



**SISTEM MONITORING SQUID PROXY MENGGUNAKAN SQUID
ANALYSIS REPORT GENERATOR DI JARINGAN INTERNET
WEBMEDIA TRAINING CENTER**

**Disusun dan Diajukan Untuk Memenuhi Persyaratan Ujian Akhir
Memperoleh Gelar Sarjana Strata-1 Pada Jurusan Sistem Komputer
Universitas Pembangunan Panca Budi
Medan**

SKRIPSI

OLEH

NAMA : MUHAMMAD KURNIAWAN RIZKY

N.P.M : 1624370676

PROGRAM STUDI : SISTEM KOMPUTER

**FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS PEMBANGUNAN PANCA BUDI
MEDAN
2019**

Sistem Monitoring Squid Proxy Menggunakan Squid Analysis Report Generator Di Jaringan Internet Webmedia Training Center

Muhammad Kurniawan Rizky*

Akhyar Lubis**

Barany Fachri**

Universitas Pembangunan Panca Budi

ABSTRAK

Penelitian ini bertujuan untuk merancang *proxy server* dengan menggunakan aplikasi *squid* yang dapat memantau aktifitas *user* di dalam jaringan. Sistem *monitoring proxy server* dirancang untuk mengontrol hak akses, *web filtering* melalui *Access Control List (ACL)* untuk membatasi situs *web* yang mengandung konten negatif yang belum sepenuhnya dapat di *filter* oleh pemerintah dan implementasi *local cache Squid* yang terhubung dengan *proxy server* di *Webmedia Training Center* mempercepat membuka situs *web* yang sering diakses. Sistem layanan *proxy server* adalah salah satu alternatif yang handal dalam menciptakan akses *internet* sehat atau positif sehingga untuk mengangkat permasalahan ini dengan membangun sistem layanan *proxy server* yang menggunakan Sistem Operasi *Ubuntu Server 14.04* dan aplikasi *proxy* nya menggunakan *Squid* dan monitoring nya menggunakan *SARG (Squid Analysis Report Generator)*.

Kata kunci : Proxy Server, Cache, Squid, Ubuntu Server, Filter

*Mahasiswa Jurusan Teknik Komputer : **Kurniawanrizky1210@gmail.com**

**Dosen Jurusan Teknik Komputer

Squid Proxy Monitoring System Using the Squid Analysis Report Generator on the Internet Network Webmedia Training Center

Muhammad Kurniawan Rizky*

Akhyar Lubis**

Barany Fachri**

University of Pembangunan Panca Budi

ABSTRACT

The purpose of this research is to design a proxy server using Squid application that can monitor user activities on the network. Proxy server monitoring system is designed to control access rights, web filtering through an Access Control List (ACL) to limit websites that contain negative content that has not been completely filtered out by the government and the implementation of the Squid local cache that is connected to the proxy server at the Webmedia Training Center open a website that is accessed frequently. The proxy server service system is one of the reliable alternatives in creating healthy or positive internet access so as to raise this problem by building a proxy server service system that uses the Ubuntu Server 14.04 Operating System and the proxy application using Squid and monitoring using SARG (Squid Analysis Report Generator)

Keyword : Proxy Server, Cache, Squid, Ubuntu Server, Filter

*Student of Computer Science : Kurniawanrizky1210@gmail.com

**Lecturer of Computer Science

DAFTAR ISI

KATA PENGANTAR.....	i
DAFTAR ISI.....	ii
DAFTAR GAMBAR.....	v
DAFTAR TABEL	vii

BAB I PENDAHULUAN

1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	3
1.3 Batasan Masalah	3
1.4 Tujuan Penelitian	3
1.5 Manfaat Penelitian	4
1.6 Metodologi Penelitian.....	4
1.7 Sistematika Penulisan	5

BAB II LANDASAN TEORI

2.1 Pengertian Jaringan Komputer	6
a. Berdasarkan Ruang Lingkup	8
b. Berdasarkan Arsitektur.....	10
c. Berdasarkan Kebutuhan.....	14
2.2 Peralatan Jaringan.....	16
2.3 Protokol Jaringan	20
2.4 Model Layer OSI.....	21
2.5 TCP/IP	24
2.6 IP Address.....	25
2.7 Kelas IP Address.....	27
2.8 Subnet Mask	29
2.9 Sistem Operasi	29
2.10 Linux.....	30
2.11 Struktur Direktori Linux	30
2.12 Squid	32
2.13 Proxy Server	32
2.14 Firewall.....	36
2.15 SARG (Squid Analysis Report Generator).....	36

BAB III ANALISIS DAN PERANCANGAN

3.1	Metode Pengumpulan Data.....	38
	a. Studi Lapangan.....	38
	b. Studi Pustaka.....	39
	c. Studi Literatur.....	40
3.2	Analisis Sistem Sedang Berjalan.....	41
3.3	Rancangan Topologi Baru.....	46
3.4	Rancangan IP Address.....	47
3.5	Perancangan Proxy Server.....	50
3.6	Perancangan Web Filter.....	52
3.7	Perancangan SARG.....	52

BAB IV HASIL DAN PEMBAHASAN

4.1	Spesifikasi Sistem.....	54
4.2	Pembuatan Server.....	56
	a. Instalasi Ubuntu Server 14.04.....	56
	b. Konfigurasi Gateway.....	57
	c. DHCP Server.....	59
4.3	Squid.....	59
	a. Instalasi Squid.....	59
	b. Konfigurasi Squid.....	60
	c. Konfigurasi Web Cache.....	60
	d. Konfigurasi Web Filter.....	61
	e. Konfigurasi IP Tables.....	62
4.4	SARG.....	63
	a. Instalasi SARG.....	63
	b. Konfigurasi SARG.....	63
4.5	Pengujian dan Analisi.....	64
	a. Pengujian Web Cache Proxy Server.....	64
	b. Pengujian Web Filter.....	67
	c. Pengujian SARG.....	68
	d. Analisis Web Cache Proxy Server.....	69
	e. Analisis SARG.....	69

BAB V PENUTUP

5.1 Kesimpulan	79
5.2 Saran	79

DAFTAR PUSTAKA

BIOGRAFI PENULIS

LAMPIRAN-LAMPIRAN

DAFTAR GAMBAR

Gambar 1 Local Area Network	8
Gambar 2 Metropolitan Area Network	9
Gambar 3 Wide Area Network	10
Gambar 4 Topologi Bus	11
Gambar 5 Topologi Ring	12
Gambar 6 Topologi Star.....	13
Gambar 7 Topologi Tree.....	13
Gambar 8 Topologi Mesh	14
Gambar 9 Jaringan Peer-to-peer.....	14
Gambar 10 Jaringan Client Server	15
Gambar 11 Kinerja Modem ADSL	16
Gambar 12 Network Interface Card.....	17
Gambar 13 Switch.....	18
Gambar 14 Kabel Coaxial.....	18
Gambar 15 Kabel UTP.....	19
Gambar 16 Kabel STP	19
Gambar 17 Fiber Optic.....	20
Gambar 18 Model Layer OSI.....	21
Gambar 19 Perbedaan Layer TCP/IP dengan OSI.....	24
Gambar 20 Konsep Proxy Server.....	35
Gambar 21 Metodologi Penyelesaian Masalah dalam Penelitian	39
Gambar 22 Tindakan yang dilakukan setelah Analisa Permasalahan.....	39
Gambar 23 Topologi Jaringan sedang berjalan.....	42
Gambar 24 Monitoring Netschool Support.....	43
Gambar 25 Grafik Pie Rincian Akses Client pada Jaringan Lama	45
Gambar 26 Topologi Jaringan Baru	46
Gambar 27 Topologi Baru di Lantai 3-1	49
Gambar 28 Flowchart Proxy Server.....	51
Gambar 29 Halaman Depan SARG	52
Gambar 30 Laporan SARG per Periode.....	53
Gambar 31 Topologi Jaringan.....	58
Gambar 32 Aktifitas File Access Log Squid	65

Gambar 33 Manajemen Direktori Squid.....	66
Gambar 34 Squid Proxy Web Block.....	67
Gambar 35 Halaman Depan SARG	68
Gambar 36 Laporan SARG Periode 16 Juli 2019.....	70
Gambar 37 Laporan SARG Periode 16 Juli 2019 Berdasarkan URL.....	72
Gambar 38 Grafik Penggunaan Kuota berdasarkan Domain.....	74
Gambar 39 Download Site	76
Gambar 40 Denied Site	78

DAFTAR TABEL

Tabel 1 Alamat IP Private	26
Tabel 2 Subnet Mask untuk tiap Kelas IP Address.....	29
Tabel 3 Studi Literatur	41
Tabel 4 Rincian Akses Client pada Jaringan Lantai 3	44
Tabel 5 Rancangan IP address	47
Tabel 6 Rincian File Access Log Squid.....	65
Tabel 7 Analisis Client Aktif	70
Tabel 8 Alamat Domain yang paling banyak di Akses.....	73
Tabel 9 Rincian Permintaan Domain yang di Blokir.....	78

BAB I

PENDAHULUAN

1.1 Latar Belakang

Jaringan komputer adalah satu atau lebih computer yang saling terhubung dan bisa saling berkomunikasi. Jaringan computer saat ini sangat dibutuhkan di berbagai kalangan dan instansi pemerintahan, kampus, dan bahkan untuk bisnis dimana banyak sekali perusahaan yang memerlukan informasi dan data-data dari kantor-kantor lainnya dan dari rekan kerja, afiliasi bisnis, dan konsumen. Namun jaringan komputer dan internet ini sering kali disalah gunakan oleh sebagian *user* untuk mengakses situs-situs yang terlarang dan juga *download* sering kali terjadi permasalahan pada jaringan komputer antara lain data yang dikirimkan lambat, rusak dan bahkan tidak sampai ke tujuan. Komunikasi sering mengalami *time-out* , hingga masalah keamanan. Penggunaan sistem jaringan komputer memerlukan konfigurasi yang tepat agar menghasilkan manfaat yang maksimal. Kesalahan konfigurasi dapat mengakibatkan berbagai masalah yang pada akhirnya menimbulkan kerugian.

Webmedia Training Center merupakan salah satu lembaga yang telah memanfaatkan sistem jaringan ini. Namun demikian, pada praktik sehari-harinya terjadi banyak permasalahan diantaranya penyalahgunaan hak akses. Penyalahgunaan ini dalam bentuk pengaksesan situs *web* terlarang, *download*, *chatting* dan sebagainya khususnya pada jam belajar.

Dibutuhkan suatu solusi yang mampu mengatasi masalah tersebut,

diantaranya membangun sebuah *proxy server* yang mampu mengontrol dan memonitoring seluruh aktifitas jaringan yang sedang berjalan. Hal ini dilakukan supaya bisa meningkatkan kinerja, kualitas dan juga menjaga nama baik sekolah.

Hasil monitoring yang diperoleh selanjutnya perlu dianalisa, dimana hasil analisa ini bisa digunakan sebagai pertimbangan pemanfaatan *internet* maupun aplikasi jaringan lainnya. Mempertimbangkan hal tersebut maka kami mencoba sebuah *proxy server* dari *Linux Ubuntu* yaitu *Squid Proxy*.

Squid adalah sebuah daemon yang digunakan sebagai *proxy server* dan *web cache*. *Squid* memiliki banyak jenis penggunaan, mulai dari mempercepat *web server* dengan melakukan caching permintaan yang berulang-ulang, *caching DNS*, *caching situs web*, dan *caching* pencarian komputer di dalam jaringan, untuk sekelompok komputer yang menggunakan sumber daya jaringan yang sama, hingga pada membantu keamanan dengan cara melakukan penyaringan(*filter*) lalu lintas, memblok situs tertentu dan juga untuk manajemen bandwidth di jaringan yang nantinya akan menschedule server pada waktu tertentu agar para siswa tidak bisa sembarangan mengakses situs-situs tertentu pada saat jam belajar, dan hasil monitoring ini akan di generate menggunakan *SARG (Squid Analysis Report Generator)*

Berdasar hal-hal tersebut maka penulis tertarik mengangkat judul: **“Sistem Monitoring Squid Proxy Menggunakan Squid Analysis Report Generator di Jaringan Internet Webmedia Training Center”**

1.2 Rumusan Masalah

Sesuai dengan latar belakang yang telah dipaparkan, maka rumusan masalah yang akan dikaji adalah :

1. Rancang bangun *system* monitoring aktifitas *user* di jaringan
2. Rancang bangun *Squid* sebagai *web cache* halaman *web*
3. Memblokir beberapa situs tertentu seperti *www.youtube.com*, *www.facebook.com* dan konten yang mengandung porno

1.3 Batasan Masalah

Agar penerapan ini dapat mencapai sasaran dan tujuan yang diharapkan, maka permasalahan yang akan dibatasi adalah :

1. Membangun sistem jaringan menggunakan *squid* untuk *proxy* dengan sistem operasi *Linux Ubuntu Server 14.04 LTS Precise Pangolin*.
2. Permasalahan yang dibahas hanya pada pengaturan *Proxy* dengan *Squid* yang mampu memfilter *web* serta *monitoring* jaringan dengan *SARG (Squid Analysis Report Generator)*.
3. Studi kasus hanya pada topologi laboratorium komputer *Webmedia Training Center*
4. Tidak membahas tentang pengamanan *security* terhadap *virus* ataupun *malware*

1.4 Tujuan Penelitian

Berdasarkan rumusan masalah yang telah disebutkan diatas, maka tujuan

dari tugas akhir ini adalah :

1. Melakukan monitoring dari hasil pengiriman dan penerimaan paket *HTTP* dari *client* yang telah direkam oleh *Squid Proxy* dengan representase hasil yang lebih *user friendly*.
2. Membuat *web cache* yang dapat menyimpan *history* dari aktifitas *client* sehingga permintaan akan paket yang sama dapat diberikan lebih cepat.
3. Mendapatkan hasil analisa yang akan dijadikan acuan dalam efesiensi penggunaan *internet* pada laboratorium komputer *Webmedia Training Center*

1.5 Manfaat Penelitian

Adapun Manfaat dari penelitian ini adalah sebagai berikut:

- a. Mengontrol situs *Web* terlarang saat jam belajar.
- b. Memonitoring seluruh aktifitas jaringan supaya bisa meningkatkan kinerja, kualitas dan nama baik *Webmedia Training Center*.

1.6 Metode Penelitian

Metode yang dilakukan adalah berupa perancangan dan mengimplementasikan dalam sebuah program. Metode yang dipergunakan dalam skripsi ini adalah sebagai berikut:

- a. Pengumpulan data yang diperlukan salah satunya data topologi jaringan di *webmedia training center* dan bahan bacaan yang kaitannya dengan judul skripsi.
- b. Analisis masalah jaringan di *Webmedia Training Center*.
- c. Perancangan Sistem Monitoring Jaringan.
- d. Penarikan kesimpulan terhadap Sistem Monitoring Jaringan.

1.7 Sistematika Penulisan

BAB I Pendahuluan

Menjelaskan mengenai latar belakang, rumusan masalah, batasan masalah, tujuan penulisan, manfaat, dan sistematika penulisan.

BAB II Landasan Teori

Berisi mengenai teori-teori yang akan menjadi acuan dalam penulisan serta dapat mendukung dalam penyusunan laporan Skripsi ini.

BAB III Perancangan Sistem

Berisi tentang prosedur perancangan teknik dan pembuatan rancangan sistem yang akan dibuat.

BAB IV Implementasi Sistem

Berisi tentang bentuk implementasi dan pengujian sistem yang telah dibuat.

BAB V Kesimpulan dan Saran

Berisi kesimpulan dari hasil laporan Skripsi, serta berisi saran-saran dari seluruh permasalahan yang dibahas

BAB II

LANDASAN TEORI

2.1 Pengertian Jaringan Komputer

Jaringan komputer dapat diartikan sebagai sebuah sistem operasi yang terdiri atas sejumlah komputer dan perangkat jaringan yang didesain untuk dapat bekerja secara bersama-sama, untuk mencapai suatu tujuan tertentu yang bersifat sebagai media komunikasi agar dapat saling bertukar informasi

Berikut adalah pengertian jaringan komputer yang telah dikemukakan para ahli di bidang teknologi :

1. Jaringan komputer adalah kumpulan dari beberapa komputer dan peralatan penunjang lainnya yang terhubung dalam satu kesatuan dan saling terkoneksi (Madcoms, 2010).
2. Jaringan komputer adalah sebuah sistem di mana komputer yang terhubung untuk berbagi informasi dan sumber daya (Izaas el Said, 2007).
3. Jaringan komputer adalah himpunan “interkoneksi” antara 2 komputer autonomous atau lebih yang terhubung dengan media transmisi kabel atau tanpa kabel (Melwin Syafrizal, 2009).
4. Jaringan komputer adalah sekumpulan komputer otonom yang saling terhubung satu dengan yang lainnya menggunakan protokol komunikasi melalui media transmisi pada suatu jaringan komunikasi data (Teguh Wahyono, 2012).

Dengan begitu maka boleh dikatakan, jaringan komputer merupakan perpaduan antara teknologi komputer dan teknologi komunikasi. Dua teknologi berbeda yang sengaja digabungkan sehingga membentuk teknologi sendiri yang kemudian menjadi dasar dari perkembangan jaringan telekomunikasi data berbasis komputer.

Jaringan mempunyai beberapa manfaat yang lebih dibandingkan dengan komputer yang berdiri sendiri (*stand-alone*), yaitu dalam hal :

1. Jaringan memungkinkan manajemen sumber daya lebih.

Pengguna atau *user* dapat saling berbagi printer tunggal dengan kualitas tinggi, dibandingkan memakai printer kualitas rendah di masing-masing meja komputer. Selain itu, lisensi perangkat lunak jaringan dapat lebih murah dibandingkan lisensi *stand-alone* terpisah untuk jumlah pengguna sama.

2. Jaringan membantu mempertahankan informasi agar tetap *update*.

Sistem penyimpanan data terpusat yang dikelola dengan baik memungkinkan banyak pengguna mengakses data dari berbagai lokasi yang berbeda, dan membatasi akses ke data sewaktu sedang diproses.

3. Jaringan membantu mempercepat proses berbagi data (*data sharing*).

Transfer data pada jaringan selalu lebih cepat dibandingkan sarana berbagi data lainnya yang bukan jaringan (*flasdisk*, disket, CD, dan lain sebagainya).

4. Jaringan memungkinkan kelompok kerja berkomunikasi dengan efisien.

Surat dan penyampaian pesan elektronik (*email*) merupakan substansi

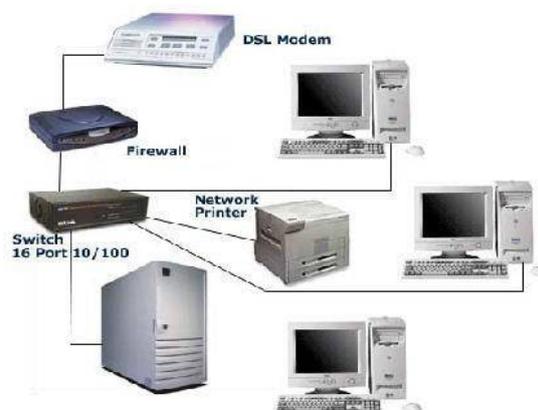
sebagian besar sistem jaringan, disamping sistem penjadwalan, pemantauan proyek, konferensi *online* dan *groupware*, dimana semuanya membantu tim bekerja lebih produktif.

a. Berdasarkan Ruang Lingkup

Jenis jaringan komputer bila di lihat berdasarkan lingkup atau luas jangkauannya, dibedakan menjadi beberapa macam :

1. *Local Area Network* (LAN)

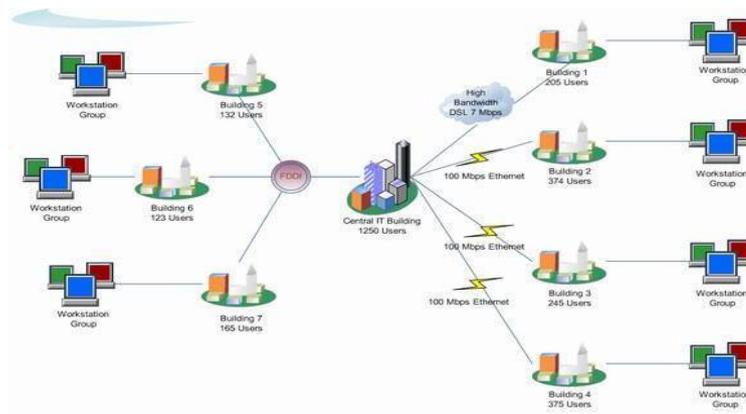
LAN merupakan jaringan komputer yang mencakup daerah yang kecil, seperti rumah, perkantoran, dan sekolah. LAN didasarkan pada teknologi *ethernet* dan *Wi-fi* dari 10 sampai 10000 Mbit/s. Untuk membuat sebuah LAN, komputer yang akan disambungkan dilengkapi dengan *Ethernet card* atau *wireless card*. Ada beberapa komputer tertentu biasanya pada laptop, sudah dilengkapi atau terintegrasi dengan *card* untuk *wireless*, sehingga tidak diperlukan penambahan *card* baru lagi untuk disambung dengan komputer lain, dapat di lihat pada gambar 1



Gambar. 1 *Local Area Network*

2. Metropolitan Area Network (MAN)

Metropolitan Area Network (MAN) merupakan jaringan komputer yang besar dan biasanya digunakan di sekolah/kampus, atau sebuah kota. Ada dua jenis koneksi yang biasanya digunakan, yaitu koneksi dengan cara *wireless* atau dengan kabel *fiber optic*. Misalnya, sebuah sekolah atau kampus mempunyai sebuah MAN yang terdiri dari beberapa LAN yang berada pada radius beberapa kilo di sekitar tempat tersebut. Kampus dengan MAN tersebut mempunyai hubungan dengan universitas-universitas lainnya membentuk sebuah WAN atau *internet*, dapat di lihat pada gambar 2



Gambar. 2 *Metropolitant Area Network*

3. Wide Area Networks (WAN)

Jaringan tipe ini terbentuk oleh keterhubungan perangkat-perangkat komputer yang mencakup regional yang tidak terbatas lagi dalam satu lingkup kota ataupun kampus namun sudah mencakup antar daerah, antar wilayah, maupun antar negara. Misalkan saja antar cabang-cabang perusahaan lingkup nasional ataupun internasional, karakteristik komunikasi antar titik dalam WAN dapat

menghemat biaya dan waktu dengan konektivitas yang terbentuk, selain itu segala proses yang dibutuhkan dapat dilakukan dengan melakukan *remote* terhadap terminal-terminal penting yang menjadi pusat layanan. WAN (*Wide Area Network*) dapat di lihat pada gambar 3.



Gambar. 3 *Wide Area Network*

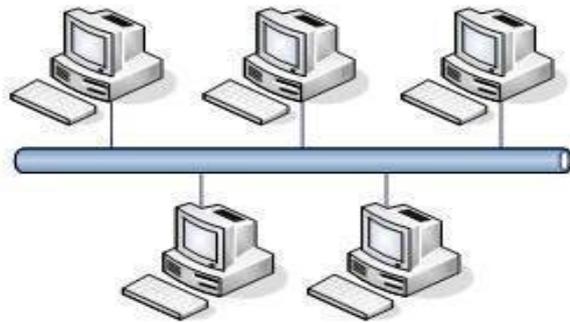
b. Berdasarkan Arsitektur

Ada beberapa jenis pemodelan jaringan yang dibagi berdasarkan arsitektur atau desain, atau sering disebut topologi jaringan. Topologi merupakan cara penghubung sejumlah *node* atau sentral dalam membentuk suatu sistem jaringan. Topologi jaringan yang umum dipakai adalah :

1. Topologi Bus

Suatu jaringan dengan topologi bus, dapat dilihat sebuah kabel atau *communication line* yang dibagi bersama oleh seluruh bagian dari jaringan, seperti terlihat pada gambar 4. Permasalahan yang dapat muncul dari topologi semacam ini adalah permasalahan koneksi dan masalah keamanan. Apabila terjadi masalah dengan kabel utama, maka seluruh komputer tidak akan mampu berkomunikasi satu sama lainnya. Di samping itu, apabila terdapat lebih dari satu komputer dengan alamat yang sama, maka semua pesan yang ditujukan komputer dengan alamat tersebut akan terlihat oleh semua komputer beralamat

sama lainnya, hal ini akan menimbulkan sebuah permasalahan keamanan jaringan.



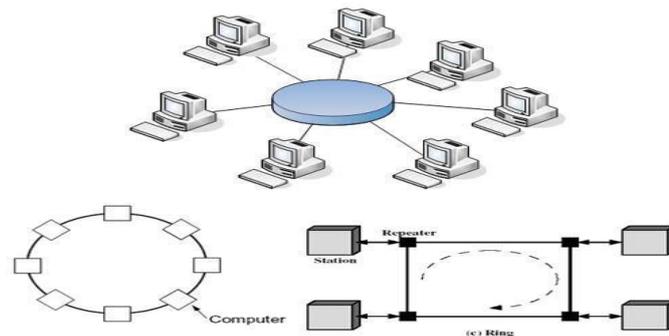
Gambar. 4 Topologi *Bus*

Topologi *bus* sudah sangat jarang digunakan dalam membangun jaringan komputer biasa karena memiliki beberapa kekurangan diantaranya kemungkinan terjadinya tabrakan aliran data, jika salah satu perangkat putus atau terjadi kerusakan pada satu bagian komputer maka jaringan langsung tidak akan berfungsi sebelum kerusakan tersebut diatasi.

2. Topologi *Ring*

Dalam topologi *ring*, setiap komputer host terhubung dengan dua komputer lainnya dan dua ujung dari kabel jaringan disambungkan sehingga secara keseluruhan membentuk sebuah lingkaran, seperti terlihat pada gambar 5. Keuntungan dari topologi ini adalah kemudahannya untuk menambahkan satu komputer host ke dalam jaringan. Namun apabila satu komputer mengalami masalah, maka secara langsung akan mempengaruhi jaringan secara keseluruhan. Transfer data antara satu komputer dengan komputer lain yang

tidak bersebelahan akan mentransmisikan data melalui komputer-komputer yang berada di antaranya. Hal ini dapat menimbulkan permasalahan keamanan jaringan.

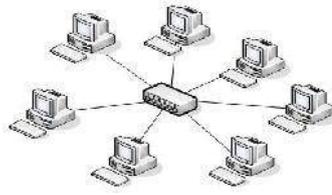


Gambar. 5 Topologi *Ring*

3. Topologi *Star*

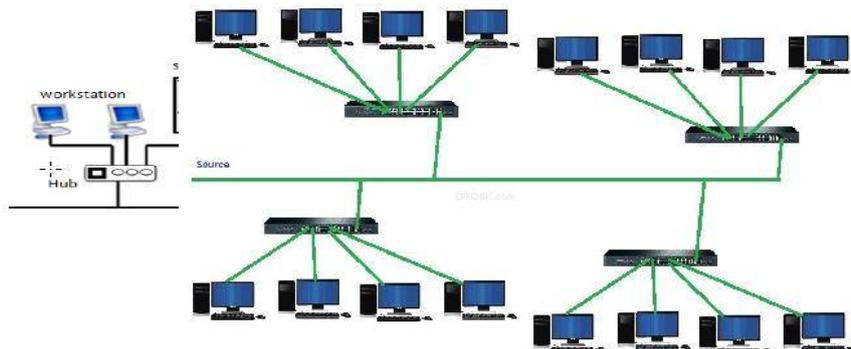
Topologi ini adalah sebuah topologi yang jika digambarkan akan dapat menyerupai bentuk bintang. Semua perangkat yang bergabung dalam jaringan berbagi sebuah perangkat sentral, umumnya *hub* dan *switch*. Topologi ini adalah topologi yang paling umum digunakan dalam implementasi jaringan komputer. Permasalahan yang paling penting dari topologi ini adalah ketergantungan jaringan terhadap perangkat sentral. Apabila perangkat tersebut rusak, maka seluruh jaringan akan *down*.

Namun kelebihan topologi ini apabila dibandingkan dengan topologi lain adalah kemudahan penanggulangan permasalahan tersebut hanya dengan segera menggantikan perangkat sentral tersebut saja tanpa harus mengubah banyak konfigurasi jaringannya. Topologi *star* digambarkan dalam gambar 6.

Gambar. 6 Topologi *Star*

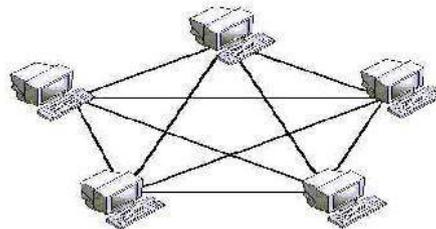
4. Topologi *Tree*

Topologi *tree* merupakan perpaduan antara topologi Bus dan *Star*, yang terdiri dari kelompok-kelompok dari *workstation* konfigurasi bintang yang terkoneksi ke kabel utama yang menggunakan topologi *bus*. Topologi ini memungkinkan untuk pengembangan jaringan yang telah ada, dan memungkinkan sebuah perusahaan mengkonfigurasi jaringan sesuai kebutuhannya, dapat di lihat pada gambar 7.

Gambar. 7 Topologi *Tree*

5. Topologi *Mesh*

Jaringan dengan topologi ini, komputer *host* dapat terhubung secara *point-to-point* ke satu, dua atau lebih komputer lainnya dalam jaringan. Komputer-komputer dalam jaringan akan menjadi *relay* terhadap komputer lainnya yang tidak terhubung langsung dengan komputer yang mengirimkan data, dapat di lihat pada gambar 8.



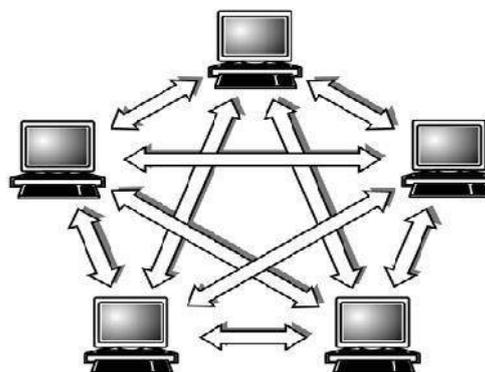
Gambar 8 *Topologi Mesh*

c. Berdasarkan kebutuhan

Dalam membangun sebuah implementasi jaringan maka di lihat juga dari segi kebutuhan hal ini didasarkan pada kebutuhan yang akan dihadapi. Berdasarkan kebutuhan jaringan komputer di bagi menjadi berapa macam:

1. Peer to peer

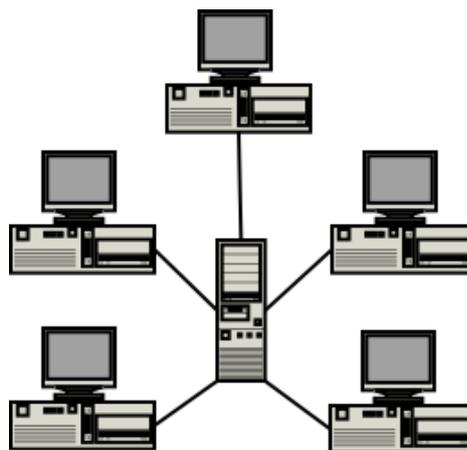
Pada jaringan ini sebuah komputer langsung dihubungkan ke komputer lainnya dan dapat saling berbagi pakai sumber daya (perangkat keras dan perangkat lunak) pada masing-masing komputer. Jenis jaringan ini biasanya hanya akan diterapkan untuk jumlah komputer yang tidak terlalu banyak karena komunikasi antar komputer menjadi susah ketika komputer yang digunakan terlalu banyak, dapat di lihat pada gambar 9.



Gambar 9 *Jaringan peer-to-peer*

2. *Client Server*

Model hubungan *Client-Server* memungkinkan jaringan untuk memusatkan fungsi dan aplikasi kepada satu atau dua *file server*. Komputer *server* merupakan komputer yang mampu menyediakan dan menerima permintaan yang dilakukan oleh komputer *client*. *Workstation* yang berdiri sendiri dapat mengambil sumber daya yang ada pada *file server*. Model hubungan ini menyediakan mekanisme untuk mengintegrasikan seluruh komponen yang ada pada jaringan dan memungkinkan banyak pengguna secara bersama-sama memakai sumber daya dari *file server*. Pada Gambar 10 terlihat bahwa *client* dan *server* akan sangat berhubungan erat di dalam jaringan. Sehingga aktifitas yang ada pada jaringan tersebut haruslah melewati *server* terlebih dahulu, tidak bisa terhubung langsung antara komputer dengan komputer lainnya secara langsung, dapat di lihat pada gambar 10.



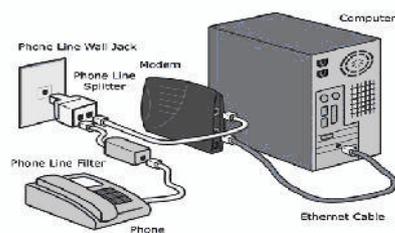
Gambar 10 *Jaringan Client-server*

2.2 Peralatan Jaringan

Peralatan jaringan digunakan sebagai media yang bisa membantu akses dan koneksi perangkat di dalam jaringan, terdapat perangkat-perangkat yang fungsinya berbeda-beda, diantaranya adalah :

1. Modem

Modem berasal dari singkatan *modulator demodulator*. *Modulator* merupakan bagian yang mengubah sinyal informasi kedalam sinyal pembawa (*carrier*) dan siap untuk di kirimkan, sedangkan *Demodulator* adalah bagian yang memisahkan sinyal informasi (yang berisi data atau pesan) dari sinyal pembawa yang diterima sehingga informasi tersebut dapat diterima dengan baik. *Modem* merupakan penggabungan kedua-duanya, artinya *modem* adalah alat komunikasi dua arah. *Modem* teknologi ADSL (*Asyetric Digital Subscribe Line*) adalah jenis *modem* yang dapat memberikan akses *internet* dan menggunakan telepon *analog* secara berbarengan. Caranya sangat mudah, untuk ADSL diberikan sebuah alat yang disebut sebagai *Splitter* atau pembagi *line*. Posisi *Splitter* ditempatkan didepan ketika *line* telepon masuk. Artinya anda tidak boleh mencabangkan *line modem* untuk ADSL dengan suara secara langsung. Alat *Splitter* berguna untuk menghilangkan gangguan ketika anda sedang menggunakan ADSL *modem*, dapat di lihat pada gambar 11.



Gambar. 11 Kinerja Modem ADSL

2. NIC (*Network Interface Card*)

NIC (network interface card) adalah *expansion board* yang digunakan supaya komputer dapat dihubungkan dengan jaringan. Sebagian besar *NIC* dirancang untuk jaringan, *protokol*, dan media tertentu. *NIC* biasa disebut dengan *LAN card (Local Area Network Card)*, dapat di lihat pada gambar 12.



Gambar 12 *Network Interface Card*

3. *Switch*

Switch merupakan suatu *device* pada jaringan yang secara konseptual berada *port-port* yang sedang saling berkirim paket pada *layer 2 (Data link Layer)* dan ada yang *layer 3 (Network Layer)*. Maksudnya, *switch* pada saat pengiriman data mengikuti *MAC address* pada *NIC (Network Interface Card)* sehingga *switch* mengetahui kepada siapa paket ini akan diterima. Jika ada *collision* yang terjadi merupakan *collision* pada data. Misalnya ketika ada pengiriman paket data dari *port A* ke *port B* dan pada saat yang sama ada pengiriman paket data dari *port C* ke *port D*, maka tidak akan terjadi tabrakan (*collision*) karena alamat yang dituju berbeda dan tidak menggunakan jalur yang sama. Semakin banyak *port* yang tersedia pada *switch*, tidak akan mempengaruhi *bandwidth* yang tersedia untuk setiap *port*, dapat di lihat pada gambar 13.



Gambar 13 *Switch*

4. Kabel

Ada beberapa jenis kabel yang banyak digunakan dan menjadi standar dalam penggunaan untuk berkomunikasi data dalam jaringan computer.

a. Kabel *Coaxial*

Kabel *coaxial* adalah jenis kabel dengan inti dari tembaga dan dikelilingi oleh anyaman halus kabel tembaga lain, diantaranya terdapat isolator. Gambar kabel *coaxial* dapat di lihat pada gambar 14 :



Gambar 14 kabel *coaxial*

b. Kabel *Twisted Pair*

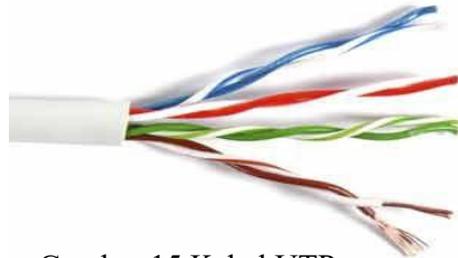
Kabel *Twisted Pair* adalah kabel jaringan yang terdiri dari beberapa kabel yang dililit perpasangan. Tujuannya dililit perpasangan ada untuk mengurangi induksi elektromagnetik dari luar maupun dari efek kabel yang berdekatan. Ada dua jenis Kabel *Twisted Pair* yaitu :

1. UTP (*Unshielded Twisted Pair*)

Kabel UTP adalah kabel *Twisted Pair* tanpa ada *foil* pelindung luar.

Kabel ini umumnya digunakan untuk instalasi *indoor* dan lalu

lintas data yang tidak sensitif Kabel UTP adalah kabel *Twisted Pair* tanpa ada *foil* pelindung luar. Kabel ini umumnya digunakan untuk instalasi *indoor* dan lalu lintas data yang tidak sensitif. Gambar kabel UTP dapat dilihat pada gambar 15 :



Gambar 15 Kabel UTP

2. STP (*Shielded Twisted Pair*)

Kabel STP menggunakan lapisan aluminium *foil* untuk melindungi setiap pasangan kabel didalamnya. Varian lain seperti S/STP juga menambahkan lapisan foil dibawah karet terluar (seperti FTP) untuk perlindungan ekstra terhadap interferensi elektromagnetik. Kabel STP dapat dilihat pada gambar 16 :

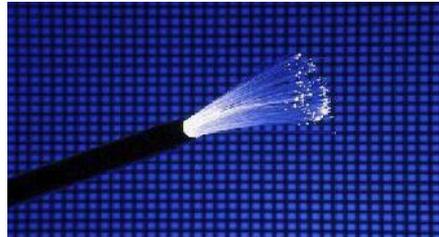


Gambar 16 kabel STP

c. *Fiber Optic Cable*

Kabel yang memiliki inti serat kaca sebagai saluran untuk menyalurkan sinyal antarterminal sering di pakai saluran *BACKBONE* karena keandalannya yang tinggi dibandingkan dengan *coaxial* kabel atau kabel

UTP. Kabel ini tidak terpengaruh oleh cuaca dan panas. Kabel *Fiber Optic* dapat dilihat di gambar 17 :



Gambar 17 kabel fiber optic

2.3 Protokol Jaringan Komputer

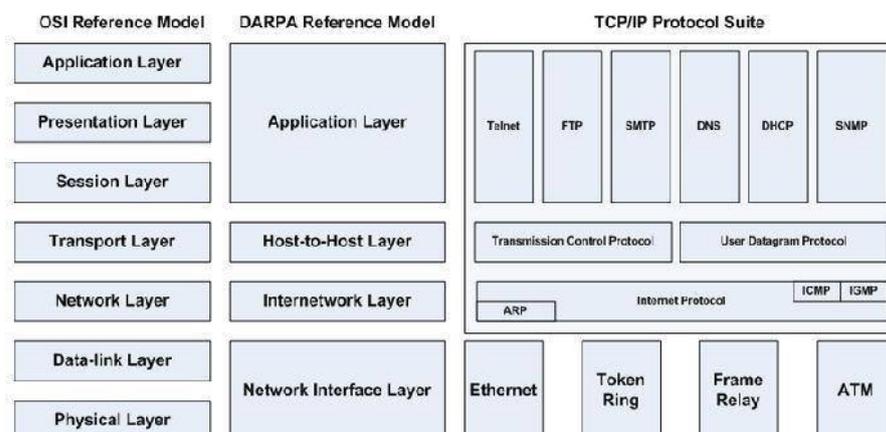
Protokol merupakan suatu aturan main (*rule*) yang mengatur komunikasi data. Dalam sebuah jaringan komputer, terjadi suatu proses komunikasi antar perangkat yang berlainan sistemnya. Perangkat yang ada tersebut dapat mengirim dan menerima data. Untuk dapat mengirim dan menerima data tersebut, dibutuhkan pengertian diantara dua belah pihak. Pengertian inilah yang disebut dengan protokol. Jadi protokol adalah himpunan aturan main yang mengatur komunikasi data. Beberapa elemen penting pada protokol adalah :

1. *Syntax* mengacu pada struktur atau format data, yaitu berkaitan dengan urutan tampilan, misalnya sebuah protokol memiliki urutan pada delapan *bit* pertama adalah pengirim, delapan *bit* kedua adalah alamat penerima, dan *bit stream* sisanya mempunyai informasi tertentu.
2. *Semantics* mengacu pada maksud atau terjemahan dari setiap *section bit* yang menyusunnya.

3. *Timing* mengacu pada waktu kapan data harus dikirim dan seberapa cepat data akan dapat terkirim.

2.4 Model Layer OSI

Model referensi *OSI (Open System Interconnection)* menggambarkan bagaimana informasi dari suatu *software* aplikasi di sebuah komputer berpindah melewati sebuah media jaringan ke suatu *software* aplikasi di komputer lain. Model ini diciptakan berdasarkan sebuah proposal yang dibuat oleh *theInternational Standards Organization (ISO)* sebagai langkah awal menuju standarisasi protokol internasional yang digunakan pada berbagai *layer*. Model ini disebut *ISO OSI (Open System Interconnection) Reference Model* karena model ini ditujukan bagi pengkoneksian *open system*. *Open System* dapat diartikan sebagai suatu sistem yang terbuka untuk berkomunikasi dengan sistem-sistem lainnya. Untuk ringkasnya model tersebut sebagai model *OSI* saja. Secara sederhana ketujuh lapis model *OSI* dapat dilihat pada gambar 18



Gambar 18 Model *Layer OSI*

Terdiri atas 7 *layer* (lapisan) yang mendefinisikan fungsi. Untuk tiap *layernya* dapat terdiri atas sejumlah *protocol* yang berbeda, masing-masing menyediakan pelayanan yang sesuai dengan fungsi *layer* tersebut. Ke-7 *layer* bekerja dari *layer* teratas menuju kebawah bawah sesuai urutan *aplication*, *presentation*, *session*, *transport*, *network*, *data-link*, dan *physical*. Berikut penjelasan masing-masing *layer* :

a. Lapisan Fisik

Lapisan fisik adalah salah satu lapisan *layer* model *OSI* yang mentransmisikan sinyal data. Lapisan fisik melakukan fungsi pengiriman dan penerimaan *bitstream* dalam *medium* fisik. Dalam lapisan ini kita akan mengetahui *spesifikasi mekanikal* dan *elektrikal* daripada media *transmisi* serta antarmukanya.

b. Lapisan *Data-Link*

Lapisan data-link adalah lapisan *layer* model *OSI* sebagai Pengiriman data yang melintasi jaringan fisik. Lapisan data *link* berfungsi *mentransformasi* lapisan fisik yang merupakan fasilitas *transmisi* data mentah menjadi *link* yang *reliabel*.

c. Lapisan *Network*

Adalah lapisan yang berfungsi untuk mengendalikan *subnet*. Hubungan lintas jaringan dan mengisolasi *layer* yang lebih tinggi. Pengalamatan dan pengiriman data. Lapisan *network* bertanggung jawab untuk pengiriman paket dengan konsep *source-to-destination*.

d. Lapisan *Transport*

Adalah lapisan yang berfungsi Menjamin penerima mendapatkan data seperti yang dikirimkan. menerima data dari *session Layer*, memecah data menjadi bagian-bagian yang lebih kecil bila perlu, meneruskan data ke *Network Layer* Lapisan *transport* bertanggung jawab untuk pengiriman *source-to-destination (end-to-end)* dari pada jenis *message* tertentu.

e. Lapisan Sesi

Adalah *layer* yang berfungsi mengijinkan para pengguna untuk menetapkan *session* dengan pengguna lainnya Hubungan antar aplikasi yang berkomunikasi. Layanan yang diberikan oleh tiga *layer* pertama (*fisik, datalink* dan *network*) tidak cukup untuk beberapa proses. Maka pada lapisan *session* ini dibutuhkan dialog *controller*.

f. Lapisan Presentasi

Adalah *layer* yang melakukan fungsi-fungsi tertentu yang diminta untuk menjamin penemuan sebuah penyelesaian umum bagi masalah tertentu Rutin *standard* mempresentasikan data. *Presentation layer* lebih pada *syntax* dan *semantic* pada pertukaran informasi dua sistem.

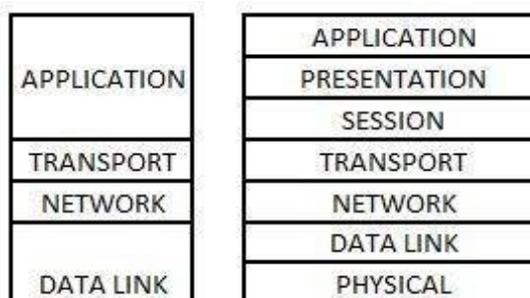
g. Lapisan Aplikasi

Aplikasi *layer* adalah *layer* yang berfungsi untuk menentukan terminal virtual jaringan *abstrak*, sehingga *editor* dan program-program lainnya dapat ditulis agar saling bersesuaian antara aplikasi yang dihadapi *user and resource* jaringan yang diakses. Sesuai namanya, lapisan ini menjembatani interaksi manusia dengan perangkat lunak/*software* aplikasi.

2.5 TCP/IP

TCP digunakan untuk aplikasi yang membutuhkan koneksi dengan pembangunan jalur *virtual* sementara. Cara kerja protokol *TCP* adalah seperti cara komunikasi telepon. Sebelum dapat berkoneksi lewat telepon, maka dibangun suatu jalur *virtual* antara penghubung dan yang dihubungi. Setelah jalur dibangun barulah komunikasi bisa berlangsung. Setelah komunikasi selesai, jalur *virtual* akan dihancurkan kembali.

IP (Internet Protocol) mengatur pengalamatan jaringan *TCP/IP*, dimana sebuah komputer diidentifikasi dengan alamat *IP*. Tiap-tiap komputer memiliki alamat *IP* yang unik, masing masing berbeda satu sama lainnya. Hal ini dilakukan untuk mencegah kesalahan pada transfer data, dapat di lihat pada gambar 19.



Gambar 19 Perbedaan *Layer TCP/IP* dengan *OSI*

Perbedaan antara model protokol *OSI* dan *TCP/IP* terletak pada *layernya*.

Penjelasan tentang *layer* tersebut adalah :

a. *Layer Application*

Lapisan ini merupakan penyatuan dari tiga buah lapisan *OSI* (*Session*, *Presentation*, *Application*). Lapisan aplikasi pada *protokol internet* ini berfungsi menangani masalah *representasi* data, manajemen hubungan, dan cara aplikasi-aplikasi saling berkomunikasi.

b. Layer Transport

Lapisan ini berfungsi mendefinisikan cara pengiriman data antara dua proses yang sedang berlangsung. Lapisan ini mengandung *TCP* yang berfungsi untuk menjamin pertukaran data antara *host-host* pada jaringan.

c. Layer Network

Layer ini adalah tempat dimana *IP (Internet Protocol)* beroperasi fungsinya mendefinisikan cara memindahkan data antara satu komputer ke komputer lainnya.

d. Layer Datalink

Lapisan ini mirip dengan lapisan fisik dan data *link* yang terdapat pada model *OSI* yang berfungsi mendefinisikan cara memindahkan data antara komputer yang terhubung ke media jaringan fisik yang sama.

2.6 IP Address

IP adalah sebuah protokol jaringan, secara umum dijalankan bersama *protocol TCP*, sehingga sering disebut *TCP/IP*. Adanya *IP Address* merupakan konsekuensi dari penerapan *Internet Protocol* untuk mengintegrasikan jaringan komputer *Internet* didunia. Seluruh *host* (komputer) yang terhubung ke *Internet* dan ingin berkomunikasi memakai *TCP/IP* harus memiliki *IP Address* sebagai alat pengenalan *host* pada *network*. Secara logika, *Internet* merupakan suatu *network* besar yang terdiri dari berbagai *sub network* yang terintegrasi. Oleh karena itu, suatu *IP Address* harus bersifat unik untuk seluruh dunia. Tidak boleh ada satu *IP Address* yang sama dipakai oleh dua *host* yang berbeda. Untuk itu, penggunaan

IP Address diseluruh dunia dikoordinasi oleh lembaga sentral Internet yang dikenal dengan *IANA (Internet Assigned Numbers Authority)* di www.iana.org.

Dalam penggunaannya alamat *IP* terbagi menjadi 2 jenis, yaitu :

a. Alamat Publik

Alamat publik adalah alamat-alamat yang telah ditetapkan oleh *Inter NIC* dan berisi beberapa buah *network identifier* yang telah dijamin unik (artinya, tidak ada dua *host* yang menggunakan alamat yang sama) jika *intranet* tersebut telah terhubung ke *Internet*.

b. Alamat *Private*

Untuk *host-host* di dalam sebuah organisasi yang tidak membutuhkan akses langsung ke *internet*, alamat-alamat *IP* yang bukan duplikat dari alamat publik yang telah ditetapkan mutlak dibutuhkan. Untuk mengatasi masalah pengalamatan ini, para desainer *internet* mereservasikan sebagian ruangan alamat *IP* dan menyebut bagian tersebut sebagai ruangan alamat pribadi. Sebuah alamat *IP* yang berada di dalam ruangan alamat pribadi tidak akan digunakan sebagai sebuah alamat publik. Alamat *private* seperti pada table 1

Tabel 1 Alamat *IP Private*

<i>Block IP</i>	<i>Range IP</i>	<i>Bit host subnetting</i>
10.0.0.0/8	10.0.0.1 s/d 10.255.255.254	24 bit
172.16.0.0/12	172.16.0.1 s/d 172.16.255.254	20 bit
192.168.0.0/16	192.168.0.1 s/d 192.168.255.254	16 bit
169.254.0.0/16	169.254.0.0 s/d 169.254.255.254	16 bit

2.7 Kelas IP Address

Untuk memudahkan pengaturan *IP address* seluruh komputer pengguna jaringan *Internet*, dibentuklah suatu badan yang mengatur pembagian *IP address*. Badan tersebut bernama *Inter NIC (Internet Network Information Center)*. *Inter NIC* membagi-bagi *IP address* menjadi beberapa kelas.

Kelas-kelas tersebut meliputi:

a. Kelas A:

Kelas A ditandai dengan bit pertama pada oktet pertama *IP Address* selalu bernilai '0'. *Range Network ID* 1-256. *Default Subnet Mask* untuk kelas ini adalah 255.0.0.0 sehingga total *Network ID* pada kelas ini sebanyak 126 *network*, dan *Host ID* sebanyak 16.777.214 *host per network*.

b. Kelas B

Kelas B ditandai dengan *bit* pertama dan *bit* kedua pada oktet pertama *IP Address* selalu bernilai '1 0'. *Range Network ID* 128-191. *Default Subnet Mask* untuk kelas ini adalah 255.255.0.0 sehingga total *Network ID* pada kelas ini sebanyak 16.384 *netwok* dan *Host ID* sebanyak 65.534 *host per netwok*.

c. Kelas C:

Kelas C ditandai dengan *bit* pertama, *bit* kedua dan *bit* ketiga pada oktet pertama *IP Address* selalu bernilai '1 1 0'. *Range Netwok ID* 192-223. *Default Subnet Mask* untuk kelas ini adalah 255.255.255.0 sehingga total *Network ID* pada kelas ini sebanyak 2.097.152 *network* dan *Host ID* sebanyak 254 *host per network*.

d. Kelas D

Merupakan alamat yang digunakan untuk multicast, *IP address* kelas D dikhususkan untuk penggunaan *multicast* dengan 4-bit pertamanya adalah 1110 dan *range* untuk *host* yang dimilikinya adalah 224.0.0.0 sampai 239.255.255.255.

e. Kelas E

Merupakan alamat yang akan dikembangkan. Untuk *IP Address* kelas E memiliki 5-bit pertama 11110 dan memiliki *range host* dari 240.0.0.0 sampai 247.255.255.255 dicadangkan untuk penggunaan di masa mendatang.

Dalam pemilihan *Network ID* dan *Host ID* perlu diperhatikan beberapa aturan dasar antara lain:

1. *Network ID* tidak boleh sama dengan 127 karena alamat tersebut secara *default* diperlukan untuk keperluan *loopback* diri *host* itu sendiri.
2. *Network ID* dan *Host ID* tidak boleh sama dengan 255 karena alamat tersebut akan dianggap sebagai alamat *broadcast*.
3. *Network ID* dan *Host ID* tidak boleh semua *bit* nya sama dengan 0 (nol) karena 0 berarti "*Only This Network*"
4. *Host ID* dalam satu *network* yang sama harus berbeda termasuk untuk *router*.
5. *IP Address* yang digunakan untuk *router* akan menjadi *default gateway* untuk jaringan
6. Selalu gunakan *subnet mask* yang sama untuk semua *host* dalam suatu jaringan. Bila tidak sama, *host-host* tersebut tidak akan bisa berkomunikasi.

2.8 Subnet Mask

Subnet mask adalah istilah teknologi informasi dalam bahasa Inggris yang mengacu kepada angka *biner* 32 bit yang digunakan untuk membedakan *network ID* dengan *host ID*, ditulis dalam bentuk decimal dengan susunan sama dengan susunan *ip address*. Tetapi ada juga yang ditulis dalam notasi CIDR (*Classless Inter-Domain Routing*). Dengan mengetahui *subnet mask* suatu *ip address* maka kita bisa menentukan jumlah *subnet*, jumlah *host* per *subnet*, blok *subnet*, dan alamat *host-broadcast*.

Tabel 2 *Subnet mask* untuk tiap kelas *IP address*

Kelas IP address	Bit Subnet mask	Subnet dalam desimal
Kelas A	11111111.00000000.00000000.00000000	255.0.0.0
Kelas B	11111111.11111111.00000000.00000000	255.255.0.0
Kelas C	11111111.11111111.11111111.00000000	255.255.255.0

2.9 Sistem Operasi

Sistem operasi adalah sekumpulan rutin perangkat lunak yang berada di antara program aplikasi dan perangkat keras. Semua perangkat lunak berjalan di bawah kendali sistem operasi, mengakses perangkat keras lewat sistem operasi, dan mengikuti aturan-aturan yang dijalankan oleh sistem operasi. Karena sistem operasi bertindak sebagai antarmuka dengan perangkat keras maka pemrogram aplikasi tidak berkomunikasi secara langsung dengan perangkat keras sehingga menyederhanakan pemrograman. Pengaksesan ke perangkat keras merupakan hal yang biasa dilakukan aplikasi, menyatukan fungsi pengaksesan perangkat keras ke

sistem operasi membuat kode itu tidak perlu dipublikasikan di masing-masing aplikasi. Karena semua aplikasi mengakses perangkat keras lewat sistem operasi, posisi sentral ini membuatnya ideal sebagai repositori rutin-rutin sistem dipakai bersama yang berguna untuk semua aplikasi.

2.10 *Linux*

Linux diciptakan pertama kali oleh seorang mahasiswa *Universitas Helsinki* di Finlandia yang bernama *Linux Torvald*. Sistem operasi yang diciptakan tersebut berbasis mesin *Unix*. Sistem operasi *linux* adalah sistem operasi pertama yang benar-benar gratis, sehingga kita bisa secara bebas mengkopinya, karena *linux* berlisensi *GNU Public License*.

2.11 Struktur direktori *Linux*

Sistem berkas *Linux* dan *Unix* diorganisir dalam struktur hirarki, seperti pohon. Level tertinggi dari sistem berkas adalah */* atau *direktori root*. Dalam filosofi disain *Unix* dan *Linux*, semua dianggap sebagai berkas, termasuk *hard disks*, partisi dan *removable* media. Ini berarti bahwa semua berkas dan direktori (termasuk cakram dan partisi lain) ada di bawah direktori *root*. Sebagai contoh, */home/bumida/squid.conf* menampilkan alur (*path*) ke berkas *squid.conf* yang ada di dalam direktori *bumida* yang mana ada di bawah direktori *home*, yang berada di bawah direktori *root (/)*.

Di bawah ini adalah daftar dari direktori umum yang berada tepat di bawah direktori *root (/)* :

- a. */bin* - aplikasi *biner* penting
- b. */boot* - berkas konfigurasi *boot*
- c. */dev* - berkas peranti (*device*)
- d. */etc* - berkas konfigurasi, skrip *startup*, dll (*etc*)...
- e. */home* - direktori pangkal (*home*) untuk pengguna
- f. */lib* - *libraries* yang diperlukan oleh sistem
- g. */lost+found* - menyediakan sistem *lost+found* untuk berkas yang berada dibawah direktori *root (/)*
- h. */media* - *mount* (memuat) *removable media* seperti *CD-ROM*, kamera *digital*.
- i. */mnt* - untuk *me-mount* sistem berkas
- j. */opt* - tempat lokasi untuk menginstal aplikasi tambahan (*optional*)
- k. */proc* - direktori dinamis khusus yang menangani informasi mengenai kondisi sistem, termasuk proses-proses (*processes*) yang sedang berjalan
- l. */root* - direktori pangkal untuk *root*, diucapkan '*slash-root*'
- m. */sbin* - sistem *biner* penting
- n. */sys* - mengandung informasi mengenai *system*
- o. */tmp* - berkas sementara (*temporary*)
- p. */usr* - tempat berkas yang sering digunakan oleh pengguna (*users*)
- q. */var* - berkas *variabel* seperti *log* dan *database* (Sugiantoro, 2012).

2.12 *Squid*

Squid adalah aplikasi yang digunakan sebagai server *cache proxy web* yang menyediakan layanan *proxy* dan *cache* untuk *HTTP*, *HTTPS*, *FTP*, *gopher* dan protokol jaringan populer. *Squid* dapat mengimplementasikan *caching Domain Name Server (DNS) lookup* dan *cache* dan *proxy Secure Socket Layer (SSL)*, dan melakukan *caching* secara transparan. *Squid* juga mendukung berbagai macam protokol *caching*, seperti *Internet Cache Protocol (ICP)*, *Hypertext Cache Protocol (HTCP)*, *Cache Array Routing Protocol (CARP)* dan *Web Cache Coordination Protocol (WCCP)* (Kadek Yota & Kadek Surya, 2014).

Squid memiliki setumpuk kendali akses yang dapat mendongkrak kecepatan *server*. *Squid* menangani semua *request* melalui sebuah proses *I/O* tunggal. *Squid* bekerja menyimpan meta data, terutama pada objek yang sering diakses. Mereka dijaga (di-*cache*) dalam *memory RAM*.

2.13 *Proxy Server*

Proxy Server adalah sebuah *server* atau program komputer yang berperan sebagai penghubung antara suatu komputer dengan jaringan internet. Atau dalam kata lain, *server proxy* adalah suatu jaringan yang menjadi perantara antara jaringan local dan jaringan internet. Cara kerja *Proxy server* sebenarnya sangat sederhana, ketika seorang pengguna layanan *proxy server* meminta berkas, file sambungan atau sumber daya dari publik *server* maka *proxy server* meneruskannya ke internet seolah-olah *proxy* tersebut yang meminta. Dan ketika *proxy server* mendapatkan apa yang di minta pengguna, dia memberikan respon

kepada pengguna seolah-olah dia adalah publik *server*. Dengan menggunakan *proxy*, maka identitas komputer menjadi tersembunyi.

Selain itu *proxy server* juga berfungsi untuk melakukan otentikasi *user*, memblokir situs-situs tertentu, memblokir *banner* dan sebagainya. *Proxy server* mempunyai 3 fungsi utama yang akan diterapkan di dalam sebuah jaringan yaitu:

a. Connection Sharing

Maksudnya yaitu *proxy* bertindak sebagai perantara antara jaringan lokal dengan jaringan publik atau internet. Pengguna tidak langsung berhubungan dengan jaringan luar atau internet, tetapi harus melewati sebuah *gateway*. *Gateway* ini sangat penting bagi para pengguna karena jaringan lokal perlu dilindungi dari bahaya yang mungkin berasal dari internet. Bahaya tersebut akan sulit diatasi jika tidak ada garis batas antara jaringan lokal dengan internet. *Gateway* juga berfungsi sebagai titik yang terhubung ke jaringan lokal dan koneksi ke jaringan luar juga terhubung ke titik tersebut. Dengan demikian, koneksi dari jaringan lokal ke *internet* akan menggunakan sambungan yang dimiliki oleh *gateway* secara bersama-sama yang disebut dengan *Connection Sharing*.

b. Filtering

Yang dimaksud dengan *filtering* disini yaitu *proxy* berfungsi untuk melindungi jaringan lokal dari bahaya yang berasal dari *internet*. *Filtering* bekerja pada layar aplikasi. Berfungsi melakukan penyaringan atas paket yang lewat dari dan ke jaringan-jaringan yang dihubungkan. Pada *filtering*, *proxy* juga

berfungsi sebagai pembatas. *Proxy* dapat dikonfigurasi untuk menolak akses ke situs *web* tertentu.

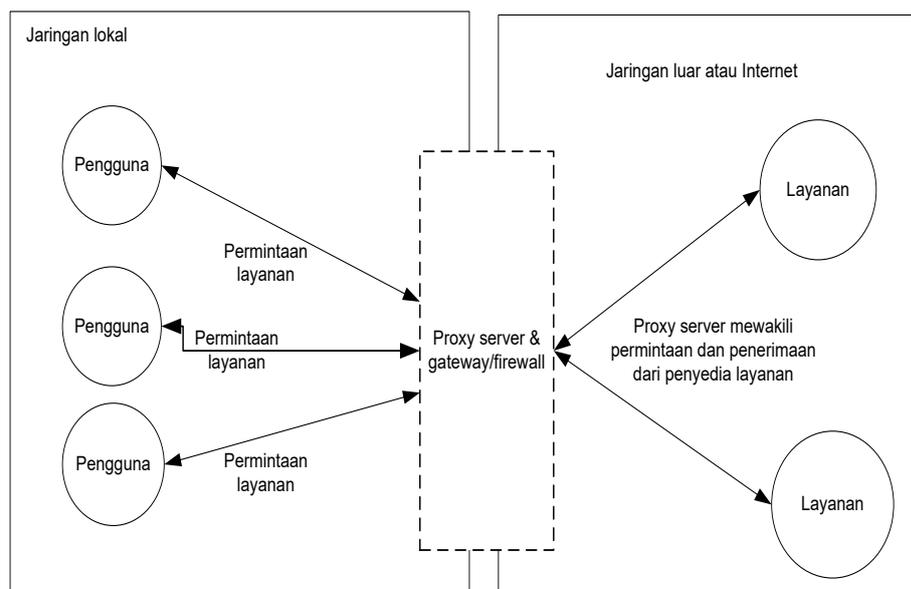
c. *Caching*

Proxy server memiliki mekanisme penyimpanan obyek-obyek yang sudah pernah diminta dari *server-server* di *internet*. *Proxy server* yang melakukan proses diatas biasa disebut *cache server*. Mekanisme *caching* akan menyimpan obyek-obyek yang merupakan hasil permintaan dari para pengguna, yang didapat dari *internet*. Disimpan dalam ruang *disk* yang disediakan (*cache*). Dengan demikian, bila suatu saat ada pengguna yang meminta suatu layanan ke *internet* yang mengandung obyek-obyek yang sama dengan yang sudah pernah diminta sebelumnya, yaitu yang sudah ada dalam *cache*, maka *proxy server* akan dapat langsung memberikan obyek dari *cache* yang diminta kepada pengguna, tanpa harus meminta ulang ke *server* aslinya di *internet*. Bila permintaan tersebut tidak dapat ditemukan dalam *cache* di *proxy server*, baru kemudian *proxy server* meneruskan atau memintakannya ke *server* aslinya di *internet*.

Proxy dalam pengertiannya adalah sebagai perantara, bekerja dalam berbagai jenis *protokol* komunikasi jaringan dan dapat berada pada level-level yang berbeda pada *hirarki layer* protokol komunikasi jaringan. Suatu perantara dapat saja bekerja pada *layer Data-Link*, *layer Network* dan *Transport*, maupun *layer Aplikasi* dalam hirarki *layer* komunikasi jaringan menurut *OSI*. Namun pengertian *proxy server* sebagian besar adalah untuk menunjuk suatu *server* yang bekerja sebagai *proxy* pada *layer* aplikasi, meskipun juga akan dibahas mengenai

proxy pada level sirkuit. Dalam suatu jaringan lokal yang terhubung ke jaringan lain atau internet, pengguna tidak langsung berhubungan dengan jaringan luar atau internet, tetapi harus melewati suatu *gateway*, yang bertindak sebagai batas antara jaringan lokal dan jaringan luar.

Gateway ini sangat penting, karena jaringan lokal harus dapat dilindungi dengan baik dari bahaya yang mungkin berasal dari internet, dan hal tersebut akan sulit dilakukan biar tidak ada garis batas yang jelas jaringan lokal dan internet. *Gateway* juga bertindak sebagai titik dimana sejumlah koneksi dari pengguna lokal akan terhubung kepadanya, dan suatu koneksi ke jaringan luar juga terhubung kepadanya. Dengan demikian, koneksi dari jaringan lokal ke internet akan menggunakan sambungan yang dimiliki oleh *gateway* secara bersama-sama (*connection sharing*). Dalam hal ini, *gateway* adalah juga sebagai *proxy server*, karena menyediakan layanan sebagai perantara antara jaringan lokal dan jaringan luar atau *internet*.



Gambar 20 Konsep *Proxy server*

Proxy ini pada umumnya digunakan untuk kegiatan menyembunyikan identitas atau untuk menghindari pemblokiran akses ke suatu *server*. (Kadek yota & Kadek surya, 2014).

2.14 Firewall

Firewall suatu cara melindungi sistem, baik dari orang-orang yang tidak berhak maupun dari komputer lain yang mengandung *virus*. *Firewall* adalah suatu cara untuk membatasi informasi yang dibolehkan masuk dan keluar dari jaringan lokal kita. Umumnya, *host firewall* terhubung ke internet dan *LAN* ke internet hanya melalui *firewall*. *Firewall* bertujuan untuk menjaga *LAN* dari akses yang tidak diinginkan dari internet. Dengan kata lain *firewall* dibuat untuk membatasi antara *LAN* dan internet (Sugiantoro, 2014).

2.15 SARG (Squid Analisis Report Generator)

SARG merupakan aplikasi berbasis *web base* yang berfungsi memberikan *report* dari *proxy server* berbasis *Squid*. Aplikasi ini dapat memberikan rincian detail dari laporan *log file Squid proxy* dengan tampilan yang lebih mudah dipahami.

Adapun kelebihan dari *SARG* adalah :

1. Aplikasi berbasis *web* sehingga dapat diakses dimana saja
2. Aplikasi ringan dan stabil tanpa memakan banyak *resource* memory
3. Rincian *report* yang diberikan sangat detail lengkap beserta grafik pendukung

4. Bersifat *open source* sehingga dapat dikonfigurasi sesuai keinginan
5. Karena berbasis *web* aplikasi ini langsung mendukung fitur *print layout*

BAB III

ANALISIS DAN PERANCANGAN SISTEM

3.1 Metode Pengumpulan Data

Dalam melakukan penelitian, penulis melakukan pengumpulan data dengan menggunakan beberapa metode, diantaranya:

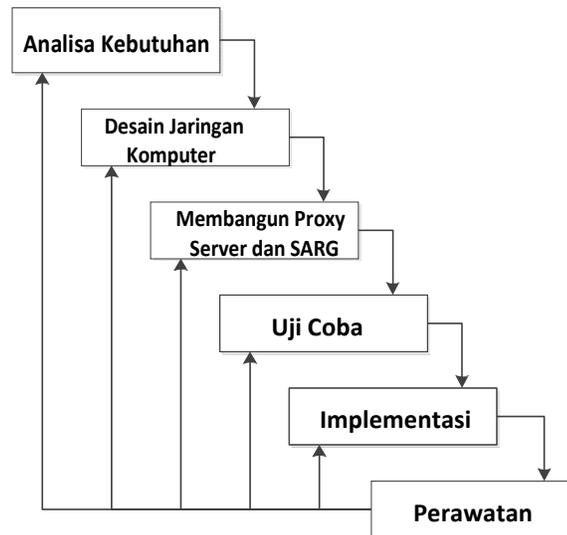
a. Studi Lapangan

1. Observasi

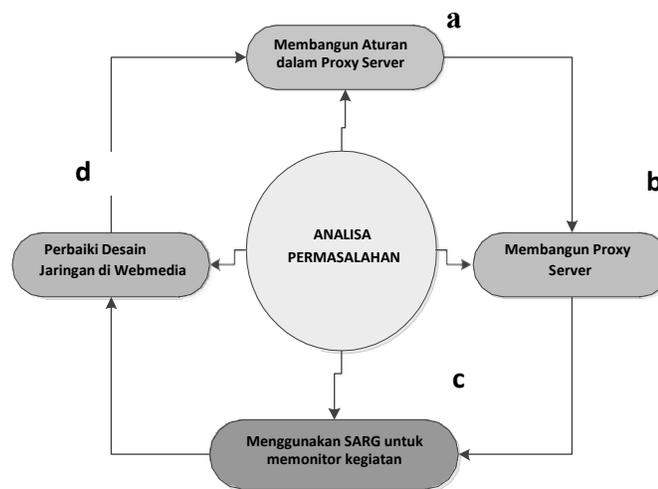
Penulis melakukan observasi pada tanggal 10-12 Desember 2018 untuk mengumpulkan dan ketersediaan data yang dapat diakses dari peralatan dan aplikasi, selain itu penulis juga mengumpulkan dokumentasi mengenai peralatan yang ada dan informasi serta konfigurasi jaringan dan topologi sebelumnya yang digunakan di *Webmedia Training Center*.

2. Wawancara

Penulis juga melakukan wawancara dengan *administrator* jaringan di *Webmedia Training Center* (10-12 Desember 2018) yaitu Bapak Nur abdilah Amd.Kom. dan teknisi jaringan sdra. Aslam Fajari untuk mendapatkan data dan informasi yang berkaitan dengan konfigurasi jaringan, jumlah user dan kegiatan yang sering dilakukan dalam penggunaan koneksi internet bersama di *Webmedia Training Center*.



Gambar 21 Metodologi Penyelesaian Masalah dalam Penelitian



Gambar 22 Tindakan yang dilakukan setelah analisa permasalahan

b. Studi Pustaka

Pengumpulan data dan informasi bersumber dari buku yang sering penulis baca dan dari situs internet yang menjadi referensi dan pedoman dalam penulisan penelitian yang dapat dilihat pada daftar pustaka.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Spesifikasi Sistem

Spesifikasi sistem yang digunakan untuk membuat virtualisasi *file server* adalah sebagai berikut :

a. Spesifikasi *Hardware*

1) Spesifikasi Laptop

Laptop yang digunakan adalah Lenovo G410. Laptop adalah perangkat yang digunakan sebagai tempat *virtualisasi proxy server*, laptop juga menjadi perangkat penghubung antara *wireless access point* dengan *proxy server virtual*, selain itu laptop juga menjadi *client*. Spesifikasi yang dimiliki perangkat ini adalah sebagai berikut :

a) *Monitor* 14" WXGA (1366 x 768)

b) *Processor core i5* 2.2-2.6GHz

c) RAM 2GB

d) HDD 500 GB

e) AMD RADEON *graphics*

f) DVD-RW *super multi*

2) Spesifikasi *Access Point*

Access point yang digunakan adalah TP-LINK TL-WA701ND. Perangkat ini berguna sebagai penghubung antara *Proxy server* virtual dengan perangkat laptop. Spesifikasi yang dimiliki perangkat ini adalah sebagai berikut :

- a) Jaringan nirkabel berkecepatan 150 Mbps
- b) Enkripsi WPA/WPA2
- c) Mendukung *AP client*, dan *repeater*
- d) QSS tombol otomatis untuk membangun koneksi aman dengan WPA2.
- e) Mendukung hingga teknologi 802.11n.

b. Spesifikasi *Software*

1) Sistem Operasi *Windows 10*

Sistem operasi *windows 10* digunakan sebagai sistem operasi pada komputer fisik (*host*). Sistem operasi ini merupakan tempat berjalannya aplikasi untuk virtualisasi *proxy server*.

2) Sistem Operasi *Ubuntu Server 14.04 LTS*

Sistem operasi *ubuntu 14.04 LTS* digunakan sebagai sistem operasi pada *proxy server virtual (guest)*. Sistem operasi ini adalah sistem operasi yang diinstal secara *virtual* di dalam komputer fisik (*host*).

3) *Oracle VM Virtual box*

Oracle vm virtual box adalah aplikasi yang digunakan sebagai *hypervisor*. Karena penelitian menggunakan mode virtualisasi *full virtualization* atau tipe 2 *hypervisor*, sehingga membutuhkan perangkat lunak yang dapat menjalankan sistem operasi (*guest*) di dalam sebuah komputer fisik (*host*).

4) *Squid versi 3.5.28*

Squid versi 3.5.28 digunakan sebagai aplikasi *proxy server* yang akan mengontrol lalu lintas paket data

5) *SARG 2.3.6 (Squid Analysis Report Generator)*

SARG 2.3.6 digunakan sebagai aplikasi yang akan *generate report* dari aktifitas *user* di jaringan.

4.2 Pembuatan Server

Pada pembahasan ini akan dibahas bagaimana cara untuk membangun sebuah *server* dengan menggunakan *Ubuntu Server* dengan *Squid* sebagai *proxy server* serta *SARG* sebagai aplikasi pendukung untuk aktifitas *monitoring*.

a. Instalasi Ubuntu Server 14.04

Instalasi ubuntu atau *distro linux* lainnya dapat dengan mudah dilakukan baik melalui media *CD* ataupun melalui media *flashdisk*. Untuk itu pertama kali yang harus dipersiapkan sebuah media *instalasi* yang sudah *bootable* sistem operasi *Ubuntu Server 14.04*. Selanjutnya melakukan *booting* pertama terhadap media tersebut sehingga muncul *interface* pertama. Pilih bahasa yang digunakan untuk penginstalan dan kemudian tekan *Enter* lalu pilih “*Instal Ubuntu Server*”.

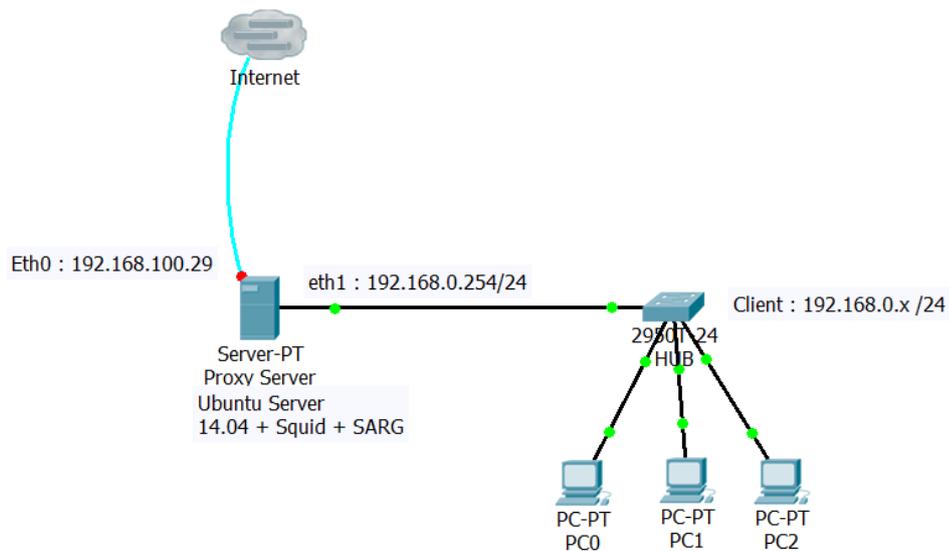
Setelah itu proses *Instalasi* akan memberikan pilihan untuk kostumisasi bahasa, lokasi dan pengaturan *keyboard*. Pilihan ini dapat diatur sesuai kebutuhan. Langkah selanjutnya adalah menentukan *primary network interface* dimana pada

konfigurasi ini *eth0* bertindak sebagai *interface* utama yang akan terhubung langsung ke sumber internet dari *ISP*. Untuk manajemen partisi sama halnya seperti *distro linux* yang lain yaitu adanya partisi *swap* yang merupakan partisi yang membantu kerja memori. Kapasitas dari partisi *swap* disarankan 2 kali dari total *RAM* dan maksimal hanya 4 GB. Pada konfigurasi ini partisi *swap* yang digunakan sebesar 2 GB serta sisa *space* sepenuhnya digunakan untuk partisi sistem dengan *formatEXT4*.

Setelah keseluruhan *Instalasi* diatas selesai maka langkah terakhir adalah memilih paket-paket aplikasi *server* yang akan di instal bersama *Ubuntu Server*. Disarankan untuk memilih semua paket pada pilihan ini agar nantinya konfigurasi dapat lebih mudah dilakukan.

b. Konfigurasi Gateway

Konfigurasi *Gateway* merupakan pengaturan arah *network interface* menerima dan mengirim paket. Tanpa konfigurasi ini *client* yang terhubung ke *server* tidak akan bisa terhubung serta *server* tidak akan meneruskan paket internet kepada *client*. Gambar 31 adalah topologi jaringan yang akan dibangun konfigurasinya :



Gambar 31 Topologi jaringan

Sumber : Hasil rancangan

Dari gambar 31 dapat diketahui bahwa *PCServer* memiliki 2 *interface network* dengan nama *eth0* dan *eth1*. *Network interface eth0* adalah *interface* yang terhubung langsung ke *internet* dimana alamat *IP* nya adalah 192.168.100.29 dengan *gateway* yang mengarah ke alamat *IP internet* yaitu 192.168.100.1 dengan *netmask* kelas C yaitu 255.255.255.0.

Sedangkan untuk *interface eth1* merupakan *interface* yang menuju ke *client* dimana untuk alamat *IP* nya akan di set *DHCP* dengan alamat 192.168.0.X /24 dengan *netmask* 255.255.255.0 yang termasuk kedalam kelas C.

Langkah selanjutnya adalah memberikan *DNS (Domain Name System)*. *DNS* digunakan sebagai penerjemah dari nama *domain* ke alamat *IP*, dan sebaliknya, yaitu dari alamat *IP* ke nama *domain*.

Sampai pada tahap ini *PC Server* sudah terkoneksi ke *internet*, selanjutnya adalah meneruskan koneksi *internet* tersebut ke *client* melalui *interface eth1* dengan mengetikkan perintah *IP TABLES* berikut pada *file /etc/rc.local*

c. DHCP Server

Untuk membuat *interface eth1* menjadi *DHCP Server*, maka terlebih dahulu harus melakukan *Instalasi* paket aplikasi yang dapat membuat *Ubuntu Server* memberikan alamat *IP* secara otomatis.

Kemudian lakukan perubahan pada baris terakhir bagian *interface* yang diapit tanda kutip dua dengan mengisi *eth1* sebagai *interface* yang akan berperan sebagai *DHCP Server*.

Sampai pada langkah ini konfigurasi *Ubuntu Server* sudah selesai, *server* sudah terkoneksi ke *internet* dan mampu meneruskan koneksi *internet* ke *client* dengan pengalamatan *IP* secara *DHCP* pada *interface eth1*.

4.3 Squid Proxy

a. Instalasi Squid

Untuk melakukan *Instalasi Squid*, sebelum disarankan untuk mengupdate *repository* dan menginstal beberapa aplikasi yang di perlukan dari *Ubuntu Server* yang digunakan.

Proses *instalasi* akan berjalan, setelah proses selesai lakukan lagi proses *update repository*.

b. Konfigurasi Squid

Sebelum *Squid proxy* dapat berjalan dengan baik, perlu dilakukan konfigurasi awal agar *squid* dapat berperan sebagai *server* terhadap segala *request* yang dilakukan oleh *client*. Konfigurasi *Squid* dilakukan dengan menuliskan *code* pada *file Squid.conf* yang terletak di direktori */etc/Squid/Squid.conf*. Untuk dapat melakukan perubahan terhadap *file* ini, maka perlu dilakukan perubahan hak akses.

Konfigurasi yang perlu pertama kali dilakukan adalah *Proxy Transparent* yaitu membuat agar permintaan *http* pada *client* langsung diarahkan ke *IP Squid* sehingga pada *client* tidak perlu lagi melakukan *setting manual IP proxy* pada *web browser*nya. Pada halaman *file /etc/rc.local* cari *code* berikut dengan menekan *Ctrl+W* pada *keyboard* , kemudian tambahkan *script* berikut sebelum *exit 0*.

Setelah itu adalah mendeklarasikan *proxy* sebagai *user* dan *group* dengan hak akses sebagai *Squid* agar *Squid* diberi hak akses oleh sistem operasi untuk menangani paket yang diterima serta meneruskannya ke *client*.

Kemudian konfigurasi selanjutnya adalah memberi info kontak *email admin* terhadap *client* jika halaman *web* yang diakses oleh *client* tidak dapat dibuka.

c. Konfigurasi Web Cache

Pada bagian ini akan dibahas tentang konfigurasi *Squid* dalam membuat *web cache* yang nantinya membuat *Squid* dapat menyimpan halaman *web* yang diakses oleh *client*. Tahap awal dalam konfigurasinya adalah membuat sebuah

direktori baru sebagai lokasi penyimpanan *web cache*. *web cache* yang diakses oleh *client* akan disimpan pada direktori “*/etc/squid/cacheku*” dengan kapasitas maksimal direktori adalah 10240 MB dengan jenis *storage* adalah *ufs* dimana jumlah sub direktori awal adalah 100 dan sub direktori kedua maksimal 256 direktori.

Langkah terakhir adalah memberikan konfigurasi *acl* terhadap *IP client* agar dapat terkoneksi dengan *Squid*. bahwa *network* 192.168.0.0/24 merupakan *rule acl* dengan nama *webmedia*, kemudian pada *code* “*http_access allow webmedia*” mendefinisikan bahwa *acl* dengan nama *webmedia* diberikan status *allow* atau diizinkan.

d. Konfigurasi Web Filter

Pada konfigurasi ini akan dijelaskan bagaimana *Squid* dapat memfilter permintaan *client* terhadap *http request* sesuai dengan *rule-rule* yang akan dibuat. Ada 3 jenis konfigurasi *web filter* yang dapat dilakukan pada *Squid*.

1. *Filter* berdasarkan alamat *domain*

Konfigurasi *filter web* berdasarkan alamat *domain* dapat dilakukan dengan 2 metode yaitu pertama dengan cara mendefinisikan seluruh alamat *domain* pada *file Squid.conf* dengan *acl* yang terletak sebelum *code http_access allow localhost* dan *http access allow all*.

untuk metode kedua adalah dengan mengarahkan *acl* terhadap *file teks* yang berisi *list* dari alamat *website* yang akan di *block*. Pada konfigurasi *code http_access deny nama-acl* berperan sebagai perintah

untuk memblock (*deny*) akses terhadap *domain* yang didefinisikan pada nama-*acl*, dengan demikian seluruh alamat *domain* yang berada pada *acl* tersebut akan di *block* oleh *Squid*.

2. *Filter* berdasarkan kata kunci pada *URL*

Untuk konfigurasi ini juga sama seperti teknik *filter* berdasarkan alamat *domain* yaitu memiliki 2 metode yang sama, pertama kata kunci dapat didefinisikan langsung pada file *Squid.conf* serta metode kedua kata kunci bisa bersumber dari file teks yang tersimpan. *Filter* berdasarkan *IP* dari alamat *domain*.

Pada konfigurasi ini *filter web* berdasarkan alamat *IP* dari *domain* yang akan di *block*, untuk itu sebelumnya alamat *IP* dari *domain* terkait harus diketahui terlebih dahulu. Untuk melihat alamat *IP* dari sebuah *domain* dapat diketahui dengan mengetikkan perintah *nslookup* alamat_*domain*.

e. Konfigurasi *IPTABLES*

Konfigurasi *IPTABLES* adalah langkah terakhir pada konfigurasi *proxy server*. Konfigurasi ini berfungsi untuk mengarahkan permintaan paket *http* yang biasanya pada *port* 80 menjadi *port Squid* yaitu 3129. Dengan cara ini *client* tidak perlu memberikan pengaturan *proxy* manual pada *browser*. Perintah *iptables* ini nantinya akan dibuat pada file *rc.local* agar dieksekusi selalu walaupun *PC server* mengalami proses *rebooting*.

4.4 SARG

Seperti yang telah dibahas sebelumnya bahwa *SARG (Squid Analysis Report Generator)* adalah sebuah aplikasi yang dapat memberikan laporan segala aktifitas *host* yang dibawah *Squid* dalam bentuk halaman *web* yang diakses melalui *localhost* pada *server*. Pada bagian ini akan dibahas tentang bagaimana memasang aplikasi *SARG* pada *PC server* penelitian ini.

a. Instalasi SARG

Sebelum melakukan *Instalasi SARG* lakukan *update* paket *repository* pada *server ubuntu*. setelah proses *Instalasi* selesai maka selanjutnya adalah tahap konfigurasi dari *SARG* agar aplikasi ini dapat berjalan dengan baik.

b. Konfigurasi SARG

Konfigurasi *SARG* dilakukan mencakup beberapa langkah dimulai dari pembuatan direktori pada *folder localhost Ubuntu*. Membuat sebuah *forlder* baru dengan nama *SARG* pada direktor */var/www/* yang merupakan *folder localhost* dari *Ubuntu*. Setelah itu maka langkah selanjutnya adalah melakukan konfigurasi pada *file SARG.conf/etc/SARG/SARG.conf*

Setelah semua perubahan, maka lakukan proses simpan agar perubahan yang dilakukan dapat diterapkan. Untuk selanjutnya adalah mengaktifkan keseluruhan dari *SARG*.

Setelah semua konfigurasi selesai maka aplikasi *SARG* dapat diakses melalui *web browser* dengan mengetikkan alamat *ip_server/nama_folder* yaitu untuk penelitian ini adalah 192.168.0.254/*sarg*

4.5 Pengujian dan Analisis

Pada pembahasan ini konfigurasi yang telah diimplementasikan selanjutnya akan dilakukan pengujiannya secara langsung kepada subjek penelitian yaitu *Webmedia Training Center* terkhusus ruangan komputer lantai 3 saja. Pengujian mencakup hasil dari *Web Cache* pada *Proxy Squid* serta hasil dari *SARG report* serta dianalisa apakah *server* yang dibangun sudah sesuai perancangan pada penelitian.

a. Pengujian *Web Cache Proxy Server*

Untuk mengetahui apakah *Squid Proxy Server* sudah berjalan dengan baik adalah dengan melihat hasil file *access.log* pada *Squid* tersebut. Untuk melihat aktifitas ini dapat dijalankan dengan perintah :

```
#tail -f /var/log/Squid3/access.log
```

Hasil *access.log squid* dapat di lihat pada gambar 32 :

192.168.0.2	TCP_MISS	http://www.pancabudi.ac.id/unpab/image/new s/medium	Diterusk an
192.168.0.2	TCP_DENIED	https://www.facebook.com/	Ditolak
192.168.0.2	TCP_REFRESH_MODI FIED	http://www.pancabudi.ac.id/	Refresh Cache Proxy
192.168.0.2	TCP_MEM_HIT	http://www.pancabudi.ac.id/unpab/image/bann er	Dari Memory cache
192.168.0.2	TCP_REFRESH_MODI FIED	http://www.pancabudi.ac.id/home/time	Refresh Cache Proxy

File-file history dari halaman *web* yang telah diakses tersebut akan tersimpan didalam manajemen direktori *Squid*. Berikut adalah menejemen direktory dari *Squid* :

```

$ ls 00
00 0D 1A 27 34 41 4E 5B 68 75 82 8F 9C A9 B6 C3 D0 DD EA F7
01 0E 1B 28 35 42 4F 5C 69 76 83 90 9D AA B7 C4 D1 DE EB F8
02 0F 1C 29 36 43 50 5D 6A 77 84 91 9E AB B8 C5 D2 DF EC F9
03 10 1D 2A 37 44 51 5E 6B 78 85 92 9F AC B9 C6 D3 E0 ED FA
04 11 1E 2B 38 45 52 5F 6C 79 86 93 A0 AD BA C7 D4 E1 EE FB
05 12 1F 2C 39 46 53 60 6D 7A 87 94 A1 AE BB C8 D5 E2 EF FC
06 13 20 2D 3A 47 54 61 6E 7B 88 95 A2 AF BC C9 D6 E3 F0 FD
07 14 21 2E 3B 48 55 62 6F 7C 89 96 A3 B0 BD CA D7 E4 F1 FE
08 15 22 2F 3C 49 56 63 70 7D 8A 97 A4 B1 BE CB D8 E5 F2 FF
09 16 23 30 3D 4A 57 64 71 7E 8B 98 A5 B2 BF CC D9 E6 F3
0A 17 24 31 3E 4B 58 65 72 7F 8C 99 A6 B3 C0 CD DA E7 F4
0B 18 25 32 3F 4C 59 66 73 80 8D 9A A7 B4 C1 CE DB E8 F5
0C 19 26 33 40 4D 5A 67 74 81 8E 9B A8 B5 C2 CF DC E9 F6

```

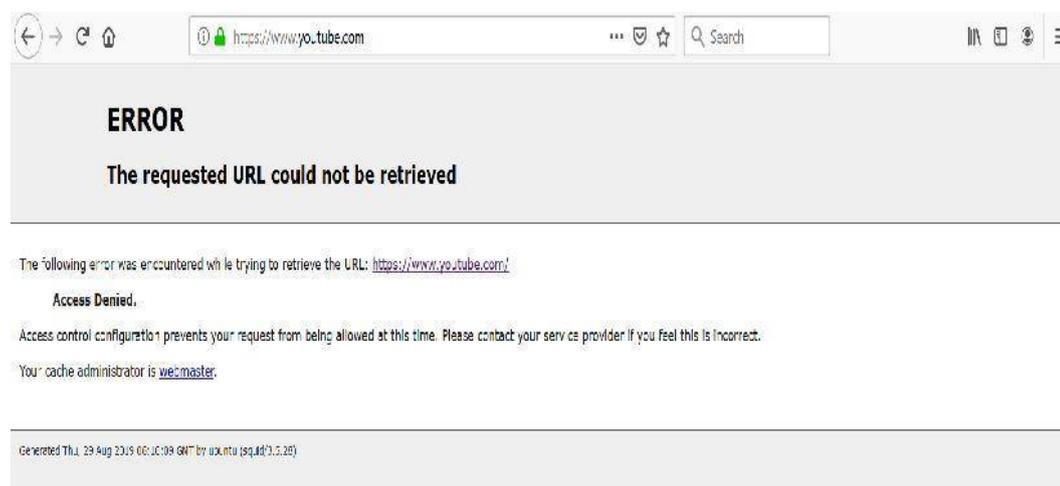
Gambar 33 Manajemen direktori *Squid*

Sumber : Hasil rancangan

Dari gambar 33 terlihat bahwa struktur direktori tempat penyimpanan halaman *web* dalam bentuk *cache* pada *Squid* dibentuk dengan pengalamatan yang *terinskripsi* sehingga dari sisi keamanan *Squid* dapat menjaga keamanan datanya. Para pengguna tidak akan dapat melihat secara langsung *cache* yang tersimpan pada *server Squid*. Namun direktori yang terbentuk tersebut dapat menjadi tolak ukur bahwa *server web proxy* dari *Squid* sudah berjalan dengan baik ditandai dengan adanya aktifitas dari *file access.log* sebelumnya.

b. Pengujian *Web Filter*

Pada pengujian *web filter* akan dicoba untuk mengakses alamat-alamat *website* yang telah di *block* melalui *rule-rule* pada *Squid*. Untuk alamat *domain* yang di *block* maka *Squid* akan mengantarkannya ke halaman *Proxy Squid*. Berikut adalah bukti bahwa *rule* untuk *web filter* telah berjalan dengan baik.



Gambar 34 *Squid Proxy web block*

Sumber : Hasil rancangan

Dari gambar 34 terlihat bahwa *proxy server* pada *Squid* bekerja dengan baik dimana terjadi penolakan akses pada halaman *web* yang telah ditentukan sebelumnya dan pada jam tertentu. Dari hasil diatas dapat di pastikan bahwa *web cache proxy server Squid* sudah berjalan sesuai konfigurasi yang diinginkan.

c. Pengujian *SARG*

Selanjutnya pada pada bagian ini akan diuji apakah aplikasi *SARG* sudah berjalan dengan baik atau tidak. Untuk mengakses *SARG* dapat dilakukan pada komputer *client* dengan mengetik di *browser* alamat *IP* dari *server* di ikuti dengan alamat *folder* tempat *SARG* terpasang dimana alamatnya adalah 192.168.0.254/*SARG*, maka hasilnya yang diperoleh :



FILE/PERIOD	CREATION DATE	USERS	BYTES	AVERAGE
16Jul2019-27Aug2019	Tue 27 Aug 2019 10:45:11 AM WIB	11	6.17G	561.05M
16Jul2019-07Aug2019	Wed 07 Aug 2019 09:23:50 AM WIB	7	2.18G	312.15M
16Jul2019-20Jul2019	Sat 20 Jul 2019 05:40:56 PM WIB	4	1.27G	317.79M
16Jul2019-16Jul2019	Tue 16 Jul 2019 02:53:07 PM WIB	1	47.50M	47.50M

Generated by [sarg-2.3.6 Arp-21-2013](#) on 27/Aug/2019-10:45

Gambar 35 Halaman depan *SARG*

Sumber : Hasil rancangan

Pada gambar 35 adalah laporan *SARG* pada objek penelitian dalam kurun waktu 5 hari. Terlihat *SARG* memberikan rincian detail tentang tanggal serta *user* yang aktif serta jumlah kapasitas permintaan data (*BYTES*) pada tanggal tersebut,

dan pada kolom terakhir (*AVERAGE*) adalah rata-rata dari penggunaan paket data. Dari data ini dapat dipastikan bahwa *SARG* telah berjalan dengan baik.

d. Analisis *Web Cache Proxy Server*

Web Cache Proxy berperan sebagai tempat penyimpanan *file cache* dari halaman yang diakses yang kemudian apabila *client* melakukan *request* terhadap halaman *web* yang sama maka *Squid* akan mengakses *server local* dan langsung memberikannya kepada *client*. Pada proses ini kecepatan akan *load* terhadap halaman *web* akan terasa lebih cepat karena *Squid Server* tidak melakukan *request* ke internet dimana akan memakan waktu yang lama.

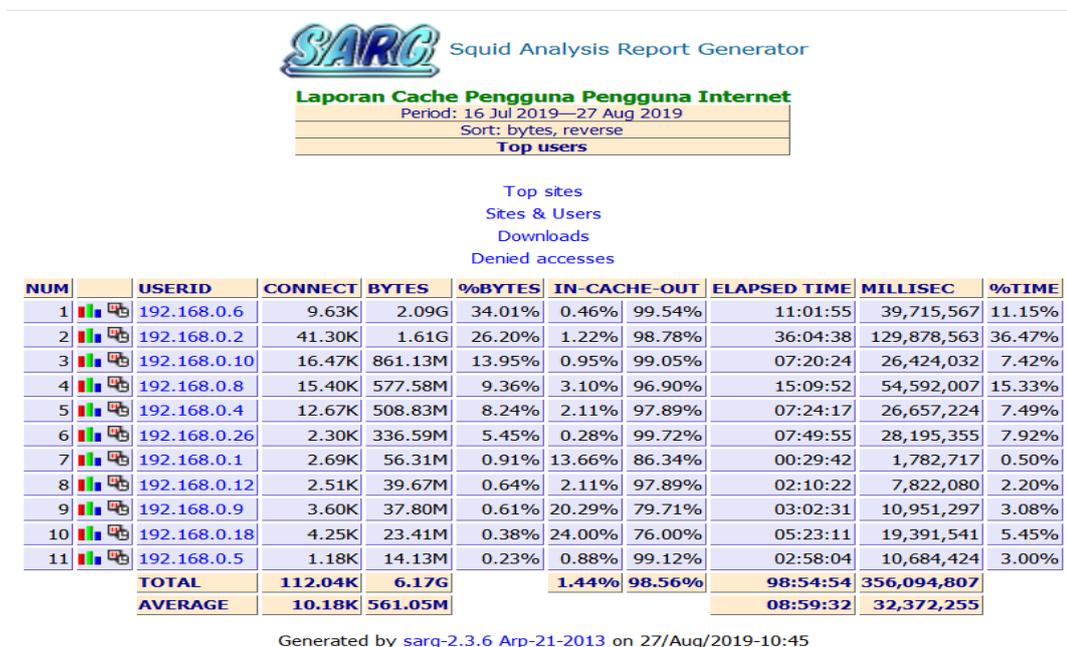
Metode ini sangat cocok bagi pengguna internet yang setiap harinya mengakses halaman yang sama. Dari data tabel 4 dapat dilihat bahwa hampir semua *client* mengakses halaman yang sama. Dengan demikian dipasangnya *Web Cache Server* dari *Squid* akan memberikan dampak yang sangat terasa terhadap kecepatan permintaan paket data halaman *website*.

e. Analisis *SARG*

Pada pembahasan ini akan dianalisa hasil laporan yang diberikan oleh aplikasi *SARG* dimana aplikasi ini akan memberikan laporan penggunaan internet terhadap situs apa saja yang diakses oleh *client* di jaringan *Squid*.

Pada gambar 37 adalah laporan *SARG* pada tanggal 16 Juli 2019 dengan jumlah *IP* aktif adalah 11. *SARG* memberi rincian detail dari aktifitas permintaan

paket atau trafik *client* serta kalkulasi jumlah paket yang diminta oleh *client* dan juga kalkulasi waktu yang digunakan oleh *client* tersebut.



Gambar 36 Laporan SARG periode 16 Juli 2019

Sumber : Hasil rancangan

Berdasarkan *data* yang dapat di lihat pada gambar 36 diketahui *client* mana yang paling aktif menggunakan internet dengan data rincian :

Sumber : Hasil rancangan

Tabel 7 Analisis *client* aktif

No	Kategori	IP Address	Nilai	Total Keseluruhan	Satuan
1	Connect	192.168.0.2	41300	112040	KB
2	Bytes	192.168.0.6	2090	6170	MB
3	Elapsed Time	192.168.0.2	36:04:38	98:54:54	Jam

Keterangan :

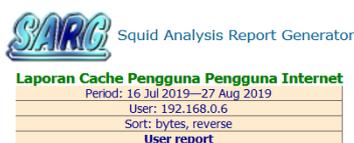
1. **Connect** : Jumlah klik yang dilakukan *client* pada halaman *website*

2. **Bytes** : Total kuota yang telah habis terpakai untuk mengakses halaman *web*
3. **Elapsed Time** : Total waktu yang dihabiskan menggunakan internet

Dari tabel 7 dapat disimpulkan bahwa :

1. Banyaknya jumlah klik yang dilakukan pada halaman *website* tidak menjadi tolak ukur bahwa *client* tersebut juga banyak menghabiskan kuota internet.
2. *Client* 192.168.0.6 adalah *client* yang paling banyak menghabiskan kuota internet pada periode 16 juli 2019 dengan total penggunaan 2090 MB dari penggunaan keseluruhan 6170 MB atau sekitar 33,88% dari keseluruhan kuota yang terpakai.
3. *Client* yang paling aktif menggunakan *internet* adalah *client* 192.168.0.2 karena jumlah *connect* yang paling banyak dalam waktu yang lebih kecil dibandingkan *client* 192.168.0.6

Selanjutnya akan diperlihatkan hasil monitoring *SARG* terhadap penggunaan internet berdasarkan alaman *domain* yang diakses. Gambar 38 akan memberikan rincian detail dari aktifitas tersebut untuk periode yang sama yaitu 16 Juli 2019.



ACCESSED SITE	CONNECT	BYTES	%BYTES	IN-CACHE	OUT	ELAPSED TIME	MILLISEC	%TIME
doc-0g-4s-docs.googleusercontent.com	2	918.77M	43.77%	0.00%	100.00%	00:39:57	2,397,735	6.04%
download888.mediafre.com	31	632.78M	30.15%	0.00%	100.00%	03:12:38	11,558,744	29.10%
fs2.letsupload.co	1	160.60M	7.65%	0.00%	100.00%	00:15:27	927,323	2.33%
k9290131.vps18.u.avcdn.net	8	101.02M	4.81%	0.00%	100.00%	00:03:10	190,927	0.48%
b1477563.iavs9x.u.avast.com	4	48.58M	2.31%	0.00%	100.00%	00:01:56	116,667	0.29%
ardownload.adobe.com	82	47.16M	2.25%	0.00%	100.00%	00:00:18	18,215	0.05%
asp-cdn.ff.avast.com	260	43.91M	2.09%	0.00%	100.00%	00:02:23	143,049	0.36%
p3357684.iavs9x.u.avast.com	32	41.26M	1.97%	0.00%	100.00%	00:01:12	72,728	0.18%
z0285146.iavs9x.u.avast.com	14	20.62M	0.98%	0.00%	100.00%	00:00:36	36,662	0.09%
m9761227.vps18tny.u.avcdn.net	8	20.45M	0.97%	0.00%	100.00%	00:00:35	35,798	0.09%
mail.google.com	303	11.53M	0.55%	1.86%	98.14%	00:03:37	217,297	0.55%
docs.google.com	155	11.26M	0.54%	17.51%	82.49%	00:01:52	112,041	0.28%
www.pcclean.io	71	7.86M	0.37%	0.04%	99.96%	00:00:37	37,807	0.10%
display.shutterstock.com	2	3.02M	0.14%	0.00%	100.00%	00:00:03	3,561	0.01%
ad4msan.com	288	1.79M	0.09%	27.45%	72.55%	00:00:16	16,861	0.04%
accounts.google.com	46	1.65M	0.08%	0.12%	99.88%	00:00:07	7,614	0.02%
drive.google.com	104	1.64M	0.08%	30.67%	69.33%	00:00:40	40,881	0.10%
www.gstatic.com	124	1.49M	0.07%	87.22%	12.78%	00:00:31	31,112	0.08%
tpc.googlesyndication.com	44	1.12M	0.05%	4.21%	95.79%	00:00:06	6,877	0.02%

ssl.gstatic.com	380	947.08K	0.05%	62.11%	37.89%	00:00:12	12,995	0.03%
fonts.gstatic.com	54	863.78K	0.04%	34.34%	65.66%	00:00:02	2,534	0.01%
imasdk.googleapis.com	8	713.47K	0.03%	0.00%	100.00%	00:00:01	1,533	0.00%
c.disquscdn.com	85	703.93K	0.03%	64.47%	35.53%	00:00:02	2,954	0.01%
hangouts.google.com	37	661.04K	0.03%	67.96%	32.04%	00:00:13	13,244	0.03%
images.taboola.com	14	630.80K	0.03%	30.04%	69.96%	00:00:02	2,881	0.01%
www.googletagservices.com	32	499.54K	0.02%	5.06%	94.94%	00:00:02	2,188	0.01%
www.google-analytics.com	722	459.13K	0.02%	0.00%	100.00%	00:00:31	31,234	0.08%
cdnjs.cloudflare.com	34	445.95K	0.02%	86.00%	14.00%	00:00:01	1,966	0.00%
contacts.google.com	14	437.82K	0.02%	29.36%	70.64%	00:00:08	8,363	0.02%
securepubads.g.doubleclick.net	36	398.80K	0.02%	0.00%	100.00%	00:00:03	3,348	0.01%
77.234.43.93	87	378.62K	0.02%	100.00%	0.00%	00:00:00	9	0.00%
77.234.42.74	87	378.62K	0.02%	100.00%	0.00%	00:00:00	0	0.00%
cdn.taboola.com	16	372.63K	0.02%	0.00%	100.00%	00:00:01	1,943	0.00%
ad4msan.win	45	367.44K	0.02%	54.47%	45.53%	00:15:06	906,772	2.28%
en.wikpedia.org	14	297.41K	0.01%	6.63%	93.37%	00:00:00	478	0.00%
s3.amazonaws.com	3	292.72K	0.01%	0.00%	100.00%	00:00:14	14,740	0.04%
kompas.api.usersider.com	5	262.11K	0.01%	0.00%	100.00%	00:00:00	388	0.00%
apis.google.com	26	259.41K	0.01%	71.04%	28.96%	00:00:03	3,832	0.01%
tempest.services.disqus.com	20	248.32K	0.01%	0.00%	100.00%	00:00:06	6,424	0.02%
www.googletagmanager.com	10	247.52K	0.01%	0.00%	100.00%	00:00:02	2,684	0.01%
drive.ad4msan.win	31	236.59K	0.01%	35.98%	64.02%	00:00:04	4,944	0.01%
www.mediafire.com	8	222.59K	0.01%	0.00%	100.00%	00:00:14	14,399	0.04%
cdns.kling.com	16	214.03K	0.01%	25.95%	74.05%	00:14:59	899,936	2.27%
celo.client-channel.google.com	70	209.73K	0.01%	56.39%	43.61%	00:59:30	3,570,031	8.99%
mc.yandex.ru	24	198.41K	0.01%	0.00%	100.00%	00:00:06	6,731	0.02%
c.aaxads.com	3	194.88K	0.01%	49.87%	50.13%	00:00:04	4,356	0.01%
play.google.com	226	193.17K	0.01%	0.00%	100.00%	00:00:14	14,032	0.04%
izz01.steplug.com	2	189.41K	0.01%	0.00%	100.00%	00:00:00	514	0.00%
ad4msan.disqus.com	6	176.77K	0.01%	0.00%	100.00%	00:00:08	8,290	0.02%
s0.2mdn.net	15	175.50K	0.01%	48.55%	51.45%	00:00:01	1,366	0.00%
adsmg.kompas.com	9	172.37K	0.01%	0.00%	100.00%	00:00:00	796	0.00%
cdn.bannerflow.com	17	169.75K	0.01%	0.00%	100.00%	00:00:00	643	0.00%
static.criteo.net	15	161.98K	0.01%	0.00%	100.00%	00:00:10	10,597	0.03%
disqus.com	44	153.25K	0.01%	0.00%	100.00%	00:00:06	6,395	0.02%
cdn.otolatrmp.com	2	150.55K	0.01%	49.94%	50.06%	00:00:02	2,119	0.01%
k259335.iavs9x.u.avast.com	1	141.38K	0.01%	0.00%	100.00%	00:00:00	288	0.00%
www.facebook.com	30	139.00K	0.01%	100.00%	0.00%	00:00:00	44	0.00%
clients6.google.com	108	137.18K	0.01%	0.00%	100.00%	00:00:32	32,991	0.08%
pagead2.googlesyndication.com	100	135.24K	0.01%	0.00%	100.00%	00:00:02	2,900	0.01%
connect.facebook.net	5	134.00K	0.01%	47.08%	52.92%	00:00:02	2,538	0.01%
h3.googleusercontent.com	37	123.49K	0.01%	15.32%	84.68%	00:00:03	3,748	0.01%
12.client-channel.google.com	29	120.95K	0.01%	37.52%	62.48%	01:11:16	4,276,684	10.77%
ak.imgfarm.com	3	120.83K	0.01%	65.62%	34.38%	00:00:01	1,695	0.00%
b357bd06b168.bitsngo.net	12	120.03K	0.01%	0.00%	100.00%	00:00:01	1,564	0.00%
translate.googleapis.com	8	119.73K	0.01%	17.25%	82.75%	00:00:00	721	0.00%
static.mediafire.com	12	118.76K	0.01%	0.00%	100.00%	00:00:04	4,162	0.01%
h6637816.iavs9x.u.avast.com	1	116.52K	0.01%	0.00%	100.00%	00:00:00	421	0.00%
23.client-channel.google.com	29	112.77K	0.01%	32.68%	67.32%	01:13:34	4,414,153	11.11%
hnwkuw.umzafkqgx.com	3	108.68K	0.01%	0.00%	100.00%	00:00:01	1,418	0.00%
firefox.settings.services.mozilla.com	3	104.57K	0.00%	0.00%	100.00%	00:00:00	151	0.00%
www.google.com	143	102.73K	0.00%	1.52%	98.48%	00:00:07	7,311	0.02%
z.moatads.com	1	98.25K	0.00%	0.00%	100.00%	00:00:01	1,891	0.00%
static0-web-prod-vidio.akamaized.net	18	97.21K	0.00%	29.06%	70.94%	00:00:00	408	0.00%
ubuntu:3128	19	95.28K	0.00%	100.00%	0.00%	00:00:00	4	0.00%
emupdate.avcdn.net	4	87.84K	0.00%	0.00%	100.00%	00:00:00	264	0.00%
ads.pubmatic.com	7	87.46K	0.00%	17.85%	82.15%	00:00:00	207	0.00%
onesignal.com	44	79.87K	0.00%	20.42%	79.58%	00:00:01	1,294	0.00%
devicemetadata.service.trafficmanager.net	38	78.15K	0.00%	0.00%	100.00%	00:00:03	3,829	0.01%
b.adnxs.com	23	73.83K	0.00%	0.00%	100.00%	00:00:05	5,915	0.01%
tekno.kompas.com	3	71.28K	0.00%	0.00%	100.00%	00:00:00	154	0.00%
ajax.googleapis.com	9	69.79K	0.00%	55.20%	44.80%	00:00:00	129	0.00%
trc.taboola.com	59	62.36K	0.00%	0.00%	100.00%	00:00:02	2,080	0.01%
chat-pa.clients6.google.com	65	61.11K	0.00%	0.00%	100.00%	00:00:18	18,810	0.05%
a1a5ab31bcd1bc68.com	15	59.25K	0.00%	12.01%	87.99%	00:00:02	2,426	0.01%
www.vidio.com	2	55.99K	0.00%	0.00%	100.00%	00:00:00	66	0.00%
fonts.googleapis.com	46	53.21K	0.00%	0.00%	100.00%	00:00:00	976	0.00%
shepherd.ff.avast.com	3	52.55K	0.00%	0.00%	100.00%	00:00:01	1,749	0.00%
TOTAL	9.63K	2.09G	34.01%	0.46%	99.54%	11:01:55	39,715,567	11.15%
AVERAGE	0	561.05M				08:59:32	32,372,255	9.09%

Generated by [sarg-2.3.6 Arp-21-2013](#) on 27/Aug/2019-10:45

Gambar 37 Laporan SARG periode 16 Juli 2019 berdasarkan url

Sumber : Hasil rancangan

Berdasarkan laporan *SARG* pada gambar 37 dapat diketahui alamat *domain* yang paling banyak diakses oleh *client* dengan rincian 3 domain teratas adalah :

Sumber : Hasil rancangan

Tabel 8 alamat *domain* yang paling banyak diakses

No	Kategori	Connect	Bytes	Keterangan
1	<i>www.google-analytics.com</i>	722	499.54	<i>Google</i>
2	<i>Ssl.gstatic.com</i>	340	947	<i>Malware</i>
3	<i>Mail.google.com</i>	303	11.53	<i>Mail google</i>

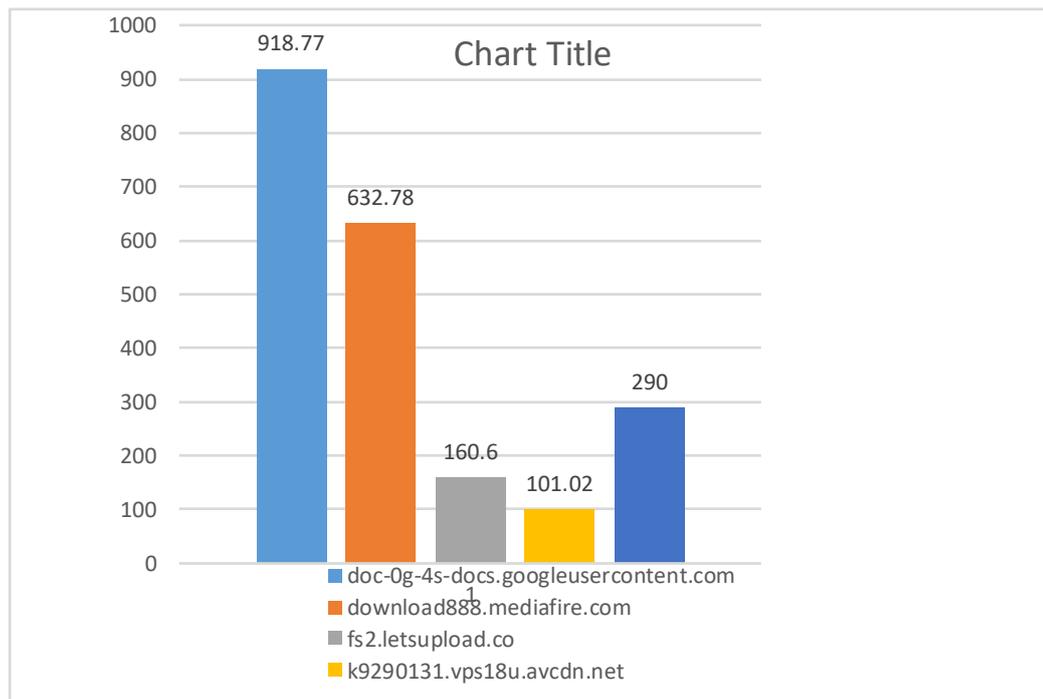
Dari data tabel 8 terlihat bahwa 3 alamat *domain* yang paling sering diakses.

Sehingga dapat di simpulkan :

1. Penggunaan internet oleh siswa webmedia lebih pada periode 16 juli 2019 lebih sering mengakses *google analytic* yaitu layanan gratis dari *Google* yang menampilkan statistik pengunjung sebuah situs *web*. *Google Analytics* dapat menelusuri pengunjung berdasarkan informasi halaman pengacu, termasuk mesin pencari, iklan, jaringan *pay-per-click*, *email marketing*, dan juga tautan yang terkandung dalam dokumen PDF
2. Komputer *client* pada jaringan tersebut sudah terinfeksi *malware* karena berdasarkan *forum* resmi milik *Microsoft* (*answers.microsoft.com*) dikatakan bahwa *ssl.gstatic* adalah sejenis *malware* yang disebut dengan *browser hijacker* yaitu program *malware* yang mengubah pengaturan *web browser* tanpa izin pengguna dan mengarahkan pengguna ke situs *web* yang tidak ingin dikunjungi oleh pengguna.

3. Jumlah akses terhadap alamat *domain* yang banyak tidak menjadi tolak ukur penggunaan kuota *internet* yang banyak.

Pada gambar 38 akan diperlihatkan grafik menggunakan kuota internet terbanyak berdasarkan alamat *domain* yang diakses pada periode 16 Juli 2019.



Gambar 38 Grafik penggunaan kuota berdasarkan *domain*

Sumber : Hasil rancangan

Selanjutnya adalah laporan *SARG* terkait aktifitas *download* yang dilakukan oleh *client* yang ada jaringan. *SARG* memberikan *report* tersebut dengan detail seperti pada gambar 39.



Squid Analysis Report Generator

Laporan Cache Pengguna Pengguna Internet

Period: 16 Jul 2019—27 Aug 2019

Downloads

USERID	IP/NAME	DATE/TIME	ACCESSED SITE
192.168.0.10	192.168.0.10	08/20/2019-10:04:45	https://redirector.gvt1.com/edgedl/widevine-cdm/4.10.1440.18-win-ia32.zip
		08/20/2019-10:04:47	http://ciscobinary.openh264.org/openh264-wm32-2e1774ab6dc6c43debb0b5b628bdf122a391d521.zip
		08/20/2019-11:16:14	https://ssl.gstatic.com/chat/sounds/incoming_message_eb39ce414e3ffba41a8e173581dc7248.mp3
		08/20/2019-11:16:14	https://ssl.gstatic.com/chat/sounds/incoming_video_short_5abdd7c1c7fa8bbd5d6b4733de315c59.mp3
		08/21/2019-12:13:14	https://ssl.gstatic.com/chat/sounds/incoming_message_eb39ce414e3ffba41a8e173581dc7248.mp3
		08/21/2019-12:13:14	https://ssl.gstatic.com/chat/sounds/incoming_video_short_5abdd7c1c7fa8bbd5d6b4733de315c59.mp3
		08/21/2019-12:21:32	https://redirector.gvt1.com/edgedl/widevine-cdm/4.10.1440.18-win-ia32.zip
192.168.0.12	192.168.0.12	07/20/2019-14:21:51	https://dl.teamviewer.com/download/TeamViewer_Setup.exe
		07/20/2019-15:27:24	https://ssl.gstatic.com/chat/sounds/incoming_message_eb39ce414e3ffba41a8e173581dc7248.mp3
		07/20/2019-15:27:24	https://ssl.gstatic.com/chat/sounds/incoming_video_short_5abdd7c1c7fa8bbd5d6b4733de315c59.mp3
192.168.0.18	192.168.0.18	08/21/2019-16:52:58	https://ssl.gstatic.com/chat/sounds/incoming_video_short_5abdd7c1c7fa8bbd5d6b4733de315c59.mp3
		08/21/2019-16:57:17	http://setup.download.avg.com/softw/avgeu.exe
		08/21/2019-17:06:53	http://www.download.windowsupdate.com/msdownload/update/v3/static/trustedr/en/authrootstl.cab
		08/21/2019-18:07:30	http://www.download.windowsupdate.com/msdownload/update/v3/static/trustedr/en/authrootstl.cab
192.168.0.2	192.168.0.2	08/01/2019-11:20:44	https://redirector.gvt1.com/edgedl/widevine-cdm/4.10.1440.18-win-x64.zip
		08/01/2019-11:21:09	http://ciscobinary.openh264.org/openh264-wm64-2e1774ab6dc6c43debb0b5b628bdf122a391d521.zip
		08/01/2019-11:35:28	http://mc.corel.com/futuretense_cs/ccurl/467/571/RunDraw.EXE
		08/01/2019-11:35:45	http://mc.corel.com/futuretense_cs/ccurl/467/571/RunDraw.EXE
		08/01/2019-16:34:37	https://ssl.gstatic.com/chat/sounds/incoming_message_eb39ce414e3ffba41a8e173581dc7248.mp3
		08/01/2019-16:34:37	https://ssl.gstatic.com/chat/sounds/incoming_video_short_5abdd7c1c7fa8bbd5d6b4733de315c59.mp3
		08/07/2019-09:15:24	https://redirector.gvt1.com/edgedl/release2/chrome/AOB27POF_VkRtq55PctNWNi_76.0.3809.100/76.0.3809.10...
		08/07/2019-09:36:54	https://web.whatsapp.com/notification_0a598282e94e87dea63e466d115e4a83.mp3
		08/07/2019-09:37:00	https://ssl.gstatic.com/chat/sounds/incoming_message_eb39ce414e3ffba41a8e173581dc7248.mp3
		08/07/2019-09:37:00	https://ssl.gstatic.com/chat/sounds/incoming_video_long_e5df1b5146e9c9dba78af43d60005200.mp3
		08/07/2019-09:37:00	https://ssl.gstatic.com/chat/sounds/incoming_video_short_5abdd7c1c7fa8bbd5d6b4733de315c59.mp3
		08/07/2019-09:37:01	https://ssl.gstatic.com/chat/sounds/incoming_message_eb39ce414e3ffba41a8e173581dc7248.mp3
		08/07/2019-09:37:01	https://ssl.gstatic.com/chat/sounds/incoming_video_long_e5df1b5146e9c9dba78af43d60005200.mp3
192.168.0.4	192.168.0.4	08/19/2019-10:07:38	http://www.download.windowsupdate.com/msdownload/update/v3/static/trustedr/en/authrootstl.cab
		08/19/2019-10:07:39	http://www.download.windowsupdate.com/msdownload/update/v3/static/trustedr/en/authrootstl.cab
		08/19/2019-10:13:16	http://setup.download.avg.com/softw/avgeu.exe
		08/19/2019-10:14:43	http://ciscobinary.openh264.org/openh264-wm32-2e1774ab6dc6c43debb0b5b628bdf122a391d521.zip
		08/19/2019-10:14:47	https://redirector.gvt1.com/edgedl/widevine-cdm/4.10.1440.18-win-ia32.zip
		08/19/2019-10:20:26	http://emupdate.bav.avcdn.net/files/bav/emupdate/20180205_wave3_avg.dll
		08/19/2019-10:27:46	https://web.whatsapp.com/notification_0a598282e94e87dea63e466d115e4a83.mp3
		08/19/2019-10:32:25	http://setup.download.avg.com/softw/avgeu.exe
		07/20/2019-15:31:08	http://setup.download.avg.com/softw/avgeu.exe
		07/20/2019-15:37:00	https://redirector.gvt1.com/edgedl/release2/chrome/ALau_fBD7PMs_75.0.3770.142/75.0.3770.142_chrome_in...
		07/20/2019-15:37:47	http://ciscobinary.openh264.org/openh264-wm32-2e1774ab6dc6c43debb0b5b628bdf122a391d521.zip
		07/20/2019-15:37:48	https://redirector.gvt1.com/edgedl/widevine-cdm/4.10.1440.18-win-ia32.zip
		07/20/2019-15:50:55	http://setup.download.avg.com/softw/avgeu.exe
		08/21/2019-12:11:59	http://sqm.microsoft.com/sqm/windows/sqmserver.dll
		08/21/2019-12:12:00	http://sqm.microsoft.com/sqm/windows/sqmserver.dll
		08/21/2019-12:12:01	http://sqm.microsoft.com/sqm/windows/sqmserver.dll
		08/21/2019-12:12:02	http://sqm.microsoft.com/sqm/windows/sqmserver.dll
		08/21/2019-12:12:03	http://www.download.windowsupdate.com/msdownload/update/v3/static/trustedr/en/authrootstl.cab
192.168.0.5	192.168.0.5	08/19/2019-11:23:52	http://au.download.windowsupdate.com/d/msdownload/update/software/ftpk/2019/08/windows10.0-kb4486153...
		08/19/2019-11:24:16	http://au.download.windowsupdate.com/d/msdownload/update/software/ftpk/2019/08/windows10.0-kb4486153...
		08/19/2019-11:24:30	http://au.download.windowsupdate.com/d/msdownload/update/software/ftpk/2019/08/windows10.0-kb4486153...
		08/19/2019-11:24:37	http://au.download.windowsupdate.com/d/msdownload/update/software/ftpk/2019/08/windows10.0-kb4486153...
		08/19/2019-11:24:42	http://au.download.windowsupdate.com/d/msdownload/update/software/ftpk/2019/08/windows10.0-kb4486153...
		08/19/2019-11:24:47	http://au.download.windowsupdate.com/d/msdownload/update/software/ftpk/2019/08/windows10.0-kb4486153...
		08/19/2019-11:24:50	http://au.download.windowsupdate.com/d/msdownload/update/software/ftpk/2019/08/windows10.0-kb4486153...
		08/19/2019-11:24:52	http://au.download.windowsupdate.com/d/msdownload/update/software/ftpk/2019/08/windows10.0-kb4486153...
		08/19/2019-11:24:54	http://au.download.windowsupdate.com/d/msdownload/update/software/ftpk/2019/08/windows10.0-kb4486153...
192.168.0.6	192.168.0.6	08/21/2019-13:36:19	http://uupdate.avast.com/files/emupdate/overseer-wave2-317.cab
		08/21/2019-13:36:20	http://uupdate.avast.com/files/emupdate/avast_disablevmm_01.cab
		08/21/2019-13:36:20	http://uupdate.avast.com/files/emupdate/disablevmm_avast.cab
		08/21/2019-13:36:20	http://uupdate.avast.com/files/emupdate/microuupdate_404.cab
		08/21/2019-13:36:20	http://uupdate.avast.com/files/emupdate/msqualitycompat_avast.cab
		08/21/2019-13:36:20	http://uupdate.avast.com/files/emupdate/msqualitycompatplus_avast.cab
		08/21/2019-13:36:21	http://uupdate.avast.com/files/emupdate/avast_antitrack_494.cab
		08/21/2019-13:36:22	http://uupdate.avast.com/files/emupdate/avast_19_r7_520.cab
		08/21/2019-13:40:27	https://www.pcclean.io/product-winutilities-pro/wuinstall.exe
		08/21/2019-13:58:03	https://ssl.gstatic.com/chat/sounds/incoming_message_eb39ce414e3ffba41a8e173581dc7248.mp3

192.168.0.8	192.168.0.8	08/21/2019-12:50:19	https://dc612.4shared.com/img/8EoKUD0gm/713fd03b/dlink_2Fdownload_2F8EoKUD0gm_2FMaudy_5FAyunda_5F...
		08/21/2019-12:51:41	https://dc746.4shared.com/img/fqY_aspQgm/ad3fa8f4/dlink_2Fdownload_2FfqY_5FaspQgm_2FBEST_5FLAGU_5FP...
		08/21/2019-12:51:55	https://dc746.4shared.com/img/fqY_aspQgm/ad3fa8f4/dlink_2Fdownload_2FfqY_5FaspQgm_2FBEST_5FLAGU_5FP...
		08/21/2019-12:52:12	https://dc746.4shared.com/img/fqY_aspQgm/ad3fa8f4/dlink_2Fdownload_2FfqY_5FaspQgm_2FBEST_5FLAGU_5FP...
		08/21/2019-12:52:18	https://dc746.4shared.com/img/fqY_aspQgm/ad3fa8f4/dlink_2Fdownload_2FfqY_5FaspQgm_2FBEST_5FLAGU_5FP...
		08/21/2019-12:53:49	https://dc746.4shared.com/img/fqY_aspQgm/ad3fa8f4/dlink_2Fdownload_2FfqY_5FaspQgm_2FBEST_5FLAGU_5FP...
		08/21/2019-12:54:01	https://dc746.4shared.com/img/fqY_aspQgm/ad3fa8f4/dlink_2Fdownload_2FfqY_5FaspQgm_2FBEST_5FLAGU_5FP...
		08/21/2019-12:54:34	https://dc746.4shared.com/img/fqY_aspQgm/ad3fa8f4/dlink_2Fdownload_2FfqY_5FaspQgm_2FBEST_5FLAGU_5FP...
		08/21/2019-12:55:57	https://dc746.4shared.com/img/fqY_aspQgm/ad3fa8f4/dlink_2Fdownload_2FfqY_5FaspQgm_2FBEST_5FLAGU_5FP...
192.168.0.9	192.168.0.9	07/22/2019-13:21:33	https://ssl.gstatic.com/chat/sounds/incoming_message_eb39ce414e3ffb41a8e173581dc7248.mp3
		07/22/2019-13:21:33	https://ssl.gstatic.com/chat/sounds/incoming_video_short_5abdd7c1c7fa8bbd5d6b4733de315c59.mp3

Gambar 39 *Download Site*

Sumber : Hasil rancangan

Pada gambar 39 adalah halaman *download site* dimana *SARG* memberikan gambaran alamat *website* apa saja yang digunakan *client* untuk mendownload file. Terlihat bahwa alamat yang paling banyak menuju ke halaman *update* sistem operasi *windows* dapat disimpulkan :

1. Aktifitas *download* yang dilakukan oleh *client* banyak bersumber dari *website 4Shared.com*.
2. Jenis *download* pada halaman *update windows* adalah jenis *download* yang dilakukan secara otomatis oleh sistem operasi *client* sehingga dapat dikatakan bahwa aktifitas *download* yang dilakukan sebagian besar bukan bersumber dari kehendak *client* melainkan oleh sistem operasi *client*.



Squid Analysis Report Generator

Laporan Cache Pengguna Pengguna Internet

Period: 16 Jul 2019—27 Aug 2019

Denied

USERID	IP/NAME	DATE/TIME	ACCESSED SITE
192.168.0.10	192.168.0.10	08/20/2019-09:57:51	https://www.facebook.com
		08/20/2019-09:57:52	https://www.facebook.com
		08/20/2019-09:58:03	https://www.facebook.com
		08/20/2019-09:58:03	https://www.facebook.com
		08/20/2019-09:58:04	https://www.facebook.com
		08/20/2019-09:58:11	https://www.facebook.com
		08/20/2019-09:58:12	https://www.facebook.com
			<i>185 more denied accesses not shown here...</i>
192.168.0.12	192.168.0.12	07/20/2019-14:21:06	https://www.facebook.com
		07/20/2019-14:21:34	https://www.facebook.com
		07/20/2019-14:21:34	https://www.facebook.com
		07/20/2019-14:21:35	https://www.facebook.com
		07/20/2019-14:21:35	https://www.facebook.com
		07/20/2019-14:21:36	https://www.facebook.com
		07/20/2019-14:21:42	https://www.facebook.com
		07/20/2019-15:27:13	https://www.facebook.com
		07/20/2019-15:27:13	https://www.facebook.com
		07/20/2019-15:27:16	https://www.facebook.com
			<i>24 more denied accesses not shown here...</i>
192.168.0.18	192.168.0.18	08/21/2019-16:52:28	https://www.facebook.com
		08/21/2019-16:52:28	https://www.facebook.com
		08/21/2019-18:29:20	https://www.facebook.com
		08/21/2019-18:29:23	https://www.facebook.com
		08/21/2019-16:52:12	https://www.youtube.com
		08/21/2019-16:52:12	https://www.youtube.com
		08/21/2019-16:52:15	https://www.youtube.com
		08/21/2019-16:52:17	https://www.youtube.com
		08/21/2019-18:24:28	https://www.youtube.com
		08/21/2019-18:24:29	https://www.youtube.com
192.168.0.2	192.168.0.2	08/07/2019-09:37:09	https://www.facebook.com
		08/07/2019-10:13:33	https://www.facebook.com
		08/07/2019-10:13:33	https://www.facebook.com
		08/07/2019-10:13:37	https://www.facebook.com
		08/07/2019-10:13:37	https://www.facebook.com
		08/07/2019-10:13:38	https://www.facebook.com
		08/07/2019-10:13:38	https://www.facebook.com
		08/07/2019-10:13:42	https://www.facebook.com
		08/07/2019-10:13:43	https://www.facebook.com
		08/07/2019-10:13:47	https://www.facebook.com
			<i>66 more denied accesses not shown here...</i>
192.168.0.4	192.168.0.4	08/19/2019-10:07:47	https://www.facebook.com
		08/19/2019-10:07:47	https://www.facebook.com
		08/19/2019-10:07:47	https://www.facebook.com
		08/19/2019-10:08:59	https://www.facebook.com
		08/19/2019-10:09:00	https://www.facebook.com
		08/19/2019-10:10:05	https://www.facebook.com
		08/19/2019-10:10:05	https://www.facebook.com
		08/19/2019-10:10:22	https://www.facebook.com
		08/19/2019-10:10:22	https://www.facebook.com
		08/19/2019-10:10:29	https://www.facebook.com
			<i>63 more denied accesses not shown here...</i>
192.168.0.6	192.168.0.6	08/21/2019-12:55:38	https://www.facebook.com
		08/21/2019-13:50:23	https://www.facebook.com
		08/21/2019-13:50:42	https://www.facebook.com
		08/21/2019-13:51:25	https://www.facebook.com
		08/21/2019-13:51:35	https://www.facebook.com
		08/21/2019-13:52:01	https://www.facebook.com
192.168.0.8	192.168.0.8	08/21/2019-12:50:22	https://www.facebook.com
		08/21/2019-12:50:33	https://www.facebook.com
		08/21/2019-12:57:36	https://www.facebook.com
		08/21/2019-13:00:51	https://www.facebook.com
		08/21/2019-13:01:30	https://www.facebook.com
		08/21/2019-13:01:39	https://www.facebook.com
		08/21/2019-13:03:36	https://www.facebook.com

192.168.0.6	192.168.0.6	08/21/2019-12:55:38	https://www.facebook.com
		08/21/2019-13:50:23	https://www.facebook.com
		08/21/2019-13:50:42	https://www.facebook.com
		08/21/2019-13:51:25	https://www.facebook.com
		08/21/2019-13:51:35	https://www.facebook.com
		08/21/2019-13:52:01	https://www.facebook.com
			22 more denied accesses not shown here...
192.168.0.8	192.168.0.8	08/21/2019-12:50:22	https://www.facebook.com
		08/21/2019-12:50:33	https://www.facebook.com
		08/21/2019-12:57:36	https://www.facebook.com
		08/21/2019-13:00:51	https://www.facebook.com
		08/21/2019-13:01:30	https://www.facebook.com
		08/21/2019-13:01:39	https://www.facebook.com
		08/21/2019-13:03:36	https://www.facebook.com
		08/21/2019-13:03:39	https://www.facebook.com
		08/21/2019-13:03:41	https://www.facebook.com
		08/21/2019-13:04:21	https://www.facebook.com
			22 more denied accesses not shown here...
192.168.0.9	192.168.0.9	07/22/2019-13:06:09	https://www.facebook.com
		07/22/2019-13:06:16	https://www.facebook.com
		07/22/2019-13:21:10	https://www.facebook.com
		07/22/2019-13:23:25	https://www.facebook.com
		07/22/2019-13:23:25	https://www.facebook.com
		07/22/2019-13:06:29	https://www.youtube.com
		07/22/2019-13:06:30	https://www.youtube.com
		07/22/2019-13:06:31	https://www.youtube.com
		07/22/2019-13:21:24	https://www.youtube.com
		07/22/2019-13:21:27	https://www.youtube.com
			7 more denied accesses not shown here...

Gambar 40 Denied SARG

Sumber : Hasil rancangan

Laporan SARG pada gambar 40 memberikan keterangan alamat *domain* yang diblok oleh *Squid*. Sesuai dengan *rule web* blok yang sudah dibuat sebelumnya bahwa alamat *domain* yang tidak diizinkan untuk diakses akan diblokir oleh *Squid*. Berikut adalah rincian jumlah *request* yang dilakukan oleh *client* terhadap alamat *domain* yang diblokir untuk periode 16 Juli 2019

Sumber : Hasil rancangan

Tabel 9 Rincian permintaan *domain* yang diblokir

Domain	Jumlah Request	Status
www.facebook.com	86	Blokir
www.youtube.com	11	Blokir

BAB V

PENUTUP

5.1 Kesimpulan

Kesimpulan yang dapat ditarik berdasarkan rancangan sistem *monitoring squid proxy* menggunakan SARG adalah sebagai berikut:

- a. Sistem *monitoring proxy server* yang diterapkan telah berhasil dibangun dengan baik, keseluruhan *system* mesin *server proxy*, *squid* dan SARG telah berhasil di implementasikan dengan baik di *webmedia training center*
- b. Konfigurasi yang dilakukan membantu perusahaan khususnya *webmedia training center* dalam pengaturan hak akses dan melarang akses (*filtering access*) ke situs yang tidak diperlukan seperti situs konten negatif, *facebook* dan *youtube*. *Web Cache* pada *squid proxy server* membuat *user* dapat mengakses halaman *web* sebelumnya dengan cepat dengan menyimpan layanan data pada *proxy server* sehingga dapat mengurangi waktu akses *user*

5.2 Saran

Setelah memaparkan beberapa kesimpulan, penulis akan memberikan beberapa saran yang akan di berikan untuk penelitian ini yaitu:

- a. Penulis menyarankan dimana kedepan nya perlu adanya pembagian *bandwith* yang merata untuk setiap ruangan yang ada di *webmedia*, karena dilihat dari hasil *monitoring* bahwa ada sebagian *user* yang memakai *bandwith* terlalu besar dalam mengakses jaringan *internet*.
- b. Sistem layanan pada *proxy server* ini diharapkan bisa lebih dikembangkan kembali fungsi layanannya seiring dengan fungsi dan kegunaannya.

DAFTAR PUSTAKA

- Andrian, Yudhi, and Purwa Hasan Putra. "Analisis Penambahan Momentum Pada Proses Prediksi Curah Hujan Kota Medan Menggunakan Metode Backpropagation Neural Network." Seminar Nasional Informatika (SNIIf). Vol. 1. No. 1. 2017.
- Azmi, Fadhillah, and Winda Erika. "Analisis keamanan data pada block cipher algoritma Kriptografi RSA." CESS (Journal of Computer Engineering, System and Science) 2.1: 27-29.
- Barus, S., Sitorus, V. M., Napitupulu, D., Mesran, M., & Supiyandi, S. (2018). Sistem Pendukung Keputusan Pengangkatan Guru Tetap Menerapkan Metode Weight Aggregated Sum Product Assesment (WASPAS). Jurnal Media Informatika Budidarma, 2(2).
- Batubara, S., Wahyuni, S., & Hariyanto, E. (2018, September). Penerapan Metode Certainty Factor Pada Sistem Pakar Diagnosa Penyakit Dalam. In Seminar Nasional Royal (SENAR) (Vol. 1, No. 1, pp. 81-86).
- Dhany, H. W., Izhari, F., Fahmi, H., Tulus, M., & Sutarman, M. (2017, October). Encryption and decryption using password based encryption, MD5, and DES. In International Conference on Public Policy, Social Computing and Development 2017 (ICOPOSDev 2017) (pp. 278-283). Atlantis Press.
- Erika, Winda, Heni Rachmawati, and Ibnu Surya. "Enkripsi Teks Surat Elektronik (E-Mail) Berbasis Algoritma Rivest Shamir Adleman (RSA)." Jurnal Aksara Komputer Terapan 1.2 (2012).
- Eka Purnama, Bambang. 2016. *Konsep Dasar Internet*. Bandung: Penerbit Teknosain
- Fachri, Barany. "Perancangan Sistem Informasi Iklan Produk Halal Mui Berbasis Mobile Web Menggunakan Multimedia Interaktif." Jurasik (Jurnal Riset Sistem Informasi dan Teknik Informatika) 3 (2018): 98-102.
- Fuad, R. N., & Winata, H. N. (2017). Aplikasi keamanan file audio wav (waveform) dengan terapan algoritma RSA. InfoTekJar: Jurnal Nasional Informatika dan Teknologi Jaringan, 1(2), 113-119.
- Ginting, G., Fadlina, M., Siahaan, A. P. U., & Rahim, R. (2017). Technical approach of TOPSIS in decision making. Int. J. Recent Trends Eng. Res, 3(8), 58-64.

- Hafni, Layla, and Rismawati Rismawati. "Analisis faktor-faktor internal yang mempengaruhi nilai perusahaan pada perusahaan manufaktur yang terdaftar di BEI 2011-2015." *Bilancia: Jurnal Ilmiah Akuntansi* 1.3 (2017): 371-382.
- Sutedjo Dharma Oetomo, Budi. 2004. *Konsep & Perancangan Jaringan Komputer*. Yogyakarta: Penerbit Andi
- Sumartono, I., Siahaan, A. P. U., & Mayasari, N. (2016). An overview of the RC4 algorithm. *IOSR J. Comput. Eng.*, 18(6), 67-73.
- Supiyandi, S., Hermansyah, H., & Sembiring, K. A. (2020). Implementasi dan Penggunaan Algoritma Base64 dalam Pengamanan File Video. *Jurnal Media Informatika Budidarma*, 4(2), 340-346.
- Syahputra, Rizki, and Hafni Hafni. "Analisis kinerja jaringan switching clos tanpa buffer." *journal of science and social research* 1.2 (2018): 109-115.
- Yota Ernanda Aryanto, Kadek & Surya Mahedy, Kadek. 2014. *Jaringan Komputer*. Penerbit Graha Ilmu
- Mauladi. 2005. *Tip dan Trik Linux*. Penerbit PT Elex Media Komputindo
- Odom, Wendell. Copyright 2005. *Computer Networking First-Step*. Penerjemah Tim Penerjemah. Yogyakarta: Penerbit Andi
- Ali Pangera, Abas. 2008. *Menjadi Administrator Jaringan Nirkabel*. Yogyakarta: Penerbit Andi
- Prakoso, Samuel. 2005. *Jaringan Komputer Linux: Konsep Dasar, Instalasi, Aplikasi, Keamanan dan Penerapan*. Yogyakarta: Penerbit Andi
- Priyo Utomo, Eko. 2011. *Membangun jaringan Komputer dan Server Internet*. Yogyakarta: Penerbit MediaKom
- Madcoms. 2015. *Panduan Lengkap Membangun Sendiri Sistem Jaringan Komputer*. Yogyakarta: Penerbit Andi
- Sugiantoro, Bambang. 2012. *Panduan Praktis Instalasi Jaringan Komputer dan Sistem Operasi Linux Ubuntu*. Yogyakarta: Penerbit Suka-Press UIN Sunan Kalijaga
- Saini, Kulbir. 2011. *Squid Proxy Server 3.1*. Mumbai : Packt Publishing
- Tasril, V. (2018). Sistem Pendukung Keputusan Pemilihan Penerimaan Beasiswa Berprestasi Menggunakan Metode Elimination Et Choix Traduisant La Realite. *INTECOMS: Journal of Information Technology and Computer Science*, 1(1), 100-109.

Wahyuni, Sri. "Implementasi Rapidminer Dalam Menganalisa Data Mahasiswa Drop Out." Jurnal Abdi Ilmu 10.2 (2018): 1899-1902.