

## SISTEM PENGAWASAN KINERJA MULTI *NET- SERVER* MELALUI MEDIA *ON-LINE*

**Oleh : Achmad Hardito**

Staf Pengajar Jurusan Teknik Elektro Politeknik Negeri Semarang  
Jl. Prof. Sudarto, SH, Tembalang, Semarang 50275

### Abstrak

*Sistem pengawasan berbasis web merupakan sebuah sistem pemantauan jarak jauh melalui jaringan internet, penggunaannya bisa diaplikasikan ke pemantauan kinerja apa saja. Aplikasi berbasis Web bisa digunakan untuk pemantauan kinerja server. Server sebagai pusat data dan pusat software perlu adanya suatu alat pemantauan terutama kinerja perangkat kerasnya. RRD-Tool digunakan untuk mengelola database yang ada pada server khususnya pada linux operating system. RRD-tool fungsinya sebagai database dan digunakan untuk salah satu kegiatan monitoring pada suatu jaringan, dengan bantuan Web data yang ada pada database bisa divisualisasikan dalam bentuk grafik. Sistem aplikasi ini tidak terbatas pada pengawasan satu jaringan saja tapi bisa dikembangkan menjadi beberapa jaringan (multy network). Penelitian ini membuat sistem aplikasi untuk mengawasi dan menganalisa kinerja multi net-server secara real time dengan menggunakan RRD-Tool. Dengan sistem aplikasi ini pemantauan dapat dilakukan melalui media on-line, sehingga semua kejadian atau kegiatan pada suatu server bisa dilihat secara waktu. Sistem aplikasi pengawasan ini juga dapat memudahkan administrator (Pengelola) dan User untuk mengetahui kondisi server setiap saat. Manfaat dari pengawasan kinerja server, dapat mengetahui kapasitas kerja server yang di tunjukan melalui beberapa plug-in seperti load, cpu, entropy, memory, swap, diskfree dan uptime. Hasil dari penelitian ini adalah berupa web yang menampilkan besaran besaran plug-in pada beberapa server yang diawasi. Pengawasan kinerja server.*

**Kata Kunci :** *Multi - Net Server, RRD -Tools, System Monitoring, Plug-in*

### 1. Pendahuluan

Sistem pengawasan *server* berbasis web sangat diperlukan terutama digunakan untuk pengawasan kinerja perangkat keras . Sistem ini dibuat untuk memudahkan administrator mengadakan pengawasan jarak jauh secara *online* melalui *internet* , sehingga jika terjadi masalah pada *server* dapat dilakukan penanganan dengan cepat. Sistem pengawasan ini dapat memantau kinerja server dalam waktu nyata.

Sistem pengawasan berbasis web sudah pernah dilakukan, dalam penelitian sebelumnya sistem aplikasi ini digunakan untuk pengawasan kinerja *server* pada jaringan *mini cluster* komputer terutama pengawasan padapenyeimbang beban dan suhu. (Hadiyanto, dkk. 2004).

Sistem pengawasan kinerja multi *net-server* adalah sistem aplikasi yang dapat dipergunakan untuk mengawasi beberapa *server* pada tempat yang berbeda pada waktu yang bersamaan dan pengawasannya secara paralel. Sistem pengawasan ini merupakan suatu proses yang digunakan untuk mengumpulkan data pada *server* saat

*online*. Data yang didapat merupakan data yang terjadi saat itu atau waktu nyata. (Akbar, dkk. 2007).

Dalam pembacaan sistem pengawasan telah diperkenalkan pendekatan baru untuk mengaktifkan mesin paralel secara terbuka dan dapat diakses oleh public untuk multi pengguna dengan pekerjaan multi blok yang berjalan pada waktu yang sama. Konsep ini diperlukan terutama untuk mesin paralel atau *cluster computer*. (Akbar, dkk., 2007).

Penyediaan data yang cepat dan tepat merupakan kebutuhan mendasar pada sebuah lembaga. *Server* perlu dilakukan pengawasan secara rutin, sebab *server* berfungsi untuk mengatur pada sistem jaringan. Sistem pengelolaan sumber daya adalah alat yang bagus untuk pemantauan kinerja *server* secara waktu nyata terutama pada penyeimbang beban secara dinamis. (Uthayopas, dkk. 1998).

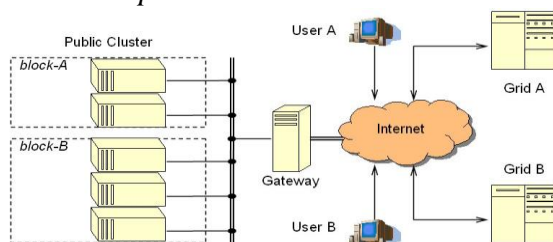
Dari latar belakang diatas penulis membuat Sistem Pengawasan Kinerja Multi Net-

server Melalui Media *On-line*. Pembuatan sistem aplikasi berbasis *web* tidak terbatas pada *cluster* komputer tapi dikembangkan untuk pengawasan kinerja *server* antar jaringan antar gedung.

## 2. Tinjauan Pustaka

### 2.1. Konfigurasi Jaringan

Sistem pengawasan kinerja multi *net-server* adalah sistem aplikasi yang dapat dipergunakan untuk mengawasi beberapa *server* pada tempat yang berbeda pada waktu yang bersamaan dan pengawasannya secara paralel. Sistem pengawasan ini merupakan suatu proses yang digunakan untuk mengumpulkan data pada *server* saat online. Data yang didapat merupakan data yang terjadi saat itu juga atau waktu nyata. Sistem pengawasan diperlukan untuk lebih menunjang suatu manajemen jaringan yang lebih baik. Salah satu aktivitas dalam sistem *monitoring* yang dapat dilakukan dalam jaringan adalah pengawasan kinerja *server* berbasis *web*. (Akbar,dkk., 2007). Gambar 2.1 memperlihatkan gambar jaringan yang memantau kinerja server terutama penyeimbang beban pada *cluster* komputer.



Gambar 2.1. Jaringan cluster komputer dengan beberapa blok server (Akbar,dkk., 2007)

Sistem *client-server*, *client* dapat mengirim sebuah *query* pada proses *server* manapun. *Client* bertanggung jawab terhadap persoalan *user-interface* dan server mengatur data dan mengeksekusi transaksi. Jadi, proses *client* dapat dijalankan melalui personal komputer dan mengirim *query* pada *server* yang berjalan pada kerangka utama. Arsitektur ini sangat populer untuk beberapa alasan: Pertama, arsitektur tersebut relatif sederhana untuk

diimplementasikan berkenaan dengan pemisahan fungsional pada *client* dan *server*-nya desentralisasi. Kedua, tidak menggunakan mesin *server* yang mahal berkaitan dengan interaksi pengguna biasa. Ketiga, pengguna dapat menjalankan *user interface* grafis yang mereka kenali daripada *user interface* pada *server*. (Ramakrishnan, 2004).

*Round Robin Database* (RRD) adalah suatu basis data yang digunakan untuk mengolah data pada *server* dan dipergunakan untuk *monitoring* serta analisa data. Pada tool ini analisis data berupa nilai rata-rata, maksimum, minimum, dan nilai terakhir pada suatu interval tertentu yang telah ditetapkan sebelumnya. *RRD-tool* adalah sederetan *tool* untuk menciptakan dan mengubah *basis data*, dari data-data itu menghasilkan sebuah grafik. *RRDtool* dipakai untuk mencatat data terhadap waktu (seperti jaringan *bandwidth*, temperatur ruang mesin, atau *load server* rata-rata) dan bisa menampilkan data itu sebagai nilai rata-rata dalam selang waktu tertentu. Kelebihan dari *RRd-tool*:

- Dapat memberikan kemudahan dalam melakukan aktivitas *log* data dengan bebas pada kurun waktu tertentu. *RRd-tool* akan otomatis melakukan interpolasi nilai dari data tersebut pada slot waktu terakhir (*latest official time slot*)
- Secara otomatis akan melakukan analisis data ketika suatu data baru dimasukkan ke dalam *RRD*, hal ini memberikan keuntungan apabila menyimpan data dengan interval waktu tertentu.
- Memberikan jaminan bahwa ukuran *RRD* tidak akan mengalami penambahan dan data yang lama secara otomatis akan dibuang. (Hadiyanto,dkk., 2004).

Monitoring jaringan menggunakan *web* dapat digunakan untuk pencatatan dan analisis data secara akurat berupa arus trafik, dan digunakan juga untuk mengidentifikasi kinerja pada jaringan lain. Monitoring jaringan yang baik menampilkan angka berupa grafik pada

saat kondisi jaringan itu sedang berjalan. Ada beberapa keuntungan melakukan sistem *monitoring* jaringan: (Friendly,2007).

- a. Anggaran jaringan dan sumber daya bisa ditekan. Sistem monitoring yang baik dapat memperlihatkan, bahwa infrastruktur jaringan (*bandwidth*, dan perangkat lunak) sesuai serta bisa menangani kebutuhan pengguna jaringan.
- b. Trafik jaringan bisa mendeteksi penyusup (virus, hackers) dan mencegah akses ke *server* dan layanan lainnya yang sangat penting. Virus jaringan dengan mudah dideteksi, ada pemberitahuan akan adanya virus jaringan, serta melakukan tindakan sebelum mereka memakan *bandwidth Internet* dan membuat jaringan tidak stabil.
- c. Kinerja jaringan bisa optimal, tanpa sistem pengawasan yang efektif mustahil untuk mengkonfigurasi alat dan protokol untuk mencapai kinerja yang terbaik.
- d. Penggunaan jaringan secara layak dapat dibagi untuk penggunaan *bandwidth* secara merata kepada semua *users* serta menjamin kalau jaringan dipakai sesuai dengan tujuan.

Penelitian di atas hanya membatasi dalam satu topologi yaitu *Cluster* dan lebih banyak digunakan dalam pengawasan penyeimbang beban dan suhu saja. Pada penelitian ini penulis membuat rancang bangun sistem aplikasi yang bisa digunakan untuk mengawasi kinerja *server* pada multi jaringan dengan *multi server*.

Sebuah *host* dapat berkomunikasi dengan LAN atau Internet, terlebih dahulu harus mengatur *TCP/IP Network Interface Card (NIC)* *host* bersangkutan. Beberapa informasi yang perlu kita berikan kepada *NIC* tersebut adalah *IP address*, *Gateway*, dan *DNS Server*. Dengan ketiga informasi tersebut sebuah *host* akan dapat berkomunikasi dengan LAN atau Internet.

Melakukan konfigurasi *TCP/IP* di *host* dengan menggunakan sistem operasi Linux tidak terlalu sulit, karena linux sudah menyediakan tampilan *Graphical User Interface (GUI)* untuk mempermudah pengguna melakukan konfigurasi, dapat juga seorang administrator melakukan konfigurasi melalui terminal Linux. Secara default masing masing *NIC* sudah dikonfigurasi untuk mendapatkan konfigurasi *TCP/IP* secara otomatis, jika di jaringan sudah tersedia *DHCP server* sebuah *host* akan dapat berkomunikasi dengan LAN atau Internet.

## 2.2. Instalasi DHCP Server

Sebuah komputer dapat melakukan koneksi ke *LAN* atau ke internet bila *host* tersebut memiliki informasi *IP-Address*, *Gateway*, dan *DNS server*. Informasi tersebut dapat diperoleh secara manual (static) maupun secara dinamis (*DHCP*). Agar sebuah *host* dapat memperoleh informasi *TCP/IP* secara dinamis maka *Dynamic Host Configuration Protocol (DHCP) server* harus sudah dipasang terlebih dahulu.

Langkah instalasi *DHCP server* pada *LINUX* :

- a. Pastikan aplikasi *DHCP server* sudah terinstal.

```
#rpm -qa | grep dhcp
Libdhcp4client-3.0.6-10.fc8
Dhcp-devel-3.0.6-10.fc8
Libdhcp6client-0.10-51.fc8
Libdhcp-1.27-3.fc8
Dhcpv6-client-0.10-51.fc8
Dhcp-3.0.6-10.fc8
```

- b. Bila *DHCP* belum terpasang, maka *DHCP* dipasang dahulu dengan perintah:

```
#rpm -ivh dhcp-3.0.6-10.fc8.i386.rpm
```

Setelah *DHCP* terpasang, kita akan melakukan konfigurasi *DHCP server*. *DHCP server* akan dikonfigurasi dengan informasi berikut:

```
Subnet yang dilayani      :
192.168.10.0/255.255.255.0
IP Address : 192.168.10.2
Gateway    : 192.168.10.2
```

IP                      DNS                      Server  
 :192.168.10.1,202.202.5.3  
 Range              Dynami              IP              :  
 92.168.10.100.192.168.10.200

mendapatkan informasi TCP/IP dari DHCP server ini.

- c. Konfigurasi file dhcp.conf  
 #vi /etc/dhcpd.conf  
 Secara default, file dhcpd.conf disimpan di direktori /etc/dhcpd.conf, sedangkan contoh konfigurasinya ada di direktori /usr/share/doc\*/dhcpd.conf.sample  
 Isi file dhcpd.conf dengan informasi berikut:

```
#### Begin of dhcpd.conf file ####
Ddns-update-style none;
Ignore client-updates;
Subnet 192.168.10.0 netmask 255.255.0
{
#- default gateway
Option routers 192.168.10.2;
Option subnet-mask 255.255.255.255.0;
Option domain name-servers
1192.168.10.1, 202.202.5.3;
Option time-offset-18000; # Eastern
Standart Time
Range dinamic-bootp 192.168.10.100
192.168.10.200;
Default-lease-time 21600;
Max-lease-time 43200;
}
#### end of dhcpd.conf file ####
```

- d. Simpan dan tutup file dhcp.conf diatas (:wq).
- e. Pastikan layanan dhcpd jalan tiap kali komputer boot.

```
# chkconfig-level 345 dhcpd on
```

- f. Jalankan layanan DHCP.  
 #Service dhcpd start

Jika layanan DHCP mengalami kegagalan ketika diulangi, kemungkinan ada kesalahan konfigurasi. Untuk itu periksa kembali dengan perintah sebagai berikut:

```
# grep dhcpd/ var/log/messages
```

Bila sudah tidak ditemukan kesalahan layanan DHCP sudah bisa digunakan untuk *host* secara otomatis. Baik oleh sitem operasi linux maupun oleh sistem operasi windows. Bila server ini berjalan dengan baik maka host akan

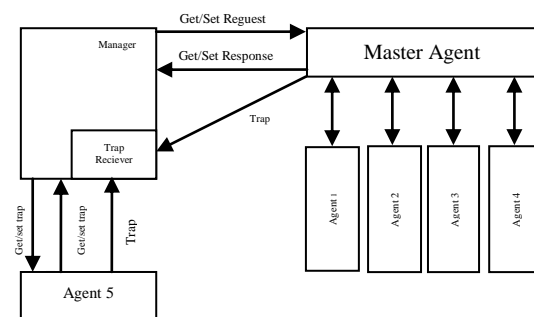
### 2.3. Simple Network Management Protocol (SNMP).

*Simple Network Management Protocol* adalah sebuah protokol internet untuk mengelola perangkat pada IP jaringan. Protokol ini digunakan untuk memonitor *divice* yang terhubung ke jaringan untuk memantau kondisi sistemnya. Sebagai contoh penggunaan CPU, penggunaan *Harddisk*, penggunaan *memory*, *traffic* jaringan dan lain lain. *Divice* yang dapat dipantau adalah *PC*, *Server*, atau *Router*.

Dengan adanya SNMP tidak perlu memeriksa satu persatu *server*, tetapi cukup mengakses satu komputer untuk melihat *kondisi* seluruh server dan router, hal ini disebabkan *server* dan *router* akan bertindak sebagai *SNMPserver* yang tugasnya menyediakan request SNMP dari komputer lain. Satu PC akan bertindak sebagai *SNMP agent* yaitu komputer yang mengumpulkan informasi dari *SNMPserver*.

Selain digunakan untuk memonitoring sebetulnya SNMP dapat digunakan unuk melakukan perubahan dan meberikan konfigurasi baru ke *server*. Tetapi pengubahan konfigurasi sistem di *server* hanya dilakukan apabila ada perubahan infrastruktur di jaringan.

Nilai-nilai *variable* yang diakses menggunakan SNMP diatur dalam bentuk hirarki. Tipe hirarki dan metadata (seperti Tipe dan diskripsi variabel) diatur oleh *Management Information Bases (MIBs)*.



Gambar. 2.2 Komunikasi Dasar SNMP.

## 2.4. Collectd Sebagai Client Dan Server.

*Collectd* dibagi menjadi dua bagian yaitu: *client* dan *server*. Masing-masing bagian mempunyai konfigurasi yang berbeda meskipun memiliki lokasi berkas konfigurasi yang sama yaitu: `/etc/collectd/collectd.conf`.

Berikut adalah penjelasan bagian penting dari konfigurasi:

### a. Bagian konfigurasi yang penting untuk PC yang bertugas sebagai server.

#### Nama server

```
Hostname "usm-collectd-server"
```

Kita nonaktifkan resolusi DNS untuk mempercepat proses

```
#FQDNLookup true
```

Jeda waktu pembacaan sensor dan atau data kiriman *Interval 10*

Jumlah *thread* yang digunakan untuk membaca sensor dan atau data

```
ReadThreads 5
```

Kita nonaktifkan *plugin syslog* untuk meminimalisir waktu tulis ke HDD

```
#LoadPlugin syslog
```

```
#<Plugin syslog>
```

```
#LogLevel info
```

```
#</Plugin>
```

Aktifkan *plugin network* supaya bisa terkoneksi ke komputer lain. *Port 25826* dibuka untuk menerima data dari klien *collectd*.

```
LoadPlugin network
```

```
<Plugin network>
```

```
Listen "*" "25826"
```

```
</Plugin>
```

### b. Konfigurasi penting collectd pada sisi klien.

#### a) Nama computer

```
Hostname "stimik0"
```

Kita nonaktifkan resolusi DNS untuk mempercepat proses

```
#FQDNLookup true
```

Jeda waktu pembacaan sensor *Interval 10* Jumlah *thread* yang digunakan untuk membaca sensor dan atau data

```
ReadThreads 5
```

#### b) Kita nonaktifkan *plugin syslog* untuk meminimalisir waktu tulis ke HDD

```
#LoadPlugin syslog
```

```
#<Plugin syslog>
```

```
#LogLevel info
```

```
#</Plugin>3
```

### c. Mengaktifkan Plugin Network.

Kirim data ke IP 180.250.149.102 (*server collectd*) di port 25826. *Time To Live* digunakan untuk mengantisipasi koneksi yang tidak akan dibuang secara otomatis.

```
LoadPlugin network
```

```
<Plugin network>
```

```
Server "173.44.133.190" "25826"
```

```
Time To Live 200
```

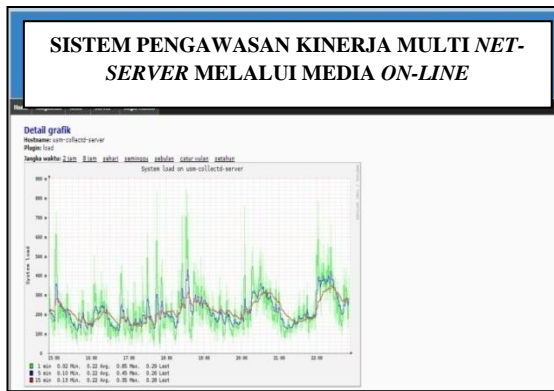
```
</Plugin>
```

## 3. Hasil Dan Pembahasan

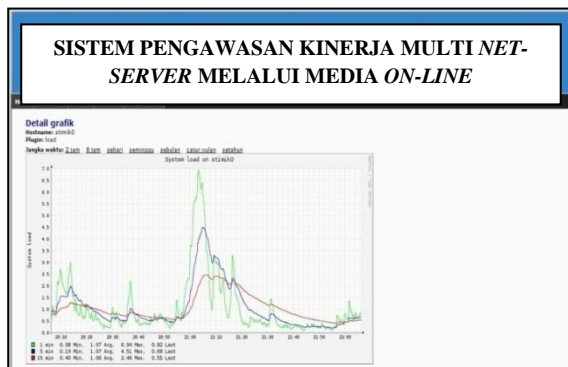
### 3.1. Hasil Pengujian Sistem Pengawasan Load Server

Gambar 3.1 menunjukkan grafik load server1, Load server pada dasarnya adalah sebuah besaran, nilainya biasanya mulai dari 0.00. Ini mengungkapkan bagaimana proses *loading* sedang menunggu dalam antrian untuk mengakses prosesor. Proses loading dihitung dalam jangka waktu tertentu dan semakin kecil angkanya, semakin baik. Semakin tinggi nilainya sering dikaitkan dengan penurunan kinerja server. Load Server nilainya selalu berubah hampir setiap waktu, karena nilai langsung dibaca dan dihitung selama periode tertentu (1, 5, 15 menit).

Gambar 3.1a dan Gambar 3.1b menunjukkan bahwa kondisi server baik dan aman. Karena nilai yang ditunjukkan masih dibawah 1 pada nilai rata-rata. Selama nilai bebannya server berada di bawah 1,00, Anda tidak akan melihat masalah seperti situs web menjadi lambat (kecuali ada beberapa masalah jaringan terkait di suatu tempat antara komputer dan server).



Gambar 3.1a Grafik Load Server1



Gambar 3.1b Grafik Load Pada Server2

Load hanyalah salah satu faktor dari banyak hal seperti penggunaan memori, penggunaan CPU, ukuran file swap dan itu sering mempengaruhi kinerja server. Jika server yang memiliki memori internal tidak cukup itu akan menggunakan *file swap*. Ini artinya akan bekerja intens dengan *hard drive*, yang akan meningkatkan penggunaan CPU dan pada gilirannya menghasilkan server load yang lebih tinggi yang akan membuat respon permintaan server menjadi lambat.

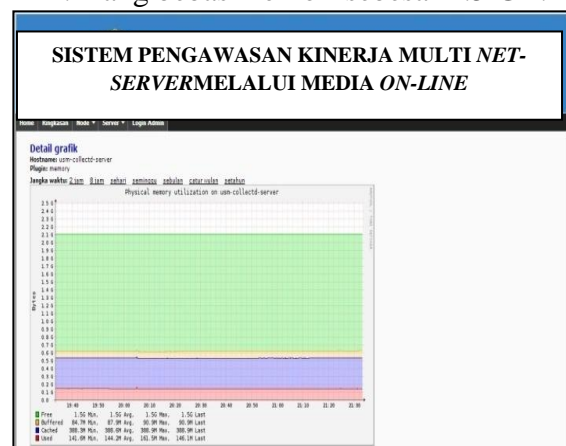
Dalam kasus seperti ini solusinya adalah: memori internal (RAM), akan menyebabkan file swap yang lebih rendah, penggunaan CPU yang lebih rendah, beban server lebih rendah, menyebabkan kecepatan respon lebih cepat. Jadi server menggunakan semua yang dibutuhkan dan lebih banyak menggunakan memori internal.

Jika beban server sekitar 2-4 seperti terlihat pada beberapa titik pada grafik diatas

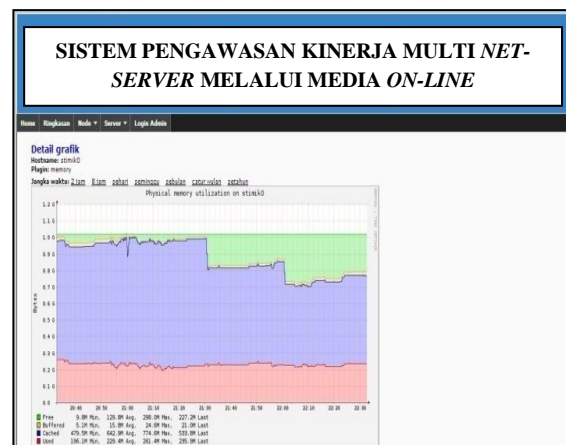
berarti ada suatu yang salah terjadi overselling atau overloading. Tapi secara keseluruhan server1 tidak pernah menghadapi masalah serius. Load server Pada Interval 1- 15 menit menunjukkan bahwa proses kerja pada server di server2 Sangat rendah.

### 3.2. Memori Server

Gambar 3.2a Menunjukkan bahwa memori pada server USM 2GB, memori yang digunakan rata rata 173.5 MB, cached memorinya 291.6 MB, Buffer memori 90.4 MB. Yang bebas memori sebesar 1.5 GB.



Gambar 3.2a Grafik Memori Pada Server1



Gambar 3.2b Grafik Memori Server Pada Server2

Gambar 3.2b menunjukkan memori server pada laboratorium STIMIK Provisi sangat rendah, karena server ini hanya dipergunakan untuk praktikum mahasiswa. Memori yg tidak digunakan rata-rata 28,7 MB, Buffer rata-rata 78 MB, Cache rata-

rata 275,5 MB, memori digunakan 267.6 MB.

#### 4. Kesimpulan Dan Saran

##### 4.1 Kesimpulan

- a. Terdapat beberapa faktor yang dapat mempengaruhi kinerja dari sistem pengawasan berbasis web, yakni kondisi dari jaringan Ethernet serta spesifikasi dari komputer yang digunakan.
- b. Program sistem pengawasan telah dilakukan uji coba pada server1 dan server2 perbedaan spesifikasi yang digunakan menghasilkan perbedaan hasil grafis yang ditampilkan.
- c. Dalam sistem pengawasan yang dibuat telah menunjukkan hasil yang baik dan sesuai dengan yang diharapkan, dalam pembacaan kinerja servernya hasilnya menunjukkan real-time.
- d. Grafis yang dihasilkan antara menggunakan jaringan kabel koaksial dengan jaringan serat optik sangat berbeda. Serat optik lebih bagus dibanding kabel koaksial.

##### 4.2 Saran

- a. Penelitian sistem pengawasan multi net-server menggunakan RRD Tool adalah Sistem informasi yang memantau masalah kinerja server, penelitian ini masih banyak kekurangan, terutama dilihat dari menu-menu yang berkaitan dengan penampilan Plug-in yang kurang lengkap.
- b. Pengembangan selanjutnya dalam penelitian ini bahwa sistem pengawasan tidak hanya terbatas pada 2 lembaga saja tapi bisa dikembangkan menjadi beberapa puluh lembaga bahkan ratusan.

#### DAFTAR PUSTAKA

Akbar, Z., Slamet, Ajinegoro, B.I., Ohara, G.J., Firmansyah, I., Hermanto, B. And Handoko, L.T., 2007. *Open And Free Cluster For Public*. International conference on information integration and web-based application and services, Jakarta, Indonesia.

Akbar, Z., and Handoko, L.T., 2007. *Web Based Interface in Public Cluster*. International conference on information integration and web-based application and services, Jakarta, Indonesia.

Daryanto, 2010. *Teknologi Jaringan Internet*, Bandung, Satu Nusa.

Friendly, H.L., 2007. *Jaringan Wireless di Dunia Berkembang*.

Hadiyanto and Handoko, L.T., 2004. *The development of Mini Open Cluster for The Computation Society in Indonesia*. J. Theor. Comput. Stud. Vol 3 (2004) 0408.

Liang, Z., Sun. Y., and Wang. C.L., 2006. *clusterProbe: An Open, Flexible and Scalable Cluster Monitoring Tool*. Research Department of Computer Science and Information Systems The University of Hong Kong Pokfulam Road, Hong Kong.

Prakoso, S., 2005. *Jaringan Komputer Linux: Konsep Dasar, Instalasi, Aplikasi, Keamanan Dan penerapan*, Yogyakarta, Andi.

Siregar, E., 2010. *Langsung Praktik Mengelola Jaringan Lebih Efektif Dan Efisien Pada Linux Fedora dan Windows XP*, Yogyakarta, ANDI.