

Aplikasi Pemantau Perangkat Jaringan Berbasis Web Menggunakan Protokol SNMP dan Notifikasi SMS

Shofa Taftazanie, Agung Budi Prasetyo, Eko Didik Widiyanto^{*}

Program Studi Sistem Komputer, Fakultas Teknik, Universitas Diponegoro
Jl. Prof. Soedarto, SH, Kampus Undip Tembalang, Semarang, Indonesia 50275

Abstract - *Device monitoring tool on a computer network is required by the administrator to perform management of the device. This research produces web applications using PHP that is capable of monitoring and collecting device information on the network. This app uses SNMP and syslog protocols to perform monitoring on the device. Device information will be collected on the monitor server and displayed in web-based applications. This app using crontab to be able to monitor every minute. This application can also send a warning message in the form of SMS and email to the administrator when an error occurs on the device so that handling is expected to be faster done. This web application can make it easier for administrators to monitor devices anywhere.*

Keywords – *Monitoring; SNMP; syslog; SMS; PHP; crontab*

Abstrak - *Pemantauan perangkat pada suatu jaringan komputer sangat diperlukan oleh administrator untuk melakukan manajemen perangkat tersebut. Penelitian ini menghasilkan aplikasi web menggunakan PHP yang mampu melakukan pemantauan dan pengumpulan informasi perangkat di jaringan. Aplikasi ini menggunakan protokol SNMP dan syslog untuk melakukan pemantauan pada perangkat. Informasi perangkat akan dikumpulkan pada server monitor dan ditampilkan pada aplikasi berbasis web. Aplikasi menggunakan crontab agar dapat melakukan pemantauan setiap menit. Aplikasi juga dapat mengirimkan pesan peringatan berupa SMS dan email kepada administrator saat terjadi kesalahan pada perangkat server sehingga penanganan diharapkan dapat lebih cepat dilakukan. Aplikasi web ini dapat mempermudah administrator dalam memantau perangkat dimana saja.*

Kata Kunci - *Pemantauan; SNMP; syslog; SMS; PHP; crontab*

I. PENDAHULUAN

Administrator jaringan komputer bertugas untuk melakukan pemantauan terhadap suatu jaringan komputer. Kesalahan atau kerusakan pada server rentan untuk terjadi pada perangkat jaringan atau server yang mengakibatkan masalah pada akses *client*. Server yang bekerja terus-menerus dapat mencapai ambang batas seperti pemakaian CPU, RAM, atau penyimpanan yang

tersedia. Manajemen perangkat yang efisien dan otomatis diperlukan dalam jaringan yang besar karena sangat sulit untuk mengelola perangkat hanya dengan usaha manusia [1]. Kerusakan pada aplikasi sering terjadi yang disebabkan oleh konfigurasi yang tidak sesuai dengan aplikasi. Terkadang pesan kesalahan yang ditampilkan oleh sistem mengarahkan pada file log aplikasi.

Informasi log dapat digunakan oleh administrator untuk melihat dan memantau aktivitas perangkat. Log file berisi satu atau lebih log atau aktivitas yang telah dijalankan oleh aplikasi yang setidaknya terdapat waktu ketika aktivitas dijalankan [2]. Manajemen log sangatlah dibutuhkan oleh administrator, guna untuk mempermudah melihat log pada setiap perangkat. Protokol *syslog* menyediakan jalur yang memungkinkan perangkat mengirimkan pesan *log* melalui jaringan IP ke *event collector* [3].

Pemantauan bukan hanya melihat aktivitas jaringan melalui log saja, namun juga dapat mengetahui kondisi perangkat. RAM, *storage*, dan *availability* perlu dilakukan pemantauan untuk menjaga perangkat dalam kondisi yang optimal. *Simple Network Management Protocol* (SNMP) adalah sebuah protokol yang digunakan untuk memonitor perangkat pada jaringan, seperti memonitor peralatan jaringan (seperti *switch*), peralatan komputer, dan perangkat lainnya [4].

Pradikta dkk. [5] memaparkan perancangan sistem pemantauan menggunakan protokol SNMP. Protokol SNMP digunakan untuk mengumpulkan data perangkat dengan melakukan permintaan informasi pada perangkat. Informasi yang dikumpulkan digunakan untuk menghitung nilai *availability* pada perangkat. Basis data digunakan untuk menampung hasil dari permintaan SNMP.

Syuhada [6] memaparkan perancangan sebuah prototipe perangkat yang dapat mengirim pesan laporan jika terjadi kesalahan pada perangkat. Pendeteksian kesalahan dilakukan dengan menggunakan *ping* pada setiap perangkat dan *service*. Penelitian tersebut menggunakan *syslog* sebagai protokol untuk mengumpulkan log perangkat. *Rsyslog* digunakan sebagai aplikasi untuk menyimpan log pada basis data yang telah ditentukan.

Taufan dkk. [7] memaparkan perancangan sistem pemantauan secara berkala setiap 10 menit. Sistem menggunakan *script* yang akan melakukan PING pada server perangkat. *Script* dijalankan dengan menggunakan *crontab* setiap 10 menit secara otomatis.

^{*}) Penulis korespondensi (Eko Didik Widiyanto)
Email: didik@live.undip.ac.id

Jika terjadi kesalahan pada server, sistem akan mengirimkan pesan peringatan pada administrator.

Peringatan kesalahan dapat membantu administrator dalam mengatasi masalah yang terjadi pada server dengan cepat. Administrator tidak dapat setiap waktu memantau seluruh aktivitas perangkat pada jaringan, sehingga diperlukan sebuah sistem peringatan untuk lebih menghemat waktu. Sistem peringatan ini dapat ditingkatkan fasilitasnya dengan pengiriman pesan peringatan melalui media SMS (*Short Message Service*) yang langsung terkirim ke administrator [8].

Prakdikta dkk. [5] menggunakan protokol SNMP untuk melakukan *pemantauan* namun tidak terdapat peringatan kondisi perangkat. Syuhada [6] menggunakan protokol *syslog* untuk mengumpulkan *log*. Sistem melakukan peringatan dengan menggunakan PING pada setiap server. Namun, sistem tidak dapat mendeteksi kesalahan dalam penggunaan RAM dan *storage* pada server. Taufan dkk. [7] membangun sistem yang dapat memonitor server setiap 10 menit. Pendeteksian masalah menggunakan PING ke server tujuan. Awaj dkk. [8] memantau kondisi lingkungan, yaitu suhu dan kelembaban, di ruang server dan mengirimkan pesan peringatan melalui SMS. Penelitian ini menghasilkan aplikasi berbasis *web* yang dapat diakses oleh administrator melalui berbagai perangkat elektronik seperti *smartphone*, laptop, dan *gadget* lainnya yang mendukung untuk mengakses *web*. Sistem dapat melakukan pemantauan penggunaan RAM dan *storage* pada perangkat. Sistem dapat melakukan peringatan kesalahan melalui SMS kepada administrator. Sistem dapat mendeteksi penggunaan RAM, *storage* dan *availability* server dengan menggunakan permintaan SNMP. Protokol *Syslog* digunakan untuk menganalisis kesalahan yang terjadi pada perangkat.

II. METODE DAN BAHAN

Metode penelitian yang digunakan dalam mengembangkan sistem pemantauan jaringan komputer terdiri dari 4 tahap yaitu identifikasi masalah, desain aplikasi dan sistem, pembuatan sistem dan aplikasi, dan implementasi. Beberapa sistem pemantauan yang telah ada dikaji, seperti Cacti dan LogAnalyzer. Cacti adalah aplikasi *open source network monitoring* dan grafik dengan menggunakan PHP, MySQL, dan RRDTool [9]. *LogAnalyzer* adalah sebuah aplikasi berbasis *web* yang digunakan untuk menampilkan data *log* pada *Rsyslog* [10].

Kebutuhan fungsional dari aplikasi pemantau yang dikembangkan adalah sistem dapat mengumpulkan *log* aktivitas dari perangkat *remote* dengan menggunakan *syslog*. Server pemantauan dapat mengambil dan mengumpulkan informasi komputer *agent* menggunakan protokol SNMP dalam interval setiap satu menit. Server pemantauan dapat mengirimkan pesan peringatan melalui SMS dan email. Kebutuhan fungsional yang terakhir yaitu sistem

Tabel 1. Spesifikasi Komputer

Komponen	Server
Prosesor	Intel Pentium CPU G620 2.60 GHz
Motherboard	ECS H61H2-M12
RAM	2 GB
HDD	320 GB
NIC	1 <i>Fast Ethernet</i>

aplikasi dapat menampilkan *log* dan informasi perangkat yang telah dikumpulkan oleh sistem.

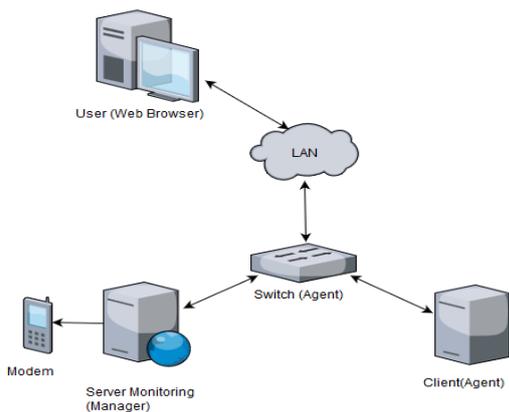
Perangkat keras digunakan sebagai wadah sistem dan penunjang agar sistem berjalan, yaitu berupa dua unit komputer dengan pembagian 1 sebagai server pemantauan dan 1 sebagai server untuk dilakukan pemantauan. Perangkat komputer memiliki spesifikasi pada Tabel 1. Unit Mikrotik Routerboard digunakan sebagai perangkat jaringan untuk pemantauan dan sebagai *switch* yang menghubungkan antar perangkat. Sebuah modem untuk mengirimkan pesan peringatan berupa SMS. Laptop Asus A43SJ difungsikan untuk melakukan konfigurasi sistem, membuat aplikasi dan sebagai klien. Media fisik komunikasi antar perangkat menggunakan kabel UTP.

Perangkat *server pemantauan* dan server yang dipantau menggunakan sistem operasi Debian 8 (*wheezy*). Aplikasi untuk mendukung fungsi menyentralisasi *log* setiap perangkat ke server pemantauan dan disimpan pada basis data menggunakan *Rsyslog*. Net SNMP digunakan untuk mengambil informasi dari setiap perangkat.

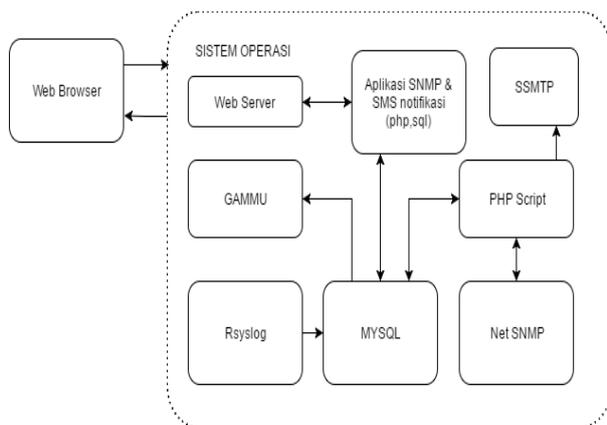
Pembuatan aplikasi pemantauan menggunakan bahasa pemrograman PHP. Pengembangan aplikasi menggunakan *framework* CodeIgniter. Apache2 digunakan sebagai server *web* untuk menampung aplikasi agar dapat diakses. Basis data pada server pemantauan menggunakan MySQL. MySQL tersedia sebagai perangkat lunak gratis di bawah lisensi GNU GPL (*General Public License*).

Sistem pemantauan terdapat fungsi peringatan dengan mengirimkan pesan SMS dan email. Pesan SMS dikirimkan dengan menggunakan aplikasi SMS *gateway* GAMMU. Sedangkan pengiriman email menggunakan *mail server* SSMTP melalui Gmail. Konfigurasi pada server dilakukan secara *remote* dengan menggunakan aplikasi Putty. Putty adalah perangkat lunak *remote console/terminal* yang digunakan untuk melakukan *remote access* ke komputer dan perangkat jaringan dengan menggunakan SSH, Telnet, atau serial. Melakukan percobaan pada aplikasi menggunakan Mozilla Firefox sebagai penjelajah *web*.

Desain topologi sistem pemantauan ini terdiri dari berbagai entitas yang terlibat dalam pembangunan sistem. Entitas-entitas tersebut dihubungkan menggunakan suatu topologi *star*. Topologi *star* mempunyai kontrol terpusat, semua link harus melewati pusat yang menyalurkan data tersebut ke semua simpul atau client yang dipilihnya [11]. Topologi ini memungkinkan server pemantauan (*manager*) untuk menerima dari *agent* dan *client (agent)* untuk mengirimkan informasi perangkat ke *manager* seperti



Gambar 1. Desain topologi fisik



Gambar 2. Desain perangkat lunak sistem

pada Gambar 1. Setiap perangkat memiliki fungsi masing-masing sebagai *manager*, *agent*, atau *client* yang menggunakan aplikasi sistem pemantauan.

Gambar 2 menunjukkan desain sistem perangkat lunak pada server dapat berkomunikasi sehingga membentuk sebuah sistem. Perangkat lunak terdiri atas komponen software yang digunakan dalam sistem pemantauan. Perancangan dilakukan pada dua tahap yaitu penggunaan protokol SNMP dan *Syslog* sebagai pengumpul informasi. Sistem dirancang untuk menjalankan proses permintaan SNMP dan pendeteksian kesalahan pada perangkat. Pengumpulan log pada aplikasi sistem pemantauan ini menggunakan *syslog*. *Rsyslog* (*Rocket-fast System for log Processing*) adalah sebuah sistem *logger* yang menggunakan protokol *syslog* berfungsi mengumpulkan log pada basis data yang ada [12]. Pendeteksian pada ketersediaan *service* menggunakan port aplikasi. *Port* merupakan bagian dari pengalamanan yang telah ditentukan yang berfungsi sebagai jalur dari aplikasi ke *transport layer* atau dari *transport layer* ke aplikasi [13]. Data ini kemudian disimpan ke database *Mysql*.

Proses kerja sistem pemantauan ditunjukkan pada Gambar 3. Diagram alir tersebut menjelaskan ng proses pengumpulan informasi perangkat menggunakan SNMP. Informasi yang dikumpulkan disimpan ke dalam basis data dan informasi digunakan untuk mendeteksi kesalahan. Diagram tersebut menjelaskan pemrosesan

data yang dikumpulkan untuk dijadikan sebagai informasi perangkat dalam kondisi baik atau tidak sesuai dengan parameter yang telah ditentukan. Peringatan berupa SMS akan dikirim jika perangkat dalam kondisi tidak baik.

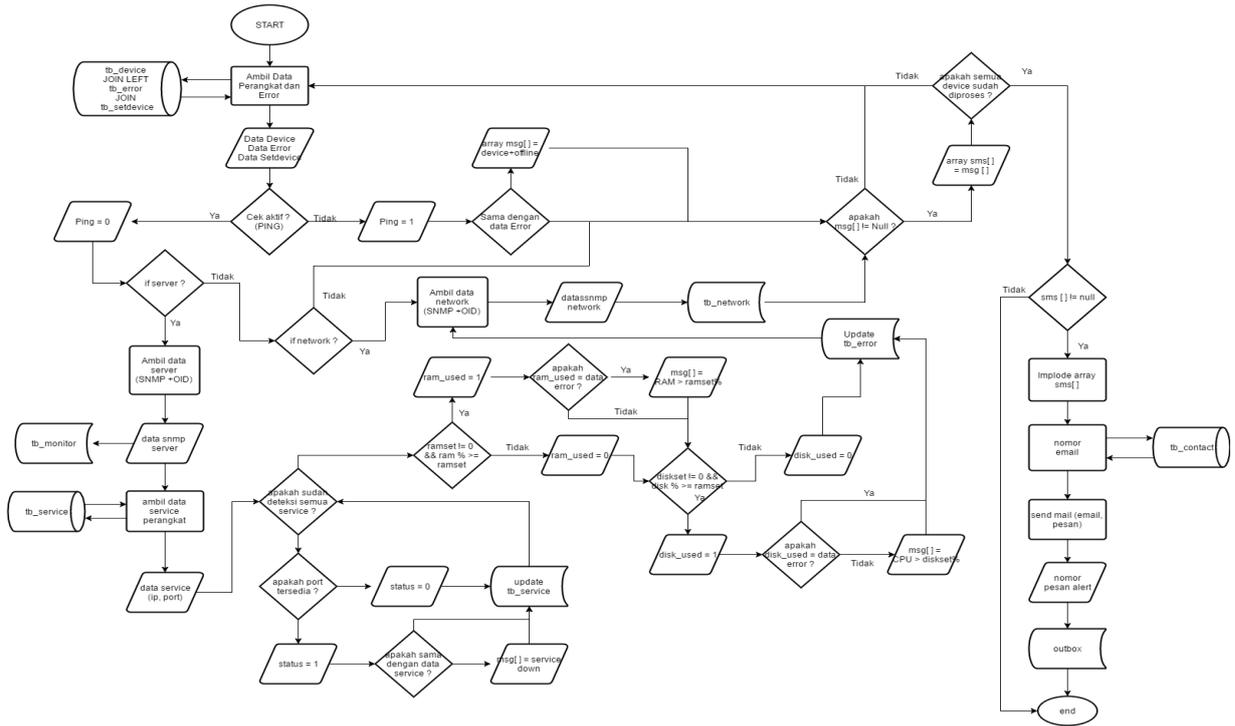
Perancangan basis data membahas tentang basis data yang digunakan oleh sistem aplikasi ini. Beberapa tabel basis data yang digunakan *default* dari aplikasi pihak kedua seperti *Gammu* dan *syslog*. Basis data akan menyimpan hasil dari pengumpulan data menggunakan protokol SNMP. Basis data digunakan untuk mempermudah dalam melakukan pengolahan serta penyimpanan data dalam jumlah banyak. *Entity Relationship Diagram* (ERD) dari aplikasi pemantauan ditunjukkan pada Gambar 4.

III. HASIL DAN PEMBAHASAN

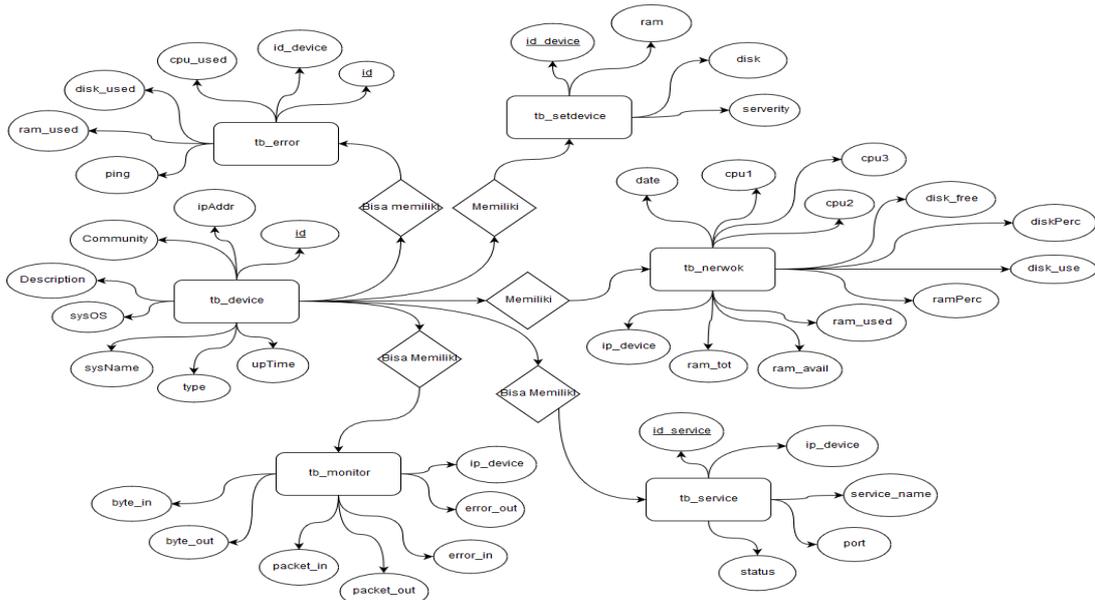
Pengujian sistem dilakukan untuk memeriksa kinerja antar komponen sistem yang diimplementasikan. Tujuan pengujian sistem adalah untuk memastikan bahwa elemen-elemen atau komponen-komponen dari sistem telah berfungsi sesuai dengan yang diharapkan. Pengujian tersebut meliputi pengujian *syslog*, sistem pemantauan SNMP, sistem peringatan perangkat, pengujian *service*, dan pengujian aplikasi secara keseluruhan.

Kebutuhan fungsional pada sistem pemantauan adalah server dapat mengumpulkan *log* aktivitas dari perangkat *agent* dengan menggunakan *syslog*. Pengumpulan *log* yang dimaksud adalah menampung *log* dari perangkat *agent* yang terdapat *syslog* kedalam basis data yang telah dibuat secara *realtime*. Sistem yang diinginkan adalah ketika perangkat *agent* menghasilkan *log*, kemudian *log* dikirim ke server pemantauan dan sistem menyimpan pada basis data. Berdasarkan Gambar 5, setelah dilakukan aktifitas pada tiap perangkat *agent* terdapat data baru pada basis data tabel *SystemEvents*. Data tersebut adalah *log* aktifitas yang dihasilkan oleh perangkat *agent* dan dikirimkan ke manajer server untuk disimpan dalam basis data. Hasil pengujian *syslog* menunjukkan sistem dapat mengumpulkan *log* tiap perangkat dan disimpan pada basis data.

Berdasarkan kebutuhan fungsional pada sistem pemantauan adalah server pemantau dapat mengambil dan mengumpulkan informasi komputer *agent* menggunakan protokol SNMP dalam waktu interval satu menit. Pengumpulan informasi komputer menggunakan protokol SNMP yang dijalankan pada *file* PHP. Program PHP bertugas untuk menyimpan data hasil dari permintaan SNMP ke dalam basis data. Sistem yang diinginkan adalah dapat melakukan penambahan data pada basis data tiap menit. Gambar 6 menunjukkan hasil pengujian bahwa sistem dapat melakukan permintaan SNMP pada perangkat *agent*. *Request* dilakukan oleh sistem pada setiap perangkat dalam interval waktu satu menit. Informasi yang dikumpulkan disimpan pada basis data tabel *tb_monitor*.



Gambar 3. Diagram alir proses pemantauan



Gambar 4. ERD basis data pemantauan

ID	CustomerID	ReceivedAt	DeviceResponseTime	Facility	Priority	FromHost	Message	NTSeverity	Importance
81568	NALL	2016-12-17 12:59:08	2016-01-06 03:46:19	5	7	MikroTik	1.3.6.1.2.1.1.3.0	NALL	NUL
81567	NALL	2016-12-17 12:59:08	2016-01-06 03:46:19	5	6	MikroTik	public	NALL	NUL
81566	NALL	2016-12-17 12:59:08	2016-01-06 03:46:19	5	6	MikroTik	from: 10.42.12.65 version: 1	NALL	NUL
81565	NALL	2016-12-17 12:59:08	2016-01-06 03:46:19	5	7	MikroTik	1.3.6.1.2.1.2.2.1.20.1	NALL	NUL
81564	NALL	2016-12-17 12:59:08	2016-01-06 03:46:19	5	6	MikroTik	public	NALL	NUL
81563	NALL	2016-12-17 12:59:08	2016-01-06 03:46:19	5	6	MikroTik	from: 10.42.12.65 version: 1	NALL	NUL
81562	NALL	2016-12-17 12:59:08	2016-01-06 03:46:19	5	7	MikroTik	1.3.6.1.2.1.2.2.1.14.1	NALL	NUL
81561	NALL	2016-12-17 12:59:08	2016-01-06 03:46:19	5	6	MikroTik	public	NALL	NUL
81560	NALL	2016-12-17 12:59:08	2016-01-06 03:46:19	5	6	MikroTik	from: 10.42.12.65 version: 1	NALL	NUL
81559	NALL	2016-12-17 12:59:08	2016-01-06 03:46:19	5	7	MikroTik	1.3.6.1.2.1.2.2.1.17.1	NALL	NUL
81558	NALL	2016-12-17 12:59:08	2016-01-06 03:46:19	5	6	MikroTik	public	NALL	NUL
81557	NALL	2016-12-17 12:59:08	2016-01-06 03:46:19	5	6	MikroTik	from: 10.42.12.65 version: 1	NALL	NUL
81556	NALL	2016-12-17 12:59:08	2016-01-06 03:46:19	5	7	MikroTik	1.3.6.1.2.1.2.2.1.11.1	NALL	NUL
81555	NALL	2016-12-17 12:59:08	2016-01-06 03:46:19	5	6	MikroTik	public	NALL	NUL
81554	NALL	2016-12-17 12:59:08	2016-01-06 03:46:19	5	6	MikroTik	from: 10.42.12.65 version: 1	NALL	NUL
81553	NALL	2016-12-17 12:59:08	2016-01-06 03:46:19	5	7	MikroTik	1.3.6.1.2.1.2.2.1.16.1	NALL	NUL
81552	NALL	2016-12-17 12:59:08	2016-01-06 03:46:19	5	6	MikroTik	public	NALL	NUL
81551	NALL	2016-12-17 12:59:08	2016-01-06 03:46:19	5	6	MikroTik	from: 10.42.12.65 version: 1	NALL	NUL
81550	NALL	2016-12-17 12:59:08	2016-01-06 03:46:19	5	7	MikroTik	1.3.6.1.2.1.2.2.1.10.1	NALL	NUL

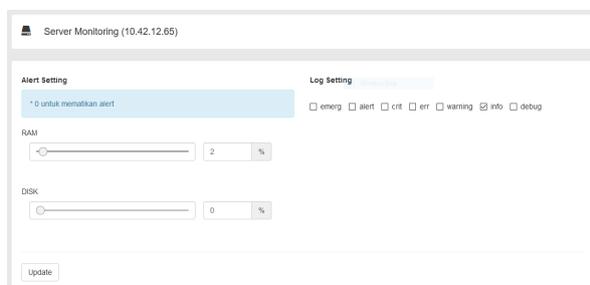
Gambar 5. Tabel SystemEvents

id	device	date	cpu1	cpu2	cpu3	ram_avail	ram_used	ramPerc	disk_free	disk_use	diskPerc	illn
4299	1	2016-12-05 01:42:03	0.07	0.03	0.05	1114820	922344	8.6322	286273136	1762204	0.580677	1074195
4298	1	2016-12-05 01:41:03	0.01	0.01	0.05	1115788	921376	8.58488	286273140	1762200	0.580676	1073783
4297	1	2016-12-05 01:40:03	0.01	0.01	0.05	110312	934052	9.20849	286273148	1762192	0.580673	1073402
4296	1	2016-12-05 01:39:04	0.01	0.01	0.05	1114156	925708	8.79968	286273176	1762164	0.580664	1073028
4295	1	2016-12-05 01:38:03	0.01	0.01	0.05	1117936	919328	8.4987	286273168	1762172	0.580666	1072652
4294	1	2016-12-05 01:37:03	0.01	0.01	0.05	1118084	910608	8.47492	286273176	1762164	0.580664	1072345
4293	1	2016-12-05 01:36:03	0.01	0.01	0.05	1116768	911416	8.39033	286273196	1762144	0.580659	1072035
4292	1	2016-12-05 01:35:03	0.02	0.02	0.05	1108484	926888	9.04696	286273184	1762156	0.580661	1071653
4291	1	2016-12-05 01:34:03	0.02	0.02	0.05	1103916	913264	9.01862	286273208	1762132	0.580653	1071271
4290	1	2016-12-05 01:33:03	0.03	0.03	0.05	1106254	913904	8.90375	286273212	1762128	0.580652	1070972
4289	1	2016-12-05 01:32:03	0.01	0.04	0.05	1107172	929992	8.85338	286273216	1762134	0.580651	1070395
4288	1	2016-12-05 01:31:03	0.01	0.04	0.05	1105668	930496	8.88431	286273228	1762112	0.580647	1069950
4287	1	2016-12-05 01:30:03	0.04	0.05	0.05	1095132	942032	9.45118	286273236	1762104	0.580644	1069440
4286	1	2016-12-05 01:29:03	0.11	0.06	0.06	1116024	925568	8.843	286273240	1762100	0.580643	1068903
4285	1	2016-12-05 01:28:03	0.2	0.06	0.06	1114208	922568	8.51615	286273264	1762076	0.580635	1068673
4284	1	2016-12-05 01:27:06	0.01	0.01	0.05	1115664	921500	8.44606	286273260	1762080	0.580636	1068444
4283	1	2016-12-05 01:27:03	0.01	0.01	0.05	1115664	921500	8.44606	286273260	1762080	0.580636	1068431
4282	1	2016-12-05 01:26:03	0.01	0.01	0.05	1105932	932132	8.96559	286273272	1762068	0.580632	1067798
4281	1	2016-12-05 01:25:03	0.02	0.02	0.05	1090768	946396	9.63084	286273284	1762076	0.580632	1067165
4280	1	2016-12-05 01:24:03	0.02	0.02	0.05	1100336	933328	8.99957	286273288	1762052	0.580627	1066815

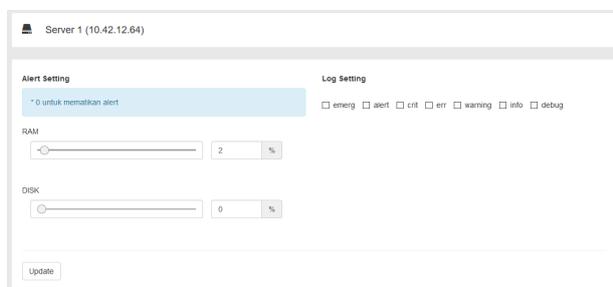
Gambar 6. Tabel tb_monitor

	id_device	ram_used	disk_used	ping
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1	0	0	0
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	2	0	0	0
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	3	0	0	0

Gambar 7. Kondisi awal status perangkat



Gambar 8. Pengaturan pada perangkat Monitoring Server



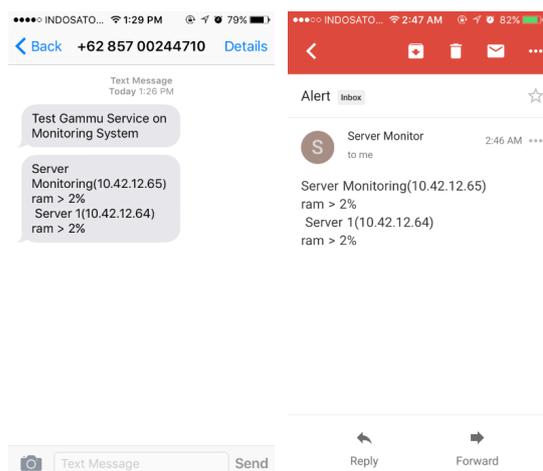
Gambar 9. Pengaturan pada perangkat Server 1

	id_device	ram_used	disk_used	ping
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1	1	0	0
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	2	1	0	0
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	3	0	0	0

Gambar 10. Kondisi status RAM perangkat melebihi batas

Berdasarkan kebutuhan fungsional pada sistem pemantauan adalah aplikasi dapat mengirimkan pesan peringatan melalui SMS dan *email*. Pesan peringatan berupa informasi perangkat dalam keadaan melebihi batas yang telah ditentukan. Sistem hanya akan mengirimkan pesan sekali jika tidak ada perubahan terjadi. Gambar 8 menunjukkan kondisi awal ketika perangkat dalam keadaan normal. Selanjutnya dilakukan pengaturan penggunaan RAM pada aplikasi pemantauan sistem. Setiap perangkat dilakukan pengaturan batas penggunaan RAM 2% dari jangkauan 1% sampai 100% seperti pada Gambar 8 dan Gambar 9. Sistem akan memberi peringatan jika penggunaan RAM sampai 2% atau lebih dari total kapasitas RAM.

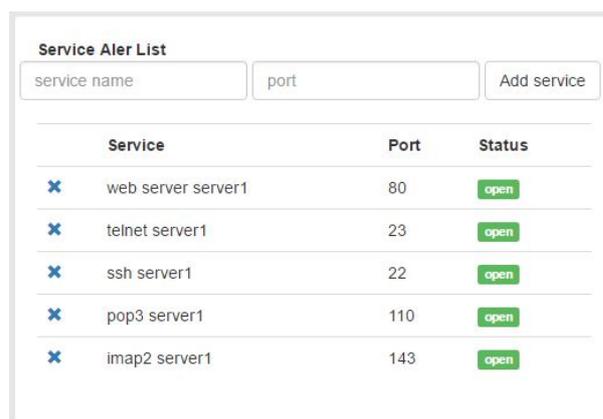
Untuk setiap perubahan status, aplikasi hanya mengirimkan 1 buah SMS saja. Gambar 10 menunjukkan bahwa nilai berubah menjadi "1" karena terjadi kesalahan pada perangkat. Perubahan nilai akan membuat sistem untuk mendeteksi, jika perubahan dari nilai "0" menjadi "1" berarti terjadi kesalahan sistem dan sistem mengirimkan pesan peringatan, sedangkan jika nilai berubah dari "1" menjadi "0" maka sistem tidak akan melakukan pengiriman pesan.



Gambar 11. Pesan peringatan SMS dan email

Tabel 2. Hasil pengujian pesan peringatan

Perangkat Monitoring Server	RAM	DISK	PING
Server 1	Berhasil	Berhasil	Berhasil



Gambar 12. Pemambahan service pada basis data

Hasil pengujian pada setiap perangkat menunjukkan sistem dapat mengirimkan pesan peringatan ke nomor dan email yang terdaftar. Ketika perangkat terjadi kesalahan status akan berubah menjadi nilai 1, namun ketika perangkat dalam kondisi normal status akan bernilai 0. Pesan peringatan dikirim hanya sekali selama belum terjadi perubahan nilai status pada perangkat.

Berdasarkan kebutuhan fungsional pada sistem pemantauan adalah dapat mengirimkan pesan peringantn melalui SMS dan email. Pesan peringatan berupa informasi availability service yang berjalan pada perangkat. Service yang dimaksud adalah aplikasi seperti web server, mail server, basis data, dan aplikasi lainnya. Service yang akan dilakukan pemantauan sesuai dengan yang terdaftar pada basis data. Berdasarkan Gambar 11 setelah diberikan kondisi pada tiap perangkat pesan peringatan terkirim melalui sms dan *email*. Pesan dikirimkan hanya sekali selama tidak terjadi perubahan pada kondisi sebelumnya. Tabel 2 menunjukkan hasil pengujian terhadap pemakaian RAM, disk, dan ketersediaan perangkat (*offline/online*). Pengujian dilakukan dengan

	id	ip_device	service_name	port	status
	2	10.42.12.65	mysql monitor	3306	0
	3	10.42.12.64	web server server1	80	0
	4	10.42.12.64	telnet server1	23	0
	5	10.42.12.64	ssh server1	22	0
	6	10.42.12.64	pop3 server1	110	0
	7	10.42.12.64	imap2 server1	143	1
	10	10.42.12.65	Web Server	80	0

Gambar 13. Perubahan basis data tb_service pada service imap2

Service Aler List

service name port

Service	Port	Status
web server server1	80	open
telnet server1	23	open
ssh server1	22	open
pop3 server1	110	open
imap2 server1	143	close

Gambar 14. Perubahan status pada aplikasi menjadi close

Tabel 3. Hasil Pengujian Service

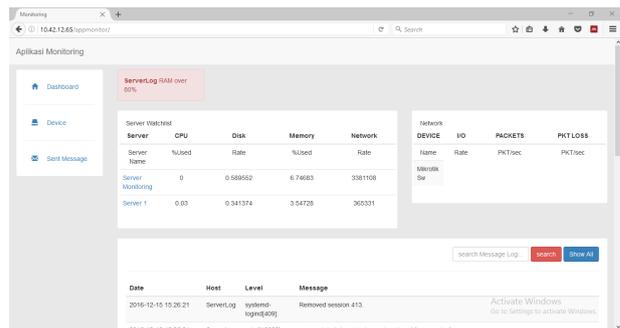
Service	Server Monitoring	Server 1
Web Server	Berhasil	Berhasil
SSH	Berhasil	Berhasil
Telnet	-	Berhasil
MySQL	Berhasil	Berhasil
IMAP2	-	Berhasil

menambahkan *service* yang terdapat pada perangkat “server 1” ke dalam basis data. Penambahan *service* meliputi nama dan port yang digunakan oleh *service*. Port digunakan untuk mendeteksi *service* dalam kondisi tersedia atau tidak. Penambahan *service* dilakukan seperti pada Gambar 12.

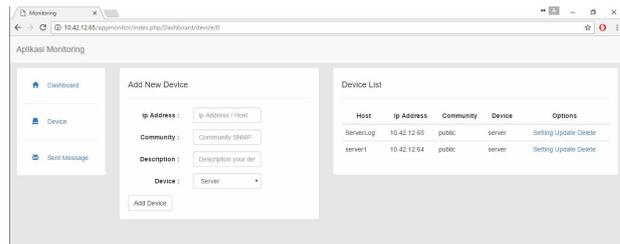
Setelah *service* ditambahkan, dilakukan penonaktifan *service* perangkat “server 1” yang sudah terdaftar. Dalam pengujian ini, *service* yang dilakukan pengujian adalah *service* IMAP2 yang digunakan sebagai server mail pada perangkat “server 1”. Penonaktifan dilakukan pada “server 1”.

Setelah *service* nonaktif, maka sistem pemantauan akan mendeteksi port yang digunakan oleh IMAP2 tidak tersedia. Sistem akan mengubah nilai status pada basis data menjadi “1” seperti pada Gambar 13. Aplikasi akan menampilkan status “close” pada tabel *service* seperti pada Gambar 14. Selain perubahan status pada aplikasi, juga terdapat pesan peringatan melalui SMS dan email.

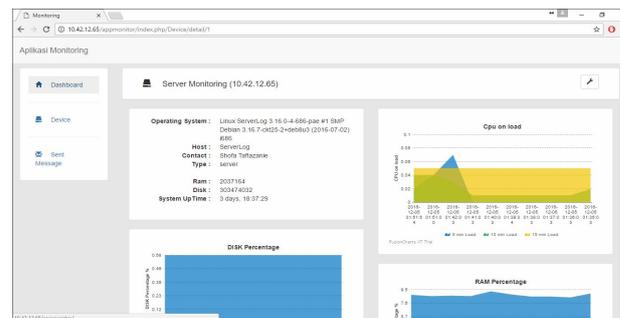
Berdasarkan hasil pengujian sistem peringatan *service*, sistem mendeteksi melalui kesalahan pada port yang digunakan *service*. Ketika port tidak tersedia atau tidak dapat diakses oleh sistem, maka sistem akan mengubah nilai menjadi “1” pada basis data. Tabel 3 menunjukkan hasil pengujian sistem terhadap beberapa *service* aplikasi pada perangkat.



Gambar 15. Tampilan Dashboard



Gambar 16. Tampilan manajemen perangkat

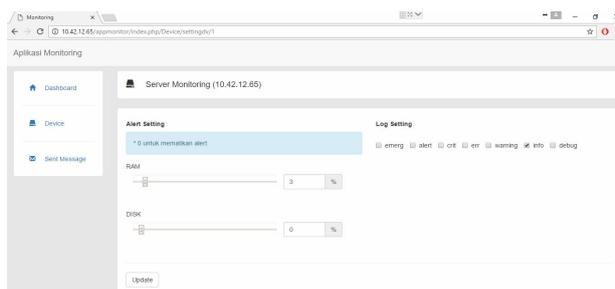


Gambar 17. Tampilan detail perangkat

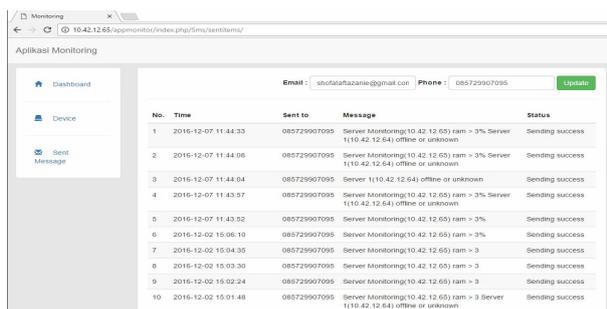
Tampilan awal pada aplikasi menampilkan “*dashboard*” yang menyediakan informasi dan aktifitas *log* setiap perangkat. Aplikasi mempunyai beberapa menu, di antaranya *dashboard*, *device*, dan *send message* seperti pada Gambar 15. Menu “*device*” mempunyai submenu berisi manajemen perangkat dan daftar perangkat yang telah disimpan. Menu manajemen perangkat terdapat *form* untuk menambah dan mengubah informasi perangkat. Pada sebelah kanan terdapat tabel yang berisi data perangkat yang terdaftar pada aplikasi. Gambar 16 menunjukkan tampilan manajemen perangkat pada aplikasi.

Fitur *detail* menampilkan informasi rinci tentang perangkat yang dipilih. Informasi yang ditampilkan adalah OS perangkat, *hostname*, kapasitas RAM, kapasitas *disk*, *total uptime* perangkat, grafik CPU on load, grafik RAM, grafik disk, dan log perangkat seperti pada Gambar 17.

Menu pengaturan *alert* dan *log* terdapat pada *detail device* dan manajemen perangkat. Menu pengaturan *alert* dan *log* terdapat pengaturan RAM dan *disk* berupa jangkauan antara 0 sampai 100 dalam satuan persen. Jika nilainya 0 maka sistem peringatan akan menjadi tidak aktif. Pengaturan *log* digunakan untuk menyeleksi *log* yang ditampilkan pada menu *detail device*. Gambar 18 menunjukkan tampilan pengaturan *alert* dan *log* perangkat.



Gambar 18. Tampilan pengaturan alert dan log perangkat



Gambar 19. Tampilan pesan alert sistem

Menu *send message* menampilkan riwayat pesan peringatan yang dikirimkan oleh sistem. Pada menu *send message* terdapat *form email* dan nomor yang dapat digunakan untuk mengubah email dan nomor tujuan pesan peringatan dikirimkan. Gambar 19 menunjukkan tampilan menu *send message* pada aplikasi sistem *pemantauan*.

Aplikasi pemonitor server telah dapat mengumpulkan informasi perangkat-perangkat yang tersambung ke jaringan dan menganalisis kesalahan sistem yang terjadi di perangkat tersebut serta mampu menampilkan hasilnya di web. Proses pengumpulan *log* dilakukan dengan menggunakan protokol SNMP [5] dan memonitor sistem menggunakan *syslog* di perangkat tersebut [2],[3],[6]. Deteksi kesalahan atau peringatan dalam penggunaan RAM dan storage di perangkat dapat dilakukan oleh aplikasi, yang berbeda dengan [6]. Proses pemantauan dan deteksi dilakukan secara periodik menggunakan *crontab* [7]. Sistem dapat melakukan peringatan kesalahan melalui SMS kepada administrator [8].

IV. KESIMPULAN

Aplikasi pemonitor server berhasil dapat mengumpulkan informasi perangkat-perangkat yang tersambung jaringan. Proses pengumpulan *log* dari berbagai perangkat pada jaringan berhasil dilakukan dengan menggunakan *rsyslog* dan dapat disimpan pada basis data. Kode PHP dapat melakukan permintaan informasi perangkat dengan menggunakan protokol SNMP pada jaringan. Data yang diambil menggunakan PHP dan protokol SNMP dapat dikumpulkan dalam

basis data. Aplikasi web mampu mengelola perangkat serta menampilkan *log*, informasi perangkat, grafik penggunaan RAM, disk, dan CPU.

Saran untuk penelitian lanjutan dapat dilakukan pengembangan aplikasi dengan penambahan OID secara dinamis. Penelitian lanjutan dapat melakukan pengumpulan data menggunakan protokol SNMP pada berbagai jenis perangkat seperti CISCO, Windows, Mac, dan perangkat lainnya.

DAFTAR PUSTAKA

- [1] K. S. Shin, J. H. Jung, J. Y. Cheon, and S. B. Choi, "Real-time network monitoring scheme based on SNMP for dynamic information," *J. Netw. Comput. Appl.*, vol. 30, no. 1, pp. 331–353, 2007.
- [2] S. O. Fageeri and R. Ahmad, "An Efficient Log File Analysis Algorithm Using Binary-based Data Structure," *Procedia - Soc. Behav. Sci.*, vol. 129, pp. 518–526, 2014.
- [3] C. Lonvick, "RFC 3164: The BSD syslog protocol," *Cisco Syst. August*, pp. 1–39, 2001.
- [4] B. Douglas Mauro and K. Schmidt Publisher, *Essential SNMP, 2nd Edition*. United States of America: O'Reilly Media, Inc, 2005.
- [5] R. Pradikta, A. Affandi, and E. Setijadi, "Rancang Bangun Aplikasi Monitoring Jaringan dengan Menggunakan Simple Network Management Protocol," *J. Tek. ITS*, vol. 2, no. 1, pp. A154–A159, 2013.
- [6] A. A. Syuhada, "Prototipe SMS Report Portabel menggunakan Raspberry Pi," *Teknol. Inf.*, pp. 1–5, 2014.
- [7] H. A. W. Taufan, D. Prayogo, K. Kushartantya, "Sistem Monitoring Jaringan Pada Server Linux Dengan Menggunakan Sms Gateway," *JMASIF*, vol. 2, no. 3, pp. 63–72, 2011.
- [8] M. F. Awaj, A. F. Rochim, and E. D. Widiyanto, "Sistem Pengukur Suhu dan Kelembaban Ruang Server," *J. Teknol. dan Sist. Komput.*, vol. 2, no. 1, pp. 40–47, 2014. doi: <http://dx.doi.org/10.14710/jtsiskom.2.1.2014.40-47>.
- [9] D. Kundu and S. M. I. Lavlu, *Cacti 0.8 Network Monitoring*. Birmingham: Packt Publishing, 2009.
- [10] "LogAnalyzer Basics," 2011. [Online]. Available: <http://logalyzer.adiscon.com/doc/basics.html>. [Accessed: 25-Nov-2016].
- [11] A. Supriyadi and D. Gartina, "Memilih Topologi Jaringan Dan Hardware Dalam Desain Sebuah Jaringan Komputer," *Informatika*, vol. 16, no. 2, pp. 1037–1053, 2007.
- [12] R. Gerhards, "rsyslog: going up from 40K messages per second to 250K," *Linux Kongress*, 2010.
- [13] N. Wells, *Sams Teach Yourself TCP/IP in 24 hours*. United States of America: Pearson Education, Inc, 2009.