PENGEMBANGAN PERANGKAT LUNAK MANAJEMEN KONEKSI INTERNET UNTUK SOHO

Muchammad Husni dan David Yulianto Santoso

Jurusan Teknik Informatika – Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

ABSTRAK

Untuk bisa mengakses Internet, komputer harus terhubung ke Internet melalui Internet Service Provider dan menggunakan IP address Internet. Jumlah IP address Internet saat ini (IPv4) sangat terbatas, oleh karena itu penggunaannya dibatasi dan diatur dengan sistem sewa yang cukup mahal. Alternatif koneksi Internet yang paling murah dan umum digunakan adalah Dial-up, komputer client akan otomatis menggunakan IP address dynamic yang disewakan selama kita terkoneksi.

Apabila pada Local Area Network terdapat beberapa komputer yang ingin menggunakan Internet secara bersamaan apakah harus dilakukan Dial-up untuk tiap komputer ? Tidak, terdapat 2 alternativ Internet-Sharing yakni Masquerading dan Proxy. Saat ini terdapat beberapa software yang menyediakan solusi Internet-Sharing dengan Masquerading atau Proxy yang berbasis sistem operasi Windows. Permasalahannya software tersebut tidak handal (lambat dan sering hang) dan tidak aman karena banyaknya celah keamanan pada sistem operasi Windows seperti Denial-of-Service dan Trojan. Penggunaan Internet yang tidak terkontrol pada suatu perusahaan dapat menyebabkan efek yang kontraproduktiv pada karyawan, oleh karena itu diperlukan manajemen akses Internet yang efektiv, efisien dan mudah digunakan.

Pada makalah ini dikembangkan solusi Internet-Sharing dan Manajemen Koneksi Internet yang handal, aman, fleksibel dan mudah digunakan. Untuk Internet-Sharing menggunakan fasilitas Masquerading yang terdapat pada kernel Linux 2.2 yang terbukti handal dan aman. Sistem manajemen koneksi dan konfigurasi didesain berbasis Web agar mudah digunakan oleh network administrator dan karyawan. Sistem ini juga cocok diaplikasikan pada Internet Café/Warung Internet.

Kata kunci : Internet, SOHO, IP, dial-up.

1. PENDAHULUAN

1.1. Latar Belakang

Internet pada dasarnya adalah kumpulan sejumlah besar komputer dan router yang saling terkoneksi secara hirarkis tersebar diseluruh dunia. Standar komunikasi yang digunakan diInternet adalah Internet Protocol (IP), yang menspesifikasikan bahwa setiap node yang terhubung ke Internet diidentifikasi berdasarkan pengalamatan IP address. IP Address adalah unik, dan jumlahnya sangat terbatas oleh karena itu penggunaannya diatur dan dibatasi bagi institusi dan Internet Service Provider dengan sistem sewa. Terdapat berbagai macam alternativ infrastruktur koneksi ke Internet, cara yang paling praktis dan ekonomis adalah dengan sistem Dial-up modem melalui jaringan telepon PSTN. Pada sistem Dial-up, komputer client secara otomatis akan mendapatkan IP Address Internet selama tetap terkoneksi, IP address ini pada umumnya bersifat dinamis, bukan IP address static.

Pada lingkungan perusahaan *SOHO* (Small Office Home Office) yakni perusahaan skala kecil yang memiliki kantor kecil dan biasanya menggunakan rumah sebagai kantor; pada umumnya telah mempunyai Intranet dengan jumlah komputer kurang dari sepuluh. Seringkali terdapat beberapa orang yang pada saat yang bersamaan membutuhkan untuk mengakses Internet, permasalahannya apakah setiap orang/komputer harus melakukan Dial-up tersendiri ? Tidak, terdapat 2 alternativ metode *Internet-Sharing* yakni Proxy dan Network-Address-Translation (NAT) atau biasa disebut Masquerading. Perbedaannya: Proxy bekerja pada Application layer sedangkan Masquerading bekerja pada Network layer. Saat ini telah beredar beberapa software *Internet-Sharing* yang menggunakan sistem Proxy ataupun Masquerading, dan sebagian besar berbasis sistem operasi Windows seperti: WinGate™, WinProxy™. Namun masih terdapat banyak keluhan bahwa kinerja software tersebut tidak stabil, sering hang, lambat, membutuhkan persyaratan hardware yang tinggi dan yang utama tidak aman. Sistem operasi Windows secara default rentan terhadap

berbagai serangan *Denial-of-Service* dan Trojan dari hacker di Internet yang dapat menyebabkan terganggunya koneksi Internet bahkan hilangnya data-data pada Intranet. Selain itu penggunaan Windows membutuhkan hardware yang berat yang otomatis membuat Total Cost of Ownership dari sistem *Internet-Sharing* tersebut menjadi mahal.

Software *Internet-Sharing* yang ada saat ini masih bersifat Admin-centric, yakni didesain untuk dioperasikan oleh Network Administrator, bukan oleh end-user. Semua fungsional mulai konfigurasi hingga koneksi ke Internet harus dilakukan oleh Administrator pada komputer gateway. Peran Administrator yang terlalu dominan menyebabkan beban tugas Administrator lebih berat, Administrator harus siap kapanpun diperlukan untuk koneksi ke Internet.

Penggunaan Internet pada perusahaan dapat memberikan banyak manfaat, namun sebaliknya jika penggunaannya tidak terkontrol dapat menyebabkan turunnya produktivitas kerja. Penggunaan Internet pada jam-jam produktif untuk hal-hal yang kurang penting seperti Internet Relay Chat (IRC), Internet Messenger (ICQ, Yahoo, MSN), Napster, Game menyebabkan menurunkan kinerja karyawan bahkan merusak budaya kerja.

1.2. Permasalahan

Permasalahan yang ingin diatasi dalam pengembangan perangkat lunak ini adalah :

1. Bagaimana merancang dan membuat sistem *Internet-Sharing* yang aman (terhadap DoS, Trojan), handal (stabil, tidak hang), tangguh (tidak bisa dimanupulasi oleh user) dan mudah digunakan.
2. Bagaimana merancang dan membuat sistem manajemen koneksi Internet yang mempunyai kontrol akses yang fleksibel yang dapat menkontrol penggunaan Internet agar lebih efektif dan efisien waktu dan biaya.
3. Bagaimana merancang dan membuat sistem manajemen koneksi Internet yang accountable, dapat menampilkan laporan penggunaan Internet oleh pengguna dan laporan koneksi Dial-up ke Internet Service Provider (ISP).
4. Bagaimana merancang dan membuat sistem manajemen koneksi Internet yang user-centric dan multiplatform.

1.3. Tujuan dan Manfaat

Tujuan dari penelitian ini adalah mengembangankan sistem *Manajemen Koneksi Internet* yang aman, handal, tangguh, fleksibel dan mudah digunakan.
Manfaat dari perangkat lunak ini antara lain ialah :

1. Seluruh komputer dalam Intranet dapat mengakses Internet secara bersamaan dengan hanya menggunakan 1 koneksi Dial-up Modem. Sehingga lebih efisien dalam penggunaan Internet dan hemat biaya.
2. Melindungi komputer gateway dan Intranet dari serangan *Denial-of-Service* dan Trojan.
3. Menyediakan mekanisme *Internet-Sharing* yang handal (stabil, tidak hang), tangguh (tidak bisa dimanipulasi oleh user) dan cepat.
4. Meringankan tugas Network Administrator untuk melakukan Dial-up rutin, setting Firewall, mengatur kontrol akses, membuat Laporan Dial-up.
5. Menyediakan sistem Kontrol Akses Internet user, Dial-up account yang mudah dikonfigurasikan dan dilengkapi laporan penggunaan Internet.
6. Memberikan laporan mendetail tentang Dial-up sehingga dapat dijadikan Cross check dengan tagihan/billing dari Internet Service Provider, dan juga sebagai ukuran untuk mengkontrol pemakaian.
7. Sebagai alternativ sistem *Internet-Sharing* yang memiliki Total Cost of Ownership (TCO) yang rendah.

2. IP MASQUERADING

IP Masquerading adalah pengembangan *one-to-many* dari Network Address Translation (NAT) yang fungsinya untuk membungkus paket data dari komputer client sehingga seakan-akan nampak bahwa paket data tersebut berasal dari komputer yang menyediakan fungsi Masquerading [2]. Aplikasi yang umum dari IP Masquerading ini adalah untuk menyediakan fungsi Internet-Sharing pada lingkungan Intranet. Komputer-komputer dalam Intranet yang tidak mempunyai IP address Internet dapat berkomunikasi ke Internet melalui server Linux yang mempunyai 1 IP address Internet (terhubung ke Internet) dan meyediakan layanan IP Masquerading. Komputer-komputer dalam Intranet dapat terhubung ke server Linux dengan topologi LAN seperti Ethernet, TokenRing, FDDI, selain alternativ koneksi yang lain seperti Dial-up PPP atau SLIP, server Linux berfungsi sebagai default gateway bagi komputer-komputer dalam Intranet.
Contoh:

Server Linux terkoneksi ke Internet melalui Dial-up PPP atau Leased line, dan koneksi ke Intranet dengan topologi Ethernet, maka komputer-komputer dalam Local Area Network yang terkoneksi ke server Linux melalui Ethernet juga dapat berkomunikasi ke Internet.

Fasilitas IP Masquerading memungkinkan komputer-komputer Intranet yang tidak mempunyai IP address Internet dapat mengakses Internet.

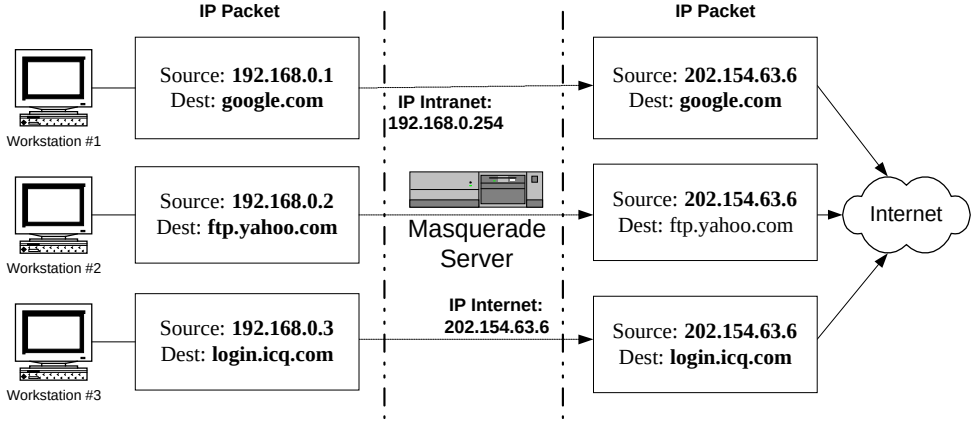
Masquerading memungkinkan komputer pada Intranet mengakses Internet secara transparan melalui gateway Masquerading. Bagi komputer/server di Internet, semua paket data nampak seperti berasal dari komputer gateway Masquerading. Selain menawarkan kemampuan Network Address Translation, IP Masquerading juga menjadi basis untuk menyediakan networking yang aman. Dengan Firewall yang dikonfigurasikan dengan benar, sangat sulit untuk menembus sistem Masquerading dan komputer-komputer Intranet didalamnya [1].

Seperti pada NAT, IP Masquerading bekerja pada Network layer dari arsitektur OSI atau Internet layer dari arsitektur TCP/IP [2]. Pada Masquerade server tidak ada port yang di-bind sebab Masquerade bekerja pada level packet IP. Karena pada tiap komputer dalam Intranet default gateway yang digunakan adalah Masquerade server, maka setiap packet IP yang akan menuju Internet akan diroutingkan ke Masquerade server. Pada Masquerade server, header packet IP akan diubah source addressnya menjadi IP address Masquerade server. Demikian pula sebaliknya apabila packet IP diterima dari Internet, Masquerade server akan merubah header tujuan packet IP agar menuju komputer client yang meminta koneksi.

Ilustrasi singkat mengenai cara kerja Internet-Sharing dengan IP Masquerade dapat dilihat pada Gambar 1.

Dalam perancangan perangkat lunak FireGate terdapat beberapa hal yang digarisbawahi yakni:

1. Teknologi *Internet-Sharing* yang digunakan
   Teknologi *Internet-Sharing* yang dipilih adalah *IP Masquerading* dengan menggunakan kernel Linux 2.2. Alasana mengapa teknologi ini dipilih karena relatif lebih handal, cepat dan transparan dibandingkan dengan menggunakan server Proxy. *IP Masquerading* bekerja pada layer Network sehingga lebih cepat dan lebih efisien dalam penggunaan resource.
2. Teknologi perlindungan Intranet terhadap serangan DoS , Probing dan Trojan
   Serangan *Denial-of-Service* yang paling mudah dan sering terjadi adalah *Nuke* yang menyerang port 139 pada Windows9x. Internet Trojan seperti Back Orifice dan NetBus akan membuka layanan tersembunyi pada port 31337, cracker dapat mengendalikan dan mencuri data pada komputer dari Internet. Probing adalah langkah-langkah enumerasi yang dilakukan cracker untuk meraba server dan komputer dalam jaringan. Semua ancaman tersebut dapat diminimalkan dengan menggunakan Packet Filtering Firewall untuk menyaring paket data yang menuju ke Intranet. Kernel Linux 2.2 memiliki fungsi Packet Filtering Firewall yang dapat dikonfigurasikan dengan menggunakan program IPCHAINS.
3. Kontrol penggunaan Internet oleh pengguna
   Perangkat lunak *Internet-Sharing* yang ada sekarang ini tidak ada yang menyediakan fasilitas untuk mengkontrol penggunaan Internet, sehingga menyebabkan kebijaksanaan koneksi ke Internet harus dilakukan secara manual oleh Administrator jaringan. Hal ini menjadikan tugas Administrator jaringan menjadi lebih berat, dan juga ketersediaan Internet menjadi bergantung sepenuhnya kepada keberadaan Administrator jaringan. Seharusnya ada suatu sistem yang dapat dengan mudah dikonfigurasikan untuk mengatur dan melayani koneksi Internet sesuai kebijaksanaan perusahaan, sehingga pengguna dapat melakukan koneksi tanpa tergantung pada



Gambar 1. Cara kerja *Internet-Sharing* dengan IP

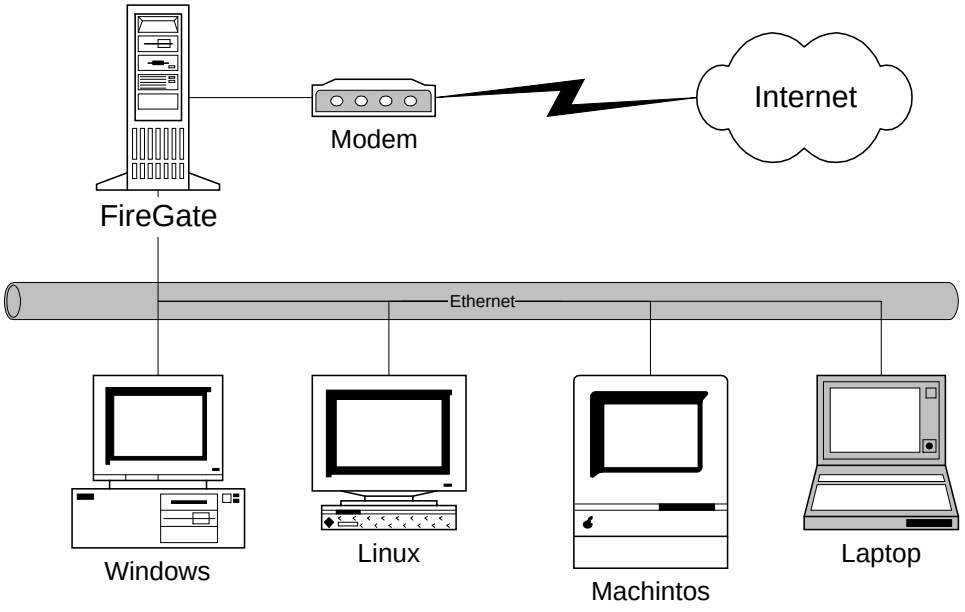
Administrator yang sedang sibuk mengerjakan tugasnya yang lain. Untuk mengkontrol penggunaan Internet oleh pengguna dalam Intranet, diterapkan pembatasan jumlah maksimal jam koneksi atau disebut Quota. Quota untuk setiap pengguna dapat disesuaikan dengan kebutuhan dan wewenangnya, sehingga pengguna tidak bisa menggunakan Internet melebihi dari quota yang telah ditetapkan. Setelah dipelajari ternyata penerapan quota semata masih cukup, bahkan dapat menjurus pada inefisiensi. Contoh kasus: Semua karyawan dalam suatu perusahaan diberikan quota 20 jam koneksi / bulan, namun apabila koneksi ke Internet dilakukan secara independen dan sporadis maka jumlah koneksi Dial-up yang terjadi justru semakin panjang. Karyawan "A" melakukan koneksi pada pukul 08:00, pada saat itu tidak ada orang lain yang ingin menggunakan Internet kecuali "A". Pada pukul 09:00 "A" mengakhiri koneksi, sesaat kemudian pada pukul 09:05 karyawan "B" melakukan koneksi sendirian hingga pukul 10:00, dan disambung karyawan "C" pada pukul 10:10 yang juga terkoneksi ke Internet sendirian hingga pukul 12:00. Hal ini menyebabkan modem terus-menerus melakukan Dial-up mulai pukul 08:00 hingga pukul 12:00 hampir tanpa putus, sehingga biaya koneksi ke ISP menjadi tinggi.

Solusi agar koneksi Internet menjadi lebih efisien adalah dengan menerapkan penjadwalan koneksi atau session. Perusahaan mendefinisikan jadwal session Internet, misalnya: Hari Senin hingga Kamis, Internet dapat diakses mulai pukul 12:00 sampai dengan 13:00, sedangkan hari Jumat dan Sabtu mulai pukul 14:00 sampai dengan 15:00. Dengan adanya pembatasan session pengaksesan Internet menjadi bersamaan, sehingga efisiensi koneksi Dial-up menjadi tinggi selain juga dapat menurunkan frekwensi dan biaya koneksi. Manfaat lainnya: mengurangi beban rutin Administrator jaringan untuk melakukan Dial-up, sehingga Administrator jaringan dapat menyelesaikan masalah lainnya yang lebih penting.

4. Laporan koneksi ISP dan koneksi pengguna
   Koneksi Dial-up yang ada sekarang ini pada umumnya tidak mempunyai fasilitas untuk menampilkan laporan yang mendetail tentang koneksi Dial-up yang terjadi. Akibatnya perusahaan kurang bisa memonitor dan mengkontrol penggunaan Internet. Selain itu laporan koneksi Dial-up juga dapat digunakan sebagai referensi silang dengan tagihan dari ISP. Bagi perusahaan/organisasi yang menerapkan tarif untuk penggunaan Internet, laporan penggunaan Internet oleh anggotanya sangat dibutuhkan, seperti pada Warung Internet, Hotel, Klub/Organisasi pengguna Internet.

3. ARSITEKTUR SISTEM

Teknologi *Internet-Sharing* dirancang



Gambar 2. Arsitektur Jaringan FireGate

menggunakan *IP Masquerading*, sehingga FireGate berperan sebagai default Gateway dalam Intranet, sekaligus berfungsi sebagai Firewall yang akan melindungi komputer didalam Intranet. Secara umum Arsitektur Jaringan Intranet yang menggunakan FireGate dapat dilihat pada Gambar 2.

Server FireGate dirancang sebagai dedicated server yang berdiri sendiri tanpa memerlukan monitor dan keyboard, cukup dilengkapi modem untuk melakukan Dial-up dan network interface card yang terhubung ke jaringan ethernet.

Agar akses Internet lebih fleksibel, dapat dilakukan dari komputer apapun: Desktop PC, Laptop dengan sistem operasi yang berbeda-beda: Windows, Linux, Machintos perangkat lunak dirancang berbasis Web.

Selain itu perawatan sistem yang berbasis web relatif lebih mudah, sehingga meringankan tugas Administrator.

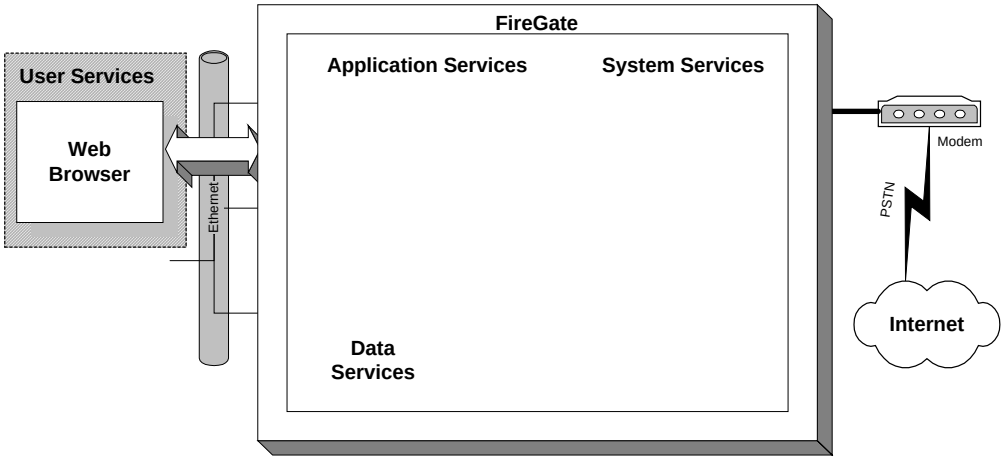
Arsitektur konseptual FireGate (Gambar 3) menjelaskan komponen internal sistem FireGate. Pengguna berinteraksi dengan sistem dengan menggunakan web browser untuk mengakses web server yang terdapat pada FireGate. Masukan dan perintah dari pengguna melalui web akan di-proses oleh program CGI Kontrol Akses dimana sebagian besar logika sistem terprogram didalamnya. Bagian Kontrol Akses akan melakukan query ke database untuk mengetahui otoritas pengguna, jumlah quota dan jadwal session. Selain itu Kontrol Akses juga bertugas melakukan konfigurasi FireWall dan mengeksekusi Dialer jika diperlukan. Bagian Dialer berfungsi melakukan koneksi ke Internet Service Provider yang menggunakan berbagai metoda otentikasi Point-to-Point Protocol yang beragam. Bagian Firewall berfungsi sebagai program bantu untuk konfigurasi Packet Filtering Firewall. Bagian Cron adalah program yang berjalan terus-menerus secara background yang bertugas untuk men-cek kestabilan koneksi, men-cek apakah quota pengguna yang sedang terkoneksi sudah habis, men-cek apakah session yang sedang berlangsung sudah habis. Jika quota seorang pengguna sudah habis maka Cron akan memutus koneksi pengguna tersebut, demikian pula jika session telah berakhir Cron akan memutus semua koneksi pengguna dan memutus koneksi ke ISP. Server DNS yang terintegrasi dengan sistem berfungsi sebagai *cache-DNS*, yang melayani permintaan layanan konversi nama domain ke IP address.

3.1. Proses Koneksi

Setelah login pengguna dapat memilih menu untuk melakukan koneksi. Urutan prosesnya, pertama akan di-check apakah saat itu terdapat session yang aktiv, kemudian di-check apakah quota pengguna masih tersisa. Jika keduanya terpenuhi masih terdapat pengecekan apakah username dan IP address yang digunakan sedang aktiv terkoneksi. Kemudian dilanjutkan dengan merubah status pengguna menjadi 'terkoneksi' dengan merubah field *Masq:=1*. Apabila ini adalah pengguna yang pertama kali melakukan koneksi proses akan dilanjutkan dengan mengeksekusi Dialer. Berikutnya dilakukan *IP Masquerading* terhadap IP address yang digunakan pengguna.

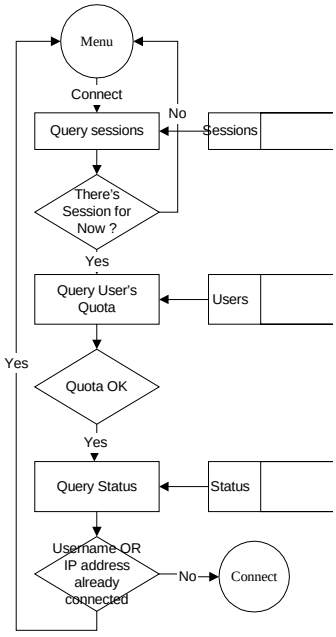
3.2. Proses Diskoneksi

Setelah selesai menggunakan Internet pengguna harus melakukan Diskoneksi, sebab kalau tidak sistem FireGate akan tetap menganggap pengguna masih terkoneksi hingga session atau quota tercapai terlebih dahulu. Proses diskoneksi diawali dengan melakukan pengecekan terhadap tabel status untuk mengetahui apakah pengguna benar-benar sedang terkoneksi. Apabila pengguna adalah satu-satunya pengguna yang sedang terkoneksi (pengguna terakhir) maka proses diskoneksi otomatis akan



Gambar 3. Arsitektur FireGate

mengakhiri koneksi Dial-up. Kemudian diikuti dengan pencabutan *IP Masquerading* untuk IP address pengguna.



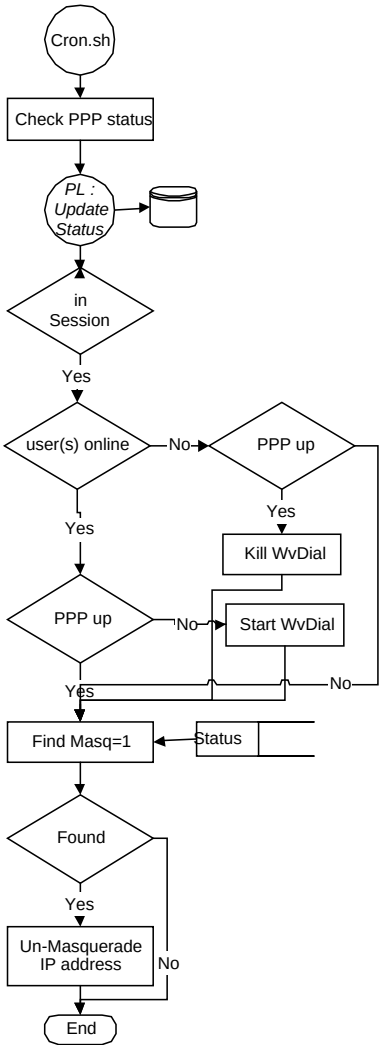
Gambar 4. Alur Proses Koneksi

3.3. Proses Crond

Proses Crond adalah proses yang dieksekusi secara transparan dan rutin dalam periode tertentu (1 menit) yang bertugas melakukan update data-data koneksi pengguna. Tujuan proses Crond ini agar data koneksi pengguna selalu valid, tidak mudah dicurangi oleh pengguna. Penjelasan proses Crond dibagi dua yakni proses cron.sh dan stored procedure *UpdateStatus.*

Urutan proses cron.sh diawali dengan pengecekan status PPP, kemudian dilanjutkan dengan mengeksekusi function *UpdateStatus* pada PostgreSQL dengan bantuan program PSQL. Pada function *UpdateStatus* , apabila session tidak aktiv atau baru saja selesai maka dilakukan diskoneksi masal terhadap pengguna yang statusnya masih terkoneksi. Apabila session sedang aktiv, proses dilanjutkan dengan melakukan update *endtime* koneksi, dan melakukan pengecekan quota pengguna yang sedang terkoneksi. Apabila terdapat pengguna yang quotanya habis maka akan dilakukan diskoneksi terhadap pengguna tersebut.

Hasil kembalian dari eksekusi function *UpdateStatus* adalah status dari session dan informasi jumlah pengguna yang aktiv. Apabila tidak ada pengguna yang aktiv namun status PPP terkoneksi maka akan dilakukan hang-up koneksi Dial-up. Sebaliknya jika terdapat pengguna aktiv namun koneksi Dial-up terputus maka akan mengeksekusi Dialer. Langkah terakhir pada cron.sh adalah melakukan query kedatabase untuk mendapatkan pengguna yang harus dihapus Masqueradingnya (Masq=1). Kemudian mengeksekusi IPCHAINS untuk mencabut Masquerading pada IP address tersebut, dan merubah field pada tabel *status* (Masq:=0).



Gambar 5. Alur Proses Cron.sh

4. KESIMPULAN

Dari uraian pengembangan perangkat lunak di atas, dapat diambil beberapa kesimpulan sebagai berikut :

1. Pada lingkungan Intranet *SOHO* dapat dilakukan pemakaian Internet secara bersama-sama dengan hanya menggunakan satu koneksi Dial-up, sehingga koneksi Dial-up dapat dimanfaatkan secara lebih efisien dan hemat biaya.

2. Packet Filtering Firewall dapat digunakan untuk melindungi Intranet yang terkoneksi ke Internet terhadap serangan *DenialOfService* dan *Trojan* oleh cracker di Internet.
3. Perangkat lunak yang diintegrasikan dengan Packet Filtering Firewall dapat digunakan untuk membatasi akses ke Internet oleh pengguna dari dalam Intranet, sehingga Internet hanya dapat digunakan oleh pengguna yang tepat.
4. Penggunaan *Quota* dan *Session* dapat membantu mengendalikan total pemakaian Internet pada lingkungan *SOHO*.
5. Internet Gateway yang *user-centric* dapat meringankan beban rutin administrator, dan memudahkan pengguna untuk melakukan koneksi ke Internet.
6. Laporan koneksi Dial-up dapat digunakan sebagai referensi silang atas tagihan dari ISP. Laporan koneksi pengguna dapat digunakan untuk evaluasi pemakaian internet pada *SOHO*, dan dapat dikembangkan menjadi sistem penagihan untuk pengguna apabila diperlukan.
7. Perangkat lunak yang dikembangkan sudah dapat digunakan untuk billing Warnet sederhana dengan sistem quota pra-bayar.

Saran

Beberapa saran untuk pengembangan selanjutnya dari perangkat lunak dijelaskan sebagai berikut :

1. Masquerading layanan kedalam Intranet.
   Internet gateway sekaligus berfungsi menyediakan Masquerading bagi layanan-layanan yang terdapat pada komputer didalam Intranet dengan cara membuka port tertentu pada server dan merelaykan data ke komputer yang menyediakan layanan didalam intranet.
2. Menggunakan Web Server dan CGI handler yang lebih ringan.
   Untuk mencapai kinerja yang tinggi dan dapat bekerja pada perangkat keras yang rendah, disarankan untuk menggunakan web server yang kecil seperti *thttpd* dengan CGI handler yang diimplementasikan dalam C/C++.
3. Layanan koneksi darurat dan koneksi otomatis.
   Pengguna dengan wewenang tertentu dapat melakukan koneksi darurat diluar jadwal session yang telah ditetapkan, sehingga tidak harus menunggu session aktiv. Dan fasilitas koneksi otomatis menurut jadwal yang ditetapkan, digunakan untuk proses pengambilan email secara otomatis oleh program didalam intranet.
4. Pengembangan untuk sistem billing Warnet.
   Perangkat lunak yang telah ada dikembangkan untuk digunakan pada Warnet dengan menambahkan kemampuan pencatatan penjualan voucer quota pra-bayar, laporan penerimaan kas dan menu untuk operator warnet.

5. DAFTAR PUSTAKA

[1]. Grennan, M. “Firewall and Proxy Server HOWTO”. Linux Documentation Project, USA. February 2000.

[2]. Ranch, D. “Linux *IP Masquerading* HOWTO”. Linux Documentation Project, USA. November 2000.

[3]. Scambray, J., McClure, S. and Kurtz, G. “Hacking Exposed”, Osborne/McGraw-Hill, California, 2001.

[4]. PostgreSQL Global Development Group, “PostgreSQL 7.1.3 Documentation”, PostgreSQL Global Development Group, 2001.