

# Aplikasi Monitoring Perangkat dan Aktivitas Pengguna pada Jaringan Menggunakan *Protocol SNMP* dan *Squid Proxy*

Danur Wijayanto<sup>1,\*</sup>, Indra Waspada<sup>1</sup>

<sup>1</sup> Fakultas Sains dan Matematika, Departemen Ilmu Komputer/ Informatika  
Universitas Diponegoro

(*coresponding author*) danurwijayanto@gmail.com<sup>\*</sup>)

**Abstract**— Management and Information Technology Services Division (UP2TI) is a unit in the Faculty of Science of Mathematics, University of Diponegoro engaged in the care and management of everything relating to information technology. UP2TI manage network devices such as routers, switches, servers, and access point in the FSM UNDIP. With a lot of number of networking devices need to manage, admin UP2TI got problem if the monitoring of network devices is done manually, simply relying based on client report and there is no system to monitor internet activity. A solution to these problems is creating a network monitoring system This application can also be referred to as the Network Management System that help system administrators to monitor and control the complex network SNMP is protocol TCP/IP that used as a protocol in the Network Management System. A system developed by using log squid proxy to determine the activities of internet users. The application development using waterfall method with PHP programming using CodeIgniter framework and MySQL database management system. Once the development is completed, it followed by black-box testing. The result of this research is an application that help admin in monitoring devices and user activities on the network Diponegoro University Faculty of Science Mathematics.

**Keyword**— *monitoring application, network management system (NMS), SNMP, Squid Proxy.*

**Intisari**— Unit Pengelolaan dan Pelayanan Teknologi Informasi (UP2TI) adalah salah satu unit di Fakultas Sains Matematika Universitas Diponegoro yang bergerak di bidang pelayanan dan pengelolaan segala sesuatu yang berkaitan dengan teknologi informasi. UP2TI mengelola perangkat jaringan seperti *router, switch, server, dan access point* di FSM UNDIP. Dengan banyaknya perangkat jaringan yang dikelola, *admin* UP2TI mengalami kesulitan jika proses *monitoring* perangkat jaringan dilakukan secara manual yaitu hanya mengandalkan laporan dari *client* jika ada permasalahan pada jaringan dan juga belum ada sistem untuk *memonitor* aktifitas pengguna internet. Solusi atas permasalahan tersebut dengan membuat aplikasi *monitoring* jaringan. Aplikasi *monitoring* bisa juga disebut sebagai *Network Management System* yaitu suatu sistem yang berfungsi untuk membantu *system administrator* dalam *memonitor* dan mengontrol perangkat jaringan yang kompleks SNMP merupakan sebuah protocol aplikasi pada jaringan TCP/ IP yang dapat digunakan sebagai *protocol* dalam *Network Management System*. Aplikasi *monitoring* yang dikembangkan juga melakukan pembacaan *log squid proxy* untuk mengetahui aktivitas pengguna internet. Pengembangan aplikasi ini menggunakan metode *waterfall* dengan bahasa pemrograman PHP dengan *framework CodeIgniter* dan sistem manajemen basis data MySQL. Setelah pengembangan selesai dilakukan, dilanjutkan pengujian secara *black-box*. Hasil akhir dari penelitian ini adalah aplikasi yang memudahkan *admin* dalam *memonitoring* perangkat dan aktivitas pengguna pada jaringan Fakultas Sains Matematika Universitas Diponegoro.

**Kata Kunci**— *aplikasi monitoring, network management system (NMS), SNMP, Squid Proxy.*

## I. PENDAHULUAN

Unit Pengelolaan dan Pelayanan Teknologi Informasi (UP2TI) adalah salah satu unit di Fakultas Sains Matematika Universitas Diponegoro yang bergerak di bidang pelayanan dan pengelolaan segala sesuatu yang berkaitan dengan teknologi informasi. UP2TI mengelola perangkat jaringan seperti *router, switch, server, dan access point* di FSM UNDIP. Dengan banyaknya perangkat jaringan yang dikelola, *admin* UP2TI mengalami kesulitan jika proses *monitoring* perangkat jaringan dilakukan secara manual yaitu hanya mengandalkan laporan dari *client* jika ada permasalahan pada jaringan.

UP2TI membutuhkan aplikasi *monitoring* jaringan yang berfungsi untuk mengetahui kondisi perangkat jaringan sehingga akan diketahui sumber masalah dalam jaringan, dan menampilkan aktifitas pengguna Internet. Selama ini belum ada sistem yang memantau kondisi perangkat jaringan di FSM Undip sehingga apabila terjadi masalah seperti

server, router, dan switch mati *admin* masih mengandalkan laporan dari pengelola perangkat yang bersangkutan, dan juga belum ada sistem untuk *memonitor* aktifitas pengguna Internet. Untuk mengatasi permasalahan di atas maka diperlukan *Network Management System* (NMS).

NMS merupakan suatu sistem yang berfungsi untuk membantu *system administrator* dalam *memonitor* dan mengontrol perangkat jaringan yang kompleks. SNMP merupakan sebuah *protocol* aplikasi pada jaringan TCP/ IP yang dapat digunakan sebagai *protocol* dalam NMS. SNMP diperkenalkan pada tahun 1988 yang merupakan standard untuk mengelola peralatan *Internet Protocol* (IP). SNMP menyediakan operasi – operasi yang memungkinkan peralatan untuk dikelola secara jarak jauh seperti mematikan *interface Ethernet*, melakukan *check* kecepatan *ethernet interface* yang sedang beroperasi, *memonitor* suhu, dan memperingatkan kepada *administrator* jaringan jika suhu terlalu panas [1].

Beberapa penelitian mengenai NMS dan SNMP telah dilakukan sebelumnya, penelitian pertama menghasilkan aplikasi *monitoring* jaringan menggunakan SNMP yang memungkinkan *administrator* jaringan untuk memantau performa peralatan jaringan, untuk mendeteksi, mengisolasi dan *recovery* kesalahan dari peralatan jaringan, untuk mendapatkan statistik data yang diterima dan dikirimkan oleh perangkat, dan untuk mengontrol jaringan secara jarak jauh [2]. Penelitian kedua menghasilkan suatu aplikasi manajemen jaringan berbasis PHP dan SNMP yang menggunakan sistem operasi *Linux Mandrive Limited Edition* 2005. Aplikasi tersebut menginformasikan status *link* dari suatu segmen jaringan, sehingga *admin* jaringan akan diketahui sumber atau letak masalah dalam suatu jaringan tersebut [3]. Penelitian selanjutnya menghasilkan aplikasi *monitoring* jaringan berbasis web yang digunakan sebagai perantara untuk mengolah nilai SNMP yang disimpan dalam database MySQL yang kemudian digunakan untuk menampilkan informasi tentang kondisi jaringan yang meliputi *availability* perangkat dan trafik pada *transport* TCP [4].

Penelitian mengenai pembacaan aktivitas pengguna Internet menggunakan *log squid proxy* sebelumnya telah dilakukan oleh Zulfikri Rifqi Hakim yang menghasilkan aplikasi pembacaan *log* memanfaatkan *file log squid web proxy* sehingga dapat dimanfaatkan untuk analisis *digital forensic* saat terjadi *cyber crime* dan analisis statistik pengguna Internet di lingkungan FSM Undip [5].

Berdasarkan penelitian yang ada, *protocol* SNMP dan pemanfaatan *squid web proxy* dirasa cukup efektif untuk digunakan dalam membangun sistem *monitoring* jaringan atau NMS. Maka dari itu, dalam penelitian ini, akan merancang aplikasi *monitoring* jaringan atau NMS menggunakan *protocol* SNMP dan *squid web proxy* yang akan digunakan untuk mempermudah *admin* dalam *memonitor* kondisi perangkat jaringan dan aktivitas pengguna Internet di lingkungan FSM Undip.

## II. TINJAUAN PUSTAKA

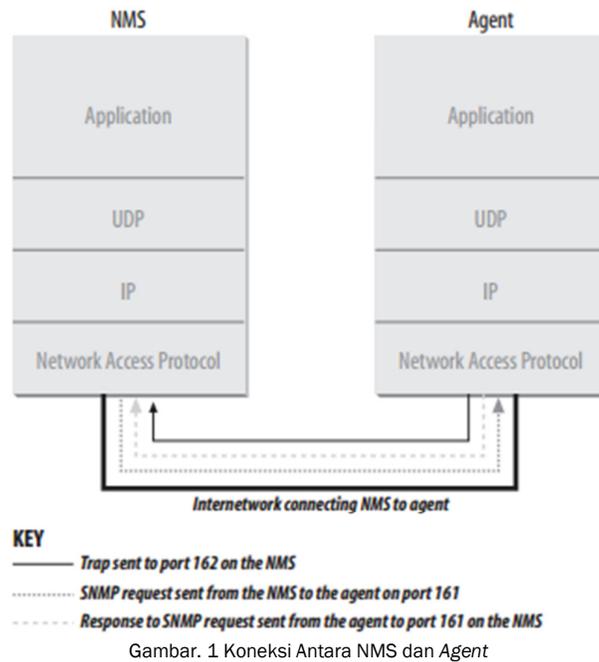
Pada bab ini menyajikan teori yang digunakan dalam penelitian, yaitu Protocol SNMP dan NET-SNMP.

### A. Protocol SNMP

NET-SNMP SNMP adalah standar protocol Internet untuk mengelola perangkat pada jaringan secara jarak jauh yang diperkenalkan pada tahun 1988. Protocol ini berada pada application layer. Beberapa perangkat mendukung protocol SNMP seperti routers, switches, dan servers. SNMP digunakan untuk melakukan *monitoring* terhadap kondisi dari routers, servers, dan perangkat jaringan lainnya yang mendukung protocol tersebut serta dapat menampilkan informasi seperti lalu lintas data yang keluar dan masuk interface dan temperature perangkat. [1].

SNMP menggunakan User Datagram Protocol (UDP) sebagai protocol transportasi untuk berkomunikasi antara manager dan agent karena sifatnya yang *connectionless* (tidak terjadi proses *acknowledgement* antara host pengirim dan penerima) sehingga mempunyai kecepatan komunikasi data yang tinggi. SNMP menggunakan UDP port 161 untuk mengirim dan menerima permintaan dan port 162 untuk menerima traps dari managed device.

Pada *application layer*, aplikasi SNMP (NMS atau *manager*) memutuskan apa yang akan dilakukan. Misalnya NMS mengirim SNMP *request* kepada *agent*, atau *agent* mengirim *trap* ke NMS. Layer aplikasi menyediakan *informasi* kepada *user*, seperti status dari *port Ethernet Switch*. Layer selanjutnya yaitu *transport layer*. Pada layer ini SNMP memanfaatkan *protocol* UDP untuk berkomunikasi. Pada layer Internet, SNMP menggunakan IP untuk menentukan tujuan *packet*. Layer terakhir yaitu *Network Access* dimana SNMP *packet* diterima oleh *host* tujuan berdasarkan MAC Address [1].



Gambar. 1 Koneksi Antara NMS dan Agent

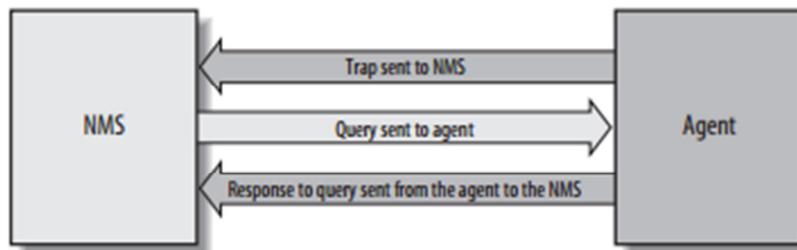
SNMP terdiri dari tiga elemen yaitu :

a. *Manager*

*Manager* adalah bagian dari SNMP selain *agents*. Sebuah *manager* merupakan sebuah server yang menjalankan beberapa macam perangkat lunak yang berfungsi melakukan pengelolaan jaringan. *Manager* bisa juga disebut sebagai *Network Management System (NMS)*. *Manager* bertanggung jawab dalam berkomunikasi dengan *agent* pada perangkat jaringan yang dimonitor untuk mendapatkan informasi. [1].

b. *Agent*

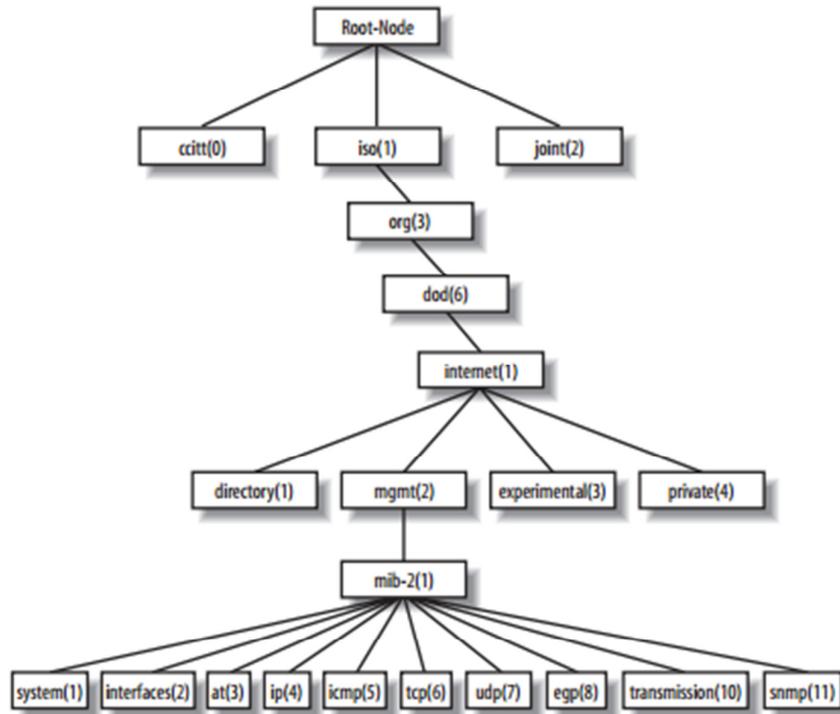
*Agent* adalah perangkat lunak yang berjalan pada perangkat jaringan yang di-*manage*. *Agent* dapat berupa perangkat lunak terpisah (*daemon*), atau yang tergabung di dalam sistem operasi (CISCO IOS pada CISCO Router). *Agent* menyediakan informasi kepada NMS dengan menjaga *track* dari banyak *operational aspect* dari perangkat. Sebagai contoh, *agent* di *router* dapat menjaga *track* dari status setiap *interface* :*interface* mana yang aktif, dan *interface* mana yang mati. Ketika *agent* melihat ada masalah terjadi, *agent* akan mengirim *trap* kepada NMS [1].



Gambar. 2 Interaksi Pesan Trap antara NMS dan Agent

c. *Management Information Base (MIB)*

MIB merupakan tempat penyimpanan informasi yang dimiliki oleh agen yang nantinya diakses oleh *Network Management System (NMS)*. Informasi yang disimpan oleh MIB menggunakan diagram pohon dan menempatkan *Object Identifier (OID)* pada setiap *node* pohon. OID berupa integer yang dipisahkan oleh dots (.).



Gambar. 3 MIB Tree

Objek – objek informasi SNMP memperoleh kedudukan di bawah *node mib-2(1)*. Untuk mengakses objek – objek tersebut maka harus dituliskan sesuai urutan node seperti yang dijelaskan pada tabel 1.

Tabel 1 MIB-II Groups

Subtree name	OID	Deskripsi
<i>sistem</i>	1.3.6.1.2.1.1	Medefinisikan objek yang berhubungan dengan sistem operasi, <i>uptime</i> , <i>system contact</i> , dan <i>system name</i>
<i>interface</i>	1.3.6.1.2.1.2	Melacak status <i>interface</i> perangkat.
<i>at</i>	1.3.6.1.2.1.3	Alamat terjemahan
<i>ip</i>	1.3.6.1.2.1.4	Melacak beberapa aspek dari IP, termasuk <i>IP Routing</i>
<i>icmp</i>	1.3.6.1.2.1.5	Melacak <i>ICMP error</i>
<i>tcp</i>	1.3.6.1.2.1.6	Melacak koneksi TCP
<i>udp</i>	1.3.6.1.2.1.7	Melacak statistik UDP.
<i>egp</i>	1.3.6.1.2.1.8	Melacak statistik dari <i>Exterior Gateway Protocol</i> (EGP) dan menyimpannya dalam EGP tabel
<i>transmission</i>	1.3.6.1.2.1.9	Tidak ada objek yang didefinisikan pada grup ini, tetapi ada beberapa perangkat medefinisikan pada objek ini.
<i>snmp</i>	1.3.6.1.2.1.10	Mengukur kinerja SNMP seperti melacak paket yang dikirim dan diterima

*Protocol Data Unit* (PDU) adalah format pesan yang digunakan oleh *manager* dan *agent* untuk mengirim dan menerima informasi. Jenis – jenis PDU dijelaskan pada table 2 berikut.

Tabel 2 Format PDU

Jenis	Versi SNMP	Keterangan
<i>get</i>	SNMPv1	Permintaan untuk mendapatkan sebuah informasi dari sebuah objek tertentu
<i>getnext</i>	SNMPv1	Permintaan untuk mendapatkan sebuah informasi dari sebuah objek tertentu berikutnya
<i>getbulk</i>	SNMPv1, SNMPv2 dan SNMPv3	Permintaan untuk mendapatkan informasi dari beberapa objek tertentu sekaligus
<i>set</i>	SNMPv1	Perintah untuk menuliskan sebuah informasi ke sebuah objek tertentu
<i>getResponse</i>	SNMPv1	Tanggapan <i>Agnet</i> SNMP terhadap permintaan NMS.
<i>trap</i>	SNMPv1	Pemberitahuan dari <i>Agent</i> SNMP kepada NMS
<i>notification</i>	SNMPv1, SNMPv2 dan SNMPv3	Membakukan PDU format untuk SNMPv1 <i>traps</i>
<i>inform</i>	SNMPv1, SNMPv2, dan SNMPv3	Pemberitahuan dari sebuah NMS kepada NSM lain
<i>report</i>	SNMPv1, SNMPv2, dan SNMPv3	Tanggapan <i>agent</i> SNMP yang berisi laporan statistik SNMP

## B. NET-SNMP

*Net-SNMP* merupakan aplikasi *open source* yang mengimplementasikan SNMP. Dikembangkan di Universitas Carnegie Mellon (sekitar tahun 1995) dan Universitas California, Davis (antara tahun 1995 dan 2000). Sekarang dikembangkan oleh *Net-SNMP development team* (sejak 2000) yang disimpan di <http://sourceforge.net/projects/net-snmp/>. *Net-SNMP* menyediakan [6] :

1. Agen dengan *MIB compiler* yang dapat digunakan untuk mengembangkan MIB.
2. Library SNMP yang bisa digunakan untuk pengembangan selanjutnya.
3. Peralatan yang digunakan untuk memperoleh atau men-set informasi dari agen.
4. Peralatan untuk menghasilkan dan menangani SNMP traps dan mendukung SNMPv1, v2, dan v3.

Tabel 3 menampilkan deskripsi beberapa perintah di Net-SNMP dan PDU yang sesuai.

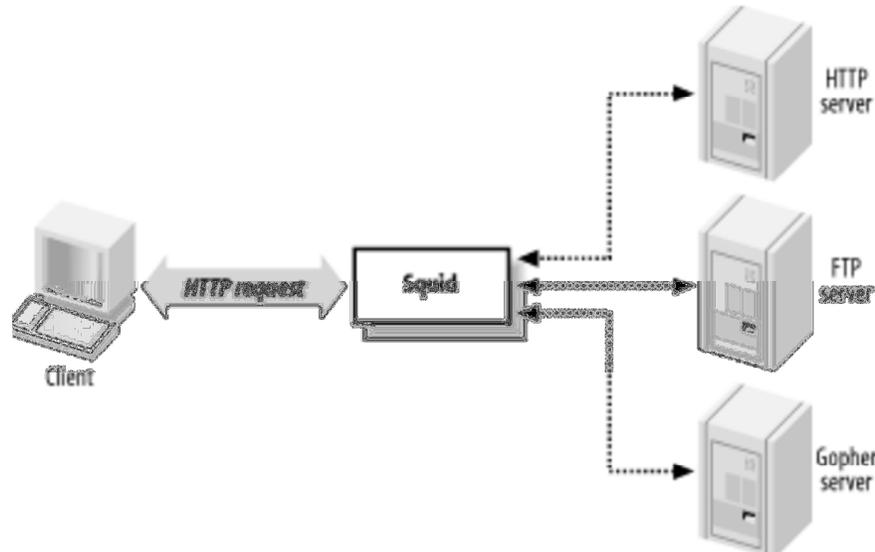
Tabel 3 Beberapa Perintah di Net-SNMP [6]

Nama	Deskripsi	PDU
<i>snmpget</i>	Menerima nilai objek pada node.	<i>GetRequest</i>
<i>snmpset</i>	Set nilai objek pada node	<i>SetRequest</i>
<i>snmpbulkget</i>	Mendapatkan banyak objek dalam satu waktu dengan <i>subtree</i> yang berbeda	<i>GetBulkRequest</i>
<i>snmpwalk</i>	Menjelajahi semua objek dibawah <i>subtree</i> MIB	<i>GetNextRequest</i>
<i>snmptrap</i>	Menggunakan permintaan TRAP untuk mengesir informasi ke <i>network manager</i> .	<i>Trap</i>
<i>snmstatus</i>	Digunakan untuk menerima beberapa statistik penting dari entitas jaringan	
<i>snmpnetstat</i>	Menampilkan nilai dari berbagai informasi yang berhubungan dengan jaringan menggunakan <i>protocol</i> SNMP	

### C. SQUID

*Squid* merupakan *open source caching proxy* yang populer. *Squid* dapat bekerja sebagai *proxy* dan *cache*. Sebagai *proxy*, *squid* berperan sebagai perantara transaksi antara *client* dan *server*. *Squid* menerima permintaan dari *client*, kemudian memproses dan meneruskannya ke server tujuan. *Request* tersebut dapat di *log*, ditolak, dan dimodifikasi sebelum diteruskan ke *server* tujuan. Sebagai *cache*, *squid* menyimpan konten web yang diterima untuk digunakan setelahnya.

Pada gambar 4 menunjukkan *squid* menerima permintaan HTTP dan HTTPS dari *client*, dan berkomunikasi dengan beberapa protokol server. Secara umum, *squid* mengetahui bagaimana berkomunikasi dengan HTTP, FTP, dan *Gopher servers*. [7]



Gambar. 4 Squid Proxy

### III. IMPLEMENTASI DAN CONTOH KASUS

Pada bab ini menyajikan implementasi pada penelitian ini berupa desain jaringan dan contoh kasus.

#### A. Desain Topologi

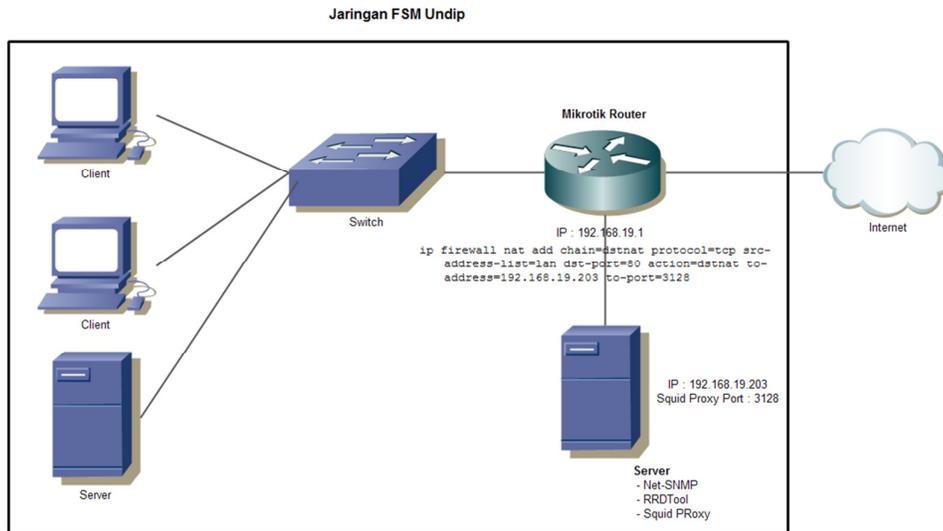
Gambar 4 memperlihatkan desain dari pengujian Aplikasi Monitoring Aktivitas dan Perangkat Jaringan.

Berikut perangkat *hardware* yang digunakan beserta spesifikasinya:

1. Server yang digunakan sebagai *manager* dengan spesifikasi :

- Intel i5-3220M 2.4 Ghz
- 4GB RAM DDR3
- 640GB HDD
- Ubuntu Desktop 14.04 LTS
- PHP versi 5.5
- Web Server Apache versi 2.4
- Net-SNMP versi 5.7.3
- Squid Proxy versi 3.1.20

2. Router Mikrotik Dinara untuk melakukan konfigurasi NAT pada *squid proxy*.



Gambar. 5 Desain Topologi

B. Contoh Kasus dan Analisa Hasil

Skenario 1

Kasus : Terjadi perubahan status perangkat

Hasil : Mengirimkan notifikasi email ke pengguna

Analisa :

Pada gambar 5 menunjukkan daftar semua perangkat yang telah didaftarkan pada aplikasi. Dalam kasus ini, perangkat yang digunakan yaitu Mikrotik Statistika. Dalam kondisi awal, Mikrotik Statistika mempunyai status *UP*. Dalam kasus ini, tiba – tiba perangkat Mikrotik Statistika mengalami perubahan status menjadi *down*. Daftar perangkat dapat dilihat pada gambar 6. Pada gambar tersebut dapat diketahui telah terjadi perubahan status perangkat Mikrotik Statistika menjadi *Down*. Selain status pada daftar perangkat berubah, aplikasi juga akan mengirim notifikasi ke email pengguna. Isi dari email dapat dilihat pada gambar 7. Dalam gambar tersebut dapat dilihat nama perangkat yang mengalami perubahan status, status lama, status baru dan waktu terjadinya perubahan status pada perangkat tersebut.

No	Nama Perangkat	Alamat IP	Status	Aksi
1	Mikrotik Dinara	182.255.0.34	Up	Edit Detail Hapus
2	Mikrotik IF Gedung E	172.20.5.3	Up	Edit Detail Hapus
3	Mikrotik IF Gedung B	172.20.5.2	Up	Edit Detail Hapus
4	Mikrotik Kimia	172.20.4.2	Up	Edit Detail Hapus
5	Mikrotik Statistika	172.20.6.2	Up	Edit Detail Hapus
6	Router Matematika	172.20.7.2	Up	Edit Detail Hapus

Gambar. 6 Tampilan Daftar Perangkat

No	Nama Perangkat	Alamat IP	Status	Aksi
1	Mikrotik Dinara	192.255.0.34	Up	Edit Detail Hapus
2	Mikrotik IF Gedung E	172.20.5.3	Up	Edit Detail Hapus
3	Mikrotik IF Gedung B	172.20.5.2	Up	Edit Detail Hapus
4	Mikrotik Kimia	172.20.4.2	Up	Edit Detail Hapus
5	Mikrotik Statistika	172.20.6.2	Down	Edit Detail Hapus
6	Router Matematika	172.20.7.2	Up	Edit Detail Hapus

Gambar. 7 Tampilan Daftar Perangkat

NMS FSM UNIDIP 4:00 PM (11 minutes ago) ☆

to me

Indonesian > English Translate message Turn off for: Indonesian x

Terjadi Perubahan Status Perangkat Pada :

No	Perangkat	Status Lama	Status Baru	Waktu
1	Mikrotik Statistika	Up	Down	2016-08-28 16:00:06

Terjadi Perubahan Status Interface Pada :

No	Perangkat	Interface	Status Lama	Status Baru	Waktu
----	-----------	-----------	-------------	-------------	-------

Gambar. 8 Notifikasi Email

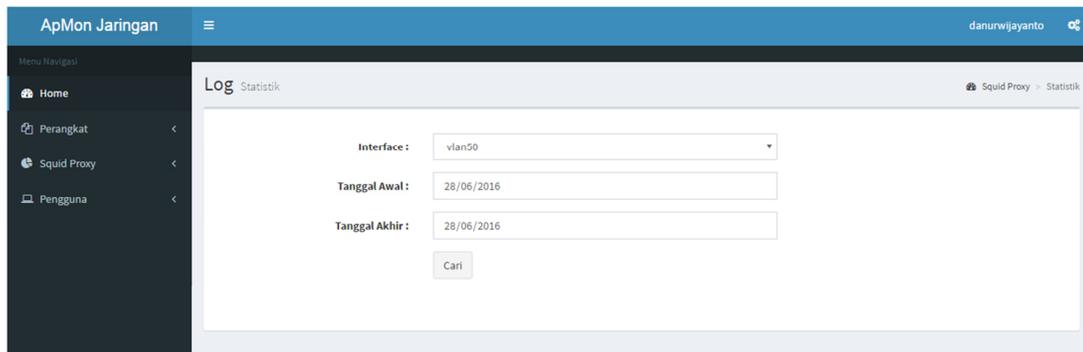
## Skenario 2

Kasus : Pencarian data *log squid*

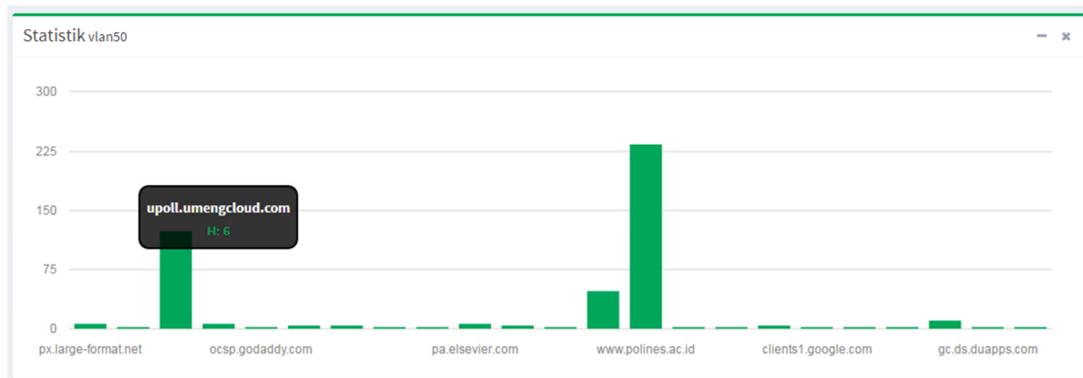
Hasil : Menampilkan statistik hasil pencarian

Analisa :

Untuk melakukan pencarian, pengguna harus memasukkan parameter pencarian pada *form* yang telah disediakan. Parameter – parameter tersebut adalah *interface*, waktu awal dan waktu akhir. Tampilan *form* dapat dilihat pada gambar 8. Setelah semua parameter pencarian telah dimasukkan kemudian pengguna mengakses tombol cari. Setelah itu aplikasi akan menampilkan informasi nama situs, dan jumlah pengakses dalam bentuk grafik yang dapat dilihat pada gambar 9.



Gambar. 9 Pilihan Pencarian



Gambar. 10 Hasil Pencarian

#### IV. KESIMPULAN

Kesimpulan yang dapat diambil dari hasil penelitian ini adalah :

- Menghasilkan aplikasi *monitoring* perangkat dan pengguna menggunakan *protocol* SNMP dan *Squid Proxy* yang diimplementasikan di Fakultas Sains dan Matematika Universitas Diponegoro.
- Aplikasi melakukan pengecekan status perangkat dan *interface* setiap 3 menit.
- Aplikasi mendukung protokol SNMP versi 1, 2 dan 3.
- Aplikasi dapat membantu *admin me-monitoring* perangkat jaringan dan memantau aktifitas pengguna Internet.
- Aplikasi dapat mengetahui informasi perangkat seperti status *interface*, waktu keadaan *up*, jumlah *traffic* masuk dan keluar, IP Address, *subnet mask*, dan jumlah *interface* pada perangkat yang dikelola
- Aplikasi dapat menyimpan *traffic* masuk dan keluar pada *interface* perangkat yang disimpan dalam RRDTOol.
- Aplikasi dapat mengetahui asal pengguna jaringan Internet dengan memanfaatkan IP asal *log squid* dan data *interface* perangkat yang dikelola.

## REFERENSI

- [1] Douglas R Mauro and Kevin J Schmidt, *Essential SNMP 2nd Edition*, 2nd ed. United States of America: O`Reilly Media, 2005.
- [2] Abubucker Samsudeen Shaffi and Mohammed Al-Obaidy, "Managing Network Component Using SNMP," *International Journal of Scientific Knowledge*, vol. 2, 2012.
- [3] Sri Puji Utami, Surya Agustian, and Iman Fauzi Aditya Sayogo, "Perancangan Online Network Monitoring Berbasis PHP dan SNMP," *SNATI 2006*, 2006.
- [4] Reza Pradikta, Achmad Affandi, and Eko Setijadi, "Rancang Bangun Aplikasi Monitor Jaringan dengan Menggunakan SNMP," *Jurnal Teknik Pomits*, 2013.
- [5] Zulfikri Hakim Rifqi, "Sistem Pembacaan Log Penggunaan Internet Menggunakan SQUID PROXY," skripsi, Departemen Ilmu Komputer/ Informatika, Universitas Diponegoro, Semarang, 2013.
- [6] Ying-Dar Lin, Ren-Hung Hwang, and Fred Baker, *Computer Networks An Open Source Approach*. New York: McGraw-Hill, 2012.
- [7] Duane Wessels, *Squid: The Definitive Guide*. USA: O'Reilly Media, 2004.