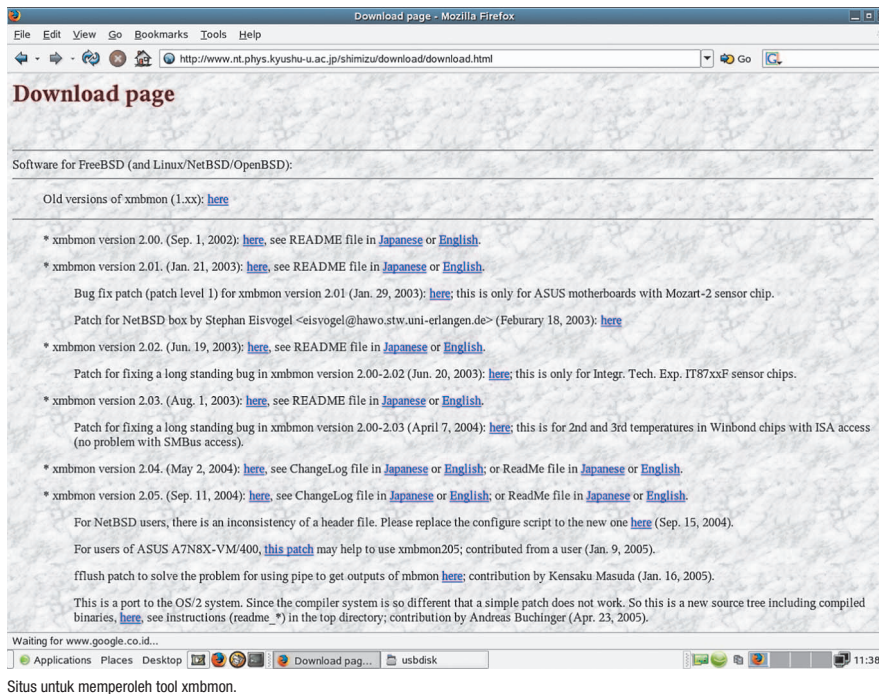


Linuxindo

PUSAT : Wisma Bisnis Indonesia Suite #415 - JAKARTA
BANDUNG: (022) 7234192 - CIREBON: (0231) 200418 - SOLO: (0271) 662318

PERINGATAN ! Linux bisa membuat Anda kecanduan, menambah PD dan belum ada obatnya. Tidak Setiap Paket Promosi tersedia di Cabang.

(021) 5362390
www.Linuxindo.com



```
# ./mbmon -T1 -c1
41.5
```

Sayangnya, informasi yang kita inginkan adalah suhu dalam tipe integer. Oleh karena itu, kita akan membuat program cut untuk membantu:

```
# ./mbmon -T1 -c1 | cut -d. -f1
41
```

Untuk memastikan bahwa semua whitespace diabaikan, gunakan bantuan program tr:

```
# ./mbmon -T1 -c1 | cut -d. -f1 | tr -d '[:space:]'
41#
```

Selanjutnya, kita bisa gunakan output perintah tersebut untuk mengisi variabel temperatur kita, sebagai contoh `TEMP=`./mbmon -T1 -c1 | cut -d. -f1 | tr -d '[:space:]'``.

Autoshutdown

Sampai di sini, kita telah siap membangun aplikasi. Aplikasi yang akan kita bangun cukup sederhana. Kita berikan nama `hwmon2`. Tugas dari program ini adalah memeriksa suhu sistem dan membuat laporan sederhana. Program `hwmon2` ini akan dijalankan

setiap menit, dan oleh karenanya, kita akan mendapatkan laporan suhu sistem setiap menit (yang disimpan dalam file `hwmon2.report`). Berikut ini adalah source code program `hwmon2`:

```
#!/bin/sh

TEMP_OK=55
TEMP=`/home/DATA/PUB/Apps/System/LowLevel/xmbmon/xmbmon205/mbmon -c1 -T1 | cut -d. -f1 | tr -d '[:space:]'`
STATUS="OK--$TEMP of $TEMP_OK"

[ $TEMP -ge $TEMP_OK ] &&
STATUS="DANGER--$TEMP of $TEMP_OK"
echo "`date`" $STATUS >> ~/BACKUP/hwmon2.report
```

Sebagai catatan, script tersebut dijalankan oleh root dan semua file disimpan di `/root/BACKUP/`. Gantilah path ke `mbmon` sesuai konfigurasi sistem Anda. Ganti pula lokasi laporan dan script Anda.

Untuk membuat autoshutdown, kita akan membuat program tambahan, sebut saja `hwmon3`, yang akan memeriksa baris terakhir pada `hwmon2.report`, dan akan melakukan tindakan yang bersesuaian apabila menemukan kata `DANGER`. Program

ini juga akan mencatat waktu shutdown dan suhu terakhir ketika shutdown dilakukan. Berikut ini adalah source code program `hwmon3`:

```
#!/bin/sh

STATUS=`tail -n1 ~/BACKUP/hwmon2.report | grep -i danger | tr -d '[:space:]'`
if [ ! -z $STATUS ]
then
    echo "`date` DANGEROUS SHUTDOWN $STATUS" >> ~/BACKUP/shutdown.time
    /sbin/shutdown -h now
fi
```

Berikut ini adalah contoh isi file `shutdown.time`:

```
Sun Jul 3 21:26:00 WIT 2005
DANGEROUS SHUTDOWN SunJul321:25:00WIT2005DANGER--55of55
```

Jangan lupa untuk menjalankan `hwmon2` dan `hwmon3` setiap menit. Tambahkan kedua baris ini ke dalam crontab Anda.

```
* * * * * /root/BACKUP/hwmon2.sh
* * * * * /root/BACKUP/hwmon3.sh
```

Dengan kerja sama antara `hwmon2` dan `hwmon3`, kita bisa mendapatkan laporan status sistem setiap jangka waktu tertentu, dan dapat pula mengambil tindakan yang diinginkan.

Satu pertimbangan penting yang mungkin perlu adalah masalah seberapa sering kita menjalankan program `hwmon2` dan `hwmon3`. Terlalu sering tentu akan berakibat pada kesibukan sistem yang mungkin tidak perlu. Namun, terlalu lama juga akan membuat aplikasi ini kurang berguna. Sesuaikan dengan kondisi server Anda dan load server yang mungkin.

Pertimbangan lainnya adalah suhu yang bisa ditoleransi (`TEMP_OK` pada `hwmon2`). Pastikan jangan terlalu rendah dan jangan lupa terlalu tinggi. Kenali pula karakteristik processor yang digunakan.

Sampai di sini dulu pembahasan kita. Source code program-program tersebut masih bisa dikembangkan lebih lanjut sesuai kebutuhan Anda. Selamat mencoba! ☺
Noprianto (noprianto@infolinux.co.id)