ISBN 979-978-96964-9-8

PROSIDING SEMINAR NASIONAL TEKNOIN 2012

Fakultas Teknologi Industri
UNIVERSITAS ISLAM INDONESIA

SEMINAR NASIONAL TEKNOIN 2012

Pengembangan Teknologi Manufaktur untuk Menunjang Penguatan Daya Saing Bangsa

TEKNIK INFORMATIKA

YOGYAKARTA, 10 NOVEMBER 2012

ISBN No. 978-979-96964-3-9

Prosiding

Seminar Nasional Teknoin 2012

"Pengembangan Teknologi Manufaktur untuk Menunjang Penguatan Daya Saing Bangsa"

Yogyakarta, 10 November 2012

Bidang Teknik Informatika

diselenggarakan oleh:

Fakultas Teknologi Industri
Universitas Islam Indonesia
Yogyakarta

Prosiding Seminar Nasional Teknoin 2012
ISBN: 979-978-96964-9-8

Diterbitkan oleh:

Fakultas Teknologi Industri
Universitas Islam Indonesia
Jl. Kaliurang Km 14,5 Yogyakarta 55584
T. 0274-895287, 0274-895007 Ext 110/200
F. 0274-895007
E. seminarteknoin@yahoo.com, teknoin@uii.ac.id
W. seminarteknoin.fit.uii.ac.id

Bidang Teknik Informatika
Yogyakarta, 10 November 2012

Organisasi Penyelenggara

Penanggung Jawab	: Ir. Gumbolo Hadi Susanto, M.Sc.	Dekan
Pengarah	: Wahyudi Budi Pramono, ST., M.Eng	Wakil Dekan
	Dr. Sri Kusumadewi, S.Si., MT.	Direktur Pascasarjana MTI
	Dra. Kamariah, MS.	Ketua Jurusan Teknik Kimia
	Drs. Mohammad mastur, MSIE	Ketua Jurusan Teknik Industri
	Yudi Prayudi, S.Si, M.Kom	Ketua Jurusan Teknik Informatika
	Tito Yuwono, ST., M.Sc	Ketua Jurusan Teknik Elektro
	Agung Nugroho Adi, ST., MT.	Ketua Jurusan Teknik Mesin
Ketua Pelaksana	: Risdiyono, ST., M.Eng., D.Eng.	
Bendahara	: 1. Yustiasih Purwaningrum, ST., MT.	
	2. Erawati Lestari, A.Md.	
Reviewer	: 1. Prof. Dr. Ir. Mauridhi Hery Purnomo, M.Eng.	
	2. Dr. Ir. Rila Mandala, M.Eng.	
	3. Ir. Muhammad Waziz Wildan, M.Sc., Ph.D.	
	4. Risdiyono, ST., M.Eng., D.Eng.	
	5. Dr. Ir. Paryana Puspaputra, M.Eng.	
	6. Ir. Erlangga Fausa, M.Cis	
	7. Ridwan Andi Purnomo, ST., M.Sc., Ph.D.	
	8. Asmanto Subagyo, M.Sc.	
	9. Izzati Muhaimmah, ST., M.Sc. Ph.D.	
	10. Hendra Setiawan, ST., MT. D.Eng.	
	11. Muhammad Ridlwan, ST., MT.	
Makalah & Prosiding:		
Koordinator	Purtojo, ST., M.Sc.	
	1. Khamdan Cahyari, ST., M.Sc.	
	2. Firdaus, ST., MT.	
	3. Hanson Prihantoro, ST., MT.	
	4. Jerri Irgo, SE., MM.	
	5. Heri Suryantoro, A.Md.	
	6. Bagus Prabawa Aji, ST.	
	7. Adi Swandono, A.Md.	
Sekretariat:		
Koordinator	M. Faizun, ST., M.Sc.	
	1. Indah Kurniasari, SP	
	2. Muhammad Susilo Atmodjo	
	3. Pangesti Rahman, SE.	
Sie. Acara dan Publikasi:		
Koordinator	Arif Hidayat, ST., MT.	
	1. Dyah Retno Sawitri, ST.	
	2. Agus Sumarjana, ST.	
	3. Suwati, S.Sos.	

Dynamic Connection Logging System for Mikrotik Router Board

Muhammad Tirta Mulia[1]
Ferry Mulyanto[2]
Jurusan Teknik Informatika, Universitas Pasundan, Jl. Setiabudi 193 Bandung 40153 [1,2)]
081221000140, mtirtamulia@gmail.com, tirta.mulia@unpas.ac.id, tirta.mulia@students.itb.ac.id
0818223978, ferrymulyanto@gmail.com, ferry@unpas.ac.id

Abstrak

Interkoneksi antar komputer baik lokal maupun internet telah menjadi kebutuhan bagi organisasi. Hal ini menyebabkan pengelolaan jaringan menjadi suatu kebutuhan pula. Pemantauan keadaan jaringan dapat dilakukan dengan melihat log aktivitas jaringan itu sendiri. Dikarenakan banyak sekali data log yang dapat dibangkitkan dalam satu hari saja, mengakibatkan hal ini akan tidak mungkin bila dilakukan tanpa alat bantu berupa software.

Perangkat-perangkat keras jaringan, dalam hal ini mikrotik router board, memiliki kemampuan untuk membangkitkan log aktvitas. Beberapa alat memiliki opsi terhadap log tersebut, salah satunya adalah opsi untuk menyimpan log demi kebutuhan penelusuran (tracing) dimasa yang akan datang. Namun fitur ini akan mengurangi perfomansi fungsional utama perangkat serta terbatas oleh kapasitas daya tampung dari perangkat itu sendiri. Oleh karena itu dibutuhkan pengolahan log file yang terpisah dari perangkat tersebut.

Pengolahan data log akan unik untuk setiap organisasi, sehingga ekstraksi informasi yang spesifik tentunya membutuhkan alat bantu yang spesifik pula. Untuk itu, membuat aplikasi sendiri dalam pengolahan log file ini dapat menjadi suatu pilihan dimana dapat ditentukan atribut-atribut penyaringan data pada log file serta perangkaian informasi yang sesuai dengan kebutuhan.

Kata kunci : interkoneksi, pemantauan, sistem log, perangkat lunak

I. Pendahuluan

Latar Belakang

Internet telah menjadi kebutuhan individu maupun organisasi. Selain memberikan manfaat, penggunaan internet juga memiliki resiko. Ketika komputer maupun jaringan komputer suatu organisasi terhubung ke internet, maka akan beresiko terkena serangan atau percobaan penyusupan (*intrussion*) baik oleh manusia maupun *botnet*. Bila ini terjadi maka akan mengancam sistem informasi organisasi tersebut.

Keamanan informasi merupakan suatu keharusan dari sebuah sistem informasi. Untuk menjamin hal ini, dikembangkan sistem deteksi (IDS) dan pencegahan (IPS). Sistem deteksi ini tidak akan bekerja tanpa adanya sistem yang mengumpulkan informasi aktivitas jaringan atau sistem *logging*.

Log berisi semua aktivitas pada jaringan, sehingga *log* sangat bermanfaat bagi administrator jaringan untuk mengetahui keadaan jaringan saat ini maupun untuk analisa statistik tertentu. Analisa statistika ini bisa membantu untuk pengambilan keputusan terhadap manajemen jaringan. Misalnya, seberapa besar *bandwidth* sebaiknya dialokasikan untuk suatu layanan atau seorang pengguna jaringan, layanan apa yang paling sering diakses dan oleh siapa.

Pada router/server Mikrotik, data dari s*ystem logging* secara default hanya ditampilkan (*echo*) saja pada antarmuka pengguna baik *console* maupun *window* (menggunakan aplikasi Winbox). Namun *log* ini juga bisa disimpan dalam beberapa pilihan seperti disimpan pada *disk*, *memory*, maupun pada *server remote*. Bila perangkat mikrotik ini berupa perangkat keras *router board*, tentunya pilihan pertama dan kedua tidak bisa kita lakukan. Untuk itu pilihan mengirim data *log* ke perangkat lain akan lebih efektif bila menginginkan kemampuan penyimpanan data *log* yang besar dan fleksibilitas dalam menganalisa data tersebut tanpa membebani proses penyimpanan pada perangkat Mikrotik itu sendiri.

Data yang besar dapat disimpan ke dalam *database management system.* Fleksibilitas dalam mengekstraksi informasi dari data *log* dapat dicapai dengan menggunakan aplikasi untuk menganalisanya. Untuk menghasilkan informasi yang spesifik terhadap kebutuhan dapat dilakukan dengan membangun aplikasi pengolahan logging sendiri.

Tujuan

Pada tulisan ini akan dipaparkan mengenai logging system pada Mikrotik, cara mengekstraksi informasinya, dan pembuatan aplikasi untuk pengolahan logging beserta algoritma untuk *parsing* data pada *log*.

Batasan Masalah

Pada tulisan ini dilakukan pembatasa metodologi yang digunakan dalam mengolah data *log.* Metodologi yang diguanakan adalah *signature-based* yang mana akan dijelaskan pada bab selanjutnya.

Metodologi Penelitian

Metodologi yang digunakan dalam pembuatan tulisan ini dilakukan melalui tahapan-tahapan sebagai berikut :

1. Studi pustaka, untuk mendapatkan informasi mengenai pemantauan jaringan, *log system*, metode umum pengolahan log serta karakteristik perangkat *router board* Mikrotik.
2. Eksplorasi perangkat, tahapan ini dilakukan untuk uji coba perangkat pada jaringan komputer yang fungsional sehingga didapat data log interkoneksi sesungguhnya.
3. Rekayasa perangkat lunak, dimulai dari tahapan analisis kebutuhan, perancangan serta implementasi perangkat lunak.

II. Landasan Teori

Logging

Logging merupakan proses pencatatan aktivitas dalam suatu sistem. Pada IDS suatu jaringan komputer, data dari sistem log dapat digunakan untuk mengkonfirmasi validitas *alert* dan memeriksa insiden [1]. Data log ini terdiri dari beberapa field. Pada Mikrotik tanggal dan waktu even, type, mac-address, protokol, alamat IP sumber, alamat IP tujuan, panjang pesan dan IP dari router Mikrotik yang mengirim pesan log ini.

Metodologi Umum Pengolahan Logging

Data pada log tidak akan memiliki arti jika tidak diolah. Pengolahan ini memiliki tujuan untuk mengekstraksi informasi dari log. Beberapa metoda yang umum digunakan dalam pengolahan log adalah sebagai berikut [1]:

Signature-based Detection

Sebuah *signature* adalah pola yang berhubungan informasi yang akan diambil. Pada IDS *signature* adalah pola *threat* yang telah dikenali sebelumnya.

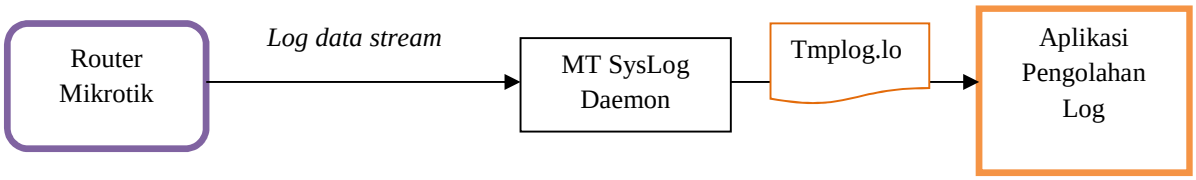
Anomaly-based detection
Metode ini membandingkan definisi seperti apa bagi suatu aktivitas dianggap normal terhadap even yang diobservasi untuk mengetahui deviasi yang signifikan.

Stateful Protocol Analysis
Adalah proses yang membandingkan profil definisi dari aktivitas protokol yang telah ditentukan sebelumnya untuk setiap keadaan protokol terhadap even yang diobservasi sehingga mendapatkan deviasinya. Tidak seperti anomaly-based detection yang menggunakan profil host atau spesifik jaringan, pada metode ini tergantung pada profil universal yang dikembangkan vendor yang menentukan bagaiaman protokol tertentu digunakan dan tidak boleh digunakan.

III. Perancangan

Arsitektur Sistem Secara Umum

Pada sistem yang dirancang, akan terdiri dari tiga bagian yaitu router Miktorik, perangkat lunak Mikrotik Syslog Daemon dan aplikasi pengolahan log. Berikut ini aliran kerja pada sistem yang akan dirancang secara umum.



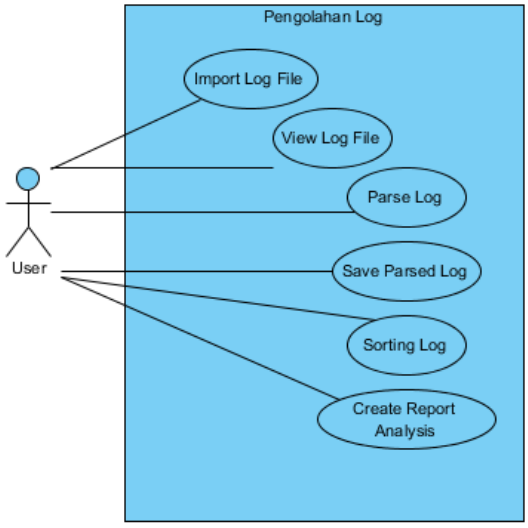
Gambar 1 Aliran kerja sistem secara umum

Seperti yang telah disebutkan pada pendahuluan, bahwa router mikrotik dapat melakukan *generate* log namun hanya sebatas menampilkan saja(*echo*). Pada keadaan ini dapat dipilih opsi untuk mengirimkan log ke perangkat komputer lain dengan fungsi *remote* dengan mendefinisikan terlebih dahulu alamat Ipnya. Komputer yang akan menampung *log* harus menjalankan aplikasi yang disediakan oleh Mikrotik, yaitu *Syslog Daemon*. Aplikasi ini berfungsi menangkap *streaming data* dari router dan menyimpannya dalam bentuk *file* dengan ekstensi '*.log*'.

File '*.log*' inilah yang akan diolah oleh aplikasi selanjutnya. Detil perancangan aplikasi pengolahan log akan dibahas pada bagian selanjutnya.

Perancangan Sistem

Dalam perancangan sistem digunakan notasi UML [2]. Fungsional dari sistem sendiri ditunjukkan oleh diagram use case pada gambar 2.



Gambar 2 Fungsional Sistem

Detil dari setiap fungsional adalah sebagai berikut :

a. **Import log file** : fungsi untuk memilih *log file* yang akan diolah. Fungsi ini dibutuhkan karena *log file* bisa didapat lebih dari satu sumber atau perangkat router.

b. **View Log File** : adalah fungsi untuk melihat isi *log file* tanpa pengolahan.

c. **Parse Log** : digunakan untuk menguraikan isi *log file* sehingga didapat token yang diinginkan seperti tanggal, waktu, alamat IP, *mac address*, IP perangkat, protokol yang digunakan.

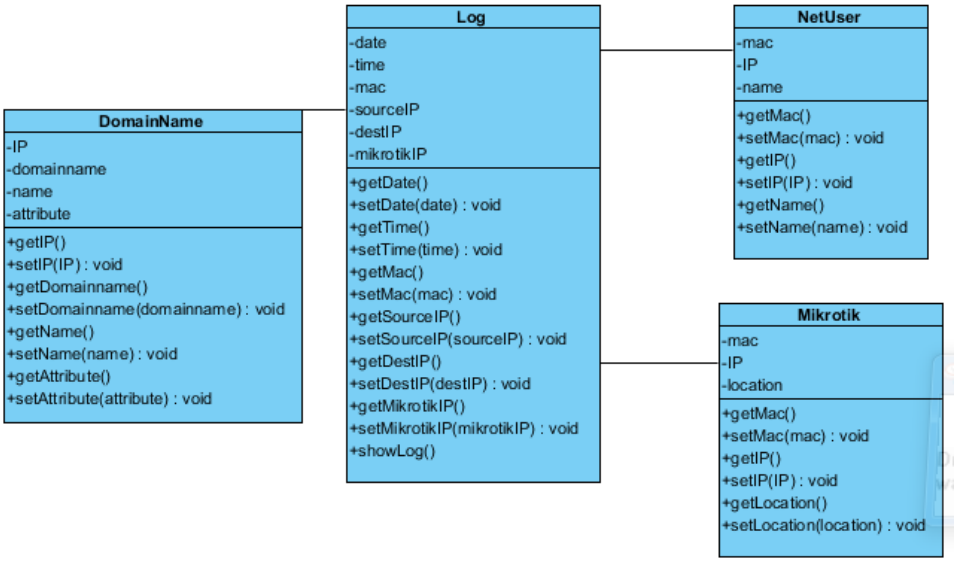
d. **Save Parsed Log** : fungsi ini digunakan untuk meyimpan hasil penguraian dari *log file* ke dalam *database*.

e. **Sorting Log** : user dapat mencari informasi dari *database* log dengan kriteria tertentu.

f. **Create Report** : user dapat mencetak hasil dari proses *sorting log*.

Entitas Sistem

Hal – hal utama pada sistem dapat digambar dengan diagram *class*. Berikut adalah diagram *class* yang menunjukkan entitas-entitas dari pada sistem yang dirancang.

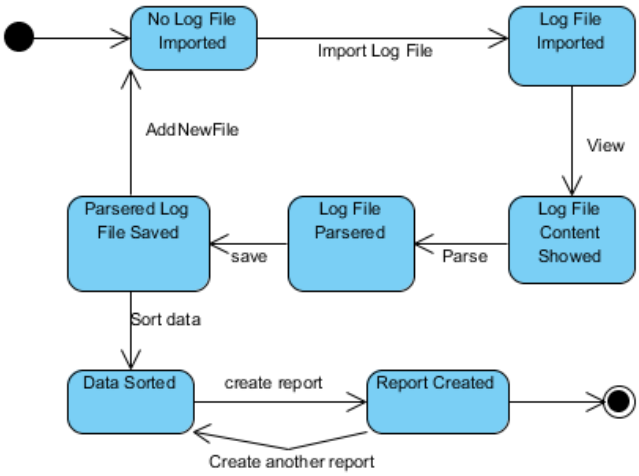


Gambar 3 Entitas-entitas pada sistem

Ada empat kelas(*class)* utama pada sistem ini. Yang pertama adalah kelas Log. Kelas ini menyimpan data mengenai aktivitas jaringan (log). Kelas kedua adalah kelas NetUser yang dirancang untuk menyimpan informasi pengguna dari local network. Kelas ketiga adalah Mikrotik yang menampung informasi mengenai perangkat router yang mengirim log. Hal ini akan memudahkan lokalisasi masalah jika terjadi insiden. Kelas terakhir adalah kelas DomainName yang mencatat semua domain beserta alamat IPnya. Sehingga bisa dilakukan penelusuran situs mana yang paling sering diakses dan oleh siapa yang mengaksesnya.

Diagram State

Alur agoritma yang digunakan dalam usecase yang telah digambarkan sebelumnya ditunjukkan oleh diagram *State* pada gambar 4.



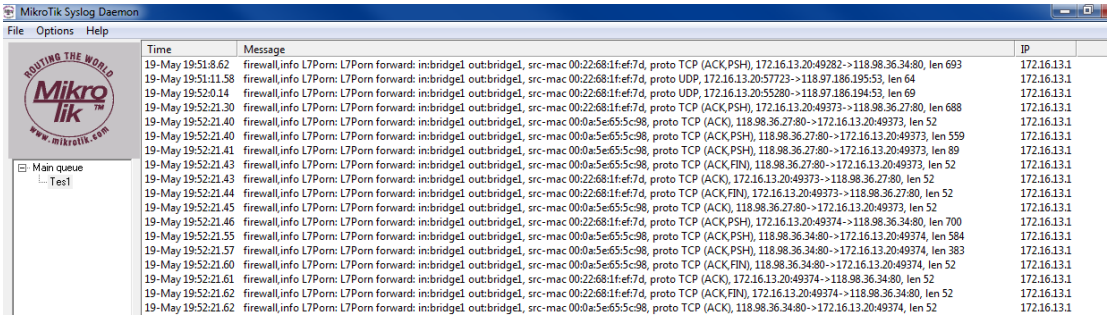
Gambar 4 Diagram state pengolahan log file

IV. Implementasi

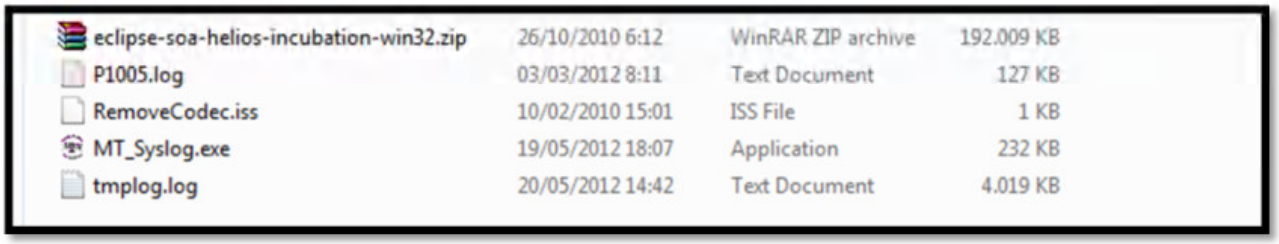
Pada bagian ini akan dibahas penerapan dari perancangan sistem. Namun konfigurasi untuk mengarahkan log router Mikrotik ke *remote server* tidak akan dibahas dikarenakan keterbatasan alokasi penulisan.

Pembangkitan *Log File*

Pembangkitan *log file* dilakukan pada komputer yang telah ditentukan melalui alamat IP, yaitu 172.16.12.93. Komputer dengan alamat tersebut akan berperan sebagai *log server*. Pada komputer ini cukup menjalankan aplikasi yang telah disediakan oleh Mikrotik yaitu *Mikrotik Syslog Daemon* untuk menangkap log dari perangkat. Saat dijalankan, aplikasi ini akan secara otomatis menerima log dari perangkat (gambar 5) dan menyimpannya berupa *file text* dengan nama *tmplog.log*. (gambar 6).



Gambar 5 Log diterima dari perangkat menggunakan aplikasi MT Syslog



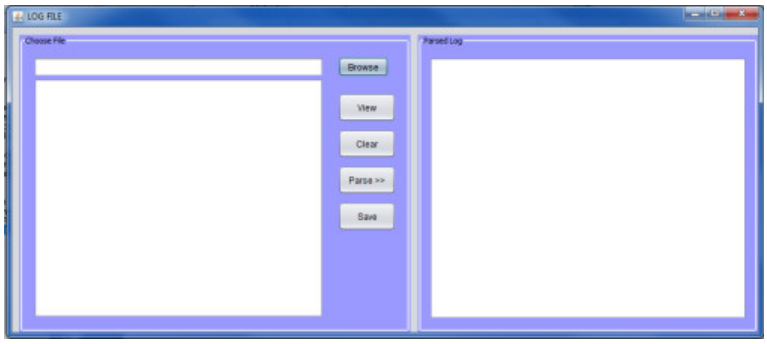
Gambar 6 Log file disimpan dengan nama tmplog.log pada direktori yang sama denga aplikasi MT SysLog

Aplikasi Pengolahan Log

Bagian terakhir dari sistem ini adalah aplikasi pengolahan log. Kriteria aplikasi ini adalah sebagai berikut :

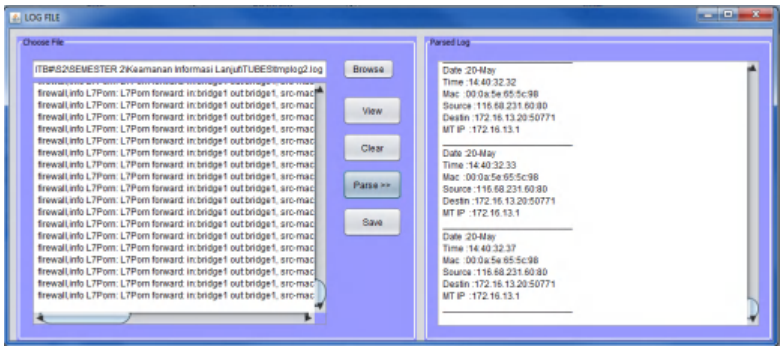
- Aplikasi ini dibangun dengan konsep berorientasi objek
- Menggunakan bahasa pemrograman Java.
- Data log database MySQL.

Antarmuka untuk keempat fungsi yang telah diimplementasikan tersebut ditunjukkan pada gambar 7.



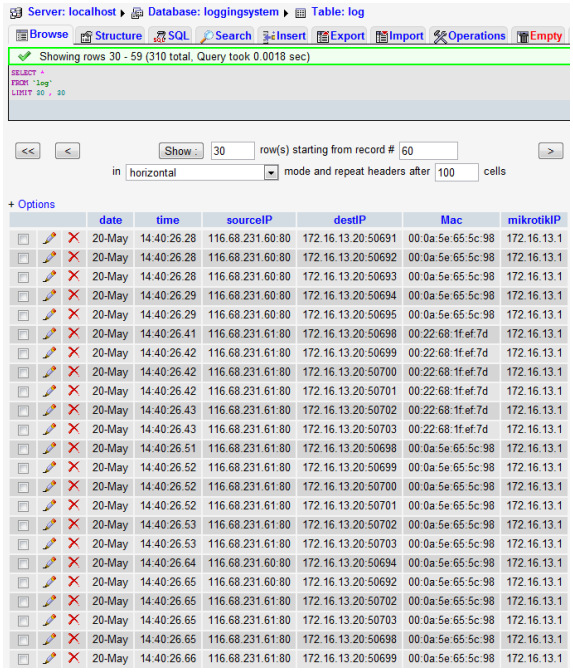
Gambar 7 Antarmuka aplikasi

Sisi kiri (gambar 7) digunakan untuk mengambil *log file* dan juga menampilkan isi dari *log file* tersebut (gambar 8). Padi sisi kanan adalah tempat untuk menampilkan hasil *parsing* dari *log file* sebelum disimpan ke *database* (gambar 8).



Gambar 8 Tampilan hasil proses View dan Parse Log File

Gambar 9 menunjukkan hasil pencacahan log file yang telah disimpan ke database.


Server: localhost ▸ Database: loggingsystem ▸ Table: log

Browse | Structure | SQL | Search | Insert | Export | Import | Operations | Empty

Showing rows 30 - 59 (310 total, Query took 0.0018 sec)

```
SELECT *
FROM `log`
LIMIT 29 , 30
```

<< < Show : 30 row(s) starting from record # 60 >

in horizontal mode and repeat headers after 100 cells

+ Options

	date	time	sourceIP	destIP	Mac	mikrotikIP
	20-May	14:40:26.28	116.68.231.60:80	172.16.13.20:50691	00:0a:5e:65:5c:98	172.16.13.1
	20-May	14:40:26.28	116.68.231.60:80	172.16.13.20:50692	00:0a:5e:65:5c:98	172.16.13.1
	20-May	14:40:26.28	116.68.231.60:80	172.16.13.20:50693	00:0a:5e:65:5c:98	172.16.13.1
	20-May	14:40:26.29	116.68.231.60:80	172.16.13.20:50694	00:0a:5e:65:5c:98	172.16.13.1
	20-May	14:40:26.29	116.68.231.60:80	172.16.13.20:50695	00:0a:5e:65:5c:98	172.16.13.1
	20-May	14:40:26.41	116.68.231.61:80	172.16.13.20:50698	00:22:68:1f:ef:7d	172.16.13.1
	20-May	14:40:26.42	116.68.231.61:80	172.16.13.20:50699	00:22:68:1f:ef:7d	172.16.13.1
	20-May	14:40:26.42	116.68.231.61:80	172.16.13.20:50700	00:22:68:1f:ef:7d	172.16.13.1
	20-May	14:40:26.42	116.68.231.61:80	172.16.13.20:50701	00:22:68:1f:ef:7d	172.16.13.1
	20-May	14:40:26.43	116.68.231.61:80	172.16.13.20:50702	00:22:68:1f:ef:7d	172.16.13.1
	20-May	14:40:26.43	116.68.231.61:80	172.16.13.20:50703	00:22:68:1f:ef:7d	172.16.13.1
	20-May	14:40:26.51	116.68.231.61:80	172.16.13.20:50698	00:0a:5e:65:5c:98	172.16.13.1
	20-May	14:40:26.52	116.68.231.61:80	172.16.13.20:50699	00:0a:5e:65:5c:98	172.16.13.1
	20-May	14:40:26.52	116.68.231.61:80	172.16.13.20:50700	00:0a:5e:65:5c:98	172.16.13.1
	20-May	14:40:26.52	116.68.231.61:80	172.16.13.20:50701	00:0a:5e:65:5c:98	172.16.13.1
	20-May	14:40:26.53	116.68.231.61:80	172.16.13.20:50702	00:0a:5e:65:5c:98	172.16.13.1
	20-May	14:40:26.53	116.68.231.61:80	172.16.13.20:50703	00:0a:5e:65:5c:98	172.16.13.1
	20-May	14:40:26.64	116.68.231.60:80	172.16.13.20:50694	00:0a:5e:65:5c:98	172.16.13.1
	20-May	14:40:26.65	116.68.231.60:80	172.16.13.20:50692	00:0a:5e:65:5c:98	172.16.13.1
	20-May	14:40:26.65	116.68.231.61:80	172.16.13.20:50702	00:0a:5e:65:5c:98	172.16.13.1
	20-May	14:40:26.65	116.68.231.61:80	172.16.13.20:50703	00:0a:5e:65:5c:98	172.16.13.1
	20-May	14:40:26.65	116.68.231.61:80	172.16.13.20:50698	00:0a:5e:65:5c:98	172.16.13.1
	20-May	14:40:26.66	116.68.231.61:80	172.16.13.20:50699	00:0a:5e:65:5c:98	172.16.13.1

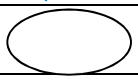
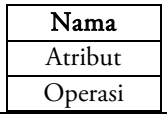
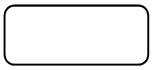
Gambar 9 Hasil parsing log file yang disimpan dalam database

V. Kesimpulan

Beberapa kesimpulan yang dapat diambil dari tulisan ini adalah :

- Log pada perangkat jaringan (dalam hal ini Mikrotik router board) dapat diolah lebih lanjut pada remote log server.
- Penyimpanan pada remote log server memungkinkan dilakukannya ekstraksi informasi dari log interkoneksi jaringan.
- Pembuatan aplikasi khusus untuk mengolah log file memberikan kebebasan dalam menentukan kriteria-kriteria informasi yang ingin diekstrak dari log.

Daftar Notasi

Notasi	Arti
Diagram *Use Case*	
	**Aktor** : seseorang atau sesuatu yang berinteraksi dengan sistem
	**Use case:** representasi fungsionalitas bagi seorang aktor
Diagram *Class*	
	**Class :** menyatakan *static view* dari sistem yang merupakan hal-hal pokok pada sistem.
	Relationhip : keterhubungan antar *class*.
Diagram *State*	
	**State :** hasil dari aktivitas sebelumnya yang dijalankan oleh objek
	**Initial :** keadaan mulai
	Event/Activity : aktivitas yang merubah *state*
	**end :** akhir dari proses

Daftar Pustaka

[1] NIST, *Guide to Intrusion Detection and Prevention Systems (IDPS).*, 2007.

[2] Magnus Penker, Brian Lyons, David Fado Hans-Erik Eriksson, *UML 2 Toolkit*. USA: Wiley, 2004.

[3] Mikrotik Indonesia. (2012, Mei) Mikrotik. [Online]. http://mikrotik.co.id/download.php

[4] Hirondelle Systems. (2012, Mei) JavaPractices - Parse Text. [Online]. http://www.javapractices.com/topic/TopicAction.do?Id=87