

IMPLEMENTASI SQUID SEBAGAI PROXY SERVER UNTUK AKSELERASI PENGGUNAAN INTERNET

Slamet¹⁾

1) Jurusan Sistem Informasi. STIKOM Surabaya, email : slamet@stikom.edu

Abstract: Every day new sites are always popping up and the number is always increasing. These sites provide information that is easily accessible by anyone, anytime. Sustainability of this process would cost computer performance by forming a log file on the server and client. If the file is not organized well, then this file will continue to grow and slow down the performance of an operating system. This means that if this happens on a computer network, this file will get bigger and become a burden that can interfere with the accumulation of information systems. Proxy or more specifically referred to as a web proxy, used for sharing Internet and as a bulwark of defense is good enough to block or filter out sites. It can also restrict access to the Internet on our network. In the solution used a Squid proxy server software that is configured as a caching proxy server, url filtering and http internet access protocol. So that only certain people are able to use internet access. This research used Ubuntu 4.10 Server as server operating system, Apache Web Server as web server software that integrates with MySQL software for user authentication management system.

Keyword: *Proxy Server, Squid, Filtering, Authentication*

Teknologi internet sudah dapat dinikmati oleh siapa saja, begitu juga informasi yang disajikan di internet adalah daya tarik tersendiri bagi seseorang untuk mengikuti perkembangan informasi terkini. Setiap hari situs-situs baru selalu bermunculan dan jumlahnya makin hari selalu meningkat dibuktikan dengan munculnya perusahaan *dotcom* yang mengambil kesempatan untuk meraih keuntungan bisnis lewat teknologi internet.

Kita mengetahui bahwa situs-situs tersebut menyajikan informasi yang mudah diakses oleh siapa saja dan dimana saja bagi mereka yang membutuhkan informasi secara cepat dan mudah. Koneksi internet masih mempunyai kendala, salah satunya adalah *bandwidth* atau lebih dikenal dengan lebar jalur data. Sampai saat ini, *bandwidth* merupakan salah satu elemen akses internet yang masih mahal sehingga penggunaannya harus dikelola secara efektif. Bila kita ibaratkan mobil sebagai data, maka *bandwidth* dapat kita umpamakan sebagai jalan rayanya. Logikanya, bila data yang diambil dari dokumen *web* pada saat melakukan *browsing* itu intensitas dan arusnya besar, maka *bandwidth* yang kita alokasikan akan terasa penuh dan berakibat adanya antrian.

Di lain hal, sebuah lingkungan intranet yang terhubung dengan internet kemungkinan besar akan mengakses suatu situs dengan alamat yang sama secara berulang-ulang untuk mendapatkan informasi. Suatu contoh, bila kita membuka sebuah situs www.stikom.edu, maka web browser akan menyalin file dari situs *stikom.edu* ke dalam komputer lokal. Apabila file ini tidak diorganisasikan secara baik, maka file ini akan terus membesar dan memperlambat kinerja sebuah sistem operasi. Artinya file ini akan semakin membesar dan menjadi beban yang mengganggu sistem informasi. Ada dua macam *web cache* yang bisa digunakan, diantaranya adalah *cache* pada *web browser* dan *cache* pada *proxy server*.

Cache jenis pertama diterapkan oleh *web browser* untuk menyalin obyek pada komputer lokal. Jenis ini sangat sederhana dan sudah ter-*install* secara otomatis pada saat kita menggunakan komputer. Sedangkan *cache* jenis kedua lebih didesain untuk menyimpan obyek yang terkoneksi ke dalam jaringan lokal dan cenderung dalam skala yang lebih besar. *Cache* jenis ini dinamakan *proxy* karena bisa digunakan secara simultan oleh beberapa pengguna sekaligus dan *di-share* untuk melakukan *browsing* ke internet.

Karena banyaknya *software cache* yang beredar dan keberagaman akan kelebihan dan kekurangan masing-masing, maka pada penulisan kali ini hanya akan disajikan implementasi *Squid* sebagai salah satu *software* untuk *proxy server*. *Squid* mempunyai kelebihan dari sisi kestabilan dan kemampuan *filtering* dalam menangani jaringan berskala besar. Keunggulan lainnya adalah mempunyai kemampuan *parent-sibling*, mudah dikonfigurasi, gratis dan mempunyai stabilitas yang cukup tinggi sehingga diharapkan akselerasi penggunaan internet dapat ditingkatkan.

TEORI PENUNJANG

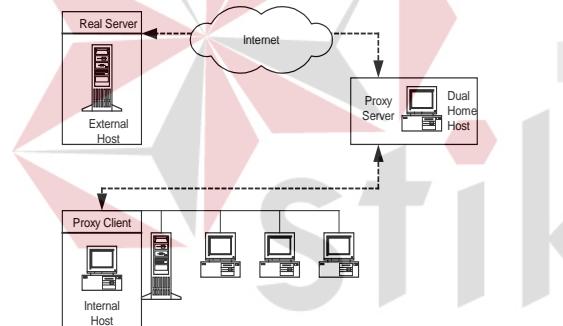
1. Linux sebagai Sistem Informasi

Linux (Sofyan: 2000) adalah sebuah Sistem Operasi (OS) turunan dari UNIX, yang merupakan implementasi independen dari standard IEEE untuk OS yang bernama POSIX (Portable Operating System Interface). OS adalah perangkat lunak (software) yang mengatur koordinasi kerja antar semua perlengkapan perangkat keras (hardware) dalam sebuah komputer. Linux memiliki kemampuan yang berbasis ke standard POSIX meliputi *true-multitasking*, *virtual memory*, *shared libraries*, *demand-loading*, *proper memory management*, dan *multiuser*. Linux seperti layaknya OS UNIX lainnya, mendukung banyak *software* mulai dari TEX, X Window, GNU C/C++ sampai ke TCP/IP.

Linux pertama kali dibuat oleh Linus Torvalds di Universitas Helsinki, Finlandia. Kemudian Linux dikembangkan lagi dengan bantuan dari banyak *programmer* dan *hacker* UNIX di seluruh dunia. Sekarang Linux bisa diperoleh dari distribusi-distribusi yang umum digunakan, misalnya: Ubuntu, RedHat, Mandrake, Debian, Slackware, Caldera, Stampede Linux, TurboLinux dan lain-lain. *Kernel* yang digunakan adalah sama-sama Linux *kernel*, sedangkan perbedaannya hanyalah paket-paket aplikasi yang disertakan, sistem penyusunan direktori, *init style* dan lain-lain.

2. Proxy Server

Proxy server (Rafiudin: 2008) adalah aplikasi khusus atau program server yang berjalan pada *host firewall*; baik pada *dual-homed host* yang memiliki sebuah *interface* ke jaringan internal dan *interface* lain ke jaringan eksternal, atau pada *bastion host* yang memiliki akses ke Internet dan dapat diakses oleh mesin internal. Program ini menangani *request-request* untuk *service-service* internet dari user dan melewatkannya ke *service* yang sebenarnya. Proxy menyediakan koneksi pengganti dan bertindak selaku *gateway* terhadap *service-service* tersebut. Oleh karena itu proxy sering juga disebut *gateway level* aplikasi.



Gambar 1. Penggunaan proxy server

Proxy server menghubungkan *user* pada jaringan internal dengan *service* pada *internet*. *User* dan *service* tersebut tidak berkomunikasi secara langsung. Masing-masing berhubungan dengan *proxy* dan *proxy* yang menangani hubungan antara *user* dan *service* di belakang layar. *Proxy server* dapat membatasi apa yang dapat dilakukan oleh *user*, karena *proxy* dapat memutuskan apakah suatu *request* dari *user* diperbolehkan atau ditolak.

3. Squid sebagai Proxy Server

Squid (Maryanto: 2001) dikenal sebagai aplikasi *cache* yang handal. Squid berlisensi GLP (GNU Public License, lihat di <http://www.gnu.org> alias *open source* yang dalam pembuatannya melibatkan banyak kepala dan jari. Dengan *open source* ini, squid bisa dikatakan selalu dalam tahap pembuatan (beta/develop version) selama hidupnya. Namun tidak berarti squid jelek dan tidak stabil, bahkan justru kebalikannya. Produk *open source* otomatis menjadi *beta tester* yang diharapkan akan memberikan kontribusi berupa input kepada team

developer tentang *bug* dan kesalahan yang ditemuinya (bahkan lebih dari itu, *beta tester* terkadang juga ikut memperbaiki dan mengirimkan *patchnya*). Dalam perkembangannya tidak hanya segelintir orang yang menjadi *beta tester*, semakin menarik dan populer sebuah produk *open source*, semakin banyak *beta tester* yang bekerja secara suka rela. Oleh sebab itu ini menjadikannya lebih banyak ditemukannya *bug* baru sekaligus dengan *bug fix-nya*. Pada akhirnya mengakibatkan membaiknya produk *open source* bisa diandalkan. Demikian *life cycle* secara umum produk *open source*.

Squid dalam sejarah pembuatannya merupakan produk turunan dari *Harvest Cache* (dibuat oleh *Harvest Project*, <http://harvest.cs..colorado.edu/>) dan nama squid dipilih oleh nama pembuatnya karena nama bagus lainnya sudah diambil orang lain. Masih dari sumber yang sama, squid dikatakan sebagai *server proxy cache* yang mempunyai *performance* tinggi dan mendukung operasional FTP, Gopher dan HTTP. Dalam memenuhi permintaan *client*, squid menangani dalam satu proses I/O (input/output) dan tidak bersifat *blocking*.

Squid hampir bersifat monolitik (seperti sendmail) dimana seluruh pekerjaan mulai dari menangani permintaan *client*, mengambil, mencari dilakukan oleh satu proses. Hampir tidak ada proses lain, kecuali untuk versi terbaru, untuk proses I/O squid menyerahkan operasionalnya pada aplikasi lain yaitu *diskd*. Squid dibuat oleh sebuah komunitas internet yang dipimpin oleh Duane Wessel dari National Laboratory for Applied Network Research dan dibayai oleh National Science Foundation. Dan seperti yang kita tahu, squid berjalan di atas Unix dan variannya, juga termasuk linux. Jenis varian Unix yang dapat sampai saat ini diketahui berhasil menjalankan squid diantaranya:

- Ubuntu
- FreeBsd
- NetBsd
- BSDI
- OSF and Digital Unix
- IRIX
- SunOS/Solaris
- NeXTStep
- SCO Unix
- AIX
- HP-UX
- Linux lain (semua distribusi)

METODE PENELITIAN

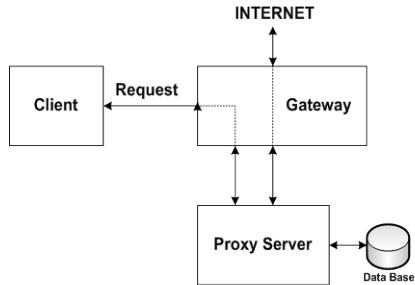
Langkah-langkah yang ditempuh dalam menyelesaikan tugas ini adalah sebagai berikut:

1. Studi Pendahuluan

Studi pendahuluan meliputi studi mengenai:

- Konsep tentang internet dan jaringan TCP/IP.
- Konsep tentang Linux sebagai basis sistem operasi yang akan digunakan.
- Konsep tentang Firewall sebagai pendekatan implementasi
- Penggunaan software Squid sebagai Proxy Server

2. Perancangan Sistem



Gambar 2. Alur Kerja Sistem

Proxy bekerja dengan mendengarkan *request* dari *client* internal dan mengirim *request* tersebut ke jaringan eksternal seolah-olah proxy server itu sendiri yang menjadi *client*. Pada waktu proxy server menerima respon dari internet, ia memberikan respon tersebut ke *client* yang asli seolah-olah ia *public server*.

Proxy server memperbarui *request* layanan pada jaringan eksternal atas nama *client* mereka pada jaringan *private*. Ini secara otomatis menyembunyikan identitas dan jumlah *client* pada jaringan internal dari jaringan eksternal. Karena posisi mereka di antara *client* internal dan *public server*, proxy juga dapat menyimpan *content* yang sering diakses dari jaringan publik untuk mengurangi akses ke jaringan publik tersebut.

3. Implementasi Sistem

Pada tahap implementasi sistem dilakukan pengaplikasian dengan perangkat lunak yang digunakan antara lain:

- Ubuntu 10.4 i386 Server, merupakan Distro yang digunakan sebagai basis Sistem Operasi.
- Squid, merupakan software berbasis linux yang diimplementasikan sebagai proxy server.
- Mysql, merupakan software berbasis linux yang dimplementasikan sebagai sistem manajemen autentikasi *user*

HASIL DAN PEMBAHASAN

Dari penelitian ini dihasilkan *log* (laporan) yang mencakup informasi sebagai berikut:

- Ringkasan permohonan (*request*) penggunaan metode (yaitu, GET, POST, dan lain-lain)
- Ringkasan penggunaan protokol (yaitu, http, ssl, dan lain-lain)
- Ringkasan penggunaan klien (yaitu, total untuk setiap PC pada jaringan Anda)
- Ringkasan penggunaan server (yaitu, total untuk situs yang dikunjungi)
- Ringkasan jenis *url* (yaitu, *request*, gambar, direktori, html, file yang dieksekusi, dan lain-lain)
- Ringkasan *url top-level domain* (yaitu, .id, .com, .org, .au, dan lain-lain)
- Grafik yang menunjukkan penggunaan internet secara real time

Sedangkan konfigurasi Squid yang dihasilkan dalam penelitian ini adalah:

```
auth_param basic program
/usr/lib/squid/mysql_auth
auth_param basic children 5
auth_param basic realm Squid proxy-caching web
server
auth_param basic credentialsttl 2 hours
auth_param basic casesensitive off
acl mysql_user proxy_auth REQUIRED
acl all src all
acl manager proto cache_object
acl localhost src 127.0.0.1/32
acl to_localhost dst 127.0.0.0/8
acl localnet src 222.124.29.224/27
#
acl SSL_ports port 443           # https
acl SSL_ports port 563           # snews
acl SSL_ports port 873           # rsync
acl Safe_ports port 80            # http
acl Safe_ports port 21            # ftp
acl Safe_ports port 443           # https
acl Safe_ports port 70            # gopher
acl Safe_ports port 210           # wais
acl Safe_ports port 1025-65535  # unregistered
ports
acl Safe_ports port 280           # http-mgmt
acl Safe_ports port 488           # gss-http
acl Safe_ports port 591           # filemaker
acl Safe_ports port 777           # multiling
http
acl Safe_ports port 631           # cups
acl Safe_ports port 873           # rsync
acl Safe_ports port 901           # SWAT
acl purge method PURGE
acl CONNECT method CONNECT
acl admin src 222.124.29.241/255.255.255.255
acl nasty_users src
222.124.29.249/255.255.255.255
acl dstubuntu dst
222.124.29.228/255.255.255.255
acl dstsiina dst 222.124.29.235/255.255.255.255
acl dstlearning dst
222.124.29.243/255.255.255.255
acl dstsicyca dst
222.124.29.229/255.255.255.255
acl wireless src 222.124.29.242/255.255.255.255
acl useragent browser -i ^.*NSPlayer.*
acl useragent browser -i ^.*player.*
acl useragent browser -i ^.*Windows-Media-Player.*
acl useragent browser -i
^.*live_suarasurabaya.*
acl useragentq rep_mime_type ^.*video.*
acl useragentq rep_mime_type ^.*audio.*
acl zipfile urlpath_regex -i \.zip$ 
acl swffile urlpath_regex -i \.swf.* 
acl youtube_getvideo urlpath_regex -i
\get_video\?
acl webtorrents urlpath_regex -i ^.*torrent*.* 
http_access allow manager localhost
http_access allow purge localhost
http_access deny manager
http_access deny purge
http_access deny !Safe_ports
http_access deny CONNECT !SSL_ports
http_access deny useragent
http_access deny useragentq
http_access deny chat_dom !wireless
http_access deny audiostream !dstlearning
#http_access deny file_ext_block !dstsicyca
!dstubuntu !dstsiina
http_access deny chat_dom
http_access deny tunnel_dom
http_access deny webproxy_dom !dst_proxy_local
http_access deny url-block_dom
http_access deny porn_dom
http_access deny youtube_getvideo
http_access deny webtorrents
http_access deny torrent_dom
```

```

delay_pools 1
delay_class 1 1
delay_parameters 1 1000/16000
delay_access 1 allow file_ext_block
delay_access 1 deny ALL
#http_access allow localnet
http_access allow mysql_user
http_access allow localhost
http_access deny all
#Allow ICP queries from local networks only
icp_access allow localnet
icp_access deny all
# Squid normally listens to port 3128
http_port 3128
cache_mem 40 MB
cache_swap_low 90
cache_swap_high 95
maximum_object_size 4096 KB
minimum_object_size 4 KB
maximum_object_size_in_memory 8 KB
hierarchy_stoplist cgi-bin ?
cache_dir ufs /home/proxy/cache 4000 16 256
access_log /var/log/squid/access.log squid
refresh_pattern ^ftp: 1440 20%
10080
refresh_pattern ^gopher: 1440 0%
1440
refresh_pattern -i (/cgi-bin/|\.?) 0 0%
0
refresh_pattern . 0 20%
4320
acl apache rep_header Server ^Apache
broken_vary_encoding allow apache
hosts_file /etc/hosts

```

SIMPULAN

Squid adalah salah satu di antara *software* yang digunakan sebagai proxy server yang sampai saat ini telah diakui dan terbukti paling tangguh dan stabil. Dari penelitian ini dapat disimpulkan beberapa hal, yaitu:

- Pemakaian *bandwidth* menjadi sangat optimal, dimana 80 % trafik internet yang dipakai oleh web dapat diatur dan direkayasa sesedikit mungkin.
- Kemudahan dalam mengatur *service* maupun *port* pada *http*, *ftp*, *icp* dan sebagainya.
- Adanya ‘*access control list*’ untuk pengaturan pengaturan akses ke internet termasuk di dalamnya untuk mencegah akses ke situs-situs porno. Di mana hal ini merupakan solusi tepat untuk pengaturan internet pada lembaga-lembaga pendidikan maupun perkantoran.
- Meningkatkan kenyamanan dalam berlayar di dunia *cyber*, karena adanya peningkatan akselerasi internet yaitu pemakaian internet menjadi lebih cepat.

DAFTAR RUJUKAN

- Maryanto, Dodi. 2001. *Optimasi Akses Internet dengan Squid*. Jakarta: Elex Media Komputindo
- Portal bagi para Squid Developer, <http://squid.sourceforge.net/>. Diakses pada tanggal 17 Juli 2010
- Purbo, Onno dan Dodi Maryanto. 2000. *Membangun Server Internet dengan FreeBSD*. Jakarta: Elex Media Komputindo
- Rafiudin, Rahmat. 2008. *Squid : Koneksi Anti Mogok*. Yogyakarta: Andi Publisher
- Sembiring, Jhoni H. 2002. *Jaringan Komputer berbasis Linux*. Jakarta: Elex Media Komputindo

Situs Resmi Squid. <http://www.squid-cache.org/>. Diakses pada tanggal 6 Juli 2010

Sofyan, Ahmad. 2000. *Server Linux*. Jakarta: Nurul Fikri Computer & Statistik – YPTE

Wahana Komputer. 2009. *Langkah Mudah Administrasi Jaringan Menggunakan Linux Ubuntu 9*, Semarang: Wahana Komputer