SISTEM APLIKASI SMS GATEWAY SEBAGAI MANAJEMEN KEAMANAN WEB SERVER MENGGUNAKAN IDS (INTRUSION DETECTION SYSTEM) DAN IPS (INTRUSION PREVENTION SYSTEM)

SKRIPSI

Disusun Oleh :

TRI ARDHANA SEPDIANTO
06.12.630

INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK KOMPUTER DAN INFORMATIKA
2011

LEMBAR PERSETUJUAN

SISTEM APLIKASI SMS GATEWAY SEBAGAI MANAJEMEN KEAMANAN WEB SERVER MENGGUNAKAN IDS (INTRUSION DETECTION SYSTEM) DAN IPS (INTRUSION PREVENTION SYSTEM)

SKRIPSI

Disusun dan diajukan untuk melengkapi dan memenuhi persyaratan guna mencapai gelar Sarjana Teknik

Disusun Oleh :

TRI ARDHANA SEPDIANTO
NIM : 06.12.630

Mengetahui,
Ketua Program Studi Teknik Elektro S-1

Ir. Yusuf Ismail Nakhoda, MT
NIP. Y . 1018800189

Diperiksa dan Disetujui,

Dosen Pembimbing I	Dosen Pembimbing II
Ir. Yusuf Ismail Nakhoda, MT	Sotyohadi, ST
NIP. Y . 1018800189	NIP.Y.1039700309

JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK KOMPUTER DAN INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG

SISTEM APLIKASI SMS GATEWAY SEBAGAI MANAJEMEN KEAMANAN WEB SERVER MENGGUNAKAN IDS (Intrusion Detection System) dan IPS (Intrusion Prevention System)

Tri Ardhana Sepdianto, NIM : 06.126.30
Dosen Pembimbing : Ir.Yusuf Ismail Nakhoda, MT dan Sotyohadi, ST

Abstrak

Keamanan jaringan komputer sebagai bagian dari sebuah sistem sangat penting untuk menjaga validitas dan integritas data serta menjamin ketersediaan layanan bagi penggunanya. Sistem Pendeteksian Intrusi (Intrusion Detection System / IDS) memegang peranan penting dalam pengamanan jaringan.

Snort merupakan salah satu produk Open Source yang menjadi pilihan ideal sebagai pendeteksi intrusi dalam jaringan. Namun perlu diketahui bahwa fungsi dari Snort bisa dikembangkan menjadi sebuah Sistem Pencegah penyusupan (Intrusion Prevention System / IPS), dengan bantuan Firewall (IPTables).Tugas tersebut telah ditambahkan pada program gambas yang menjalankan fungsi dari snort dan iptables yang dikombinasikan.

Dengan adanya trigger sistem otomatisasi SMS digunakan. Sistem seperti ini administrator dapat menjalankan tugasnya bila saat administrator tidak ada pada tempatnya dengan diberikan suatu informasi lebih dini serta dapat melakukan pencegahan melalui SMS.

Kata Kunci : Snort, SMS Gateway, Iptables, Gambas, MYSQL

Abstract

Computer network security as part of a system is very important to maintain the validity and integrity of data and ensure the availability of services for its users. Intrusion Detection System (Intrusion Detection System / IDS) plays an important role in network security.

Snort is one Open Source product that becomes an ideal choice as a network intrusion detection. But keep in mind that the functions of the Snort could be developed into an intrusion Prevention System (Intrusion Prevention System / IPS), with the help of Firewall (iptables). The task has been added to the squash programs that perform the function of snort and iptables are combined.

With existence of trigger automatization system is applied. Such a system is expected for administrators to out their duties as an administrator when there is no place to be given more early an information and can do prevention through SMS.

***Keywords :** Snort, SMS Gateway, Iptables, Gambas, MYSQL*

KATA PENGANTAR

Dengan mengucap syukur kehadirat Tuhan Yang Maha Esa yang dengan segala Kasih dan Anugerah–Nya, telah memberikan kekuatan, kesabaran, bimbingan dan perlindungan sehingga penulis dapat menyelesaikan laporan skripsi dengan judul:

" SISTEM APLIKASI SMS GATEWAY SEBAGAI MANAJEMEN KEAMANAN WEB SERVER MENGGUNAKAN IDS (Intrusion Detection System) dan IPS (Intrusion Prevention System)"

Pembuatan skripsi ini disusun guna memenuhi syarat akhir kelulusan pendidikan jenjang Strata I di Institut Teknologi Nasional Malang. Dalam penyusunan skripsi ini penulis banyak mendapat bantuan baik moril maupun materiil, saran dan dorongan semangat dari berbagai pihak, untuk itu penulis mengucapkan terima kasih kepada :

1. Bapak Prof. Dr. Ir. Abraham Lomi, MSEE selaku rektor ITN Malang
2. Bapak Ir. Sidik Noertjahjono, MT selaku Dekan Fakultas Teknologi Industri.
3. Bapak Ir.Yusuf Ismail Nakhoda, MT selaku Ketua Jurusan Teknik Elektro S-1 ITN Malang.
4. Bapak Ir.Yusuf Ismail Nakhoda, MT selaku Dosen Pembimbing I
5. Bapak Sotyohadi, ST selaku Dosen Pembimbing II
6. Bapak dan ibuku tercinta yang selalu memberikan semangat untuk selalu meraih cita-cita dengan ilmu dan amal soleh.
7. Teman – teman yang memberikan support dan nasihat untuk segera menyeleseikan skripsi
8. Dan semua pihak yang telah membantu penulis dalam menyelesaikan skripsi ini yang tidak bisa penulis sebutkan satu persatu.

Penulis menyadari bahwa laporan ini masih banyak yang perlu disempurnakan. Oleh sebab itu kritik dan saran yang membangun sangat diharapkan. Akhir kata, penulis mohon maaf kepada semua pihak bilamana selama penyusunan skripsi ini penyusun membuat kesalahan secara tidak sengaja dan semoga skripsi ini dapat bermanfaat bagi kita semua.

Malang, Februari 2011

Penulis

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

BAB I
PENDAHULUAN

1.1 Latar Belakang

Dengan berkembangnya teknologi informasi, hampir seluruh informasi yang penting bagi suatu institusi dapat diakses oleh penggunanya. Keterbukaan akses tersebut memunculkan berbagai masalah baru,. Di satu sisi mempermudah pekerjaan manusia menjadi lebih ringan, tetapi di sisi lain jumlah insiden keamanan sistem informasi meningkat cepat sehingga pada hakekatnya sisi keamanan jaringan berada pada posisi yang sangat terancam. Keamanan jaringan komputer adalah bagaian dari sistem yang sangat penting untuk menjaga validitas dan integritas data serta menjamin ketersediaan layanan bagi penggunanya. Sistem harus dilindungi dari segala macam serangan dan usaha-usaha penyusupan atau pemindaian oleh pihak yang tidak berhak.

Sistem pertahanan terhadap aktivitas gangguan yang ada saat ini umumnya dilakukan secara manual oleh administrator. Hal ini mengakibatkan integritas sistem bergantung pada ketersediaan dan kecepatan administrator dalam merespon gangguan yang terjadi. Apabila gangguan tersebut telah berhasil membuat jaringan mengalami malfungsi, administrator tidak dapat lagi mengakses sistem secara remote. Sehingga administrator tidak dapat melakukan pemulihan sistem dengan cepat.

Serta sistem deteksi penyusupan jaringan yang ada saat ini umumnya mampu mendeteksi berbagai jenis serangan tetapi tidak mampu mengambil tindakan lebih lanjut. Selain itu sistem yang ada pada saat ini tidak memiliki interaktivitas dengan administrator pada saat administrator tidak sedang mengadministrasi sistemnya. Hal ini merupakan suatu hal yang tidak efektif terutama pada saat sistem berada dalam kondisi kritis

Karena itu dibutuhkan suatu sistem yang dapat menanggulangi ancaman-ancaman yang mungkin terjadi secara optimal dengan diberikan suatu informasi secara dini saat ada intrusion dan memungkinkan administrator mengakses sistem. Hal ini akan mempercepat proses penanggulangan gangguan serta pemulihan sistem atau layanan.

.Pada sistem ini akan didesain dan diimplementasikan suatu sistem deteksi penyusupan jaringan yang memiliki kemampuan untuk mendeteksi adanya aktivitas jaringan yang mencurigakan, melakukan tindakan penanggulangan serangan lebih lanjut, serta mampu berinteraksi dengan administrator menggunakan media SMS (Short Message Service).

1.2. Rumusan Masalah

Berdasarkan latar belakang di atas maka timbul suatu permasalahan bagaimana permasalahan yang ditangani :

1. Sistem keamanan jaringan secara manual mengandung resiko keterlambatan respon terhadap intrusi jaringan.
2. Memberikan suatu peringatan lebih dini kepada administrator bahwa ada tindakan yang berusaha masuk ke dalam sistem dalam bentuk SMS.
3. Administrator dapat melakukan suatu tindakan pencegahan melalui SMS jika admin tidak ada pada tempatnya.

1.3. Tujuan

Tujuan Skripsi ini adalah membangun sebuah sistem pelaporan intrusi yang masuk pada sebuah server sebagai alert kepada Administrator agar diberi suatu peringatan dan melakukan tindakan penanggulangan serangan lebih lanjut.

1.4. Batasan Masalah

Batasan masalah yang diambil pada penulisan skripsi ini diharapkan mampu membatasi pembahasan agar sesuai dengan tujuan penelitian itu sendiri. Adapun batasan masalah yang diajukan adalah sebagai berikut :

1. Mendeteksi adanya gangguan jaringan, dan setelah terdeteksi kemudian sistem akan otomatis mengirimkan SMS (Short Messaging Service) kepada administrator.
2. Tidak membahas masalah biaya yang dikeluarkan dalam mengirim dan menerima sms.

3. Operasi sistem yang digunakan adalah Linux Ubuntu 10.04
4. Mendeteksi serangan dan Memblock paket/data yang lewat berdasarkan Rule.
5. Menjelaskan tipe serangan berdasarkan rule yang ada pada snort.
6. Menggunakan serangan berupa ping of death,Login FTP,telnet.
7. Serangan yang digunakan menggunakan linux / Ubuntu.
8. Tidak membahas handphone dengan komputer secara detail.

1.5. Metodologi

Metodologi yang diterapkan dalam pengerjaan Tugas Akhir ini adalah sebagai berikut:

1. Studi literatur

   Pengumpulan data yang dilakukan dengan mencari bahan-bahan kepustakaan dan referensi dari berbagai sumber sebagai landasan teori yang ada hubunganya dengan permasalahan untuk meningkatkan pemahaman mengenai topik Tugas Skripsi yang dikerjakan.

2. Analisa Kebutuhan Sistem

   Data dan informasi yang telah diperoleh akan dianalisa agar didapatkan kerangka global yang bertujuan untuk mendefinisikan kebutuhan sistem baik hardware maupun software, di mana nantinya akan digunakan sebagai acuan perancangan sistem

3. Perancangan sistem

   Berdasarkan data dan informasi yang telah diperoleh serta analisa kebutuhan untuk membangun sistem ini, akan dibuat rancangan kerangka global yang menggambarkan mekanisme dari sistem yang akan dibuat.

4. Implementasi Sistem

   Tahapan ini semua rancangan yang sudah dihasilkan deterjemahkan dengan menggunakan bahasa pemrograman Gambas dan database mysql.

5. Eksperimen dan Evaluasi

   Pada tahap ini, sistem yang telah selesai dibuat akan diuji coba, yaitu pengujian sistem dilakukan untuk menguji dan mengetahui apakah system berjalan dengan baik dan benar sesuai dengan persyaratan.

1.6. Sistematika Pembahasan

Penyusunan laporan Tugas Akhir ini menggunakan kerangka pembahasan yang terbentuk dalam susunan bab, yang dapat dijelaskan sebagai berikut :

Bab I : PENDAHULUAN

Bab ini merupakan dasar penyusunan laporan Skripsi yang di dalamnya berisi tentang latar belakang masalah, rumusan masalah, tujuan Skripsi, batasan masalah, metodologi pengembangan sistem, dan sistematika penulisan.

Bab II : DASAR TEORI

Bab Dasar Teori memuat berbagai pengetahuan dasar yang diperoleh dari literatur. Pengetahuan dasar tersebut berkaitan dengan penyelesaian persoalan pada skripsi ini.

BAB III : PERANCANGAN DAN PEMBUATAN SISTEM

Bab ini berisi tentang tahap analisis yaitu identifikasi dan analisis masalah dan analisis kebutuhan sistem untuk menyelesaikan masalah yang dihadapi berdasarkan teori pada Bab II dan Bab III. Bab ini juga berisi hasil perancangan yaitu proses kelanjutan dari tahap analisis meliputi proses akusisi pengetahuan.

BAB IV : IMPLEMENTASI DAN PENGUJIAN

Bab ini berisi uraian mengenai lingkungan implementasi, batasan implementasi, metode implementasi, tahapan implementasi, dan proses pengujian terhadap hasil implementasi

BAB V : PENUTUP

Bab ini membahas mengenai kesimpulan yang dapat diambil dari pelaksanaan dan saran yang dapat digunakan untuk pengembangan skripsi lebih lanjut.

BAB II
DASAR TEORI

2.1 Network Security Secara Umum

Memiliki suatu kebijakan keamanan jaringan merupakan langkah awal yang paling penting untuk melindungi dan mengamankan jaringan. Kebijakan – kebijakan ini merupakan dasar untuk menentukan keamanan jaringan komputer dari masalah yang tidak di inginkan. Pada dasarnya security adalah sistem yang digunakan untuk melindungi sistem dalam suatu jaringan keamanan agar tetap terjaga.

Host atau komputer yang terhubung ke network, mempunyai ancaman keamanan lebih besar daripada host yang tidak terhubung kemana-mana. Dengan mengendalikan network security, risiko tersebut dapat dikurangi. Namun network security biasanya bertentangan dengan network access, yaitu bila network access semakin mudah, maka network security makin rawan, dan bila network security makin baik, network access makin tidak nyaman. Suatu network didesain sebagai komunikasi data highway dengan tujuan meningkatkan akses ke sistem komputer, sementara security didesain untuk mengontrol akses. Penyediaan network security adalah sebagai aksi penyeimbang antara open access dengan security.

Disini network dikatakan sebagai highway, karena menyediakan akses yang sama untuk semua, baik pengguna normal ataupun tamu yang tidak diundang. Sebagai analogi, keamanan di rumah dilakukan dengan cara memberi kunci di pintu rumah, tidak dengan cara memblokir jalan di depan rumah. Hal seperti ini juga diterapkan pada network security. Keamanan dijaga untuk (setiap) host-host tertentu, tidak langsung pada network-nya.

Keamanan untuk daerah dimana orang saling mengenal, pintu biasanya dibiarkan terbuka . Sedangkan di kota besar, pintu rumah biasanya menggunakan mekanisme keamanan tambahan. Begitu pula yang dilakukan pada network. Untuk jaringan yang menghubungkan host-host yang aman dan dikenal, tingkat keamanan host bisa tidak dijaga terlalu ketat. Bila jaringan terhubung ke jaringan lain yang lebih terbuka, dan membuka peluang akses oleh host yang tidak aman atau tidak dikenal, maka tidak bisa tidak, host-host di jaringan tersebut membutuhkan pengamanan lebih. Ini bukan berarti keterbukaan hanya membawa akibat buruk,

sebab banyaknya fasilitas yang ditawarkan dengan keterbukaan jaringan ini merupakan nilai lebih yang sangat membantu kemajuan network. Jadi network security merupakan harga yang harus dibayar dari kemajuan jaringan komputer. Keamanan atau Security haruslah memiliki beberapa bagian penting di dalamnya, yaitu :

1. Availability yaitu menjaga akses untuk masuk ke dalam informasi
2. Confidentianlity yaitu menjaga informasi secara rahasia dan hanya dapat dibuka oleh yang memiliki hak resmi untuk mengaksesnya.
3. Anonymity yaitu menyembunyikan identitas dari entitas yang terlibat dalam prosesnya
4. Privacy yaitu memiliki hak dan kewajiban yang mengatur akusisim rahasia pribadi, dan informasi rahasi yang lain.
5. Identification and Authentication yaitu cara mengetahui identitas user dalam jaringan komputer.

2.2 Sumber Lubang Keamanan

Lubang keamanan (*security hole*) dapat terjadi karena beberapa hal. salah disain (*design flaw*), salah implementasi, salah konfigurasi, dan salah penggunaan.

2.2.1 Salah Desain

Lubang keamanan yang ditimbulkan oleh salah disain umumnya jarang terjadi. Akan tetapi apabila terjadi sangat sulit untuk diperbaiki. Akibat disain yang salah, maka biarpun dia diimplementasikan dengan baik, kelemahan dari sistem akan tetap ada.

Contoh sistem yang lemah disainnya adalah algoritma enkripsi ROT13 atau *Caesar cipher*, dimana karakter digeser 13 huruf atau 3 huruf. Meskipun diimplementasikan dengan *programming* yang sangat teliti, siapapun yang mengetahui algoritmanya dapat memecahkan enkripsi tersebut.

Contoh lain lubang keamanan yang dapat dikategorikan kedalam kesalahan disain adalah disain urutan nomor (*sequence numbering*) dari paket TCP/IP. Kesalahan ini dapat dieksploitasi sehingga timbul masalah yang dikenal dengan nama "IP spoofing", yaitu sebuah *host* memalsukan

diri seolah-olah menjadi *host* lain dengan membuat paket palsu setelah mengamati urutan paket dari *host* yang hendak diserang. Bahkan dengan mengamati cara mengurutkan nomor *packet* bisa dikenali sistem yang digunakan.

Mekanisme ini digunakan oleh program nmap dan queso untuk mendeteksi *operating system* (OS) dari sebuah sistem, yang disebut *fingerprinting*.

2.2.2 Implementasi Kurang Baik

Lubang keamanan yang disebabkan oleh kesalahan implementasi sering terjadi. Banyak program yang diimplementasikan secara terburu-buru sehingga kurang cermat dalam pengkodean. Akibatnya cek atau testing yang harus dilakukan menjadi tidak dilakukan. Seringkali batas (*bound*) dari sebuah *array* tidak dicek sehingga terjadi yang disebut out-of-bound array atau buffer overflow yang dapat dieksploitasi (misalnya *overwrite* ke *variable* berikutnya). Lubang keamanan yang terjadi karena masalah ini sudah sangat banyak, dan yang mengherankan terus terjadi, seolah-olah para programmer tidak belajar dari pengalaman.

2.2.3 Salah Konfigurasi

Meskipun program sudah diimplementasikan dengan baik, masih dapat terjadi lubang keamanan karena salah konfigurasi. Contoh masalah yang disebabkan oleh salah konfigurasi adalah berkas yang semestinya tidak dapat diubah oleh pemakai secara tidak sengaja menjadi "*writeable*". Apabila berkas tersebut merupakan berkas yang penting, seperti berkas yang digunakan untuk menyimpan password, maka efeknya menjadi lubang keamanan. Kadangkala sebuah komputer dijual dengan konfigurasi yang sangat lemah. Ada masanya workstation Unix di perguruan tinggi didistribusikan dengan berkas /etc/aliases (berguna untuk mengarahkan e-mail), /etc/utmp (berguna untuk mencatat siapa saja yang sedang menggunakan sistem) yang dapat diubah oleh siapa saja. Contoh lain dari salah konfigurasi adalah adanya program yang secara tidak sengaja diset menjadi "setuid root" sehingga ketika dijalankan pemakai memiliki akses seperti super user (root) yang dapat melakukan apa saja.

2.2.4 Salah Menggunakan Program atau Sistem

Salah penggunaan program dapat juga mengakibatkan terjadinya lubang keamanan. Kesalahan menggunakan program yang dijalankan dengan menggunakan account root (super user) dapat berakibat fatal. Sering terjadi cerita horor dari sistem administrator baru yang teledor dalam menjalankan perintah "rm -rf" di sistem UNIX (yang menghapus berkas atau direktori beserta sub direktori di dalamnya). Akibatnya seluruh berkas di system menjadi hilang mengakibatkan Denial of Service (DoS). Apabila *system* yang digunakan ini digunakan bersama-sama, maka akibatnya dapat lebih fatal lagi. Untuk itu perlu berhati-hati dalam menjalan program, terutama apabila dilakukan dengan menggunakan account administrator seperti root tersebut.

Kesalahan yang sama juga sering terjadi di sistem yang berbasis MS-DOS. Karena sudah mengantuk, misalnya, ingin melihat daftar berkas di sebuah direktori dengan memberikan perintah "dir *.*" ternyata salah memberikan perintah menjadi "del *.*" (yang juga menghapus seluruh file di direktori tersebut). Insiden-insiden yang mungkin terjadi dalam keamanan jaringan komputer salah satunya adalah *Denial of Service* (DoS).

2.3 Linux

Linux adalah nama yang diberikan kepada sistem operasi komputer bertipe Unix. Linux merupakan salah satu contoh hasil pengembangan perangkat lunak bebas/gratis dan sumber terbuka utama. Seperti perangkat lunak bebas dan sumber terbuka lainnya pada umumnya, kode sumber Linux dapat dimodifikasi, digunakan dan didistribusikan kembali secara bebas oleh siapapun.

Para pengamat teknologi informatika beranggapan kesuksesan Linux dikarenakan Linux tidak bergantung kepada vendor (*vendor independence*), biaya operasional yang rendah, dan kompatibilitas yang tinggi dibandingkan versi UNIX tak bebas, serta faktor keamanan dan kestabilannya yang tinggi dibandingkan dengan sistem operasi lainnya seperti Microsoft Windows. Ciri-ciri ini juga menjadi bukti atas keunggulan model pengembangan perangkat lunak sumber terbuka (*open source software*).

Sistem operasi Linux yang dikenal dengan istilah distribusi Linux (*Linux distribution*) atau distro Linux umumnya sudah termasuk perangkat-perangkat lunak pendukung seperti server web, bahasa pemrograman, basisdata, tampilan desktop (*desktop environment*), dan paket aplikasi perkantoran (*office suite*) seperti OpenOffice.org, KOffice, Abiword, dan Gnumeric.Dan berikut ini adalah beberapa fakta dari hal-hal yang menguntungkan dengan menggunakan program dan file-file Linux/UNIX :

- Pengoperasian tidak memerlukan lisensi.
- Linux merupakan sistem operasi bebas dan terbuka. Sehingga dapat dikatakan, tidak terdapat biaya lisensi untuk membeli atau menggunakan Linux.
- Linux/UNIX menyediakan servis untuk membuat, memodifikasi program, proses dan file (*opensource software*).
- Linux/UNIX mendukung struktur file yang bersifat hirarki.

2.3.1 Distro Ubuntu 10.04

Di dunia distribusi linux tidak banyak distro yang mampu menarik perhatian ketika kali pertama dirilis dan mampu bertahan selama bertahun-tahun. Situs distrowatch.com, yang menjadi referensi dunia distribusi Linux, mencatat selama bertahun- tahun. Ubuntu adalah distribusi linux terpopular, dilihat dari jumlah hit per hari.

Ini membuktikan bahwa Ubuntu mampu menggeser popularitas OpenSUSE, Fedora, Mandriva, ataupun Debian. Keempatnya sudah ada sejak awal-awal pengembangan linux, OpenSUSE dan Fedora di dukung oleh perusahaan besar. Mandriva dari awal rilis nya sudah terkenal dengan kemudahan penggunaannya. Sementara Debian adalah distro merupakan distro paling lengkap dengan dukungan paket programnya yang banyak, dan telah diturunkan menjadi distrodistro popular saat ini. Kata Ubuntu berasal dari bahasa bantu di Afrika Selatan, yang dapat diartikan sebagai kemanusiaan untuk semua orang. Konsep mulia ini diterapkan pada distribusi Linux yang pengembangannya disponsori oleh Canonical Ltd yang didirikan oleh Mark Shuttleworth, seorang *entrepreneur* berkebangsaan Afrika Selatan dan Inggris. Mark sendiri adalah mantan

developer Debian. Distribusi Linux Ubuntu, yang dikembangkan berbasiskan pada Debian *unstable*, kali pertama dirilis pada 20 Oktober 2004, sebagai versi 4.10 Semenjak itu, Ubuntu dirilis relative teratur setiap lebih kurang enam bulan. Sejak laporan tugas akhir ini ditulis, ubuntu telah mencapai rilis 10.04 LTS (Lucid Lynx). Pada gambar 2.1 terlihat tampilan *Lucid Lynx* nya yang membedakan dengan rilis sebelumnya.

Gambar 2.1 Lucid Lynx

2.4 Gambas

Gambas merupakan sebuah IDE (Integrated Development Environment) yang berorientasi pada RAD (Rapid Application Development) seperti halnya microsoft visual basic. Dengan gambas kita dapat menggunakan aplikasi berbasis gui (graphical user interface) di linux. Sama seperti visual basic gambas juga merupakan bahasa pemprograman. Bahasa pemrograman adalah perintah perintah yang dimengerti oleh komputer untuk melakukan tugas-tugas tertentu. gambas memiliki perbedaan di bidang code dengan visual basic tetapi secara logis penggunaan algortmanya sama dengan visual basic.

2.5 Trigger

Trigger merupakan sekumpulan perintah atau sintaks yang akan secara otomatis dijalankan jika terjadi operasi tertentu dalam tabel atau view. Trigger digunakan untuk memanggil satu atau beberapa perintah SQL secara otomatis

sebelum atau sesudah terjadi proses INSERT, UPDATE atau DELETE dari suatu tabel. Trigger sering digunakan, antara lain untuk:

- Melakukan update data otomatis jika terjadi perubahan.
- Trigger dapat digunakan untuk mengimplementasikan suatu sistem log. Setiap terjadi perubahan, secara otomatis akan menyimpan ke tabel log.
- Trigger dapat digunakan untuk melakukan validasi dan verifikasi data sebelum data tersebut disimpan.

2.6 Database MySQL

MySQL merupakan Relational Database Management Sistem (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (General Public License).Di mana setiaporang bebas untuk menggunakan MySQL, namun tidak boleh dijadikan produk turunan yang bersifat closed source atau komersial.MySQL sebenarnya merupakan turunan salah satu konsep utama dalam database sejak lama, yaitu SQL (Structure Query Language).

SQL adalah sebuah konsep pengoperasian database, terutama untuk pemilihan/seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.Keandalan suatu system database (DBMS) dapat diketahui dari cara kerja optimizer-nya dalam melakukan proses perintah – perintah SQL, yang dibuat oleh user maupun program-program aplikasinya.Sebagai database server, MySQL dapat dikatakan lebih unggul dibandingkan dengan database server yang lainnya dalam query data.

2.7 SMS (Short Messaging Service)

2.7.1 Short Messaging Service

Short Messaging Service yang lebih dikenal dengan SMS, adalah sebuah teknologi yang memungkinkan untuk menerima maupun mengirim pesan antar telepon bergerak (ponsel). Teknologi baru ini pertama kali diperkenalkan pada tahun 1992 di Eropa oleh ETSI (*Europan Telecommunications Standards Institute*), dan pada awalnya menjadi suatu standar untuk telepon *wireless* yang berbasis GSM (*Global System for Mobile Communication*). Namun, teknologi lain seperti CDMA dan TDMA pun memasukkan SMS ini sebagai fitur standar mereka.

Sebagai mana namanya, SMS yang berarti layanan pesan pendek, maka besar data yang dapat ditampung oleh SMS ini sangatlah terbatas. Untuk satu SMS yang dikirimkan, hanya dapat menampung paling banyak sebesar 140 bites atau sekitar 1120 bites. Bila diubah kedalam bentuk karakter, maka untuk satu SMS hanya dapat berisi paling banyak 160 karakter untuk karakter latin, dan 70 karakter untuk non-latin seperti karakter CIna maupun Jepang. Berikut ini adalah jenis data yang dapat dibawa oleh SMS :

- Pesan *text*
- Gambar, dapat berupa *wallpaper*, logo operator, maupun animasi
- *Ringtone* ponsel
- *Bussiness Card* deperti *VCards*.
- Konfigurasi *WAP*

Beberapa alasan menggunakan SMS atau alasan yang menjadikan SMS begitu popular saat ini :

- SMS dapat dibaca maupun dikirimkan, kapanpun dan dimanapun anda berada selama ada jaringan
- Anda dapat mengirimkan SMS ke nomor tujuan, meskipun nomor ponsel tujuan anda tidak aktif. Hal ini dikarenakan SMS memiliki masa tunggu. Jadi selama masa tunggu SMS tersebut belum habis, SMS akan terus dikirmkan ke nomor tujuan meskipun terlambat.
- SMS dianggap lebih murah dan praktis, disbanding berbicara langsung melalui telepon.

2.7.2 Gammu sebagai SMS Gateway

Gateway dalam bahasa inggris berarti pintu gerbang. Namun pada dunia komputer, *gateway* dapat berarti juga sebagai jembatan penghubung antar satu system dengan system lain yang berbeda, schingga dapat terjadi suatu pertukaran data antar system tersebut. Dengan demikian, SMS *gateway* dapat diartikan sebagai penghubung untuk lalu lintas data-data SMS, baik yang dikirmkan maupun yang diterima.

Gammu merupakan salah satu pustaka atau library opensource yang dibuat sebagai *gateway* antara handphone dengan perangkat komputer. Pengembangan Gammu awalnya dari pendahulunya yaitu gnooki yang dari

segi konsep masih sangat sederhana dan rumit. Namun pada gammu proses instalasi dan penerapan pembangunan aplikasi sms semakin mudah. Ini merupakan jasa dari beberapa developer yang telah membangun gammu. Gammu dibuat menggunakan gabungan phyton dan C. ada juga versi lain yang telah dikemas menjadi aplikasi jadi bernama *wammu* yang dibangun menggunakan bahasa C++. Penulis tidak membahas gammu secara mendetil, mengenai kode-kode pembangunnya, melainkan disini penulis hanya menulis penerapannya saja. Gammu dapat di gunakan di semua Operating System. Baik itu linux, unix, windows. Namun setiap OS memiliki proses instalasi yang berbeda-beda.

2.8 Sistem Pendeteksi dan Pencegahaan Adanya Serangan

2.8.1 Sistem Pendeteksi Adanya Serangan

Sistem pemantau (*monitoring system*) digunakan untuk mengetahui adanya tamu tak diundang (intruder) atau adanya serangan (*attack*). Nama lain dari sistem ini adalah "intruder detection system" (IDS). Sistem ini dapat memberitahu administrator melalui e-mail maupun melalui mekanisme lain seperti melalui HP.

IDS (Intrusion Detection System) merupakan sistem untuk mendeteksi adanya "intrusion" yang dilakukan oleh "intruder" atau "pengganggu atau penyusup" di jaringan. IDS (Intrusion Detection System) sangat mirip seperti alarm, yaitu IDS (Instrusion Detection System) akan memperingati bila terjadinya atau adanya penyusupan pada jaringan. IDS (Intrusion Detection System) dapat didefinisikan sebagai kegiatan yang bersifat anomaly, incorrect, inappropriate yang terjadi di jaringan atau host. IDS (Intrusion Detection System) adalah sistem keamanan yang bekerja bersama Firewall untuk mengatasi Intrusion.

IDS juga memiliki cara kerja dalam menganalisa apakah paket data yang dianggap sebagai intrusion oleh intruser. Cara kerja IDS dibagi menjadi dua, yaitu:

- Knowledge Based (Misuse Detection)

  Knowledge Based pada IDS (Intrusion Detection System) adalah cara kerja IDS(Intrusion Detection System) dengan mengenali adanya

penyusupan dengan cara menyadap paket data kemudian membandingkannya dengan database rule pada IDS (Intrusion Detection System) tersebut. Database rule tersebut dapat berisi signature – signature paket serangan. Jika pattern atau pola paket data tersebut terdapat kesamaan dengan rule pada database rule pada IDS (Intrusion Detection System), maka paket data tersebut dianggap sebagai seranganm dan demikian juga sebaliknya, jika paket data tersebut tidak memiliki kesamaan dengan rule pada database rule pada IDS(Intrusion Detection System), maka paket data tersebut tidak akan dianggap serangan.

- Behavior Based (Anomaly Based)

  Behavior Base adalah cara kerja IDS (Intrusion Detection System) dengan mendeteksi adanya penyusupan dengan mengamati adanya kejanggalan – kejanggalan pada sistem, aatu adanya keanehan dan kejanggalan dari kondiri pada saat sistem normal, sebagai contoh : adanya penggunaan memory yang melonjak secara terus menerus atau terdapatnya koneksi secara paralel dari satu IP dalam jumlah banyak dan dalam waktu yang bersamaan. Kondisi tersebut dianggap kejanggalan yang selanjutnya oleh IDS (Intrusion Detection System) Anomaly Based ini dianggap sebagai serangan.

Intrusion itu sendiri didefinisikan sebagai kegiatan yang bersifat anomaly, incorrect, inappropite yang terjadi di jaringan atau di host tersebut. Intrusion tersebut kemudian akan diubah menjadi "rules" ke dalam IDS (Intrusion Detection System). Sebagai contoh, intrusion atau gangguan seperti port scanning yang dilakukan oleh intruder. Oleh karena itu IDS (Intrusion Detection System) ditujukan untuk meminimalkan kerugian yang dapat ditimbulkan dari intrusion.

Seperti dijelaskan, IDS(Intrusion Detection System) melakukan deteksi gangguan keamanan dengan melihat Anomali pada jaringan. Anomali dapat dijelaskan sebagai traffic atau aktifitas yang tidak sesuai dengan kebijakan yang dibuat (policy).

Contoh Anomali yang dijelaskan sebagai Traffic / aktivitas yang tidak sesuai dengan policy :

- Akses dari atau menuji ke host yang terlarang
- Memiliki Content atau Patern terlarang (virus)
- Menjalankan program terlarang.

Ada berbagai cara untuk memantau adanya intruder. Ada yang sifatnya aktif dan pasif. IDS cara yang pasif misalnya dengan memonitor logfile. Contoh *software* IDS antara lain:

1. **Autobuse**, mendeteksi *probing* dengan memonitor *logfile*
2. **Shadow** dari SANS
3. **Snort**, mendeteksi pola (*pattern*) pada paket yang lewat dan mengirimkan *alert* jika pola tersebut terdeteksi. Pola-pola atau rules disimpan dalam berkas yang disebut library yang dapat dikonfigurasi sesuai dengan kebutuhan.

2.8.1.1 Snort

2.8.1.1.1 Gambaran Umum

Snort yang dapat diperoleh di http://www.snort.org biasanya di sebut sebagai *Network Intrusion Detection System* (NIDS). Snort sendiri adalah *Open Source* yang tersedia di berbagai variasi Unix (termasuk Linux) dan juga Microsoft Windows.

Sebuah NIDS akan memperhatikan seluruh segmen jaringan dimana dia berada, berbeda dengan *host* based IDS yang hanya memperhatikan sebuah mesin dimana software *host* based IDS tersebut di pasang. Secara sederhana, sebuah NIDS akan mendeteksi semua serangan yang dapat melalui jaringan komputer (Internet maupun IntraNet).

Snort IDS merupakan IDS open source yang secara defacto menjadi standar IDS (Intrusion Detection System) di industri. Snort merupakan salah satu software untuk mendeteksi instruksi

pada system, mampu menganalisa secara real-time traffic dan logging IP, mampu menganalisa port dan mendeteksi segala macam intrusion atau serangan dari luar seperti buffter overflows, stealth scan, CGI attacks, SMP probes, OS fingerprinting.

2.8.1.1.2 Fungsi – fungsi Snort

1. NIDS (Network Intrusion Detection System)

   NIDS (Network Intrusion Detection System) adalah sistem pendeteksi intrusi yang menagkap paket data yang berjalan pada media jaringan (kabel atau nirkabel), dan menggabungkannya ke sebuah signature database. Bergantung pada paket apa yang masuk, signature penyusup (intruder), alert yang dihasilkan akan masuk ke database ataupun pada file log yang sudah di atur. Fungsi utama snort adalah sebagai NIDS.

2. HIDS (Host-based Intrusion Detection System)

   Host-based Intrusion Detection System dimasukkan sebagai sebuah agen pada sebuah host. Sistem-sistem pendeteksi intrusi ini dapat dilihat pada system dan aplikasi log untuk mengetahui aktifitas intruder. Sistem ini reactive, artinya bahwa sistem-sistem ini memberitahukan ketika hanya terjadi suatu intrusi.

3. Signature

   Signature adalah pola yang dapat ditemui pada sebuah paket data. Signature digunakan untuk mendeteksi satu atau bermacam-macam bentuk serangan. Contohnya kehadiran "scripts/iisadmin" dalam sebuah paket data yang akan menuju web server. Signature bias dihadirkan dalam bagian-bagian yang berbeda dari sebuah paket data, tergantung pada karakter atau sifat serangan. Contohnya pada IP header, Transport Layer header (TCP dan UDP header) dan header layer aplikasi. Biasanya IDS bergantung pada signature untuk

mengetahui aktifitas dari intruder. Beberapa vendor IDS membutuhkan update dari vendor untuk menambah signature baru dimana tipe serangan ditemukan.

4. Alert

   Alert adalah semacam pemberitahuan user dari aktifitas intruder. Ketika IDS mendeteksi adanya intruder, IDS harus memberitahukan seorang network administrator tentang hal ini dengan menggunakan alert. Alert bisa dalam bentuk logging, e-mail notification, dan sebagainya. Alert bsa disimpan dalam sebuah file ataupun database.

   Snort dapat menghasilkan alerts dalam berbagai bentuk dan dikontrol oleh outpiu plug-in. Misalnya, alert dapat ditampilkan kedalam halaman web, yang mana nantinya akan dapat di akses oleh seorang network administrator untuk kemudian dapat mengetahui serangan yang terjadi.

5. Logs

   Pesan log biasanya disimpan dalam file. Secara default, snort akan menyimpan file tersebut kedalam direktori /var/log/snort. Akan tetapi lokasi dari pesan log ini dapat diubah sesuai keinginan user. Pesan log dapat berupa file text atau binary format.

6. Sensor

   Sensor merupakan sebuah sistem deteksi intrusi yang dijalankan dan digunakan untuk "sense" sebuah jaringan. Sensor yang dimaksud disini adalah berhubungan dengan alamat IP sebuah komputer tujuan lain dimana snort dijalankan.

2.8.1.1.3 Keuntungan dan Kelebihan Snort

Keuntungan :

1. Memberikan perlindungan yang luas dalam pengamanan sistem

2. Membantu memahami apa yang terjadi dalam system

   Dukungan teknisi :

   - Melacak aktivitas pemakai dari awal sampai akhir
   - Mengenal dan melaporkan usaha-usaha modifikasi file
   - Mengetahui kelemahan konfigurasi sistem
   - Mengenali bahwa sistem telah atau potensial untuk diserang.

3. Memungkinkan operasional pengamanan sistem dilakukan oleh staf tanpa keahlian spesifik
4. Membantu penyusunan kebijakan dan prosedur pengamanan system.

Kelemahan :

1. Bukan solusi total untuk masalah keamanan system
2. Tidak bisa mengkompensasi kelemahan
3. Masih memerlukan keterlibatan manusia
4. Banyak berasumsi pada teknologi jaringan konvensional, belum bisa menangani teknologi baru (misalnya: fragmentasi paket pada jaringan ATM).

2.8.1.1.4 Kriteria Pemilihan Snort

Pada perancangan sistem keamanan ini digunakan *snort* dengan alasan *snort* mempunyai kemampuan menjadi sensor dan analyzer serta sesuai untuk diterapkan pada rancangan sistem keamanan. Kriteria Pemilihan Snort adalah :

1. Snort adalah program yang *free* dan *open source*
2. Snort dapat berjalan secara kontinu pada sistem dengan sesedikit mungkin campur tangan dari manusia
3. Snort tidak mengakibatkan overhead terhadap sistem IDS yang mengakibatkan komputer menjadi lambat dan mengakibatkan terjadinya drop paket yang seharusnya diterima oleh sistem.
4. Logging Snort bisa dikirim ke database (mysql).

2.8.2 Sistem Pencagahan Adanya Serangan

IPS (Intrusion Prevention System) merupakan jenis metode pengamanan jaringan baik software atau hardware yang dapat memonitor aktivitas yang tidak diinginkan atau intrusion dan dapat langsung bereaksi untuk mencegah aktivitas tersebut. IPS (Intrusion Prevention System) merupakan pengembangan dari dari IDS (Intrusion Detection System) .Sebagai pengembangann dari teknologi "firewall" IPS melakukan kontrol dari suatu sistem berdasarkan aplikasi konten atau pattern, tidak hanya berdasarkan ports atau IP address seperti firewall umumnya. . Intrusion Detection System Selain dapat memantau dan monitoring, IPS (Intrusion Prevention System) dapat juga mengambil kebijakan dengan memblock paket yang lewat dengan cara 'melapor' ke firewall.

Mekanisme IPS :

Formula yang umum digunakan untuk mendefinisikan IPS adalah: IPS = IDS + Firewall.

Firewall adalah sebuah sistem pengaman, jadi firewall bisa berupa apapun baik hardware maupun software. Firewall dapat digunakan untuk memfilter paket-paket dari luar dan dalam jaringan di mana ia berada. Jika pada kondisi normal semua orang dari luar jaringan anda dapat bermain-main ke komputer anda, dengan firewall semua itu dapat diatasi dengan mudah.

Firewall merupakan perangkat jaringan yang berada di dalam kategori perangkat Layer 3 (Network layer) dan Layer 4 (Transport layer) dari protocol 7 OSI layer. Seperti diketahui, layer 3 adalah layer yang mengurus masalah pengalamatan IP, dan layer 4 adalah menangani permasalahan port-port komunikasi (TCP/UDP). Pada kebanyakan firewall, filtering belum bisa dilakukan pada level data link layer atau layer 2 pada 7 OSI layer. Jadi dengan demikian, sistem pengalamatan MAC dan frame-frame data belum bisa difilter. Maka dari itu, kebanyakan firewall pada umumnya melakukan filtering dan pembatasan berdasarkan pada alamat IP dan nomor port komunikasi yang ingin dituju atau diterimanya.

Dengan lingkungan open source seperti misalnya operating system Linux, ada satu aplikasi firewall yang digunakan. Aplikasi ini juga tidak akan membuat biaya yang cukup mahal. Bahkan Anda bisa mendapatkannya gratis karena aplikasi ini pada umumnya merupakan bawaan default setiap distro Linux. Aplikasi dan system firewall di sistem open source tersebut dikenal dengan nama IPTables.

Fitur yang dimiliki IPTables :

1. Connection Tracking Capability yaitu kemampuan unutk inspeksi paket serta bekerja dengan icmp dan udp sebagaimana koneksi TCP.
2. Menyederhanakan perilaku paket-paket dalam melakukan negosiasi built in chain (input,output, dan forward).
3. Rate-Limited connection dan logging capability. Kita dapat membatasi usaha-usaha koneksi sebagai tindakan preventif serangan Syn flooding denial of services(DOS).
4. Dengan kendali yang dimiliki terhadap semua paket data yang masuk kedalam kernel, maka keamanan system juga akan dapat terjaga. Kita dapat memilih paket-paket yang bias masuk dan tidak bias masuk ke dalam kernel kita.

Ada beberapa metode IPS (Intrusion Prevention System) melakukan kebijakan apakah paket data yang lewat layak masuk atau keluar dalam jaringan tersebut.

- Signature-based Intrusion Detection System

  Pada metode ini, telah tersedia daftar signature yang dapat digunakan untuk menilai apakah paket yang dikirimkan berbahaya atau tidak. Sebuah paket data akan dibandingkan dengan daftar yang sudah ada. Metode ini akan melindungi sistem dari jenis-jenis serangan yang sudah diketahui sebelumnya. Oleh karena itu, untuk tetap menjaga keamanan sistem jaringan komputer, data signature yang ada harus tetap ter-update.

- Anomaly-based Intrusion Detection System

  Pada metode ini, terlebih dahulu harus melakukan konfigurasi terhadap IDS (Intrusion Detection System) dan IPS (Intrusion Prevention System), sehingga IDS (Intrusion Detection System) dan IPS (Intrusion Prevention System) dapat mengatahui pola paket seperti apa saja yang akan ada pada sebuah sistem jaringan komputer. Sebuah paket anomali adalah paket yang tidak sesuai dengan kebiasaan jaringan komputer tersebut. Apabila IDS (Intrusion Detection System) dan IPS (Intrusion Prevention System) menemukan ada anomali pada paket yang diterima atau dikirimkan, maka IDS (Intrusion Detection System) dan IPS(Intrusion Prevention System) akan memberikan peringatan pada pengelola jaringan (IDS) atau akan menolak paket tersebut untuk diteruskan (IPS). Untuk metode ini, pengelola jaringan harus terus-menerus memberi tahu IDS (Intrusion Detection System) dan IPS (Intrusion Prevention System) bagaimana lalu lintas data yang normal pada sistem jaringan komputer tersebut, untuk menghindari adanya salah penilaian oleh IDS (Intrusion Detection System) atau IPS (Intrusion Prevention System).

BAB III

PERANCANGAN DAN PEMBUATAN SISTEM

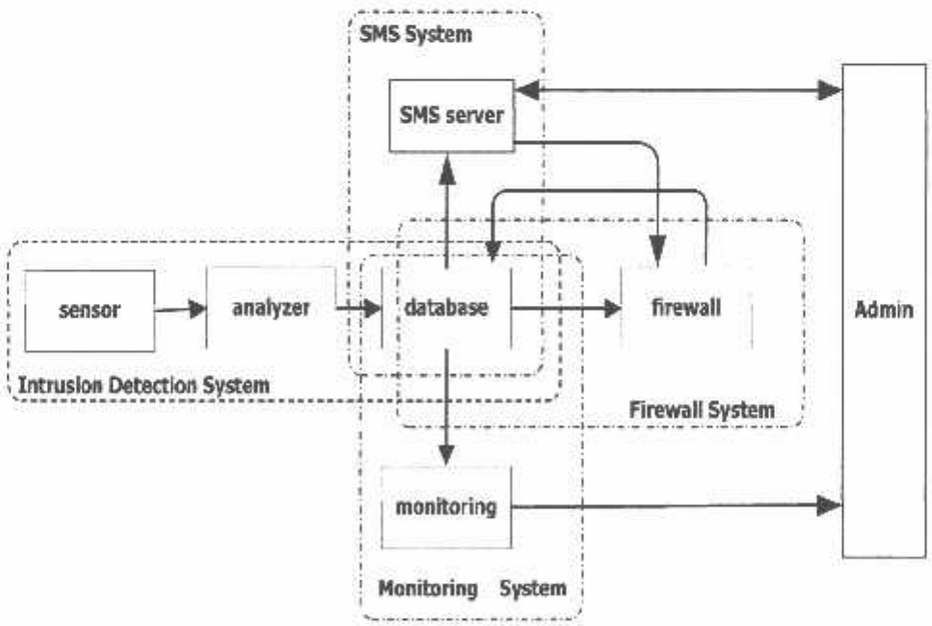
3.1 PERANCANGAN DAN PEMBUATAN SISTEM IDPS (Intrusion Detection and Prevention System)

3.1.1 Arsitektur Sistem

Sistem ini merupakan suatu metode kemanan jaringan yang bertujuan untuk membentuk suatu arsitektur sistem keamanan yang terintegrasi antara *Intrusion Detection System* (IDS) , *Firewall System*, *Database System* dan *Monitoring System*. Sistem keamanan ini bertujuan melindungi jaringan dengan kemampuan merespon sesuai dengan kebijakan keamanan.

Untuk mewujudkan metode ini perlu dirancang komponen-komponen sistem keamanan jaringan berupa :

1. Intrusion Detection System (IDS)
    a . Sensor modul
    b. Analyzer modul
2. Firewall/IPtables sistem
3. Database Sistem
4. SMS Sistem



Gambar 3.1 Arsitektur Sistem

Gambar 3.1 menunjukkan bahwa keseluruhan sistem akan dihubungkan dengan dengan *database system*, termasuk SMS *System* yang akan berjalan sesuai dengan keadaan *database system*.

Intrusion Detection and Prevention System (IDPS) pada implementasi skripsi ini tediri dari komponen-komponen :

1. Sensor

   Berfungsi untuk mengambil data dari jaringan. Sensor merupakan bagian dari sistem deteksi dini dari sistem keamanan yang dirancang dan berfungsi untuk mendeteksi nomor IP. Untuk itu digunakan suatu program yang berfungsi sebagai *intrusion detector* dengan kemampuan *packet logging*.

2. Analyzer

   Berfungsi untuk analisa paket yang lewat pada jaringan. Informasi dari *analyzer* yang akan menjadi input bagi sistem lainnya.

3. Database System

   Merupakan tempat untuk menyimpan log atau hasil dari *monitoring* yang dilakukan pada sistem ini.

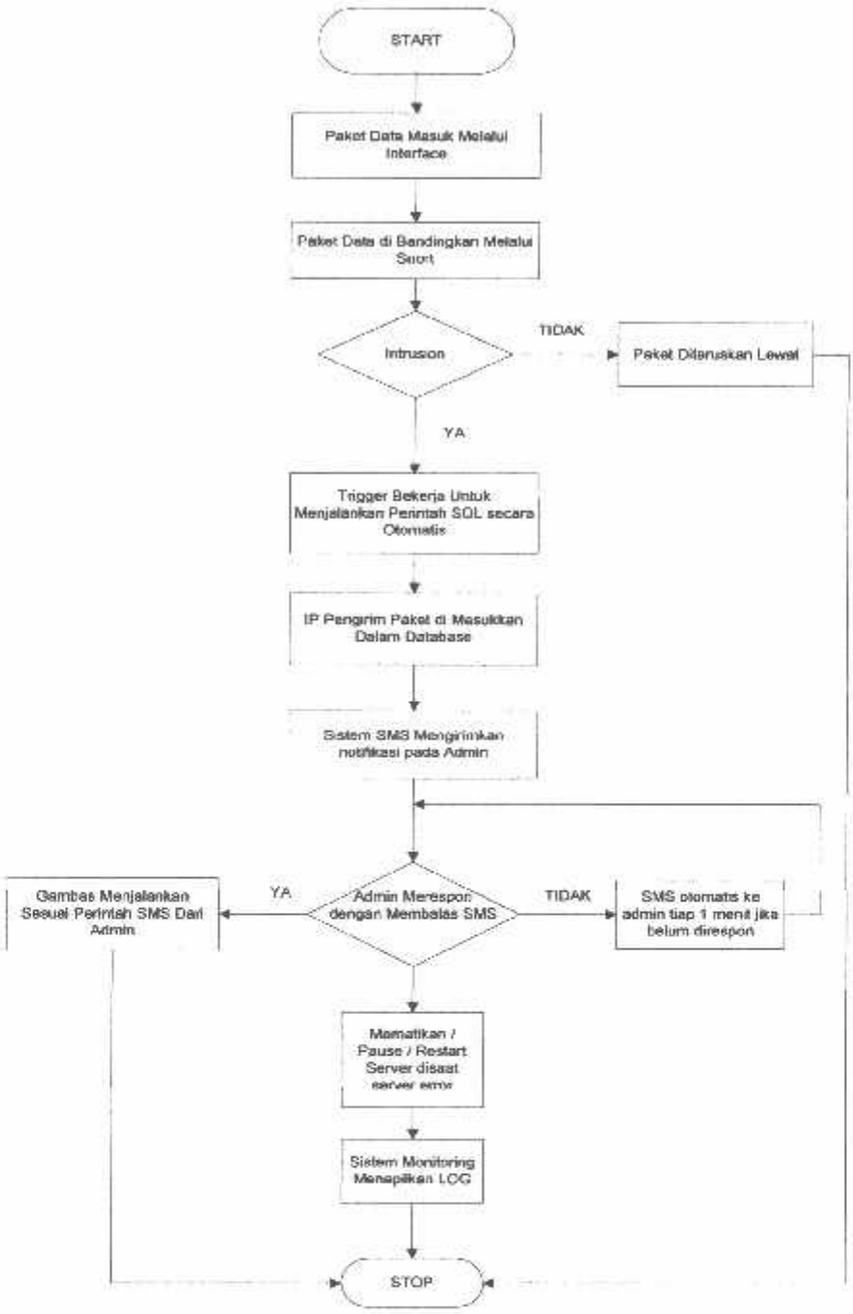
4. Firewall

   Program firewall otomatis yang dibuat pada dasarnya adalah program yang menganalisa output dari Intrusion Detection System (IDS) serta memutuskan tindakan yang harus diambil untuk host pengirim paket yang dianalisa tersebut. Apabila paket tersebut oleh IDS dikategorikan sebagai paket berbahaya atau mengandung resiko keamanan jaringan, maka program firewall akan memicu program "iptables" untuk menambahkan sebuah rule yang memblok paket yang berasal dari host paket yang mencurigakan tersebut yang akan dibolehkan atau diblock oleh admin.

5. SMS System

   Berfungsi untuk mengirimkan informasi kepada seorang *administrator* agar intrusi yang di deteksi dapat segera diketahui. Sehingga administrator bisa sesegera mungkin mengatasinya, walaupun administrator tidak ada pada tempatnya.

Flowchart proses program dan interaksi dengan user dapat dilihat sebagai berikut :



Gambar 3.2 Flowchart Proses Sistem

3.1.2 Snort

Snort merupakan suatu perangkat lunak untuk mendeteksi penyusup dan mampu menganalisis paket yang melintasi jaringan secara *real time traffic* dan *logging* kedalam database serta mampu mendeteksi berbagai serangan yang berasal dari luar jaringan. Snort tersedia gratis di internet,

www.snort.org. Snort bisa digunakan pada *platform* sistem operasi Linux, BSD, solaris, Windows dan sistem operasi lainnya. Secara *default*-nya, snort mempunyai 3 fungsi :

1. paket sniffer, seperti tcpdump, iptraf dll
2. paket logger, yang berguna untuk paket traffic dll
3. NIDS, deteksi intrusi pada network.

Untuk sistem deteksi dan analisis paket digunakan snort yang menempatkan rule-rule nya pada sebuah list (daftar) dengan metode pengolahan paket. Misalnya ada paket yang ditemukan sesuai dengan salah satu rule yang ditetapkan maka sistem akan masuk ke salah satu mode (mis:alert,log), jika tidak maka paket tersebut disesuaikan dengan rule lain yang ada pada daftar rule.

Untuk mendapatkan suatu *Intrusion Detection and prevention System* yang baik, sebaiknya instalasi *software* yang betul-betul dibutuhkan, supaya tidak memberi beban tambahan terhadap sistem, seperti :

libpcre3 libpcre3-dev libpcrecpp0 libpcap0.8 libpcap0.8-dev mysql-server libmysqlclient15-dev libphp-adodb libgd2-xpm libgd2-xpm-dev php5-mysql php5-gd php-pear apache2 php5 php5-xmlrpc php5-mysql php5-gd php5-cli php5-curl mysql-client

Setelah semua yang dibutuhkan untuk instalasi snort pada sistem operasi Ubuntu 10.04, kemudian instalasi snort itu sendiri. Snort harus terintegrasikan dengan MySQL, hal ini dikarenakan dalam perancangan sistem IDPS. log yang dicatat oleh snort akan langsung masuk kedalam database. Dengan menambahkan *option* —with –mysql, snort sudah akan terintegrasi dengan database MySQL.

Pembuatan sistem ini dilakukan untuk mendeteksi ketika terdapat suatu intrusi yang masuk kedalam sistem jaringan komputer *server*, oleh karena itu hal yang sangat perlu dilakukan adalah melakukan konfigurasi pada file /etc/snort/snort.conf. File tersebut yang menentukan apa saja

intrusi yang akan dideteksi,serta membuat folder rules didalam /etc/snort/rules. Dibawah ini adalah bagian yang penting dalam konfigurasi /etc/snort/snort.conf :

```
# variable jaringan yang satu network dengan server
var HOME_NET 192.168.1.0/24
# variable jaringan luar
var EXTERNAL_NET any
# variabel rule
var RULE_PATH /etc/snort/rules
# konfigurasi output plugin
# bentuk log adalah syslog
output alert_syslog : LOG_LOCAL7
config logdir: /usr/rofx/log config alert_with_interface_name
config checksum_mode: none config show_year
config interface: eth1
output database: log, mysql, user=snort password=1234
dbname=snort host=localhost
# path file rule
  Include $RULE_PATH/ftp.rules

  include $RULE_PATH/telnet.rules

  include $RULE_PATH/ICMP.rules
```

Menjalankan Snort dengan 3 mode, yaitu :

1. Snort dengan paket Sniffer :

```
File  Edit  View  Terminal  Help
root@IDS:~# snort -v
Running in packet dump mode

         -- Initializing Snort ++--
Initializing Output Plugins!
Verifying Preprocessor Configurations!
***
*** interface device lookup found: eth1
***

Initializing Network Interface eth1
Decoding Ethernet on interface eth1

        --== Initialization Complete ==--

   ,,_     -*> Snort! <*-
  o"  )~   Version 2.8.4.1 (Build 38)
   ''''    By Martin Roesch & The Snort Team: http://www.snort.org/team.html
           Copyright (C) 1998-2009 Sourcefire, Inc., et al.
           Using PCRE version: 7.8 2008-09-05

Not Using PCAP_FRAMES
01/08-13:08:56.060623 192.168.1.2 -> 192.168.1.1
ICMP TTL:64 TOS:0x0 ID:0 IpLen:20 DgmLen:84 DF
Type:8  Code:0  ID:53255   Seq:58  ECHO
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

01/08-13:08:56.060773 192.168.1.1 -> 192.168.1.2
ICMP TTL:64 TOS:0x0 ID:15859 IpLen:20 DgmLen:84
Type:0  Code:0  ID:53255  Seq:58  ECHO REPLY
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

01/08-13:08:57.059632 192.168.1.2 -> 192.168.1.1
ICMP TTL:64 TOS:0x0 ID:6 IpLen:20 DgmLen:84 DF
Type:8  Code:0  ID:53255   Seq:59  ECHO
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++

01/08-13:08:57.059773 192.168.1.1 -> 192.168.1.2
ICMP TTL:64 TOS:0x0 ID:35060 IpLen:20 DgmLen:84
Type:0  Code:0  ID:53255  Seq:59  ECHO REPLY
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

Gambar 3.3 Tampilan Snort paket sniffer

Gambar 3.3 menunjukkan snort berjalan dengan mode dump, yang berarti snort mendeteksi semua nomor IP dan Protokol yang lewat tanpa memasukkan log yang tersimpan kedalam database.

2. Snort dengan Paket Logger

Gambar 3.4 Tampilan snort paket logger

Snort berjalan dengan membuat file log yang ditentukan, seperti gambar 3.4 log yang masuk berisi paket – paket yang masuk.

3. NIDS, Sistem Deteksi Penyusup di jaringan

Gambar 3.5 Tampilan snort NIDS

3.2 Perancangan Database (MySQL)

3.2.1 MySQL

Sistem keamanan ini menggunakan prinsip sentralisasi database untuk menyimpan semua alert yang berasal dari sensor maupun log dari firewall.Informasi yang tersimpan pada data base ini juga merupakan input untuk pengawasan keamanan jaringan yang dilakukan oleh firewall system, monitoring system serta sistem notifikasi SMS.

Pada perancangan system ini, database diinstall menyatu dengan snort untuk Intrusion Detection System dan Intrusion Prevention System. Berikut adalah struktur database yang akan menjadi output log dari snort:

Tabel 3.1 Struktur Data

Nama Tabel	Field	Type	Null
Daemons	Start	Text	no
	Info	Text	no
Data	Sid	Int(10)	No
	cid	Int(10)	No
	data_payload	text	Yes
Detail	Detail_type	Tinyint(3)	No
	Detail_text	text	No
Encoding	Encoding_type	Tinyit(3)	No
	Encoding_text	text	No
Event	Sid	Int(10)	No
	cid	int(10)	No
	signature	int(10)	No
	timestamp	datetime	No
Hpadmin	Id	Varchar(1)	Yes
	nomer	Varchar(30)	Yes
Icmphdr	Sid	Int(10)	No
	cid	Int(10)	No
	icmp_type	Tinyint(3)	No
	icmp_code	Tinyint(3)	No
	icmp_csum	Smallint(5)	Yes

	icmp_id	Smallint(5)	Yes
	icmp_seq	Smallint(5)	Yes
Iphdr	Sid	Int(10)	No
	cid	Int(10)	No
	ip_src	Int(10)	No
	ip_dst	Int(10)	No
	ip_ver	Tinyint(3)	Yes
	ip_hlen	Tinyint(3)	Yes
	ip_tos	Tinyint(3)	Yes
	ip_len	Smallint(5)	Yes
	ip_id	Smallint(5)	Yes
	ip_flags	Tinyint(3)	Yes
	ip_off	Smallint(5)	Yes
	ip_ttl	Tinyint(3)	Yes
	ip_proto	Tinyint(3)	No
	ip_csum	Smallint(5)	Yes
Iptable	Ips	Varchar(20)	No
	status	Varchar(20)	Yes
Opt	Sid	Int(10)	No
	cid	Int(10)	No
	optid	Int(10)	No
	opt_proto	Tinyint(3)	No
	opt_code	Tinyint(3)	No
	opt_len	Smallint(6)	Yes
	opt_data	text	Yes
Outbox_multipart	Text	Text	Yes
	Coding	Enum	No
	UDH	Text	Yes
	Class	Int(11)	Yes
	TextDecoded	Varchar(160)	Yes
	ID	Int(10)	No
	SequencePosition	Int(11)	No
pbk	GroupID	Int(11)	No

	Name	Text	No
	Number	Text	No
Pbk_groups	Name	Text	No
	ID	Int(11)	No
Phones	Id	Text	No
	updatedLnDB	Timestamp	No
	insertIntoDB	Timestamp	No
	TimeOut	Timestamp	No
	Send	Enum	No
	Receive	Enum	No
	IMEI	Varchar(35)	No
	Client	Text	No
	Battery	Int(11)	No
	Signal	Signal(11)	No
	Sent	Int(11)	No
	received	Int(11)	No
reference	Ref_id	Int(10)	No
	Ref_system_id	Int(10)	No
	Ref_tag	Text	No
reference_system	Ref_system_id	Int(10)	No
	Ref_system_name	Varchar(20)	Yes
report	Sid	Int(10)	Yes
	Waktu	Datetime	Yes
	Ips	Varchar(20)	No
schema	Vseq	Int(10)	No
	Ctime	Datetime	No
sensor	Sid	Int(10)	No
	Hostname	Text	Yes
	Interface	Text	Yes
	Filter	Text	Yes
	Detail	Tinyint(4)	Yes
	Encoding	Tinyint(4)	Yes
	Last_cid	Int(10)	No

sentitems	UpdateInDB	Timestamp	No
	insertIntoDB	Timestamp	No
	SendingDateTime	Timestamp	No
	DeliveryDateTime	Timestamp	Yes
	Text	Text	No
	DestinationNumber	Varchar(20)	No
	Coding	Enum	No
	UDH	Text	No
	SMSCNumber	Varchar(20)	No
	Class	Init(11)	No
	textDecoded	Varchar(160)	No
	ID	Int(10)	No
	SenderID	Varchar(255)	No
	sequencePosition	Int(11)	No
	status	Enum	No
	StatusError	Int(11)	No
	TPMR	Int(11)	No
	RelativeValidity	Inti(11)	No
	CreatorID	text	No
serangan	id_serangan	Int(11)	No
	nama	Varchar(100)	Yes
	tanggal	Date	Yes
	ips	Varchar(20)	Yes
	status	Varchar(1)	No
signature	Sig_id	Int(10)	No
	Sig_name	Varchar(255)	No
	Sig_class_id	Int(10)	No
	Sig_priority	Int(10)	Yes
	Sig_rev	Int(10)	Yes
	Sig_sid	Int(10)	Yes
	Sig_gid	Int(10)	Yes
sigclass	Sig_class_id	Int(10)	No
	Sig_class_name	Varchar(60)	No

sig_reference	Sig_id	Int(10)	No
	Ref_seq	Int(10)	No
	Ref_id	Int(10)	No
tcphdr	Sid	Int(10)	No
	cid	Int(10)	No
	tcp_sport	Smallint(5)	No
	tcp_dport	Smallint(5)	No
	tcp_seq	Int(10)	Yes
	tcp_ack	Int(10)	Yes
	tcp_off	Tinyint(3)	Yes
	tcp_res	Tinyint(3)	Yes
	tcp_flags	Tinyint(3)	No
	tcp_win	Smallint(5)	Yes
	tcp_csum	Smallint(5)	Yes
	tcp_urp	Smallint(5)	Yes
udphdr	Sid	Int(10)	No
	cid	Int(10)	No
	udp_sport	Smallint(5)	No
	udp_dport	Smallint(5)	No
	udp_len	Smallint(5)	Yes
	udp_csum	Smallint(5)	Yes

3.3 Perancangan Sistem untuk keamanan FTP & HTTP

a. Penganalan FTP server (File Transfer Protocol)

Pengiriman file (file transfer). File Transfer Protokol (FTP) memungkinkan pengguna komputer yg satu untuk dapat mengirim ataupun menerima file ke komputer jaringan. Karena masalah keamanan data,maka FTP seringkali memerlukan nama pengguna (user name) dan password, meskipun banyak juga FTP yg dapat diakses melalui anonymous, alias tidak berpassword.

Tujuan menyerang FTP server ini rata-rata adalah untuk mendapatkan command shell ataupun untuk melakukan Denial Of

Service. Serangan Denial Of Service akhirnya dapat menyebabkan seorang user atau attacker untuk mengambil resource didalam network tanpa adanya autorisasi, sedangkan command shell dapat membuat seorang attacker mendapatkan akses ke sistem server dan file-file data yang akhirnya seorang attacker bisa membuat anonymous root-acces yang mempunyai hak penuh terhadap system bahkan network yang diserang.

b. Pengenalan HTTP server (Hypertext Transfer Protocol)

HTTP ini merupakan protocol yg digunakan dalam World Wide Web (WWW) antar komputer yg terhubung dalam jaringan di dunia ini. Untuk mengenal protocol ini jelas sangan mudah sekali dimana setiap kali anda mengetik http://... anda telah menggunakannya, dan membawa anda ke dunia internet.

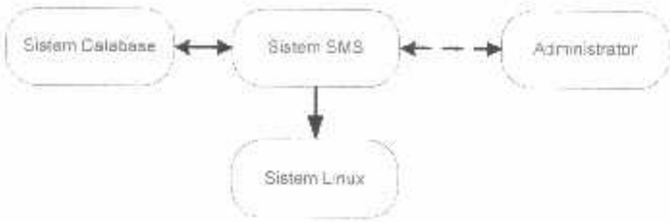
Data yg di passing dari browser ke Web server disebut sebagai HTTP request yg meminta web page dan kemudian web server akan mencari data HTML yg ada dan di kemas dalam TCP protocol dan di kirim kembali ke browser. Data yg dikirim dari server ke browser disebut sebagai HTTP response. Jika data yg diminta oleh browser tidak ditemukan oleh si Web server maka akan meninbulkan error yg sering anda lihat di web page yaitu Error : 404 Page Not Found. Web server merupakan salah satu titik eksploitasi yang paling terkenal. Dengan mengeksploitasi *bugs* yang ada di web server ini, para penyusup bisa mendapatkan hak akses setara *root*. Hal ini mengakibatkan data-data *homepage* yang ada di web server itu diacak-acak dan diganti sesuka hati penyusupnya.

Gambar 3.6 Flow chart Sistem FTP dan HTTP

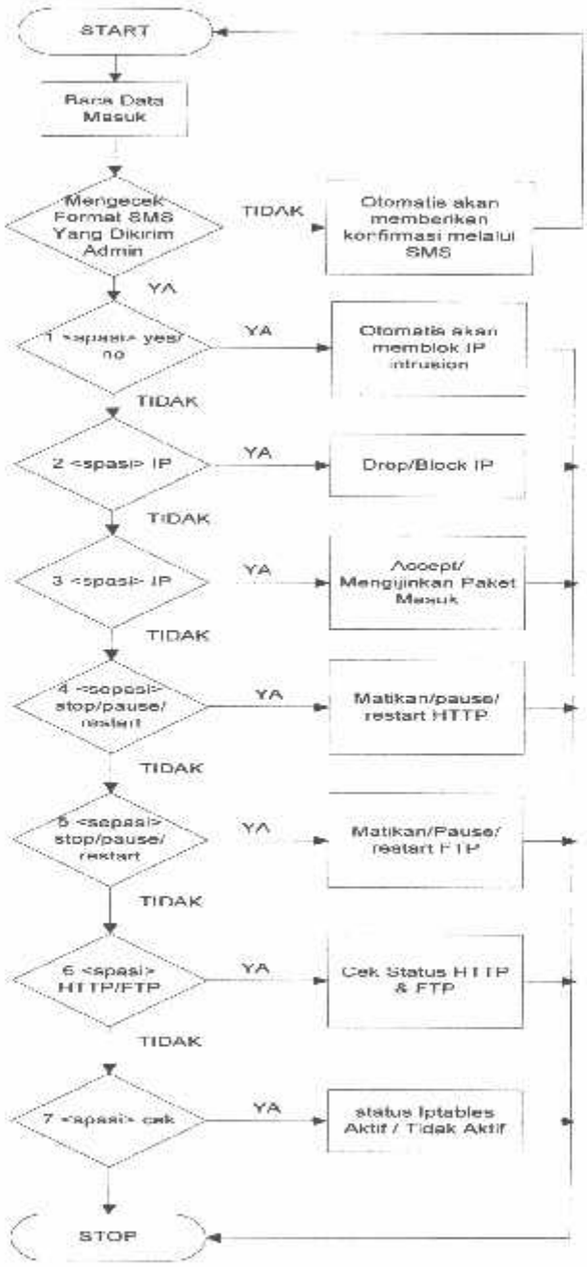
3.3 Perancangan Sistem Notifikasi SMS (Short Messaging Service)

Sitem notifikasi SMS ini dirancang sebagai bagian yang memberikan fungsi interaktif antara sistem dengan administrator. Alasan digunakannya SMS sebagai media interaktif adalah sebagai berikut :

- SMS dianggap lebih murah dan praktis, disbanding berbicara langsung melalui telepon.
- Biaya yang relatif murah
- SMS dapat dibaca maupun dikirimkan, kapanpun dan dimanapun anda berada selama ada jaringan
- Tidak tergantung pada jaringan data host



Gambar 3.7 Diagram Blok Interkoneksi Sistem Notifikasi SMS



Gambar 3.8 Flow Chart Interaksi Admin jika ada Intrusion

Fungsi dasar dari sistem SMS ini sebenarnya memberikan notifikasi atau pemberitahuan kepada administrator sesegera mungkin ketika terjadi

suatu event yang mentrigger firewall untuk memblok IP address suatu host dan di-log dalam database. Proses insertion dalam database inilah trigger yang berfungsi untuk mengirimkan pesan SMS kepada administrator. Pesan yang dikirimkan berisi tentang IP address dan jenis serangan dari host yang dilog oleh sistem.

Manfaat penerapan sistem SMS dengan interaktivitas penuh adalah dapat mengembalikan kondisi sistem apabila blocking IP address yang terjadi adalah karena kekeliruan admin saat melakukan administrasi atau testing pada host secara remote.

Pada fungsi normalnya, program sistem SMS akan mengakses database SNORT untuk mengecek kondisi tabel blocking IP address serta untuk mengambil sintaks perintah sistem yang harus dieksekusi sebagai respon admin pada suatu kondisi tertentu.

3.3.1 Konfigurasi koneksi HP dengan PC

a. Instalasi Gammu

Gammu merupakan salah satu pustaka atau library opensource yang dibuat sebagai *gateway* antara handphone dengan perangkat komputer. Pengembangan Gammu awalnya dari pendahulunya yaitu gnooki yang dari segi konsep masih sangat sederhana dan rumit. Namun pada gammu proses instalasi dan penerapan pembangunan aplikasi sms semakin mudah. Ini merupakan jasa dari beberapa developer yang telah membangun gammu. Gammu dibuat menggunakan gabungan phyton dan C. ada juga versi lain yang telah dikemas menjadi aplikasi jadi bernama *wammu* yang dibangun menggunakan bahasa C++. Penulis tidak membahas gammu secara mendetil, mengenai kode-kode pembangunnya, melainkan disini penulis hanya menulis penerapannya saja. Gammu dapat di gunakan di semua Operating System. Baik itu linux, unix, windows. Namun setiap OS memiliki proses instalasi yang berbeda-beda. Dapat anda download di :

http://www.mwiacek.com/zips/gsm/gammu/gammu.tar.gz

Pada sistem ini, koneksi antara PC dengan *Mobile Phone* dilakukan dengan menggunakan gnokii sebagai perangkat lunak yang menghubungkan, dan Sony Ericsson w200i sebagai perangkat keras

dari pembuatan SMS *Gateway*. Konfigurasi gammu berada pada folder /etc/smsdrc dan yang dilakukan adalah mengatur port dan koneksi antara PC dan *mobile phone* dengan konfigurasi sebagai berikut :

[gammu]

```
port = /dev/ttyACM0 ##port ttyACM0 sebagai driver port
USB
connection = at115200 ##menggunakan koneksi USB
```

```
root@IDS:~# gammu --identify
Manufacturer          : Sony Ericsson
Model                 : W200i/W200c (AAB-1022044-BV)
Firmware              : R4HA014  R4HA014     prgCXC1250720_CHINA_ME
IMEI                  : 355576029642261
Product code          : AAB-1022044-BV
SIM IMSI              : 510019041027149
```

Gambar 3.7 Koneksi HP dengan gammu

Berikut adalah struktur database gammu yang digunakan pada sistem :

Tabel 3.2 Struktur Database Gammu

Nama Tabel	**Field**	**Type**	**Null**
Gammu	Version	Init(11)	No
Outbox	UpdateInDB	Timestamp	No
	insertIntoDB	Timestamp	No
	SendingDateTime	Timestamp	No
	Text	Text	Yes
	DestinationNumber	Varchar(20)	No
	Coding	Enum	No
	UDH	Text	Yes
	Class	Int(11)	Yes
	textDecoded	Varchar(160)	No
	ID	Int(10)	No
	MultiPart	Enum	Yes
	RelativeValidity	Int(11)	Yes
	SenderID	Varchar(255)	Yes
	SendingTimeOut	Timestamp	Yes
	DeliveryReport	Enum	Yes

	CreatorID	text	No
Inbox	UpdateInDB	Timestamp	No
	receivingDateTime	Timestamp	No
	text	Text	No
	senderNumber	Varchar(20)	No
	coding	Enum	No
	UDH	Text	No
	SMSCNumber	Varchar(20)	No
	Class	Int(11)	No
	textDecoded	Varchar(160)	No
	ID	Int(10)	No
	recipientID	Text	No
	processed	enum	No

3.3.2 Sistem untuk Otomatisasi kirim SMS

Gammu menyediakan *daemon* (sejenis *service*) dalam paket instalasi *gammu*, *service* itu bernama *smsdrc*. *Smsdrc* akan menangani SMS yang masuk dan dapat menyimpannya kedalam *database*. Proses yang berlangsung yaitu *smsdrc* akan mengirimkan sms secara otomatis ketika tabel outbox diberi masukan. Untuk itu perlu membuat sebuah *trigger* pada database, sehingga ketika ada *event* yang masuk pada tabel snort, tabel outbox juga akan di beri masukan, schingga sms daemon akan bekerja.

Dibawah ini adalah *syntax trigger* yang dibuat :

```
DROP TRIGGER IF EXISTS 'replay';
DELIMITER //CREATE TRIGGER `replay` AFTER INSERT ON `event` FOR EACH ROW BEGIN
SELECT
nomer INTO @nomers FROM `hpadmin` WHERE ID='1' Limit 1;SELECT convert(inet_ntoa
( iphdr.ip_src ),char) INTO @ssip FROM iphdr ORDER BY `iphdr`.`cid` DESC limit 1;
if (select count(*) from iptable where ips=@ssip) = 0 then SELECT sig_name INTO
@snama FROM `signature` WHERE sig_id =New.signature; SELECT sig_name INTO @snama
FROM `signature` WHERE sig_id =New.signature;
if (select count(*) from `report` where ((now() - waktu)<60) ORDER BY waktu DESC ) >0
```

```
then if (select sid  from 'report' where ((now() - waktu)<60) ORDER BY waktu DESC
LIMIT 0,1)<> New.sid then INSERT INTO `snort`.`report` (`sid` ,`waktu`,`ips`)
VALUES (New.sid, NOW( ),trim(concat('' ,@ssip)) );INSERT INTO `snort`.`serangan`
(`nama`, `tanggal`, `ips`)VALUES (New.sid, NOW( ),trim(concat(' ' ,@ssip)) );
INSERT INTO `snort`.`outbox` (`DestinationNumber`,`TextDecoded`)VALUES
(trim(concat(' ' ,@nomers)),concat('Ada Intruder : ' ,@snama,  ' Oleh IP : ',@ssip) );
end if;
else INSERT INTO `snort`.`report` (`sid` ,`waktu`,`ips`)VALUES (New.sid, NOW( ),
trim(concat('' ,@ssip)) );INSERT INTO `snort`.`serangan` (`nama`, `tanggal`, `ips`)VALUES
(New.sid, NOW( ),trim(concat(' ' ,@ssip)) );INSERT INTO `snort`.`outbox`
(`DestinationNumber`,`TextDecoded`)VALUES (trim(concat(' ' ,@nomers)),
concat('Ada Intruder : ' ,@snama,  ' Oleh IP : ',@ssip) );
end if;
end if;
END//DELIMITER ;
```

Trigger diatas berarti bahwa ketika terdapat data yang masuk pada tabel event (db snort), sistem akan mengecek waktu pada tabel event dan signature, jika sama maka tabel outbox akan terisi dengan masukan :

1. Pada *nomer* berisi nomor tujuan (Hp Admin)
2. Tabel *event* pada kolom signature (id alert)
3. Tabel *signature* pada kolom sig_name (pesan alert)
4. Table *report* pada kolom waktu (waktu terdeteksi intrusion yang masuk)

BAB IV

IMPLEMENTASI DAN PENGUJIAN

Pada implementasi IDPS (Intrusion Detection And Prevention System) ini, perlu beberapa hal yang perlu diperhatikan. antara lain kebutuhan sistem akan perangkat keras (*hardware*) dan perangkat lunak (*sofiware*), serta langkah-langkah yang harus dilakukan untuk dapat melakukan instalasi aplikasi agar dapat berfungsi sebagaimana mestinya.

4.1 Perangkat Lunak

Perangkat lunak yang diperlukan pada perancangan dan pembuatan aplikasi ini adalah:

1. Sistem Operasi Ubuntu 10.04
2. Gammu
3. Apache Web Server
4. Snort
5. MySQL
6. Gambas

4.2 Pengujian IDPS (Intrusion Detection and Prevention System)

Pada pengujian IDPS pada jaringan komputer, dilakukan hanya melakukan deteksi dan mengatasi adanya intrusi yang lewat pada sistem.

PC intruder melakukan serangan ping of death, Telnet, FTP, sedangkan pada PC sistem menjalankan snort, dengan perintah :

Snort –v -c /etc/snort/snort.conf

Perintah diatas akan menjalankan snort dengan konfigurasi yang ada pada file snort.conf dan memasukkan pada log. Untuk mengetahui apakah terdapat intrusi atau tidak,dapat diketahui dari log yang masuk dalam database. Untuk mengetahui apakah terdapat intrusi atau tidak, dapat diketahui dari log yang masuk dalam *database*, seperti pada gambar 4.1 :

sig_id	sig_name
3	ICMP PING speedera
4	ICMP Large ICMP Packet
20	(ftp_telnet) FTP traffic encrypted
15	(ftp_telnet) Invalid FTP Command
16	(ftp_telnet) FTP command parameters were malformed
17	(ftp_telnet) TELNET CMD on FTP Command Channel
19	FTP UserLogin

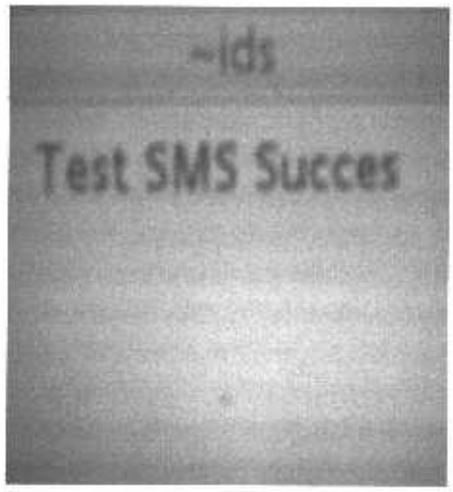
Gambar 4.1 LOG Snort

4.3 Pengujian Sistem SMS (Short Messaging System)

Pengiriman SMS pada sistem ini menggunakan gammu sebagai *tool* yang bekerja. Pada pengujian otomatisasi pengiriman SMS, dilakukan dengan memasukkan sebuah data pada database sehingga gammu akan melakukan pengiriman pesan. Karena sistem kerja dari gammu ini adalah, sms akan langsung dikirim ketika tabel outbox terisi, dan pesan yang dikirim adalah isi dari tabel outbox tersebut.

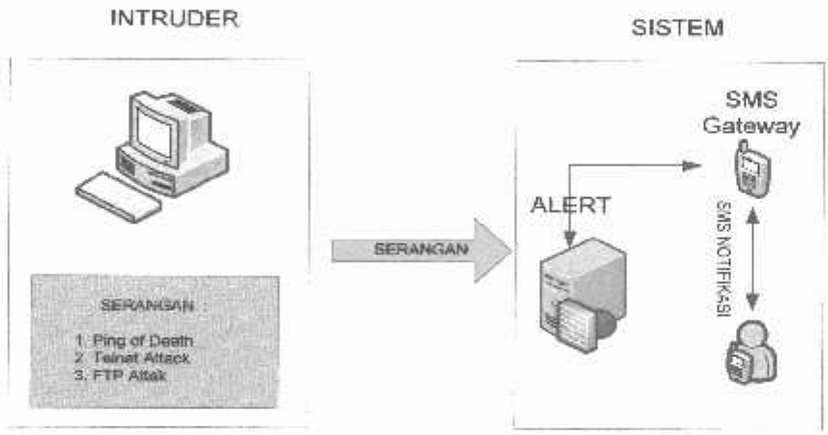
Pengujian SMS ini agar bisa memberikan suatu notifikasi sms dengan perintah *sudo gammu-smsd -c /etc/smsdrc* Kemudian jalankan program seperti pada gambar :

Gambar 4.2 Test SMS ke Admin



Gambar 4.3 Pengujian SMS Berhasil

4.4 Pengujian IDPS (Intrusion Detection and Prevention System) dengan Notifikasi SMS



Gambar 4.4 Diagram Blok Pengujian Sistem

Pengujian sistem IDS dilakukan seperti pada gambar 4.1, yaitu PC intruder dengan IP 192.168.1.2 melakukan 3 jenis intrusi yang akan ditujukan pada PC sistem dengan IP 192.168.1.1. Sistem IDS akan mendeteksi serangan tersebut, kemudian mengirimkan pesan atau alert ke sistem SMS, dan selanjutnya akan mengirimkan notifikasi berupa pesan alert ke administrator serta menampilkan alert tersebut pada log yang terdapat diprogram. Intrusi yang dilakukan untuk pengujian sistem ini adalah :

a. Ping of Death : Menggunakan program utility ping yang ada di sistem operasi komputer. Biasanya ping digunakan untuk men-cek berapa waktu yang dibutuhkan untuk mengirimkan sejumlah data tertentu dari satu

komputer ke komputer lain. Panjang maksimum data yang dapat dikirim menurut spesifikasi protokol IP adalah 65,536 byte. Pada Ping of Death data yang dikirim melebihi maksimum paket yang di izinkan menurut spesifikasi protokol IP. Konsekuensinya, pada sistem yang tidak siap akan menyebabkan sistem tersebut crash (tewas), hang (bengong) atau reboot (booting ulang) pada saat sistem tersebut menerima paket yang demikian panjang.

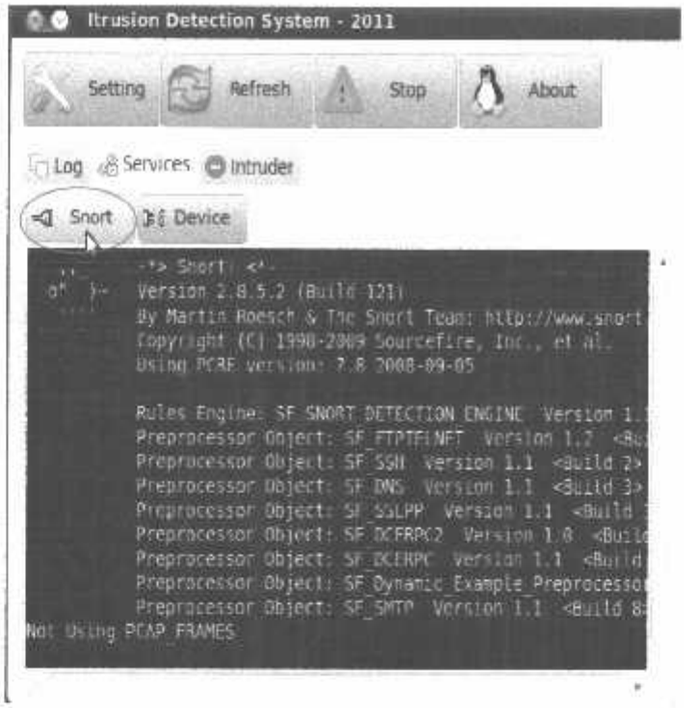
b. Telnet Attack : Telnet (Telecommunication Network) adalah sebuah protokol jaringan yang digunakan di koneksi Internet atau Local Area Network. Telnet memungkinkan kita untuk menghubungkan "terminal" kita dengan host remote yang berada di luar jaringan. Telnet biasanya digunakan untuk "remote login" dari PC ke PC lain dalamjaringan. Remote login semacam ini memungkinkan anda untuk menggunakan aplikasi yang berada dalam sistem remote. Remote login semacam ini hanya menyediakan koneksi text only,biasanya dalam bentuk command line prompt, seakan-akan anda duduk di terminal yang terhubung pada mesin remote. Telnet tidak mengenkripsi informasi yang dikirimkannya. Semua dikirimkan dalam bentuk plain text, termasuk kata sandi.

c. FTP attack : FTP (singkatan dari File Transfer Protocol) adalah sebuah protokol Internet yang berjalan di dalam lapisan aplikasi yang merupakan standar untuk pentransferan berkas (file) komputer antar mesin-mesin dalam sebuah internetwork.memanfaatkan kelalaian admin situs atau penyedia hosting yang memungkinkan pengguna anonymous di ijinkan mengakses FTP.

Untuk menjalankan sistem ini pertama kali yang harus dilakukan adalah seperti pada gambar sebagai berikut :

Gambar 4.5 Start Program

Gambar 4.6 Setting Program



Gambar 4.7 Menjalankan Service Snort

4.4.1 Pengujian Sistem yang akan dilakukan dengan menggunakan 3 metode bentuk serangan

a. Ping of Death

Pada pengujian dengan Ping of Death ini, dilakukan uji coba ping dengan jumlah paket yang 1000 pada komputer intruder. 1000 Paket, 10 Count (10 kali ping) , tiap 10 detik. Menggunakan perintah : ping 192.168.1.1 –c 10 –i 10 –s 1000

Gambar 4.8 Tampilan Serangan

Gambar 4.9 Notifikasi SMS

b. Telnet Attack

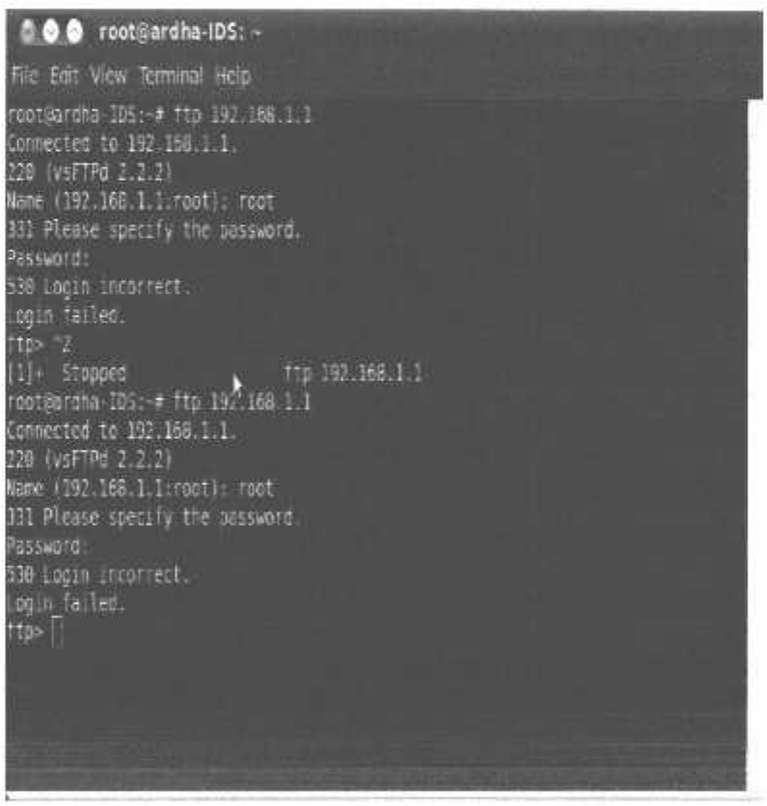
Pengujian telnet ini juga dilakukan dengan uji melakukan untuk masuk dalam port 21 (FTP), Login beberapa usaha untuk masuk kedalam FTP.

Gambar 4.10 Tampilan Serangan

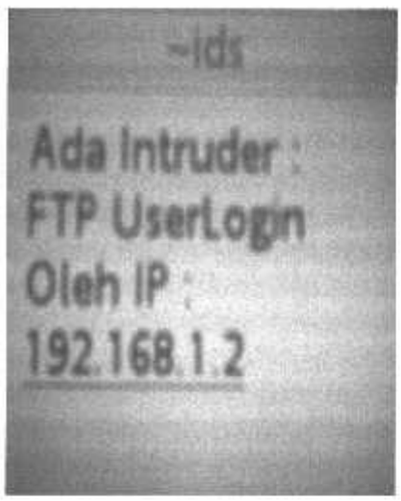
Gambar 4.11 Notifikasi SMS

c. FTP attack

Pengujian telnet ini juga dilakukan dengan uji login FTP dengan perintah ftp 192.168.1.1



Gambar 4.12 Tampilan Serangan



Gambar 4.13 Notifikasi SMS

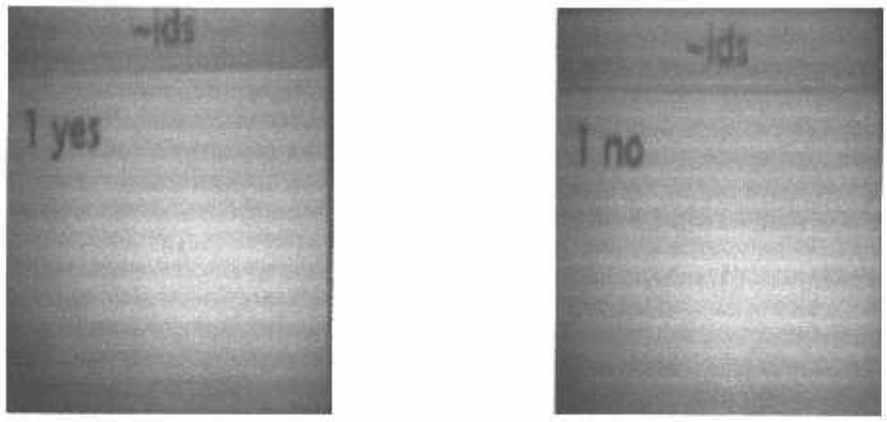
4.4.2 Administrataor Melakukan Pencegahan Terhadap Intrusion

Pengujian serangan diatas akan mengirimkan notifikasi berupa pesan singkat (SMS) secara otomatis kepada administrator. kemudian admin melakukan suatu tindakan pencagahan terhadap intrusion. Berikut ini adalah gambar yang menunjukkan Adanya suatu serangan yang dikirmkan secara otomatis melalui SMS.

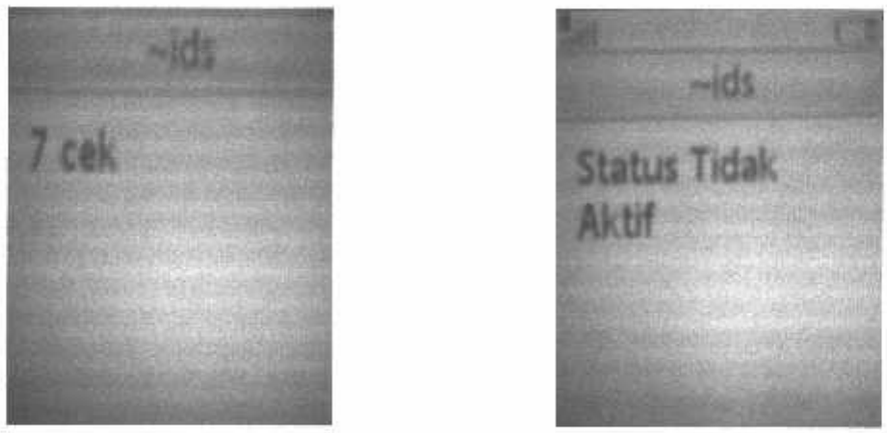
Gambar 4.14 Notifikasi Serangan

Pada gambar 4.13 terdapat suatu serangan yang telah dilakukan oleh penyerang. Kemudian jika administrator tidak memberikan suatu respon maka tiap satu menit akan diberikan notifikasi SMS. Admin dapat melakukan suatu bentuk tindakan pengeblokan IP secara manual atau otomatis, dimaksudkan secara manual admin memberikan sutu perintah tindakan melalui SMS. Jika pengeblokan IP dijalankan secara Otomatis maka IP yang dianggap sebagai intrusion akan di

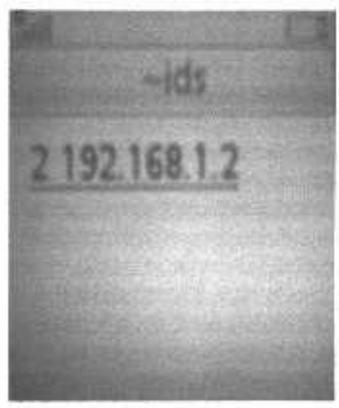
Blok secara otomatis. Berikut ini gambar dari tindakan administrator Secara Manual:



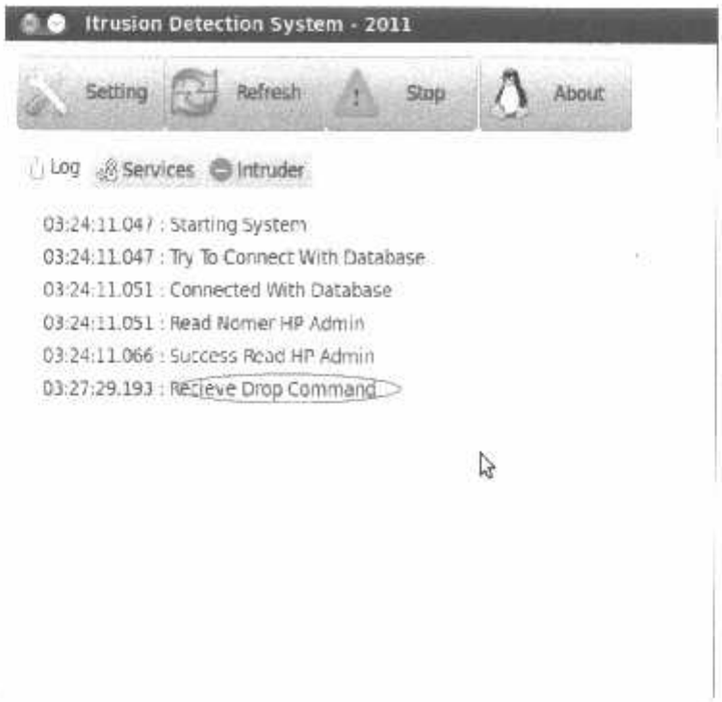
Gambar 4.15 Perintah SMS Secara Otomatis (kiri) atau manual (kanan)



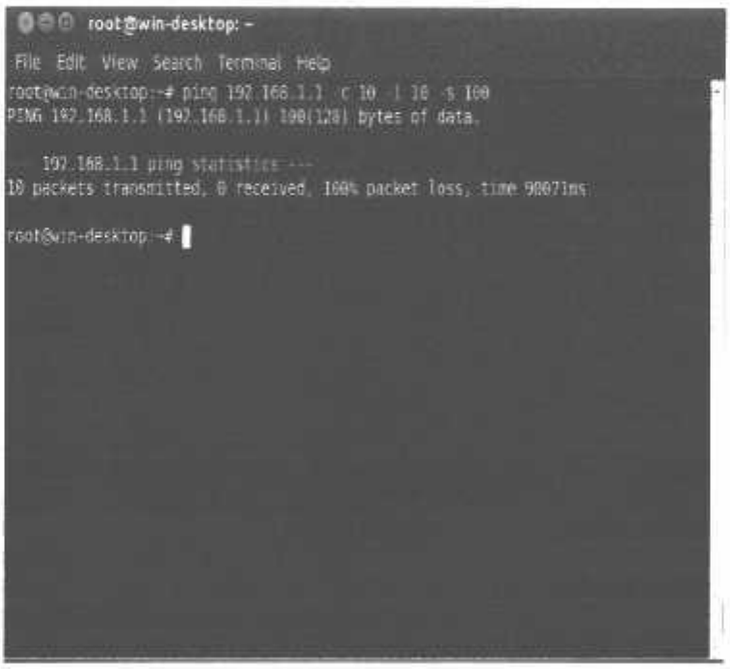
Gambar 4.16 Cek Status Manual (kiri) dan Balasan Cek Status (kanan)



Gambar 4.17 SMS yang dikirimkan Admin secara manual



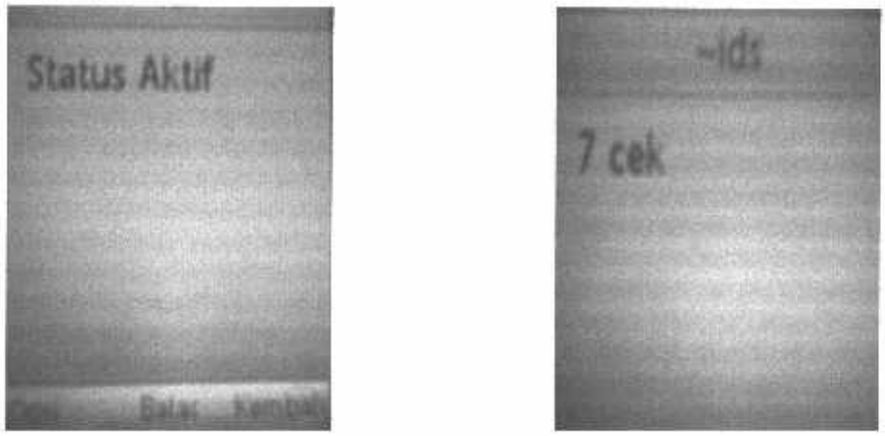
Gambar 4.18 Log yang dilakukan tindakan Admin



Gambar 4.19 Intrusion Telah Digagalkan Admin

Dari penjelasan gambar tersebut maka serangan yang telah dilakukan oleh attacker telah digagalkan oleh admin. Dengan suatu tindakan secara manual pada saat sistem mengalami serangan yang dapat membuat server crash, hang atau reboot maka alangkah lebih baik dilakukan secara otomatis sehingga tidak

membuat sistem mengalami crash, hang atau reboot. Berikut ini adalah gambar sistem yang admin lakukan secara otomatis :

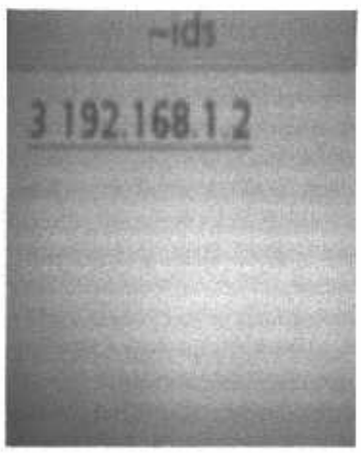


Gambar 4.20 Notifikasi cek Sistem Otomatis (kanan) dan Balasan Cek Status (kiri)

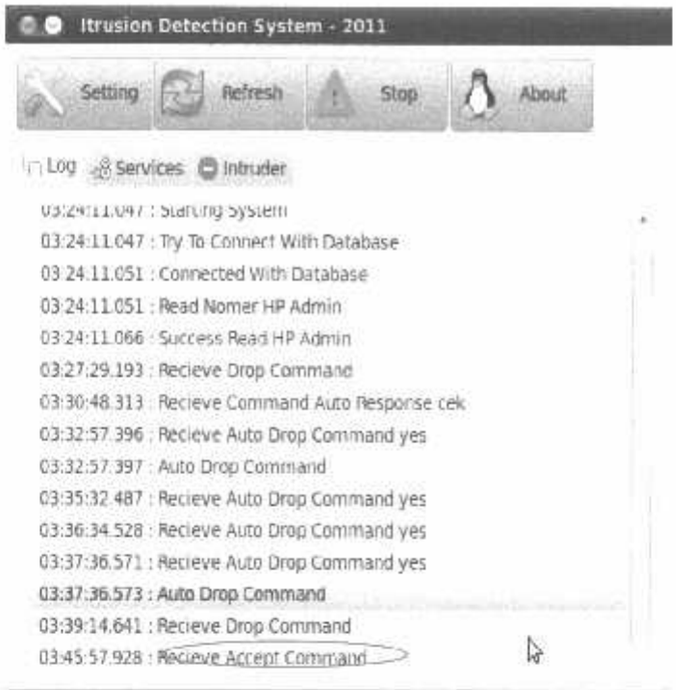
Gambar 4.21 Log yang dilakukan Admin

Gambar 4.22 Intrusion telah digagalkan secara otomatis

Dari Penjelasan gambar diatas admin juga dapat mengijinkan suatu paket melewati jaringan yang telah terdeteksi dengan alasan serangan tersebut tidak membahayakan atau admin melakukan suatu kesalahan dalam memasukkan password untuk masuk kedalam sistem. Berikut adalah gambar yang diijinkan oleh administrator :



Gambar 4.23 Perintah SMS paket diijinkan

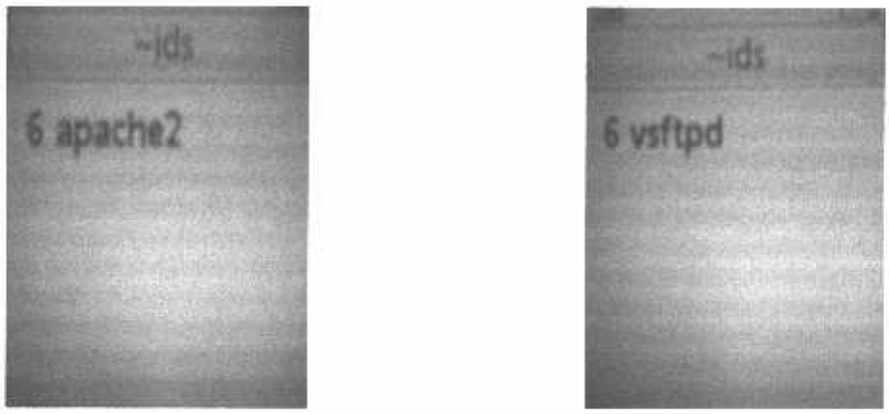


Gambar 4.24 Log perintah paket diijinkan lewat

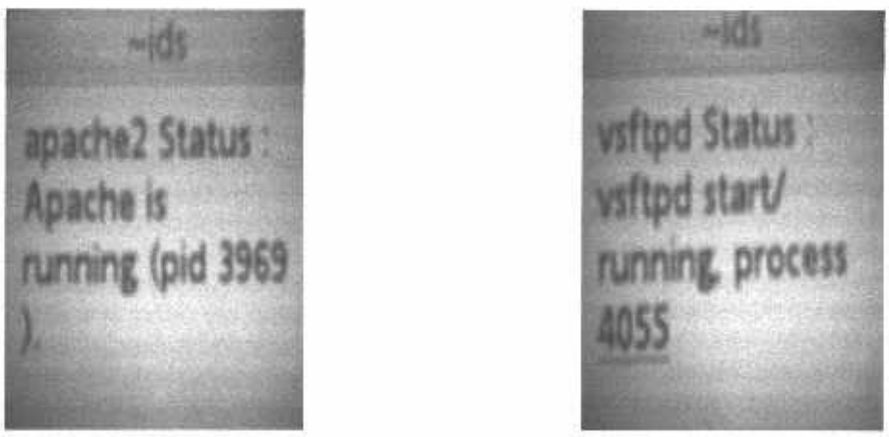
Dalam mengamankan suatu web server suatu sistem yang pada saat server mengalami masalah admin juga dapat memberikan suatu tindakan dengan melakukan stop, restart, start service untuk HTTP dan FTP. Berikut adalah gambar suatu tindakan yang dilakukan oleh administrator :

Gambar 4.25 Perintah SMS HTTP untuk melakukan stop (kiri), start (Kanan), restart (tengah)

Gambar 4.26 Perintah SMS FTP untuk melakukan stop (kiri), start (Kanan), restart (tengah)



Gambar 4.27 Perintah Cek status HTTP (kiri) dan FTP (kanan)

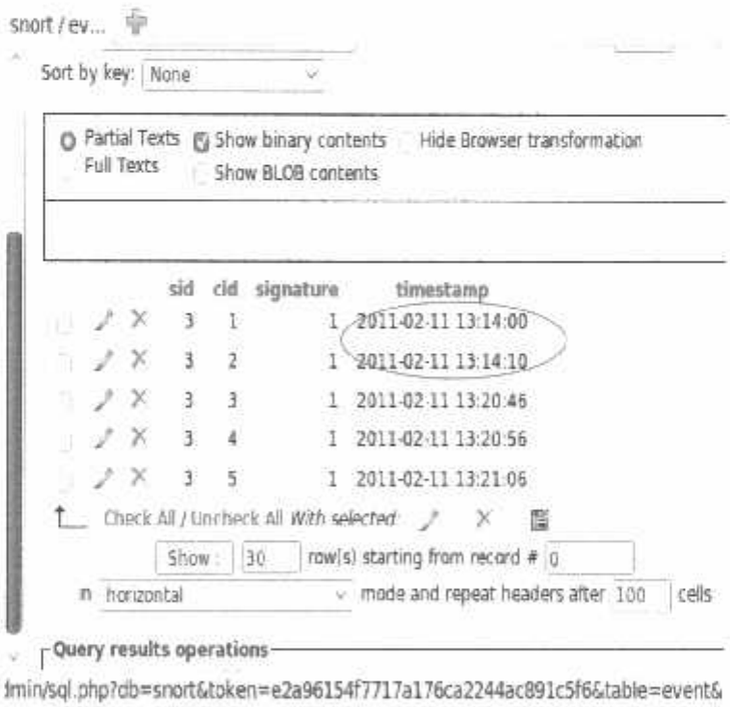


Gambar 4.28 Notifikasi SMS cek HTTP (kiri) dan FTP (kanan)

4.5 Analisa Sistem

Pengujian dari hasil diatas memberikan hasil yang akan tersimpan dalam database sistem dan mengirimkan notifikasi berupa pesan singkat (SMS) secara otomatis kepada administrator. Sistem ini terdapat dalam dua tipe yaitu Manual dan Otomatis untuk melakukan suatu pencegahan intrusion. Dimana sistem ini jika dijalankan secara otomatis maka dengan otomatis sistem ini akan melakukan pengeblokan pada intrusion. Dengan begitu Admin pada saat melakukan suatu kegiatan lain maka seorang admin dapat menggunakan sistem secara otomatis atau jika suatu serangan membuat server crash, hang atau reboot . Pada saat seorang admin ingin melakukan suatu tindakan secara manual pada saat ingin menganalisa suatu serangan admin dapat menjalankan sistem dengan cara manual.

Analisa ini menjelaskan tentang selisih waktu pada saat sistem menyimpan database dan SMS dikirmkan serta diterima dan admin melakukan tindakan. Gambar dibawah ini menunjukkan *timestamp* atau waktu pada saat terjadinya intrusi dan alert pada tabel event :



Gambar 4.29 Database snort.table event

sig_id	sig_name
3	ICMP PING speedera
4	ICMP Large ICMP Packet
20	(ftp_telnet) FTP traffic encrypted
15	(ftp_telnet) Invalid FTP Command
16	(ftp_telnet) FTP command parameters were malformed
17	(ftp_telnet) TELNET CMD on FTP Command Channel
19	FTP UserLogin

Gambar 4.30 Database snort . signature

Pada dua gambar diatas bisa diketahui tentang *signature* atau jenis intrusi yang terjadi, sebagai contoh pada tabel *event* terdapat kolom *signature* 3, angka tersebut merupakan ID dari tabel *signature* yang mempunyai nama ICMP PING speedera. Pada kolom *timestamp* dapat diketahui intrusi terjadi pada pukul 13:14:00 dan sms yang dikirm yaitu pada tabel outbox pada pukul 13:14:00, seperti gambar dibawah :

	2011-02-11 13:14:00	085649770665	Ada Intruder : ICMP PING speedera Oleh IP : 192.16...

Gambar 4.31 Table outbox (sms yang dikirim)

Analisa diatas akan memberitahukan kepada admin jika ada intrusion masuk schingga administrator akan dengan segera mengetahui adanya intrusi masuk kedalam sistem.

	ips	status
	192.168.1.2	DROP

Check All / Uncheck All *With se*

Gambar 4.32 Table iptable

Pada gambar 4.32 setelah admin diberikan peringatan melalui SMS maka admin melakukan DROP pada intrusion agar tidak dapat melanjutkan serangan.

Dari analisa diatas dapat dibuktikan bahwa IDS mengenali adanya penyusupan dengan cara menyadap paket yang lewat terhadap jaringan kemudian membandingkan dengan database snort/rule. Database rule tersebut berisi signature atau jenis – jenis paket serangan. Jika terdapat suatu pola paket data memiliki

kesamaan maka paket data tersebut dianggap sebagai serangan kemudian akan diberikan suatu pemberitahuan / alert agar seorang admin dapat mengetahui dan melakukan suatu tindakan secara tepat. Demikian juga sebaliknya, jika paket data tersebut tidak memiliki kesamaan atau tidak terdapat dalam database rule pada IDS (Intrusion Detection System) maka hal tersebut diperbolehkan lewat atau tidak dianggap suatu serangan. Berikut ini adalah tabel pengujian yang dilakukan :

Tabel 5.1 Tabel Pengujian

No.	Serangan yang digunakan	Proses yang dilakukan	Database	Keterangan
1	Ping of Death	Mengirimkan dengan jumlah paket data yang melebihi dari 65,536byte dari ketentuan protokol IP yang bisa mengakibatkan kesibukan dalam jaringan	ADA	Terdeteksi dengan jenis serangan ICMP (Internet control Message Protokol)
2	Telnet	Melakukan remote login dengan menjalankan dalam terminal	ADA	Terdeteksi dengan jenis serangan Telnet CMD (command line)
3	FTP attack	Melakukan login kedalam FTP server untuk mencoba masuk dengan memasukkan password	ADA	Terdeteksi dengan jenis serangan FTP login

BAB V

PENUTUP

5. Kesimpulan dan Saran

5.1 Kesimpulan

Setelah dilakukan perancangan dan pengujian sistem, dapat diambil kesimpulan bahwa :

1. Trafik SMS pada jaringan operator cellular berpengaruh terhadap waktu yang dibutuhkan dalam pengiriman SMS.
2. Deteksi serangan yang dilakukan oleh sistem bergantung pada rule yang dimasukkan pada konfigurasi snort sebagai Intrusion Detection System. dan jika serangan yang dalam database tidak ada maka serangan tersebut dapat terlewati.
3. Otomatisasi pengirim notifikasi terjadi karena adanya trigger pada database yang dibuat.
4. Intrusion Prevention System menggunakan sistem Iptables yang membantu untuk memblok atau meneruskan paket berdasarkan IP address.
5. Sistem notifikasi berupa SMS dapat memberikan administrator informasi lebih dini yang memungkinkan administrator mengatasi secara langsung walaupun pada saat administrator tidak ada pada tempatnya melalui media SMS.
6. Dengan ketergantungan sistem pada Database, apabila pada saat database down, maka sistem SMS tidak akan dapat berfungsi sama sekali.

5.2 Saran

Sistem yang telah dibuat sebagai Skripsi oleh penulis masih memiliki kelemahan dan kekurangan, oleh karena itu penulis mengharapkan adanya masukan, saran dan kritik yang nantinya dapat menyempurnakan sistem tersebut menjadi lebih bagus dan sempurna. Hal – hal yang dapat dikembangkan atau diberikan sebagai saran berkaitan dengan perancangan dan pembuatan sistem ini. Bila dari Sistem Aplikasi Sms Gateway Sebagai Manajemen Keamanan Web Server Menggunakan IDS (intrusion

detection system) dan IPS (Intrusion Prevention System), hendaknya terlebih dahulu dapat memperhatikan dari faktor :

- Diharapkan untuk selalu update rule agar serangan baru dapat terdeteksi oleh IDS dan dapat dicegah agar serangan tidak masuk.
- Kualitas device seperti HP dan kabel data yang digunakan mempengaruhi proses transfer data.

DAFTAR PUSTAKA

Dony Ariyus, M.Kom, 2007, *Intrusion Detection System*, Penerbit Andi, Yogyakarta.

www.miftahstever.com/blog/download/gammu_penjelasan.doc

http://tutorial.jalak.web.id/2009/windows/winxp/monitoring-serangan-hacker-ke-jaringan dengan-snort/

Mahdi Ridho, 2006, *Panduan Aplikatif Pemrograman Gambas*, Penerbit Andi, Yogyakarta

http://janita04if.wordpress.com/2009/05/03/intrusion-prevention-system-ips/

www.unsri.ac.id/upload/arsip/iptables.doc

http://bebas.vlsm.org/v10/onno-ind-2/network/network-security/snort-untuk-mendeteksi penyusup-4-2002.doc

http://foxhound-61.blogspot.com/2010/02/contoh-penggunaan-iptables-untuk.html

Tom Thomas. 2005 .*Network Security first-step*. Penerbit ANDI , Yogyakarta.

www.unsri.ac.id/upload/arsip/Handry%20Fratama%20Diansyah.doc

http://www.unsri.ac.id/upload/arsip/NURHIDAYAT%20(08053111051).doc

http://mtsox.wordpress.com/2010/03/01/sumber-lubang-keamanan/

http://www.bestlib.co.cc/2009/08/ensiklopedia-serangan-denial-of-service.html

Rahmat Rafiudin. 2010 .*Mengganyang Hacker dengan SNORT*. Penerbit Andi, Yogyakarta.

PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG

INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNONOLGI INDUSTRI
FAKULTAS TEKNOLOGI SIPIL DAN PERANCANGAN
PROGRAM PASCASARJANA MAGISTER TEKNIK

PT. BNI (PERSERO) MALANG
BANK NIAGA MALANG

Kampus I : Jl. Bendungan Sugura-gura No.2 Telp. (0341) 551431 (Hunting) Fax. (0341) 553015 Malang 65145
Kampus II : Jl. Raya Karanglo Km 2 telp. (0341) 417636 Fax. (0341) 417634 Malang

BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI

Nama : Tri Ardhana Sepdianto
NIM : 06.12.630
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Komputer dan Informatika
Judul Skripsi : "SISTEM APLIKASI SMS GATEWAY SEBAGAI MANAJEMEN KEAMANAN WEB SERVER MENGGUNAKAN IDS (INTRUSION DETECTION SYSTEM) DAN IPS (INTRUSION PREVENTION SYSTEM)"

Dipertahankan dihadapan Majelis Penguji Skripsi Jenjang Strata Satu (S-1) pada :

Hari : Sabtu
Tanggal : 12 Februari 2011
Dengan Nilai : 82,75 (A)

Panitia Ujian Skripsi :

Ketua Majelis Penguji

Ir. Yusuf Ismail Nakhoda, MT
NIP.Y. 1018800189

Sekretaris Majelis Penguji

Dr. Eng. Aryuanto S, ST. MT
NIP.Y. 1030800417

Anggota Penguji :

Penguji I

I Komang Somawirata, ST, MT
NIP. P. 1030100361

Penguji II

Irmalia Suryani Faradisa, ST, MT
NIP.P. 1030000365

PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG

INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNONOLGI INDUSTRI
FAKULTAS TEKNOLOGI SIPIL DAN PERANCANGAN
PROGRAM PASCASARJANA MAGISTER TEKNIK

PT. BNI (PERSERO) MALANG
BANK NIAGA MALANG

Kampus I : Jl. Bendungan Sugura-gura No.2 Telp. (0341) 551431 (Hunting) Fax. (0341) 553015 Malang 65145
Kampus II : Jl. Raya Karanglo, Km 2 telp. (0341) 417636 Fax. (0341) 417634 Malang

LEMBAR PERBAIKAN SKRIPSI

Dalam pelaksanaan ujian skripsi jenjang Strata satu (S-1) jurusan Teknik Elektro konsentrasi Teknik Komputer dan Informatika, maka perlu adanya perbaikan skripsi untuk mahasiswa :

Nama	: Tri Ardhana Sepdianto
NIM	: 06.12.630
Jurusan	: Teknik Elektro S-1
Konsentrasi	: Teknik Komputer dan Informatika
Masa Bimbingan	: 30 November 2010 s/d 30 Mei 2011
Judul Skripsi	:"SISTEM APLIKASI SMS GATEWAY SEBAGAI MANAJEMEN KEAMANAN WEB SERVER MENGGUNAKAN IDS (INTRUSION DETECTION SYSTEM) DAN IPS (INTRUSION PREVENTION SYSTEM)"

No	Tanggal	Uraian	Paraf
1	Penguji I 12 Februari 2011	Flowchart sistem	
2	Penguji II 12 Februari 2011	Rumusan Masalah	

Mengetahui ,

Dosen Pembimbing I

Ir.Yusuf Ismail Nakhoda, MT
NIP. Y . 1018800189

Dosen Pembimbing II

Sotyohadi, ST
NIP. Y . 1039700309

Dosen Penguji ,

Penguji I

I Komang Somawirata, ST, MT
NIP. P. 1030100361

Penguji II

Irmalia Suryani Faradisa, ST, MT
NIP.P. 1030000365

INSTITUT TEKNOLOGI NASIONAL
Jl. Bendungan Sigura-gura No. 2
MALANG

FORMULIR BIMBINGAN SKRIPSI

Nama : Tri Ardhana Sepdianto
Nim : 06.12.630
Masa Bimbingan : 30 November 2010 s/d 30 Mei 2011
Judul Skripsi : **SISTEM APLIKASI SMS GATEWAY SEBAGAI MANAJEMEN KEAMANAN WEB SERVER MENGGUNAKAN IDS (Intrusion Detection System) dan IPS (Intrusion Prevention System)**

No.	Tanggal	Uraian	Paraf Pembimbing
1	25-01-2011	Pengujian program/Demo program	
2	26-01-2011	Konsultasi makalah Seminar	
3	27-01-2011	Perbaikan gambar untuk diperbesar	
4	28-01-2011	Perbaikan abstrak	
5	29-01-2011	Acc makalah seminar	
6	09-02-2011	Konsultasi laporan skripsi	
7	11-02-2011	Perbaikan penulisan laporan sesuai ketentuan	
8			
9			
10			

Malang,

Dosen Pembimbing I

Ir.Yusuf Ismail Nakhoda, MT

NIP. Y . 1018800189

Form S-4b

INSTITUT TEKNOLOGI NASIONAL
Jl. Bendungan Sigura-gura No. 2
MALANG

FORMULIR BIMBINGAN SKRIPSI

Nama : Tri Ardhana Sepdianto
Nim : 06.12.630
Masa Bimbingan : 30 November 2010 s/d 30 Mei 2011 34
Judul Skripsi : **SISTEM APLIKASI SMS GATEWAY SEBAGAI MANAJEMEN KEAMANAN WEB SERVER MENGGUNAKAN IDS (Intrusion Detection System) dan IPS (Intrusion Prevention System)**

No.	Tanggal	Uraian	Paraf Pembimbing
1	11-01-2011	Konsultasi Bab 1-3	
2	27-01-2011	Demo Program	
3	29-01-2011	Konsultasi Makalah seminar hasil	
4	10-02-2011	Konsultasi Laporan Skripsi	
5			
6			
7			
8			
9			
10			

Malang,

Dosen Pembimbing I

Sotyohadi, ST

NIP. Y . 1039700309

Form S-4b

INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO

Formulir Perbaikan Ujian Skripsi

Dalam pelaksanaan Ujian Skripsi Jarjang Strata 1 Jurusan Teknik Elektro Konsentrasi T. Energi Listrik / T. Elektronika / T.Infokom, maka perlu adanya perbaikan skripsi untuk mahasiswa :

NAMA : TRI ARDIANTO
NIM : 06-02.623
Perbaikan meliputi :

Malang,

(_____)

INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO

Formulir Perbaikan Ujian Skripsi

Dalam pelaksanaan Ujian Skripsi Jenjang Strata 1 Jurusan Teknik Elektro Konsentrasi T. Energi Listrik / T. Elektronika / T.Infokom, maka perlu adanya perbaikan skripsi untuk mahasiswa :

NAMA : TRI ARDANA
NIM : 06.12630
Perbaikan meliputi :

RUMUSAN MASALAH.

Malang, 200

( )

LAMPIRAN

```
DROP TRIGGER IF EXISTS `replay`;

DELIMITER //CREATE TRIGGER `replay` AFTER INSERT ON `event` FOR EACH ROW BEGIN
SELECT

nomer INTO @nomers FROM `hpadmin` WHERE ID='1' Limit 1;SELECT convert(inet_ntoa

( iphdr.ip_src ),char) INTO @ssip FROM iphdr ORDER BY `iphdr`.`cid` DESC limit 1;

if (select count(*) from iptable where ips=@ssip) = 0 then SELECT sig_name INTO

@snama FROM `signature` WHERE sig_id =New.signature; SELECT sig_name INTO @snama

FROM `signature` WHERE sig_id =New.signature;

if (select count(*) from `report` where ((now() - waktu)<60) ORDER BY waktu DESC ) >0

then if (select sid from `report` where ((now() - waktu)<60) ORDER BY waktu DESC

LIMIT 0,1)<> New.sid then INSERT INTO `snort`.`report` (`sid` ,`waktu`,`ips`)

VALUES (New.sid, NOW( ),trim(concat(" ,@ssip)) );INSERT INTO `snort`.`serangan`

(`nama`, `tanggal`, `ips`)VALUES (New.sid, NOW( ),trim(concat(' ' ,@ssip)) );

INSERT INTO `snort`.`outbox` (`DestinationNumber`,`TextDecoded`)VALUES

(trim(concat(' ' ,@nomers)),concat('Ada Intruder : ' ,@snama, ' Oleh IP : ',@ssip) );

end if;

else INSERT INTO `snort`.`report` (`sid` ,`waktu`,`ips`)VALUES (New.sid, NOW( ),

trim(concat(" ,@ssip)) );INSERT INTO `snort`.`serangan` (`nama`, `tanggal`, `ips`)VALUES

(New.sid, NOW( ),trim(concat(' ' ,@ssip)) );INSERT INTO `snort`.`outbox`

(`DestinationNumber`,`TextDecoded`)VALUES (trim(concat(' ' ,@nomers)),

concat('Ada Intruder : ' ,@snama, ' Oleh IP : ',@ssip) );

end if;

end if;

END//DELIMITER ;
```

Gambas class file

```
  PUBLIC $Query AS String
  PUBLIC $Result AS Result
  PUBLIC $Str AS String

PUBLIC SUB Button2_Click()

  $Query = "TRUNCATE TABLE `hpadmin` "
TRY $Result = MODMain.$Con.Exec($Query)
  $Query = "INSERT INTO `hpadmin` (`ID`,`nomer`)VALUES ('1','" & hpadmin.Text & "' )"



  ModMain.$HP = hpadmin.Text
  ModMain.$Status = status.Value
  ModMain.$Autos = Autos.Value
CATCH
 Message.Error("Update Data Gagal!.")

END

PUBLIC SUB Form_Open()
  status.Value = ModMain.$Status
  hpadmin.Text = ModMain.$HP
END

PUBLIC SUB Button1_Click()
  $Query = "INSERT INTO `snort`.`outbox` (`DestinationNumber`,`TextDecoded`)VALUES ('" &
ModMain.$HP & "','Test SMS Succes' )"
  $Result = MODMain.$Con.Exec($Query)
  Message.Info("Success Send Messages")
END
```

```
' Gambas module file
PUBLIC $Con AS NEW Connection
PUBLIC $HP AS String
PUBLIC $Status AS Boolean
PUBLIC $Autos AS Boolean
PUBLIC $Query AS String
PUBLIC $Result AS Result
PUBLIC $Str AS String



PUBLIC PROCEDURE Connect()
        $Con.Close()
        $Con.Type = "mysql"
        $Con.Host = "localhost" '
        $Con.Login = "root"
        $Con.Port = "3306"
        $Con.Name = "snort"
        $Con.Password = "12345678"
        $Con.Open() '

END
PUBLIC SUB main()
TRY
 FMain.addLogs("Starting System")
 FMain.addLogs("Try To Connect With Database")
 Connect()
 FMain.addLogs("Connected With Database")
 FMain.addLogs("Read Nomer HP Admin ")
 $Query = "SELECT * FROM `hpadmin` WHERE ID='1' LIMIT 1 "
 $Result = $Con.Exec($Query)
 $Status = TRUE
 $Autos = FALSE

 FormSplash.Show
 IF $Result.Count > 0 THEN
  $HP = $Result!nomer
  FMain.addLogs("Success Read HP Admin")
 ELSE
  FMain.addLogs("Nomer HP Admin Tidak Ada")
 ENDIF


CATCH
 FMain.addLogs("Koneksi Database Gagal")
 Message.Error("Koneksi Database Gagal")
END
```

```
' Gambas class file


  PUBLIC $Query AS String
  PUBLIC $Result AS Result
  PUBLIC $Str AS String
  PUBLIC $countDetect AS Integer
  PUBLIC $countDrop AS Integer
  PUBLIC $cStatus AS String


PUBLIC SUB _new()
  $hProcess = EXEC ["bash", "--noediting"] FOR READ WRITE
END

PUBLIC SUB Form_Open()
  DIM $i AS Integer
  tbVDrop.Columns.Count = 4
  tbVDrop.Columns.Width = 100
  tbVDrop.Rows.Count = 2

  tbVDrop[0, 0].Text = "No"
  tbVDrop[0, 1].Text = "Tanggal"
  tbVDrop[0, 2].Text = "IP"

  tbVDrop[0, 3].Text = "Serangan"

  tbVDetect.Columns.Count = 4
  tbVDetect.Columns.Width = 100
  tbVDetect.Rows.Count = 2
  tbVDetect[0, 0].Text = "No"
  tbVDetect[0, 1].Text = "Tanggal"
  tbVDetect[0, 2].Text = "IP"
  tbVDetect[0, 3].Text = "Serangan"


  FOR $i = 0 TO tbVDrop.Columns.Count - 1
    tbVDrop[0, $i].BackColor = &H00F0F0F0&
    tbVDrop[0, $i].Font.Bold = TRUE
    tbVDetect[0, $i].BackColor = &H00F0F0F0&
    tbVDetect[0, $i].Font.Bold = TRUE
  NEXT
  $countDetect = 0
  $countDrop = 0
  getData()
  Timer1.Enabled = TRUE
END

PUBLIC SUB cekCount()

  DIM $i AS Integer
  $Query = "SELECT `id_serangan`, `nama`, `ips`, `tanggal`, `status`FROM `serangan` WHERE status ='0'"
  $Result = ModMain.$Con.Exec($Query)
  IF $Result.Count > $countDetect THEN
    getData
  ENDIF
```

```
 $Query = "SELECT `id_serangan`, `nama`, `ips`, `tanggal`, `status`FROM `serangan` WHERE status ='1'"
 $Result = ModMain.$Con.Exec($Query)
 IF $Result.Count > $countDrop THEN
   getData


 ENDIF
 END


PUBLIC SUB getData()
 DIM $i AS Integer
 DIM $ipx AS String

 $Query = "SELECT `id_serangan`, `nama`, `ips`, `tanggal`, `status`FROM `serangan` WHERE status ='0'"


 $Result = ModMain.$Con.Exec($Query)
 tbVDetect.Rows.Count = $Result.Count + 1
 IF $Result.Count > 0 THEN
   $countDetect = $Result.Count
   FOR $i = 0 TO $Result.Count - 1
      tbVDetect[$i + 1, 0].Text = $i + 1
      tbVDetect[$i + 1, 1].Text = $Result!tanggal
      tbVDetect[$i + 1, 2].Text = $Result!ips
      tbVDetect[$i + 1, 3].Text = $Result!nama
      $Result.MoveNext
   NEXT
   $ipx = tbVDetect[$Result.Count, 2].Text
   IF ModMain.$Autos = TRUE THEN
   $Query = "SELECT  * FROM `snort`.`iptable` WHERE `ips` ='" & $ipx & "'"
   $Result = ModMain.$Con.Exec($Query)

   IF $Result.Count = 0 THEN
     SHELL "iptables -A INPUT -s " & $ipx & " -j DROP"
     logs.Add(logs.Count + 1, Time & " : Auto Drop Command")
     $Query = "INSERT INTO `snort`.`iptable` (`ips`,`status`)VALUES ('" & $ipx & "','DROP' )"
     $Result = MODMain.$Con.Exec($Query)
     $Query = "Update `serangan` set status ='1' where `ips`='" & $ipx & "'"
     $Result = ModMain.$Con.Exec($Query)

   ELSE
     SHELL "iptables -A INPUT -s " & $ipx & " -j DROP"
     $Query = "Update `serangan` set status ='1' where `ips`='" & $ipx & "'"
     logs.Add(logs.Count + 1, Time & " : Auto Drop Command")

    ENDIF
   ENDIF
 ENDIF

 $Query = "SELECT `id_serangan`, `nama`, `ips`, `tanggal`, `status`FROM `serangan` WHERE status ='1'"

 $Result = ModMain.$Con.Exec($Query)
 tbVDrop.Rows.Count = $Result.Count + 1
 IF $Result.Count > 0 THEN
   $countDrop = $Result.Count
   FOR $i = 0 TO $Result.Count - 1
```

```
        tbVDrop[$i + 1, 0].Text = $i + 1
        tbVDrop[$i + 1, 1].Text = $Result!tanggal
        tbVDrop[$i + 1, 2].Text = $Result!ips
        tbVDrop[$i + 1, 3].Text = $Result!nama
        $Result.MoveNext
    NEXT
  ENDIF
END


PUBLIC SUB addLogs(isi AS String)
  logs.Add(logs.Count + 1, Time & " : " & isi)
END


PUBLIC SUB Timer1_Timer()
 DIM $ip AS String
 DIM $isOk AS Boolean
 DIM $i AS Integer
 DIM $h AS Long
 IF ($cStatus = "1") THEN
  $cStatus = "0"
  sendStatus("XXX")
 ENDIF
 IF MODMain.$Status = TRUE THEN
  cekCount
  $Query = "SELECT * FROM `inbox` WHERE Mid( `ReceivingDateTime` , 1, 10 ) = mid( current_date, 1,
10 ) ORDER BY `inbox`.`ReceivingDateTime` DESC LIMIT 1 "
  $Result = MODMain.$Con.Exec($Query)

  IF $Result.Count > 0 THEN
    $Str = Trim($Result!TextDecoded)
    '$Query = "select ips from `report`  ORDER BY waktu DESC LIMIT 0,1"
    '$Result = MODMain.$Con.Exec($Query)

    $isOk = FALSE
    IF Mid($Str, 2, 1) = " " THEN
      $isOk = TRUE
    ENDIF


    IF $isOk = FALSE THEN
       $Query = "INSERT INTO `snort`.`outbox` (`DestinationNumber`,`TextDecoded`)VALUES ('" &
MODMain.$HP & "','Perintah Tidak Dikenali' )"
       $Result = MODMain.$Con.Exec($Query)
       logs.Add(logs.Count + 1, Time & " : Recieve BAD Command")
       $Query = "TRUNCATE TABLE `inbox` "
      $Result = MODMain.$Con.Exec($Query)

      $Query = "TRUNCATE TABLE `report` "
      $Result = MODMain.$Con.Exec($Query)

    ELSE
      '$ip = Trim($Result!ips)
      $ip = Lower(Trim(Mid($Str, 3)))
```

```
IF $Str <> "" THEN
SELECT Mid($Str, 1, 1)
CASE "1"
IF ($ip = "yes") THEN
    ModMain.$Autos = TRUE

 ELSE
    ModMain.$Autos = FALSE
ENDIF
 logs.Add(logs.Count + 1, Time & " : Recieve Auto Drop Command " & $ip)


 getData()
CASE "2"

   $Query = "SELECT  * FROM `snort`.`iptable` WHERE `ips` ='" & $ip & "'"
   $Result = ModMain.$Con.Exec($Query)

  IF $Result.Count = 0 THEN
    SHELL "iptables -A INPUT -s " & $ip & " -j DROP"
    logs.Add(logs.Count + 1, Time & " : Recieve Drop Command")
    $Query = "INSERT INTO `snort`.`iptable` (`ips`,`status`)VALUES ('" & $ip & "','DROP' )"
    $Result = MODMain.$Con.Exec($Query)
    $Query = "Update `serangan` set status ='1' where `ips`='" & $ip & "'"
    $Result = ModMain.$Con.Exec($Query)

  ELSE
    SHELL "iptables -A INPUT -s " & $ip & " -j DROP"
    $Query = "Update `serangan` set status ='1' where `ips`='" & $ip & "'"
    $Result = ModMain.$Con.Exec($Query)
    logs.Add(logs.Count + 1, Time & " : Recieve Drop Command")

  ENDIF

  CASE "3"
   $Query = "SELECT  * FROM `snort`.`iptable` WHERE `ips` ='" & $ip & "'"
   $Result = ModMain.$Con.Exec($Query)

  IF $Result.Count = 0 THEN
    SHELL "iptables -F"
    SHELL "iptables -A INPUT -s " & $ip & " -j ACCEPT"
    logs.Add(logs.Count + 1, Time & " : Recieve Accept Command")
    $Query = "INSERT INTO `snort`.`iptable` (`ips`,`status`)VALUES ('" & $ip & "','ACCEPT' )"
    $Result = MODMain.$Con.Exec($Query)

  ELSE
    SHELL "iptables -F"
    SHELL "iptables -A INPUT -s " & $ip & " -j ACCEPT"
    $Query = "Update `serangan` set status ='1' where `ips`='" & $ip & "'"
    $Result = ModMain.$Con.Exec($Query)
    logs.Add(logs.Count + 1, Time & " : Recieve Accept Command")

  ENDIF

CASE "4"
 SHELL "service apache2 " & $ip
 logs.Add(logs.Count + 1, Time & " : Recieve HTTP " & $ip & " Commad")
```

```
    CASE "5"
      SHELL "service vsftpd " & $ip
      logs.Add(logs.Count + 1, Time & " : Recieve FTP " & $ip & " Command")
    CASE "6"
     ' SHELL $ip & " -status"
     'GetBash("service " & $ip & " status")
      SHELL "service " & $ip & " status > /log.log"
      FOR $i = 0 TO 10000000
        $h = $h * $i
      NEXT
      Load("/log.log")
      'ScStatus = "1"

    CASE "7"
       IF ModMain.$Autos = TRUE THEN
        $Query = "INSERT INTO `snort`.`outbox` (`DestinationNumber`,`TextDecoded`)VALUES ('" &
MODMain.$HP & "','Status Aktif' )"
        $Result = MODMain.$Con.Exec($Query)
       ELSE
        $Query = "INSERT INTO `snort`.`outbox` (`DestinationNumber`,`TextDecoded`)VALUES ('" &
MODMain.$HP & "','Status Tidak Aktif' )"
        $Result = MODMain.$Con.Exec($Query)
       ENDIF

      logs.Add(logs.Count + 1, Time & " : Recieve Command Auto Response " & $ip)
    CASE ELSE
      $Query = "INSERT INTO `snort`.`outbox` (`DestinationNumber`,`TextDecoded`)VALUES ('" &
MODMain.$HP & "','Perintah Tidak Dikenali' )"
      $Result = MODMain.$Con.Exec($Query)
      logs.Add(logs.Count + 1, Time & " : Recieve BAD Command")
    END SELECT

   ENDIF
  $Query = "TRUNCATE TABLE `inbox` "
   $Result = MODMain.$Con.Exec($Query)

   $Query = "TRUNCATE TABLE `report` "
   $Result = MODMain.$Con.Exec($Query)

  ENDIF
 ENDIF
ENDIF
END

PUBLIC SUB ToggleButton1_Click()
 '$Query = "INSERT INTO `snort`.`outbox` (`DestinationNumber`,`TextDecoded`)VALUES ('" &
hpadmin.Text & "','konfirmasi' )"
 '$Result = MODMain.$Con.Exec($Query)
 'logs.Add(logs.Count + 1, Time & " : Send Messages")
END

PUBLIC SUB status_Click()

END

PUBLIC SUB Button1_Click()
 GetBash("service apache2 status")
```

```
 END

 PUBLIC SUB BtnSetting_Click()

   FormKonfig.ShowDialog()

END

PUBLIC SUB Button2_Click()

END

PUBLIC SUB BtnStop_Click()
 IF BtnStop.Text = "Start" THEN
  ModMain.$Status = FALSE
  BtnStop.Text = "Stop"
 ELSE
  ModMain.$Status = TRUE
  BtnStop.Text = "Start"
 ENDIF

END

PUBLIC SUB BtnAbout_Click()

 Message.Info("hh")

END

PRIVATE $hProcess AS Process
PRIVATE $sText AS String

PUBLIC SUB Process_Read()

 DIM sStr AS String

   READ #LAST, sStr, -256

   $sText = $sText & sStr

 UpdateConsole

END


PUBLIC SUB Process_Error(sStr AS String)

 $sText = $sText & sStr
 UpdateConsole

END
PRIVATE SUB sendStatus(statuss AS String)
DIM $ip AS String
DIM $isOk AS Boolean
   $Query = "SELECT * FROM `inbox` WHERE Mid( `ReceivingDateTime` , 1, 10 ) = mid( current_date,
1, 10 ) ORDER BY `inbox`.`ReceivingDateTime` DESC LIMIT 1 "
```

```
    $Result = MODMain.$Con.Exec($Query)

    IF $Result.Count > 0 THEN
     $Str = Trim($Result!TextDecoded)

     $isOk = FALSE
     IF Mid($Str, 2, 1) = " " THEN
      $isOk = TRUE
     ENDIF
     IF $isOk THEN
      $ip = Lower(Trim(Mid($Str, 3)))

      logs.Add(logs.Count + 1, Time & " : Recieve Cek " & $ip & " Status Command")
      $Query = "INSERT INTO `snort`.`outbox` (`DestinationNumber`,`TextDecoded`)VALUES ('" &
MODMain.$HP & "','" & $ip & " Status : " & statuss & "')"
      $Result = MODMain.$Con.Exec($Query)
     ELSE

     ENDIF
    ENDIF


END


PRIVATE SUB UpdateConsole()

 DIM iPos AS Integer
 DIM sStr AS String
 DIM $ip AS String
 DIM $isOk AS Boolean
 DO

  iPos = InStr($sText, "\n")
  IF iPos = 0 THEN RETURN

  sStr = Normalize(Left$($sText, iPos))
  $sText = Mid$($sText, iPos + 1)

  txtConsole.Pos = txtConsole.Length
  txtConsole.Insert(sStr)
  IF ($cStatus = "1") THEN

    tStatus.Text = sStr

    $cStatus = "0"
  ENDIF



 LOOP

END


STATIC PRIVATE FUNCTION Normalize(sStr AS String) AS String
```

```
  DIM sNorm AS String
  DIM iInd AS Integer
  DIM iCar AS Integer
  DIM bEsc AS Boolean

  FOR iInd = 1 TO Len(sStr)

   iCar = Asc(sStr, iInd)

   IF iCar = 27 THEN
    bEsc = TRUE
    CONTINUE
   ENDIF

   IF bEsc THEN
    IF iCar < 32 THEN bEsc = FALSE
    CONTINUE
   ENDIF

   IF iCar < 32 AND iCar <> 10 THEN iCar = 32

   sNorm = sNorm & Chr$(iCar)

  NEXT

  RETURN Conv$(sNorm, System.Charset, Desktop.Charset)

END

PUBLIC SUB Form_Close()
  $hProcess.Kill
END
PUBLIC SUB GetBash(command AS String)
  DIM sLig AS String
  sLig = command & gb.NewLine
  txtConsole.Insert("# " & sLig)
  sLig = Conv$(sLig, Desktop.Charset, System.Charset)
  PRINT #$hProcess, sLig;

END

PUBLIC SUB Load(sPath AS String)
  DIM sData AS String

  sData = File.Load(sPath)
  tStatus.Text = Conv(sData, System.Charset, Desktop.Charset)
' Message.Info(sData)

CATCH
  Message.Error(sPath & "\nUnable to load file.\n" & Error.Text)
END

PUBLIC SUB BtnVSnort_Click()
  GetBash("snort -v -c /etc/snort/snort.conf")
END
```

```
PUBLIC SUB BtnVDevice_Click()
  GetBash("gammu --identify")
END

PUBLIC SUB tStatus_Change()


DIM $ip AS String
DIM $isOk AS Boolean

   $Query = "SELECT * FROM `inbox` WHERE Mid( `ReceivingDateTime` , 1, 10 ) = mid( current_date,
1, 10 ) ORDER BY `inbox`.`ReceivingDateTime` DESC LIMIT 1 "
    $Result = MODMain.$Con.Exec($Query)


     $isOk = FALSE
     IF Mid($Str, 2, 1) = " " THEN
      $isOk = TRUE
     ENDIF
     IF $isOk THEN
      $ip = Lower(Trim(Mid($Str, 3)))

      logs.Add(logs.Count + 1, Time & " : Recieve Cek " & $ip & " Status Command")
      $Query = "INSERT INTO `snort`.`outbox` (`DestinationNumber`,`TextDecoded`)VALUES ('" &
MODMain.$HP & "'," & $ip & " Status : " & tStatus.Text & "')"
      $Result = MODMain.$Con.Exec($Query)
     ELSE

     ENDIF
    ENDIF

END
```