

IMPLEMENTASI MONITORING JARINGAN MENGUNAKAN RASPBERRY PI DENGAN MEMANFAATKAN PROTOKOL SMTP (SIMPLE MAIL TRANSFER PROTOCOL)

Samsul Arifin

Teknik Informatika, Institut Teknologi Nasional Malang
arifinsam.sam@scholar.itn.ac.id

ABSTRAK

Penggunaan jaringan komputer dalam skala yang luas atau besar tentunya dibutuhkan suatu manajemen jaringan yang baik agar jaringan tersebut dapat berfungsi dengan baik. Dalam jaringan yang besar seringkali terjadi gangguan seperti *router crash* dan kabel terputus yang tidak dapat diprediksi kapan terjadinya dengan spesifik, yang menyulitkan pengontrolan dan pemantauan jaringan dilapangan. Ada berbagai macam protokol atau program untuk memonitoring jaringan, antara lain *SNMP* dan *ICMP*.

Dalam penelitian ini penulis memanfaatkan protokol *ICMP* (*Internet Control Message Protocol*) dengan menggunakan program *ping* untuk monitoring *node/ip* apakah *node* tersebut dalam keadaan hidup atau mati. Kemudian protokol SMTP (*Simple Mail Transfer Protocol*) digunakan untuk mengirimkan notifikasi apabila ada *node* yang mati.

Dari hasil pengujian fungsional sistem dapat berjalan pada perangkat *raspberry pi* dan fungsi pada website monitoring dapat berjalan dengan normal. Pada pengujian notifikasi dengan memanfaatkan protokol SMTP (*Simple Mail Transfer Protocol*) sesuai dengan skema pengujian pada tabel 2 didapat bahwa sistem dapat memberikan notifikasi dengan menggunakan protokol tersebut dengan rata – rata waktu pengiriman 15 detik. Pengujian user didapat dengan menyebar kuisioner kepada 15 responden didapat jawaban responden dengan rata – rata jawaban Ya sebanyak 89.3 %, Belum sebanyak 10.7 %, dan Tidak sebanyak 0 jawaban.

Kata kunci: *monitoring, jaringan komputer, icmp, raspberry pi, SMTP*

1. PENDAHULUAN

1.1. Latar Belakang

Teknologi informasi yang semakin berkembang banyak membantu dan mempermudah pekerjaan diberbagai bidang, salah satu bidang yang mengalami kemajuan adalah jaringan komputer. Namun, hal ini banyak menimbulkan masalah baru, karena semakin banyak komputer yang terhubung ke dalam suatu jaringan atau internet maka semakin banyak pula masalah yang akan muncul dan tidak dapat diprediksi secara spesifik kapan waktunya akan terjadi, seperti *router crash* dan putusnya kabel atau matinya *host* tujuan. Sebuah komputer dapat memiliki banyak fungsi, salah satunya adalah sebagai server. Fungsi komputer sebagai server digunakan untuk memberikan layanan kepada client, seperti *Web Server*, (*File Transfer Protocol*) *FTP Server*, (*Domain Name System*) *DNS Server*, *Server monitoring* dan lain sebagainya. Namun hal ini tidak efisien jika layanan yang digunakan untuk server tersebut tidak begitu memakan resource yang begitu besar dan karena dimensi yang besar serta membutuhkan daya yang besar, maka dibutuhkan suatu alternatif untuk pengganti komputer tersebut. Dengan memanfaatkan mini PC sebagai server monitoring jaringan menjadikan lebih efisien dalam segi ukuran karena mini PC yang seukuran kartu kredit dan daya yang dipakai oleh mini PC tersebut. *Raspberrypi* memiliki fitur ethernet yang digunakan untuk terhubung ke jaringan. Dengan memanfaatkan fitur ini *raspberrypi* digunakan

untuk server monitoring untuk memonitoring jaringan dan sebagai web server untuk antar muka dari server monitoring. (Hakim, 2013)

Kegiatan monitoring jaringan merupakan kegiatan yang dilakukan untuk mengelola suatu sistem jaringan dilokasi atau area tertentu. Sistem monitoring ini digunakan untuk mempermudah tim teknis dalam melakukan pemantauan secara rutin kondisi jaringan dilapangan. Terdapat berbagai macam Protokol monitoring jaringan yang umum digunakan, antara lain *SNMP* (*Simple Network Management Protocol*) dan *ICMP* (*Internet Control Message Protocol*).

Dalam penelitian yang berjudul “ Implementasi Monitoring Jaringan Menggunakan Raspberrypi dengan Memanfaatkan Protokol SMTP (*Simple Mail Transfer Protocol*) ” penulis memanfaatkan *protocol ICMP* (*Internet Control Message Protocol*) untuk memonitoring jaringan (*Node / IP*) dan SMTP (*Simple Mail Transfer Protocol*) yang digunakan untuk mengirimkan notifikasi pesan kesalahan ke *email* pada *node / IP* yang *down*, yang diimplementasikan pada sebuah mini PC *raspberrypi*.

1.2. Rumusan Masalah

Dalam penelitian ini penulis merumuskan masalah sebagai berikut :

1. Bagaimana merancang sebuah sistem yang dapat memonitor jaringan berbasis web?

2. Bagaimana sebuah sistem monitoring jaringan dapat mengirimkan notifikasi ke email apabila ada *node* yang *down* pada sebuah jaringan?
3. Bagaimana pemanfaatan mini PC *rasberry pi* sebagai *server* monitoring jaringan?

1.3. Batasan Masalah

Agar penelitian lebih fokus dan tidak meluas dari pembahasan, penulis membatasi pada ruang lingkup penelitian sebagai berikut :

1. Monitoring jaringan menggunakan bahasa pemrograman PHP.
2. Menggunakan protokol ICMP dalam memonitor *node/IP*.
3. Notifikasi apabila ada *node/IP* yang *down* menggunakan email.
4. Notifikasi yang tersimpan di log sistem adalah notifikasi *down* pada sebuah sistem.
5. Menggunakan mini PC *Raspberry Pi 2* model B+ yang digunakan untuk *server* monitoring.

1.4. Tujuan

Adapun tujuan yang ingin dicapai yaitu :

1. Membangun sistem monitoring jaringan dengan memanfaatkan protokol SMTP (*Simple Mail Transfer Protocol*) yang diimplementasikan pada *rasberry pi*.
2. Membantu tugas administrator dalam memonitoring jaringan.
3. Untuk menghasilkan sebuah sistem yang dapat mengirimkan notifikasi apabila *node/IP down*.

1.5. Manfaat

Manfaat yang ingin dicapai dari Implementasi monitoring jaringan menggunakan *Raspberry pi* dengan protokol SMTP (*Simple Mail Transfer Protocol*) yaitu :

1. Sebuah sistem yang dapat memonitor keadaan *node/IP* yang *down*.
2. Sebuah sistem yang dapat mengirimkan notifikasi secara langsung apabila kondisi suatu *node/IP down*
3. Mengimplementasi *Raspberry Pi* sebagai *server* monitoring jaringan.
4. Penyimpanan *log* hasil monitoring jaringan di sistem.

2. TINJAUAN PUSTAKA

2.1. Penelitian yang pernah dilakukan

Dalam menjaga jaringan seorang administrator harus setiap saat dapat memantau keadaan jaringan secara terus menerus karena gangguan pada jaringan tidak dapat diketahui secara spesifik kapan terjadinya. Untuk mengatasi masalah tersebut, perlu adanya sistem monitoring jaringan yang dapat mengirimkan notifikasi SMS apabila terjadi gangguan pada jaringan.(Sefrianto,2015)

Raspberry Pi merupakan komputer mini yang berfungsi layaknya komputer pada umumnya, namun

berukuran kecil. Karena dapat berfungsi sebagai komputer *rasberry Pi* juga dimungkinkan untuk dijadikan sebagai *web server*. Dalam penelitian yang berjudul “Kelayakan *Raspberry Pi* Sebagai *Web Server*: Perbandingan kinerja *Nginx*, *Apache*, dan *Lighttpd* pada Platform *Raspberry Pi* “. Pengujian tersebut dilakukan dengan parameter kinerja : *maximum request* dan *reply time*. Hasil dari penelitian menunjukkan bahwa *Raspberry Pi* layak untuk menjalankan aplikasi *web server*. (Dawood, 2014)

2.2. Jaringan Komputer

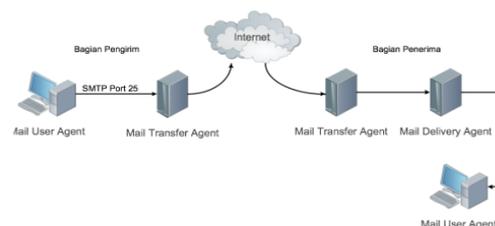
Jaringan komputer adalah kumpulan beberapa perangkat komputer yang saling terhubung satu sama lain melalui media perantara kabel ataupun media tanpa kabel (*nirkabel*). sehingga masing masing komputer yang terhubung saling bertukar data atau berbagi perangkat keras.(Tampi, 2013)

Para ahli kemudian membagi jaringan menjadi beberapa klasifikasi, antara lain:

1. Berdasarkan area atau skala
2. Berdasarkan media penghantar
3. Berdasarkan fungsi.

2.3. Simple Mail Transfer Protocol (SMTP)

SMTP (*Simple Mail Transfer Protocol*) adalah protokol aplikasi yang berfungsi untuk proses transfer surat elektronik. Standard protokol ini didefinisikan dalam RFC 821 dan disempurnakan pada RFC 5321. Dalam protokol ini didefinisikan beberapa bagian dalam proses pengiriman *email*. *Mail User Agent* (*MUA*) bagian yang bertugas untuk menangani komunikasi *Mail Server Agent* (*MSA*) dengan *mail client* melalui *port* TCP 25. *Mail Transfer Agent* (*MTA*) berfungsi untuk menangani proses *transfer* data *mail* dari *MSA* pengirim menuju *MSA* tujuan. Komunikasi keduanya dilakukan dengan *port* TCP 587. (Subhan, 2009). Berikut penjelasan mengenai aliran proses dalam SMTP pada pengiriman sebuah email. Ditunjukkan pada Gambar 1 :



Gambar 1. Aliran proses pengiriman email (Subhan, 2009)

2.4. Internet Control Message Protocol (ICMP)

PING (*Packet internet Gopher*) program yang berjalan pada protokol ICMP (*Internet Control Message Protocol*) untuk melakukan pengecekan komunikasi antar komputer dalam sebuah jaringan melalui pemodelan TCP/IP. PING (*Packet internet Gopher*) akan

mengirimkan *echo request messages* pada ip address komputer yang dituju dan meminta *respons* dari komputer tujuan. (Akbar, 2013).

2.5. IP Address

IP Address digunakan sebagai identitas untuk bisa berkomunikasi dalam sebuah jaringan komputer. Dalam proses pengiriman data *IP Address* akan dibungkus dalam bentuk paket data dengan label berupa *IP Address* pengirim dan *IP Address* Penerima. Apabila IP penerima melihat pengiriman paket tersebut dengan *IP Address* yang sesuai, maka paket disalurkan ke protokol melalui port, dimana aplikasi menunggunya. (Puspitasari, 2007).

2.6. Raspberry pi

Raspberry Pi sebuah komputer seukuran kartu kredit yang dapat digunakan layaknya sebuah *Personal Computer* (PC). Perangkat mini ini dikembangkan pertama kali oleh *Raspberry Pi Foundation* untuk menarik minat pelajar yang ingin mempelajari bahasa pemrograman komputer.(Rahmadina, 2016)

Layaknya sebuah PC *Raspberry Pi* membutuhkan *operating system* agar dapat digunakan. OS yang digunakan *Raspberry Pi* merupakan varian dari Linux. *Raspberry Pi* dirilis dengan lisensi open source hardware sehingga telah dipergunakan untuk berbagai fungsi, diantaranya *media centre, networked compute, take pictures and record videos* serta sebagai *web server*.(Rahmadina, 2016)

2.7. GNU/Linux

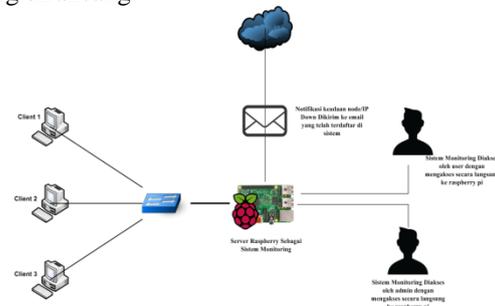
Seorang mahasiswa helsinki Linus Torvalds yang pada waktu itu tahun 1991 membuat sistem operasi baru yang gratis dan yang *sourcecode* nya bisa diakses oleh umum. Sistem operasi ini diadopsi dari MINIX yang dibuat oleh Andrew Tanenbaum untuk tujuan pendidikan. Sistem operasi buatan Linus Torvalds kemudian diberi nama *Linux*. Dalam membangun *Linux*, Linus Torvalds menggunakan *Tool – Tool* dari *Free Foundation Software* yang berlisensi GNU. Kemudian untuk menjadikan *Linux* sebuah sistem operasi yang utuh, Linus Torvalds memasukkan program – program yang juga berlisensi GNU. (Pamungkas, 2008)

Karena sistem dan pustakanya berasal dari GNU, yang diumumkan tahun 1983 oleh Richard Stallman. Kontribusi GNU adalah dasar dari munculnya nama alternatif GNU/Linux.

3. METODE PENELITIAN

3.1. Blok Diagram

Diagram blok ini berfungsi untuk memudahkan pemahaman dalam memahami alur kerja dari sistem yang dirancang.

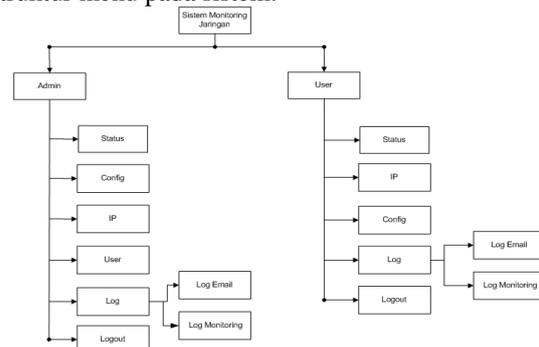


Gambar 2. Blok Diagram

Pada Gambar 2 merupakan gambaran diagram blok yang akan dibuat dimana *raspberry* digunakan untuk server monitoring. Server monitoring digunakan untuk memonitoring jaringan (*node/ip*) dan hasil dari monitoring akan disimpan di server kemudian apabila ada *node/IP* yang *down notifikasi* akan dikirimkan ke email yang telah terdaftar pada sistem.

3.2. Struktur Menu

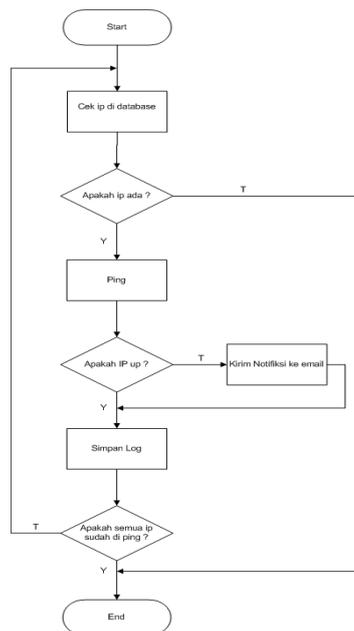
Struktur menu pada sistem yang dibangun menjadi 2 bagian, yaitu admin dan user. Dimana menu pada level admin mempunyai hak akses penuh dan level user hanya dapat melihat konfigurasi ip, email dan Log Monitoring. Gambar 3 menunjukkan struktur menu pada sistem.



Gambar 3. Struktur Menu

3.3. Flowchart Sistem

Adapun *Flowchart* sistem ditunjukkan pada Gambar 4 berikut.

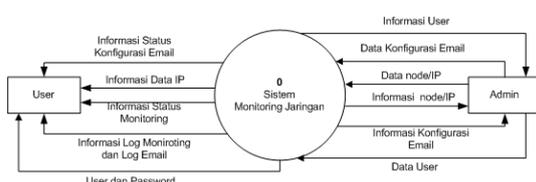


Gambar 4. Flowchart Sistem

Pada gambar 4 menjelaskan bagaimana alur sistem bekerja. Pada saat sistem *running/server* monitoring hidup, sistem akan mengambil data IP dari *database* kemudian melakukan *ping*, *output* dari *ping* apakah *node/IP* dalam keadaan *up/down*, tersambung atau tidak jika tersambung maka sistem akan *record* ke *database* hasil dari *ping*, jika tidak maka sistem akan mengirimkan *notifikasi* ke *email* menggunakan protokol SMTP (*Simple Mail Transfer Protocol*) bahwa keadaan IP yang dimonitoring dalam keadaan *down* dan hasil pengiriman *email* akan disimpan kedalam *database*.

3.4. DFD Level 0

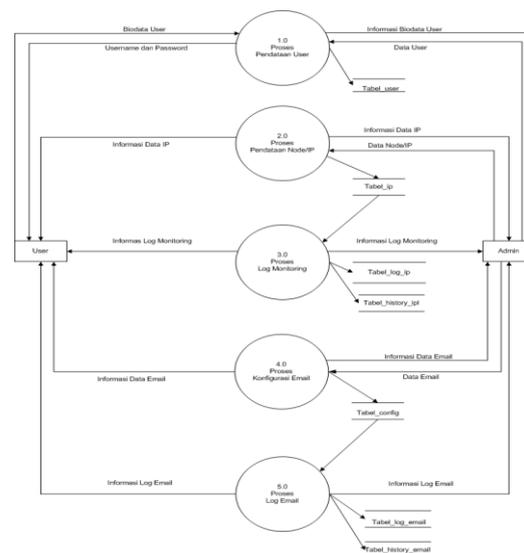
DFD Level 0 pada sistem ditunjukkan pada Gambar 5 berikut. Dalam DFD level 0 ditunjukkan 2 entitas *user* dan *admin* yang saling terhubung dengan sistem monitoring jaringan.



Gambar 5. DFD Level 0

3.5. DFD Level 1

DFD Level 1 pada sistem ditunjukkan pada Gambar 6 berikut. DFD level 1 terdapat 2 entitas dan beberapa proses, diantaranya proses pendataan *user*, proses pendataan *node/ip*, proses *log* monitoring, proses *log email*, proses *history email*, proses *history ip*, dan proses *config email*.



Gambar 6. DFD Level 1

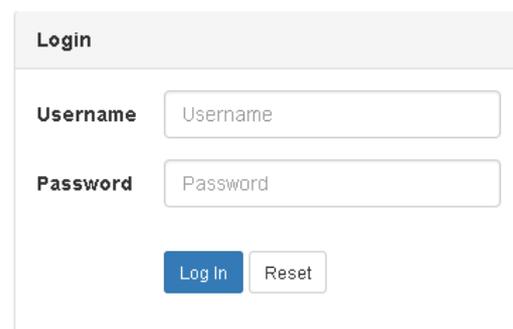
4. HASIL DAN PEMBAHASAN

4.1. Hasil Perancangan

Tahap hasil perancangan merupakan penerapan rancangan dari deteksi sistem monitoring yang sudah dirancang sebelumnya serta desain aplikasi yang diaplikasikan ke dalam bahasa pemrograman yang sudah ditentukan.

4.2. Halaman Login

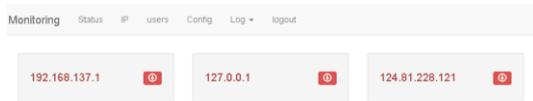
Untuk dapat melakukan pengolahan data maupun untuk melihat monitoring *user* dan *admin* diharuskan untuk login terlebih dahulu. Halaman *login* seperti ditunjukkan pada Gambar 8 berikut.



Gambar 7. Halaman login

4.3. Menu status

Dalam menu ini terdapat sebuah tampilan monitoring yang digunakan untuk melihat status dari kondisi *node / ip*. Apabila keadaan *ip down* maka status akan berwarna merah dan apabila *up* berwarna hijau.



Gambar 8. Halaman Status

4.4. Menu IP

Pada halaman menu *ip* ditampilkan seluruh *ip* yang terdapat pada sistem. Dalam menu ini tampilan dari user dan admin dibedakan berdasarkan hak aksesnya, admin dapat melakukan pengolahan data ip secara penuh, *add, update, delete* sedangkan *user* hanya dapat melihat *ip* yang tersimpan.



Gambar 9. Halaman IP

4.5. Menu user

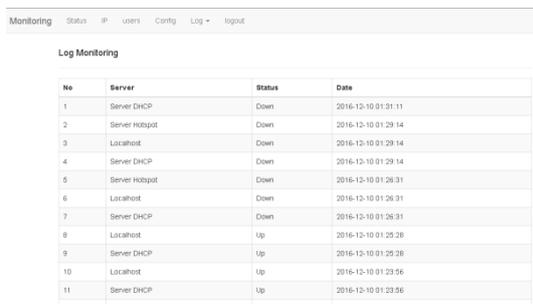
Pada halaman menu *user* ini hanya terdapat pada *admin*. Pada halaman ini digunakan untuk mengelola *user* dari sistem monitoring.



Gambar 10. Halaman User pada admin.

4.6. Menu log monitoring

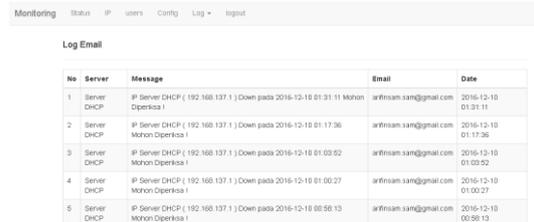
Pada halaman menu *log monitoring* terdapat pada hak akses *user* dan *admin*. Halaman ini digunakan untuk melihat *log* dari hasil monitoring sistem monitoring. Seperti ditunjukkan pada Gambar 11.



Gambar 11. Halaman log monitoring

4.7. Menu log email

Pada halaman menu *log email* terdapat pada hak akses *user* dan *admin*. Halaman ini digunakan untuk melihat *log* dari hasil *notifikasi* apabila ada *node/ip* dalam keadaan *down*. Seperti ditunjukkan pada Gambar 12.



Gambar 12. Halaman log email

4.8. Pembahasan atau pengujian dari fungsi Halaman website.

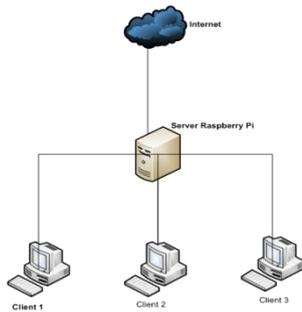
Pembahasan atau pengujian halaman pada tiap tiap halaman website dilakukan agar dapat memenuhi kebutuhan fungsional dari sistem. Dari pengujian yang dilakukan pada tiap – tiap halaman didapatkan hasil seperti pada tabel 1 berikut.

Tabel 1. Hasil Pengujian Aplikasi pada tiap – tiap halaman.

No.	Level	Nama Pengujian	Hasil
1	Admin	Halaman Login	Berhasil
2		Admin melihat halaman status	Berhasil
3		Melakukan <i>create, update, delete ip</i>	Berhasil
4		Admin mengelola halaman <i>config</i>	Berhasil
5		Mengelola halaman <i>user (create, update, delete)</i>	Berhasil
6		Admin melihat halaman <i>log monitoring node/ip log notifikasi email</i>	Berhasil
8	User	User hanya dapat melihat halaman <i>config</i>	Berhasil
9		User melihat Halaman <i>Log Monitoring</i>	Berhasil
10		User melihat halaman <i>log notifikasi email</i>	Berhasil
11		User dan <i>admin logout</i> dari sistem	Berhasil
12		User hanya dapat melihat halaman <i>ip</i>	Berhasil

4.9. Topologi yang digunakan untuk pengujian

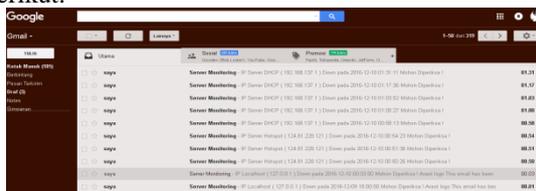
Topologi yang digunakan untuk pengujian sistem monitoring ini adalah *topologi star*, dimana *switch* yang digunakan menjadi sentral dari jaringan. Dalam *topologi* pengujian sistem digunakan 3 buah *client*, 1 *switch* dan 1 buah *server raspberry pi* yang terkoneksi dengan *internet*. Gambar *topologi* seperti ditunjukkan pada Gambar 13 berikut.



Gambar 13. Topologi Pengujian Sistem

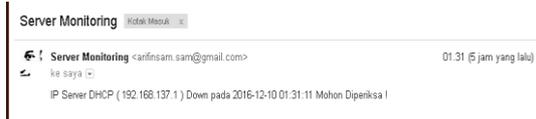
4.10. Pembahasan atau pengujian untuk pengiriman email.

Berikut hasil pengujian yang dilakukan apabila ada *node / ip* dalam keadaan down. Dari hasil pengujian didapatkan bahwa sistem dapat mengirimkan *notifikasi* ke *email* apabila ada *node / ip* yang *down*. Seperti ditunjukkan pada Gambar 14 berikut.



Gambar 14. Email Notifikasi

dan berikut adalah tampilan yang terdapat pada sebuah notifikasi pada email. Seperti ditunjukkan pada Gambar 15 berikut.



Gambar 15. Isi dari notifikasi email

4.11. Pembahasan atau pengujian pengiriman notifikasi dengan SMTP (Simple Mail Transfer Protocol)

Pada pengujian ini sistem memonitoring jaringan terhadap 5 *client* yang terhubung ke server monitoring raspberry pi. Pengujian ini dimaksudkan apakah sistem dapat mengirimkan notifikasi dengan SMTP apabila ada kondisi *client* yang *down*.

Tabel 2. Pengujian notifikasi dengan SMTP (Simple Mail Transfer Protocol)

No.	Jumlah Client	Response Time (detik)	Hasil
1	1 Down	5	Berhasil
2	2 Down	10	Berhasil
3	3 Down	15	Berhasil
4	4 Down	20	Berhasil
5	5 Down	25	Berhasil
Total		75	
Rata-Rata = Total / Jumlah Client		15	

Dari hasil pengujian yang dilakukan dengan penggunaan bandwidth 1024 kb didapatkan bahwa sistem dapat mengirimkan notifikasi dengan SMTP (*Simple Mail Transfer Protocol*) terhadap skema pengujian pada Tabel 2. dengan rata – rata response time 15 detik

4.12. Pembahasan atau pengujian log

Pengujian ini dilakukan untuk mengetahui apakah sistem dapat menyimpan log monitoring dan log notifikasi.

Tabel 3. Pengujian log notifikasi.

No.	Jumlah Client	Hasil
1	1 Down	Berhasil
2	2 Down	Berhasil
3	3 Down	Berhasil
4	4 Down	Berhasil
5	5 Down	Berhasil

Pada pengujian log notifikasi sistem dapat menyimpan log apabila sistem mengirimkan notifikasi ke email dengan presentase (5/5) x 100 = 100 %.

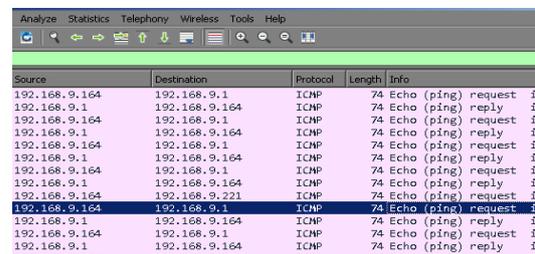
Tabel 4. Pengujian log monitoring

No.	Jumlah Client	Hasil
1	1 Client	Berhasil
2	2 Client	Berhasil
3	3 Client	Berhasil
4	4 Client	Berhasil
5	5 Client	Berhasil

Dari hasil pengujian monitoring sistem dapat menyimpan hasil monitoring kondisi client up maupun down dengan presentase (5/5) x 100 = 100 %.

4.13. Pembahasan atau pengujian monitoring

Pengujian yang dilakukan untuk mengetahui proses berjalannya monitoring digunakan tool bantu wireshark. Seperti ditunjukkan pada Gambar 16 berikut.

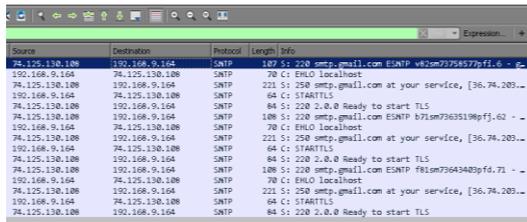


Gambar 16. Pengujian Protocol ICMP

Dari hasil pengujian protokol bahwa monitoring menggunakan protokol ICMP (*Internet Control Message protocol*) untuk monitoring jaringan.

4.14. Pembahasan atau pengujian notifikasi dengan SMTP (Simple Mail Transfer Protocol)

Sama halnya dengan Pengujian yang dilakukan untuk mengetahui proses berjalannya monitoring . pengujian yang dilakukan untuk menguji proses notifikasi dari protokol SMTP (*Simple Mail Transfer Protocol*) digunakan tool bantu wireshark. Seperti ditunjukkan pada Gambar 17 berikut.



Gambar 17. Pengujian Protocol SMTP

Dari hasil pengujian saat terjadinya *node/ip* yang *down* sistem monitoring mengirimkan notifikasi menggunakan protokol SMTP (*Simple Mail Transfer Protocol*)

4.15. Pembahasan atau pengujian user

Pengujian ini dimaksudkan untuk mengetahui apakah sistem yang dibangun sudah sesuai dengan rumusan masalah yang diangkat pada penelitian ini. Pengujian user dilakukan dengan menyebar kuisioner kepada 10 *responden*.

Tabel 5. Pengujian user.

No	Pertanyaan	Jawaban		
		Y	B	T
1	Apakah tampilan monitoring jaringan mudah dimengerti dan di operasikan ?	15	-	-
2	Apakah sistem monitoring ini dapat digunakan untuk memonitoring jaringan ?	15	-	-
3	Apakah sistem monitoring ini memudahkan administrator dalam memonitoring jaringan ?	11	4	-
4	Apakah monitoring jaringan ini sesuai diterapkan pada sebuah perangkat raspberry pi ?	13	2	-
5	Apakah sesuai monitoring jaringan ini menggunakan email sebagai media notifikasi.	13	2	-

Keterangan Jawaban : Y = Ya, B = Belum, T = Tidak

Dari hasil pengujian dengan menyebar kuisioner didapatkan kesimpulan bahwa dari jawaban 15 responden didapat total jawaban rata – rata Ya sebanyak $(67 / 75) \times 100 = 89.3$, Belum Sebanyak $(8 / 75) \times 100 = 10.7$ dan Tidak sebanyak 0 responden.

5. KESIMPULAN DAN SARAN

5.1. Kesimpulan

1. Aplikasi yang telah dibangun dengan 5 percobaan dapat memonitoring *node/ip* dengan hasil 100% berhasil.
2. Aplikasi yang telah dibangun dapat mengirimkan notifikasi ke email dengan memanfaatkan protokol SMTP (*Simple Mail Transfer Protocol*) dengan rata – rata waktu response time 15 detik.
3. Dari hasil pengujian sistem monitoring dapat berjalan di perangkat *raspberry pi*.
4. Dalam pengujian sistem dilakukan dengan topologi star pada jaringan LAN (*Local Area Network*).

5.2. Saran

Berdasarkan kesimpulan, penulis memberikan saran agar sistem monitoring dapat dikembangkan dengan menggunakan protokol SNMP (*Simple Network Management Protocol*)

DAFTAR PUSTAKA

- [1] Akbar, T., 2013. TA: Analisis Perbandingan Kinerja FreeNAS dan NAS4Free Sebagai Sistem Operasi Jaringan Network Attached Storage (NAS) Pada Local Area Network (LAN) (Doctoral dissertation, STIKOM Surabaya).
- [2] Dawood, R., Qiana, S.F. and Muchallil, S., 2014. Kelayakan Raspberry Pi sebagai Web Server: Perbandingan Kinerja Nginx, Apache, dan Lighttpd pada Platform Raspberry Pi. Jurnal Rekayasa Elektroika, 11(1), pp.25-29.
- [3] Pamungkas, D.W.L., Al-Azam, M.N. and MMT, S.K., Analisa Perbandingan File Sistem Ext2, Ext3, Dan ReiserFS Pada Sistem Operasi Linux.
- [4] Puspitasari, N.F., 2007. Implementasi Mikrotik sebagai solusi router murah dan mudah.
- [5] Rahmadina, F., 2016. Sistem Informasi Lalu Lintas Berbasis Raspberry Pi PC Board. Jurnal Nasional Teknik Elektro, 5(1).
- [6] Sefrianto, M.A., Ernawati, E. and Funny, F.C., 2015. Implementasi Protokol Snmp Dalam Sistem Monitoring Jaringan Komputer Berbasis Website Dan Sms Gateway (Doctoral dissertation, Universitas Bengkulu).
- [7] Subhan, A., 2009. Implementasi Metode Store-Forward Dengan Protokol SMTP Untuk Pengiriman Fax Pada Jaringan IP Sebagai Alternatif Migrasi Layanan Facsimile Pada Next Generation Network. In Industrial Electronic Seminar.
- [8] Tampi, B.A., Najoran, M.E., Sinsuw, A.A. and Lumenta, A.S., 2013. Implementasi Routing Pada IP Camera Untuk Monitoring Ruang di Universitas Sam Ratulangi. Jurnal Teknik Elektro Dan Komputer Unsrat, 2(2).