Perancangan Dan Analisis Keamanan Jaringan Nirkabel Menggunakan *Wireless Intrusion Detection System*

Randy Mentang, Alicia A. E. Sinsuw, ST., MT., Xaverius B. N. Najoan, ST., MT.
Jurusan Teknik Elektro-FT, UNSRAT, Manado-95115, Email: gnatnemydnar@gmail.com

Abstrak - Sistem keamanan jaringan menjadi hal yang sangat penting dalam menjaga sebuah jaringan, serangan yang bisa mengganggu bahkan merusak sistem koneksi antar perangkat yang terhubung akan sangat merugikan. Untuk mendapatkan keamanan dalam sebuah jaringan terkadang kita harus merasakan ketidak nyamanan dalam penggunaannya, hal inilah yang seringkali menjadi pertimbangan dalam penerapan sebuah sistem keamanan jaringan.

Metode WIDS (*Wireless Intrusion Detection System)* mampu mendeteksi serangan DOS (*Denial of Service)*. Melakukan penerapan pada sistem operasi *Linux* menggunakan *Snort* sebagai mesin sensor dan *Iptables* sebagai penanganan serangan dapat menjadi solusi keamanan jaringan nirkabel dari serangan yang dapat mengancam.

Konfigurasi sistem dibangun dalam jaringan WAN (*Wide Area Network*) yang dirancang untuk merepresentasikan pengujian. Hasil analisis dari setiap pengujian yang dilakukan menyimpulkan bahwa setiap tindakan yang dilakukan oleh penyerang terhadap jaringan dapat diketahui, sehingga dapat dilakukan penanganan sebelum terjadi kerusakan lebih luas.

Kata Kunci : *Denial of Service, Iptables*, Keamanan Jaringan, *Linux, Snort, Wireless Intrusion Detection System*

Abstract - Network security system becomes very important in maintaining a network, attacks that can disrupt and even destroy the system of connections between the connected devices will be very detrimental. To get the security in a network sometimes we have to feel discomfort in use, this is often a consideration in the implementation of a network security system.

Methods WIDS (Wireless Intrusion Detection System) is able to detect a DOS attack (Denial of Service). Do application on the Linux operating system using Snort as engine sensors and iptables as the handling of the attack can be a solution to the security of wireless networks from attacks that can threaten.

System configuration built in WAN networks that are designed to represent testing. Analytical results from any tests performed concluded that any actions taken by attackers on the network can be known, so that the treatment can be done before more extensive damage

Keywords : Denial of Service, Iptables, Linux, Network Security, Snort, Wireless Intrusion Detection System

I. PENDAHULUAN

Jaringan komputer adalah sekelompok komputer otonom yang saling berhubungan antara satu dengan lainnya, dengan menggunakan protokol komunikasi melalui media komunikasi sehingga dapat saling berbagi informasi, program-program, penggunaan bersama perangkat keras seperti printer, hardisk, dan sebagainya. Selain itu, jaringan komputer dapat diartikan sebagai kumpulan sejumlah terminal komunikasi yang berada diberbagai lokasi yang terdiri dari lebih satu komputer yang saling berhubungan.

Teknologi *wireless* (tanpa kabel / nirkabel) saat ini berkembang sangat pesat terutama dengan hadirnya perangkat teknologi informasi dan komunikasi. Komputer, *Notebook*, telepon seluler dan periperalnya mendominasi pemakaian teknologi *wireless*. Penggunaan teknologi *wireless* yang diimplementasikan dalam suatu jaringan lokal sering dinamakan *WLAN* (*Wireless Local Area Network*). Namun perkembangan teknologi *wireless* yang terus berkembang sehingga terdapat istilah yang mendampingi *WLAN* seperti *WMAN* (*Metropolitan*), *WWAN* (*Wide*) dan *WPAN* (Personal / *Private*). Keamanan jaringan adalah proses untuk melindungi sistem dalam jaringan dengan mendeteksi penggunaan yang berhak dalam jaringan. Pengelolahan terhadap pengendalian keamanan jaringan dapat dilihat dari sisi pengelolahan resiko (*risk management*). *Intrusion Detection System* (*IDS*) dapat didefinisikan sebagai *tool*, metode, sumber daya yang memberikan bantuan untuk melakukan identifikasi, memberikan laporan terhadap aktifitas jaringan komputer. *Intrusion Detection System* secara khusus berfungsi sebagai proteksi secara keseluruhan dari sistem yang telah di instal *IDS*. *IDS* tidak berdiri sendiri dalam melindungi suatu sistem.

II. LANDASAN TEORI

A. *Konsep dan Arsitektur Jaringan*

Komputer adalah sistem elektronik untuk memanipulasi data secara tepat dan cepat. (Kuswayatno, 2006:24) Kata komputer semula dipergunakan untuk menggambarkan orang yang pekerjaannya malakukan perhitungan aritmatika dengan atau tanpa alat bantu, tetapi arti kata ini kemudian dipindahkan kepada mesin itu sendiri. Asal mulanya, pengolahan informasi hampir ekslusif berhubugan dengan masalah aritmatika, tetapi komputer moderen digunakan untuk banyak tugas yang tidak berkaitan dengan aritmatika.

Jaringan komputer merupakan gabungan antara teknologi komputer dan teknologi komunikasi. (Sopandi, 2006:5) Gabungan teknologi ini melahirkan pengolahan data yang dapat didistribusikan, mencakup pamakaian *database*, *software* aplikasi dan aplikasi peralatan *hardware* secara bersamaan, untuk membantu proses otomatisasi perkantoran dan peningkatan kearah efisiensi kerja. Tujuan dari jaringan komputer adalah membagi sumber daya, seperti printer, *CPU* (*Central Processing Unit*), memori, *Harddisk* dan Komunikasi, contohnya surat elektronik, *instant messaging*, *chatting* serta akses informasi, contohnya *web browsing*

B. *Keamanan Jaringan*

Jaringan komputer adalah sekelompok komputer otonom yang saling berhubungan dengan lainnya menggunakan protokol komunikasi melalui media komunikasi sehingga dapat saling berbagi informasi, aplikasi dan perangkat keras secara bersama-sama. Jaringan komputer dapat diartikan juga sebagai kumpulan sejumlah terminal komunikasi yang berada di berbagai lokasi yang terdiri lebih dari satu komputer yang saling berhubungan. (Sukmaaji&Rianto, 2008:1)

Keamanan dalam jaringan komputer sangat penting dilakukan untuk memonitor akses jaringan dan mencegah penyalahgunaan sumber daya jaringan yang tidak sah. Tugas keamanan jaringan dikontrol oleh administrator jaringan. Segi-segi keamanan didefinisikan lima poin, yaitu *Confidentiality*, Mensyaratkan bahwa informasi (data) hanya bisa diakses oleh pihak yang memiliki wewenang, *Integrity*, Mensyaratkan bahwa informasi hanya dapat diubah oleh pihak yang memiliki wewenang, Availability, Mensyaratkan bahwa informasi tersedia untuk pihak yang memiliki wewenang ketika dibutuhkan, *Authentication*, Mensyaratkan bahwa pengirim suatu informasi dapat diidentifikasi dengan benar dan ada jaminan bahwa identitas yang didapat tidak palsu, *Nonrepudiation*, Mensyaratkan bahwa baik pengirim maupun penerima informasi tidak dapat menyangkal pengiriman dan penerimaan pesan.

Serangan (gangguan) terhadap keamanan dapat dikategorikan dalam empat kategori utama, yaitu *Interruption*, Suatu aset dari suatu sistem diserang sehingga menjadi tidak tersedia atau tidak dapat dipakai oleh yang berwenang. Contohnya adalah perusakan / modifikasi terhadap piranti keras atau saluran jaringan, Interception, Suatu pihak yang tidak berwenang mendapatkan akses pada suatu aset. Pihak yang dimaksud bisa berupa orang, program, atau sistem yang lain. Contohnya adalah penyadapan terhadap data dalam suatu jaringan, *Modification*, Suatu pihak yang tidak berwenang dapat melakukan perubahan terhadap suatu aset. Contohnya adalah perubahan nilai pada file data, modifikasi program sehingga berjalan dengan tidak semestinya, dan modifikasi pesan yang sedang ditransmisikan dalam jaringan, *Fabrication*, Suatu pihak yang tidak berwenang menyisipkan objek palsu ke dalam sistem. Contohnya adalah pengiriman pesan palsu kepada orang lain.

Penggunaan sistem jaringan komputer dalam skala kecil maupun luas akan membutuhkan pengaturan-pengaturan mulai dari tingkat fisik maupun non fisik. Pengaturan-pengaturan tersebut melibatkan proses pengontrolan. Ada beberapa definisi mengenai administrasi jaringan ini antara lain : *Controlling corporate strategic* (*assets*), *Controlling complekxity*, *Improving service*, *Balancing various needs*, *Reducing downtime* dan *Controlling costs*.

Pemahaman tentang routing teori maupun konfigurasi harus di kuasai dengan baik agar mampu membangun jaringan dengan baik hal ini sangat diperlukan terutama jika komputer ataupun sub organisasi perusahaan sangat banyak. Pengetahuan tentang sistem keamanan komputer terutama jaringannya (*network security*) akan sangat membantu dan memberikan nilai lebih. Selain kemampuan teori maupun praktek yang harus dikuasai dengan baik hal lain adalah memiliki etika profesional, tanpa etika dan sikap seorang profesional yang baik maka semua kemampuan teori maupun praktek yang dikuasai tidak akan berarti banyak.

Keamanan jaringan secara umum adalah komputer yang terhubung ke *network*, mempunyai ancaman keamanan lebih besar dari pada komputer yang tidak terhubung kemana-mana. Dengan pengendalian yang teliti, resiko tersebut dapat dikurangi, namun *network security* biasanya bertentangan dengan *network access*, dimana bila *network access* semakin mudah, maka *network security* semakin rawan, begitu pula sebaliknya. (Ariyus, 2007:3)

C. *Snort*

Snort merupakan suatu perangkat lunak untuk mendeteksi penyusup dan mempu menganalisis paket

yang melintasi jaringan secara *real time traffic* dan *logging* kedalam *database* serta mampu mengidentifikasi berbagai serangan yang berasal dari luar jaringan. (Ariyus, 2007:45) Program *snort* dapat dioprasikan dengan tiga mode, (Ariyus, 2007:146) yaitu Paket *Sniffer* yang berfungsi untuk melihat paket yang lewat di jaringan, Paket *Logger* yang berfungsi untuk mencatat semua paket yang lewat di jaringan untuk dianalisis di kemudian hari dan NIDS (*Network Intrusion Detection System*), pada mode ini *snort* akan berfungsi untuk mendeteksi serangan yang dilakukan melalui jaringan komputer

Fitur-fitur *Snort* adalah sebagai berikut : (Slameto 2007:7) Karena *Snort* bersifat *opensource*, maka penggunaannya betul-betul gratis. Oleh karena itu, *Snort* merupakan pilihan yang sangat baik sebagai NIDS ringan yang *cost-effective* dalam suatu organisasi yang kecil jika organisasi tersebut tidak mampu menggunakan NIDS. Karena *Snort* bersifat *opensource*, maka penggunaannya bebas, sehingga dapat diterapkan dalam lingkungan apa saja. Kode sumbernya bisa didapatkan sehingga *Snort* boleh secara bebas di modifikasi sendiri sesuai keperluan. Selain itu, karena *Snort* merupakan *software* yang bebas, maka telah terbentuk suatu komunitas *Snort* yang membantu memberikan berbagai macam dukungan untuk penggunaan, pengembangan, penyempurnaan dan perawatan *software Snort* itu. *Snort* memiliki bahasa pembuatan *rules* yang relatif mudah dipelajari dan fleksibel. Ini berarti bahwa pengguna dapat dengan mudah dan cepat membuat berbagai *rules* baru untuk mendeteksi tipe-tipe serangan yang baru. Selain itu, berbagai *rules* khusus dapat dibuat untuk segala macam situasi. *Snort* sudah memiliki sebuah *database* untuk berbagai macam *rules* dan *database* ini secara aktif terus dikembangkan oleh komunitas *Snort* sehingga tipe-tipe serangan yang baru dapat dideteksi dan dicatat.

Snort merupakan *software* yang ringkas dan padat, sehingga tidak memakan banyak *resources* tetapi cukup canggih dan fleksibel untuk digunakan sebagai salah satu bagian dari NIDS yang terpadu (*Integrated* NIDS). *Snort* dapat melakukan *logging* langsung ke sistem *database* (MySQL). *Snort* sebagai NIDS dapat menyembunyikan dirinya dalam jaringan komputer sehingga keberadaannya tidak bisa terdeteksi oleh komputer manapun. Ini disebut sebagai *stealth mode*.

Komponen-komponen Snort meliputi : *Rule Snort*, merupakan *database* yang berisi pola-pola serangan berupa *signature* jenis-jenis serangan. *Rule Snort* IDS ini harus di *update* secara rutin agar ketika ada suatu teknik seragan yang baru. *Snort Engine*, merupakan program yang berjalan sebagai proses yang selalu bekerja untuk membaca paket data dan kemudian membandingkan dengan *rule snort*. *Alert*, merupakan catatan serangan pada deteksi penyusupan. Jika *snort engine* menghukumi paket data yang lewat sebagai serangan, maka *snort engine* akan mengirimkan *alert* berupa *log file*. Untuk kebutuhan analisa, *alert* dapat disimpan di dalam *database* sebagai contoh ACID (*Analysis Console for Intrusion Datebase*) sebagai modul tambahan pada snort

D. BASE (Basic Analysis and Security Engine)

BASE adalah sebuah *interface* web untuk melakukan analisis dari intrusi yang *snort* telah deteksi pada jaringan. (Orebaugh, 2008:217) BASE ditulis oleh Kevin Johnson adalah program *analisis* sistem jaringan berbasis PHP yang mencari dan memproses *database* dari *security event* yang dihasilkan oleh berbagai program monitoring jaringan, *firewall* atau sensor IDS. (Kuhlenberg, 2007:424). Ini menggunakan otentifikasi pengguna dan sistem peran dasar, sehingga sebagai admin keamanan dapat memutuskan apa dan berapa banyak informasi yang setiap pengguna dapat melihat.

BASE adalah sistem manajemen basis data berbasis desktop yang lengkap, didesain untuk memenuhi kebutuhan yang luas dari pengguna, mulai dari melacak koleksi CD pribadi Anda, hingga menghasilkan laporan penjualan bulanan BASE menawarkan panduan untuk membantu pengguna yang baru terhadap desain basis data (baru terhadap BASE) untuk membuat Tabel, *Query*, *Form*, dan *Report*, bersama dengan sekumpulan definisi tabel yang sudah didefinisikan untuk melacak *Asset*, Konsumen, Penjualan, *Invoice*, dan banyak lagi.

Ketika Anda hanya memerlukan basis data personal, BASE menawarkan mesin basis data relasional HSQL, dikonfigurasi untuk pengguna tunggal, dengan data tersimpan pada dokumen BASE, beserta dengan dukungan *native* untuk dokumen BASE. Untuk kebutuhan yang lebih besar, BASE mendukung berbagai basis data yang populer secara *native* : MySQL, Adabas D, *Microsoft Access*, dan *Postgre* SQL. Sebagai tambahan, dukungan untuk *driver* standar JDBC dan ODBC juga mengijinkan Anda untuk terhubung secara virtual pada sembarang basis data yang ada.

E. DOS (Denial Of Service)

DoS attack adalah jenis serangan tehadap sebuah komputer atau *server* atau *router* atau mesin didalam jaringan internet dengan cara menghabiskan *resource* yang dimiliki oleh komputer tersebut sampai komputer tersebut tidak dapat menjalankan fungsinya dengan benar, sehingga secara tidak langsung mencegah pengguna lain untuk memperoleh akses layanan dari komputer yang di serang (*server*) tersebut.

DoS (*Denial of Service*) bertujuan untuk mencegah pengguna mendapatkan layanan dari sistem. Serangan

DoS dapat terjadi dalam banyak bentuk. Penyerangan dapat membanjiri jaringan dengan data yang sangat besar atau dengan sengaja menghabiskan sumber daya yang memang terbatas, seperti *proces control block* atau *pending network connection*. Penyerang juga mungkin saja mengacaukan komponen fisik dari jaringan atau memanipulasi data yang sedang dikirim termasuk data yang terenkripsi.

Ciri-ciri Serangan DOS awalnya terjadi pada jaringan dimana *hacker* atau *cracker* mencoba mengeksploitasi kelemahan *protocol* TCP (*Transmision Control Protocol*) inilah yang disebut dengan *SYN Flooding Attack*, kemudian seiring perjalanan diciptakan juga serangan-serangan untuk eksploitas kelemahan Sistem Operasi, layanan jaringan atau Aplikasi sistem bahkan *tools* yang digunakan pun semakin banyak bahkan bisa didapatkan secara gratis. Serangan DOS bersifat satu lawan satu sehingga hanya perlu sebuah komputer / *host* yang kuat baik *hardware*, OS dan aplikasi agar mampu membanjiri lalu lintas *host* korban sehingga mencegah user resmi atau valid untuk mengakses *server*. Sedangkan Serangan DDoS (*Distributed Denial Of Service*) ini menggunakan teknik yang lebih canggih dibandingkan dengan serangan DOS, yakni meningkatkan serangan dengan menggunakan beberapa buah komputer sekaligus, sehingga dapat mengakibatkan *server* atau keseluruhan segmen jaringan dapat menjadi mati total.

DDOS bisa dikatakan adalah hal yang sederhana tetapi dampaknya bisa sangat kritis dan sangat merepotkan administrator jaringan dalam melakukan perbaikan.

Menjalankan *tools* yang secara otomatis akan melakukan scanning jaringan untuk menemukan *host-host* yang memiliki celah (*vulnerable*). Setelah celah / kelemahan *host* ditemukan, *tool* tersebut dapat menginstalasikan salah satu jenis dari *Trojan Horse* yang disebut sebagai DDoS *Trojan*, yang akan mengakibatkan *host* tersebut dapat dikontrol dari jarak jauh oleh sebuah komputer master yang digunakan oleh *hacker* atau *cracker* untuk melancarkan serangan.

Denial of Service adalah tipe serangan yang sangat powerful untuk "melukai" infrastruktur dari suatu organisasi jaringan. Serangan ini termasuk tipikal serangan yang mematikan dan berbahaya, Serangan DOS tujuanya adalah satu membuat server tidak bisa menerima layanan untuk user yang sah dan pada akhirnya *service* nya *down*. DDOS adalah *Distributed Denial of Service*, maksudnya adalah komputer yang digunakan untuk melakukan serangan lebih dari satu, dengan tujuan satu. Untuk melakukan DDOS biasanya dipakai komputer yang di namai dengan *zombie network*, *zombie* disini mengacu pada grup dari komputer yang terinfeksi yang kemudian diberi perintah menyerang satu sasaran. Secara singkat tipe DDOS ada 3 yaitu : *Applicaton Layer DDOS Attack, Protocol DOS Attack, Volume Based DDOS Attack.* Untuk melakukan serangan DOS umumnya menggunakan aplikasi yang menjadi pelontar serangan tersebut, salah satu yang terkenal adalah LOIC (*Low Orbit Ion Canon*), tool ini digunakan oleh grup *hacker Anonymous* untuk melakukan serangannya.

F. Digital Blaster

Digital blaster adalah sebuah *Flooder internet* dan jaringan komputer yang bisa didapatkan melalui beberapa media seperti CD/DVD, maupun disitus internet. Digital blaster disingkat menjadi DigiBlast merupakan *hack tool* gratis dan bebas untuk disebarluaskan dengan syarat tidak untuk konsumsi profit seperti menjual atau membelinya dari seseorang. Prinsip kerja program ini adalah mengirimkan paket secara berkala kesebuah alamat IP dan *port-port* yang ditentukan. DigiBlast dapat mengirimkan paket ke alamat IP target ke sebuah *port* yang ditentukan oleh *user* (*Singel Port Flooder*) maupun ke banyak *port* yang terbuka (*Multi Port Flooder*). Yang perlu memastikan bahwa alamat IP target aktif dan terhubung ke internet.

G. IP Tables

IPTables adalah *firewall* yang secara *default* diinstal pada semua distribusi *linux*, seperti Ubuntu, Fedora dan lainnya. Pada saat melakukan instalasi pada *linux*, *iptables* sudah langsung ter-install, tetapi pada umumnya *iptables* mengizinkan semua *traffic* untuk lewat. (Purbo, 2008:188)

Iptables adalah suatu *tools* dalam sistem operasi *linux* yang berfungsi sebagai alat untuk melakukan *filter* (penyaringan) terhadap lalu lintas data. Secara sederhana digambarkan sebagai pengatur lalu lintas data. Dengan *iptables* inilah kita akan mengatur semua lalu lintas dalam komputer kita, baik yang masuk ke komputer, keluar dari komputer, ataupun *traffic* yang sekedar melewati komputer kita.

IPTables memiliki tiga macam daftar aturan bawaan dalam tabel penyaringan, daftar tersebut dinamakan rantai *firewall* atau sering disebut *firewall chain*. Ketiga *chain* tersebut adalah *INPUT*, *OUTPUT* dan *FORWARD*, dan *iptables* juga memiliki tige buah tabel, yaitu *NAT, MANGLE* dan *FILTER*. *IPTables* merupakan salah satu *firewall* popular dan *powerfull* yang tersedia di sistem oprasi *Linux*. Fungsi *IPTables* adalah untuk konfigurasi, merawat dan memeriksa rules tables (tabel aturan) tentang *filter* paket IP yang terdapat di kernel *linux*. Filter berfungsi untuk melakukan penyaringan paket data apakah paket tersebut akan di *DROP, LOG, ACCEPT* atau *REJECT*. Nat berfungsi melakukan *Network Address Translation* yang

merupakan pengganti alamat asal atau tujuan dari paket data. *Mangle* berfungsi untuk melakukan penghalusan paket data seperti *TTL*, *TOS* dan *MARK*. Raw berfungsi untuk mengkonfigurasi pengecualian dari *connection tracking* bersama-sama *NOTRACK*.
Pada tabel terdapat *chains* (rantai) yang berisi *rules* yang berbeda-beda. *Chains* pada *filter table* yaitu, *INPUT* : Untuk paket yang disiapkan untuk soket lokal atau komputer kita sendiri atau untuk mengatasi paket data yang masuk. *FORWARD* : Untuk paket yang diarahkan / *routing* ke box atau untuk mengalihkan paket yang datang. *OUTPUT* : Untuk paket yang *generate* / dibuat sendiri atau untuk menghasilkan paket data yang akan diteruskan.

H. IDS

Intrusion Detection System dapat didefinisikan sebagai *tool*, metode, sumber daya yang memberikan bantuan untuk melakukan identifikasi, memberikan laporan terhadap aktivitas jaringan komputer. (Ariyus, 2007:27)

Intrusion Detection System secara khusus berfungsi sebagai proteksi secara keseluruhan dari sistem yang telah di instal IDS. IDS tidak berdiri sendiri dalam melindungi suatu sistem.

Jenis *Intrusion Detection System* ada 2, yaitu NIDS (*Network Intrusion Detection System*) dan HIDS (*Host Intrusion Detection System*). (Junior, 2009:5) NIDS (*Network Intrusion Detection System*) : akan melakukan pemantauan terhadap seluruh bagian pada jaringan dengan mengumpulkan paket-paket data yang terdapat pada jaringan tersebut serta melakukan analisa dan membentuk apakah paket–paket tersebut merupakan paket normal atau paket serangan. HIDS (*Host Intrusion Detection System*) : hanya melakukan pemantauan pada perangakat komputer tertentu dalam jaringan.

HIDS biasanya akan memantau kajadian seperti kesalahan login berkali-kali dan melakukan penecekan pada file.

III. PERANCANGAN SISTEM

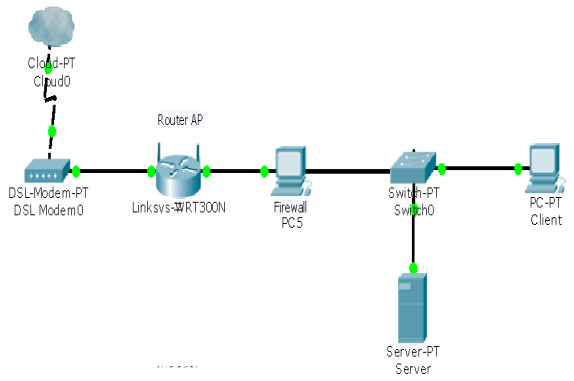
A. Metode Pengembangan Sistem

Dalam proses pengembangan sistem banyak metode atau model yang ada. Pada penelitian akan dibangun sistem *IDS* dimana lingkup pembahasan mengenai jaringan sehingga metode atau model pengembangan sistem yang digunakan dalam penelitian ini adalah *SPDLC* (*Security Policy Development Life Cycle*).

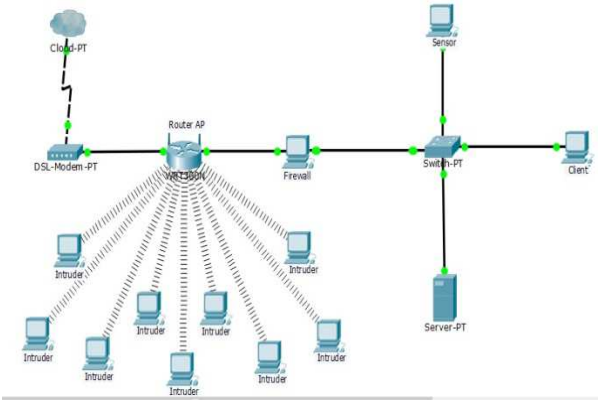
SPDLC adalah metode yang menetapkan strategi untuk melakukan pembaharuan suatu organisasi dari sistem jaringan, siklus pengembangan sistem jaringan didefinisikan pada sejumlah fase . Menurut Luay A. Wahsheh and Jim Alves Foss (2008:1120) Pengembangan sistem *SPDLC* yang diambil melakukan penelitian dalam 5 tahap, yaitu tahap *Analysis* : Pada tahap ini dilakukan perumusan masalah, mengidentifikasi konsep dari IDS, *Ethereal* dan beberapa perangkat jaringan, mengumpulkan data dan mengidentifikasi kebutuhan seluruh kompunen sistem tersebut, sehingga spesifikasi kebutuhan sistem IDS dan *Snort* dapat diperjelas dan perinci. Pada tahapan analisis yang dilakukan adalah *Identification, Understand* dan *Report*. Tahap *Design* : Pada tahap ini yang dilakukan adalah Merancang topologi jaringan untuk simulasi WAN sebagai representasi lingkungan jaringan sebenarnya dan merancang penggunaan sistem operasi dan aplikasi pada *server, client* dan komputer penyusup. Rancangan topologi jaringan dibangun dengan menggunakan *Cisco Packet Tracer* yang di instal baik dalam kondisi sebelum diterapkan IDS yang dapat dilihat pada gambar 1. Dan kondisi pada saat telah diterapkan IDS yang dapat dilihat pada gambar 2. Implementation : Fase selanjutnya adalah implementasi atau penerapan detail rancangan topologi dan rancangan sistem pada lingkungan nyata sebagai simulasi *wireless.* Detail rancangan akan digunakan dapat dilihat pada gambar 3. sebagai intruksi atau panduan tahap implementasi agar sistem yang dibangun dapat relevan dengan sistem yang sudah dirancang. Proses implementasi terdiri dari instalasi dan konfigurasi. Dengan mengumpulkan seluruh perangkat yang dibutuhkan dilaboraturium riset. Tahap *Enforcement* : Setelah tahap implementasi adalah tahap *Enforcement* dimana tahap ini penting. Proses pelaksanaan atau penyelenggaraan dilakuakan melalui aktivitas pengoprasian dan pengamatan sistem yang sudah dibangun dan diterapkan apakah sistem IDS sudah berjalan dengan benar dan baik

B. Alur Metode Penelitian

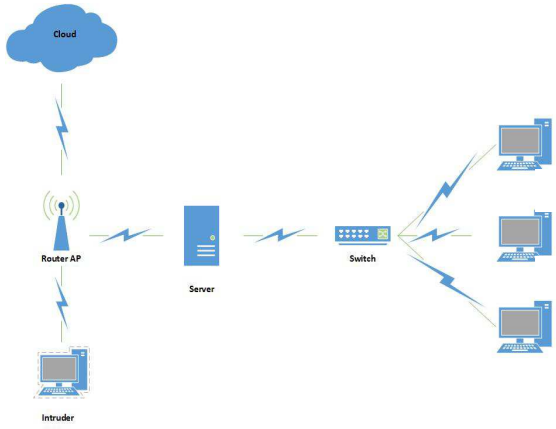
Tahapan dalam penelitian ini telah dicantumkan dalam diagram alur penelitian yang sebelum memasuki dan keluar dari model atau metode pengembang dari SPDLC terdapat beberapa tahap yang harus dilakukan. SPDLC adalah metode yang menetapkan strategi untuk melakukan pembaharuan suatu organisasi dari sistem jaringan yang dapat dilihat pada gambar 4.



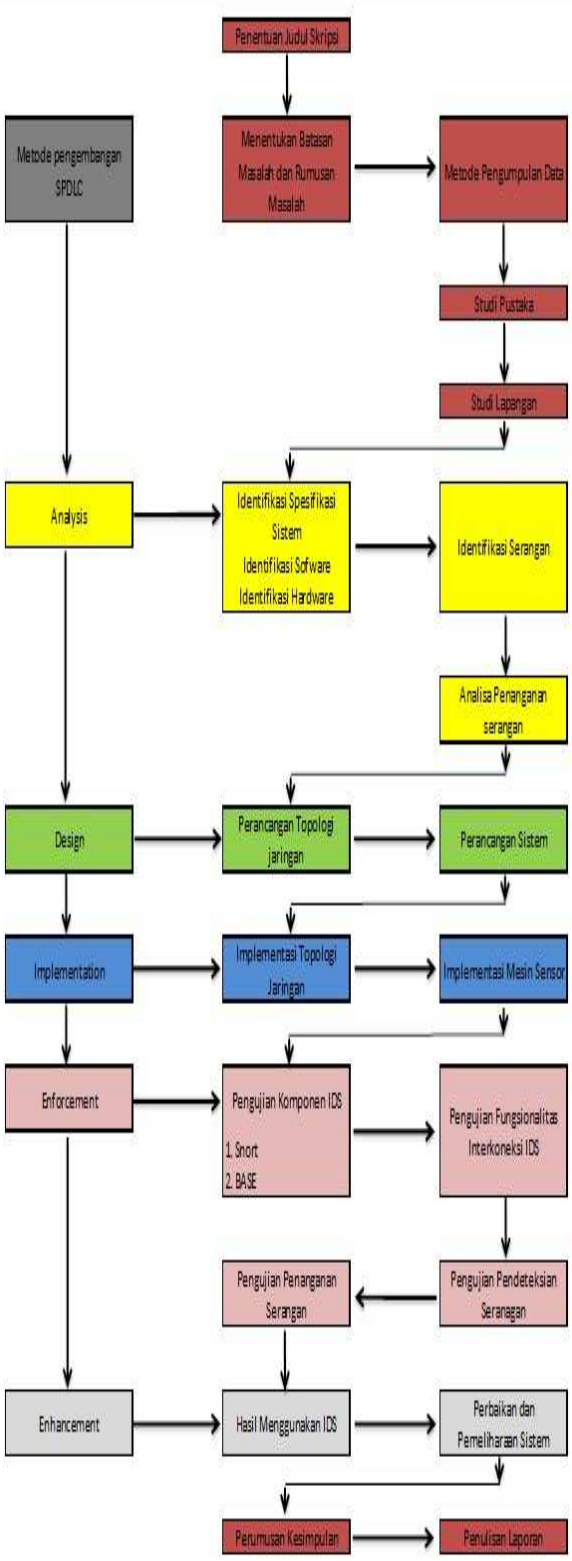
Gambar 1. Topologi Jaringan Sebelum Diterapkan IDS



Gambar 2. Topologi Jaringan Sesudah Diterapkan IDS



Gambar 3. Implementasi Topologi Jaringan



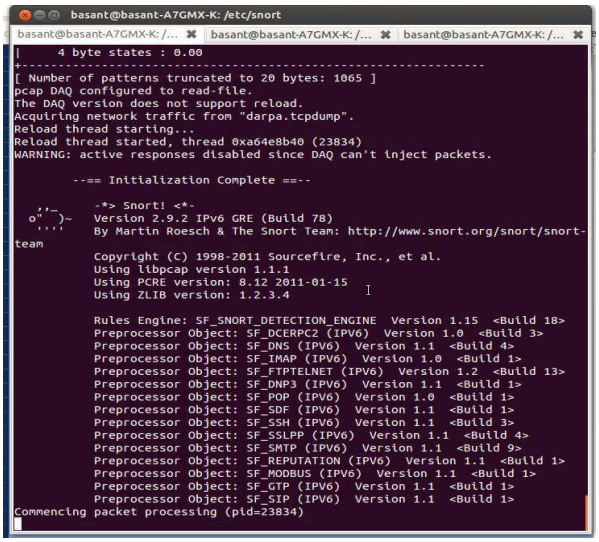
Gambar 4. Tahapan SPDLC

IV. HASIL DAN PEMBAHASAN

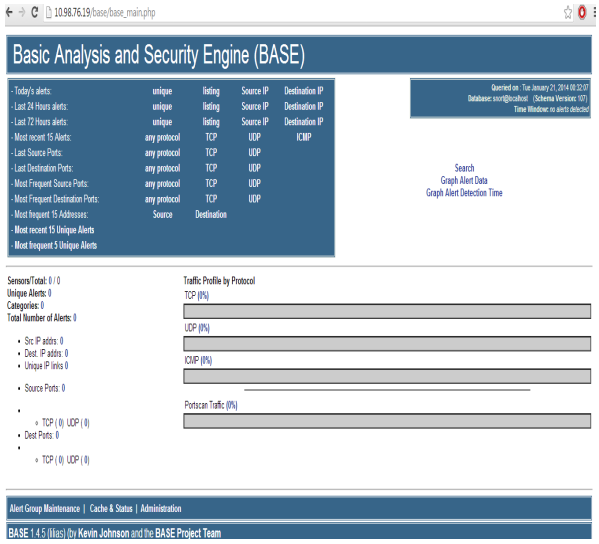
Pada bab ini menjelaskan proses implementasi sistem monitoring keamanan jaringan yang mengintegrasikan *IDS* dan *Snort* yang berbasis *open source*.

A. *Pengujian Snort*

Pengujian *Snort* pada mesin Sensor dilakukan dengan menggunakan rules sederhana (sebagai representasi dari definisi jenis serangan tertentu) dan memastikan *Snort* dapat mendeteksi *rules* tersebut. *Snort* diaktifkan dengan perintah berikut, agar dapat mencetak hasilnya langsung ke layar *console* : *snort -c /etc/snort/snort.conf -i eth0*, yang dapat dilihat di gambar 5.



Gambar 5. Pengujian Fungsi Snort



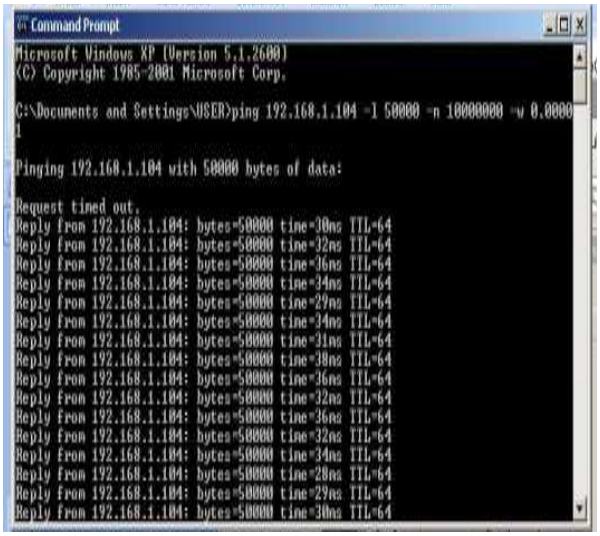
Gambar 6. Pengujian Fungsi BASE

B. *Pengujian BASE*

Pengujian fungsionalitas ACID BASE yang dilakukan dengan mengakses dan mengeksporasi sistem ACID BASE secara keseluruhan. Hasilnya, ACID BASE telah berhasil di implementasikan dan dapat menampilkan event snort ketika bekerja dalam memonitoring setiap kegiatan dalam jaringan yang bisa dilihat di gambar 6.

C. *Fungsionalitas Interkoneksi IDS*

Menggunakan studi kasus untuk menguji sistem IDS dalam melindungi *server*. Pengujian ini direpresentasikan dengan melakukan simulasi serangan *ping attack* dan *digital* blaster yang bisa dilihat pada gambar 7 dan gambar 8.



Gambar 7. Pengujian Serangan Ping Attack

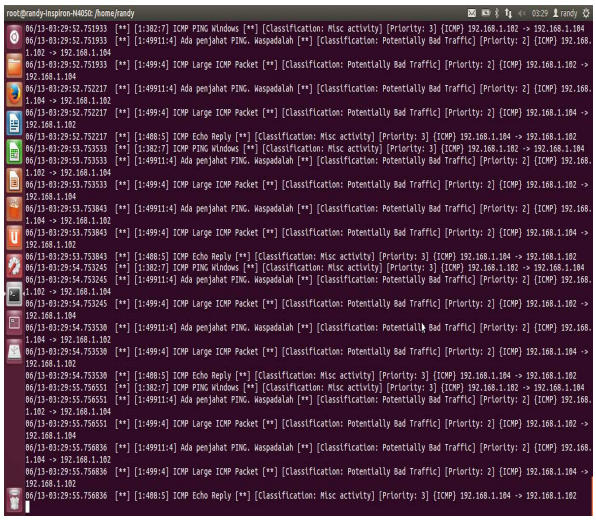
Gambar 8. Pengujian Serangan Digital Blaster

D. Pengujian Snort dan BASE Dalam Pendeteksian

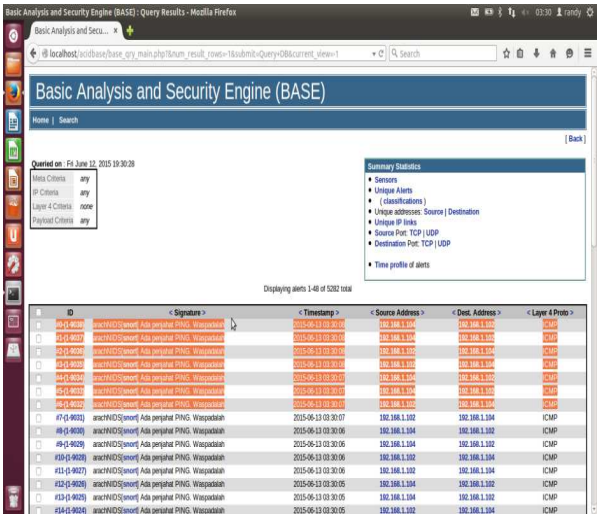
Fase ini menguji keefektifan dari fungsionalitas pendeteksian serangan yang dilakuan dengan proses serangan *ping attack* dan *digital blaster* oleh *snort* dan di tampilkan secara detail oleh *acidbase* yang prosesnya dapat dilihat pada gambar 9, gambar 10 dan gambar 11.

E. Pengujian Fungsionalitas Penanganan Serangan

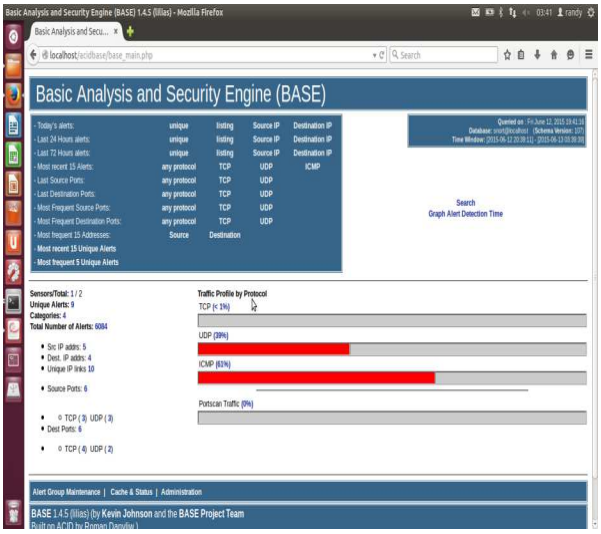
Pada sub bab ini, akan menguji keefektifan dari fungsionalitas penaganan dari serangan yang telah dilakukan. Ini merupakan sistem terintergrasi yang berfungsi melakukan tidakan yang mencagah dan menangani masalah yang terjadi. Untuk menguji sistem IDS dalam melindungi *server* dan merepresentasikannya dengan melakukan simulasi serangan yang kemudian dilakukan pemblokiran serangan menggunakan *IP Tables* dan *MAC filtering* yang dapat dilihat pada gambar 12 dan pada gambar 13.



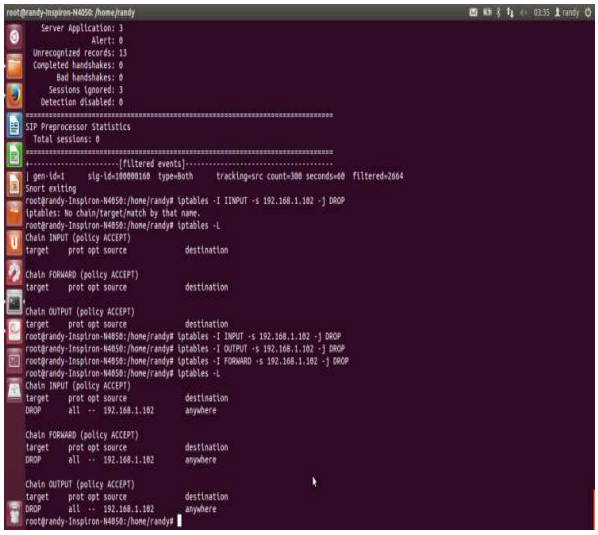
Gambar 9. Pendeteksian Serangan Snort



Gambar 11. Pendeteksian Serangan BASE



Gambar 10. Pendeteksian Serangan BASE

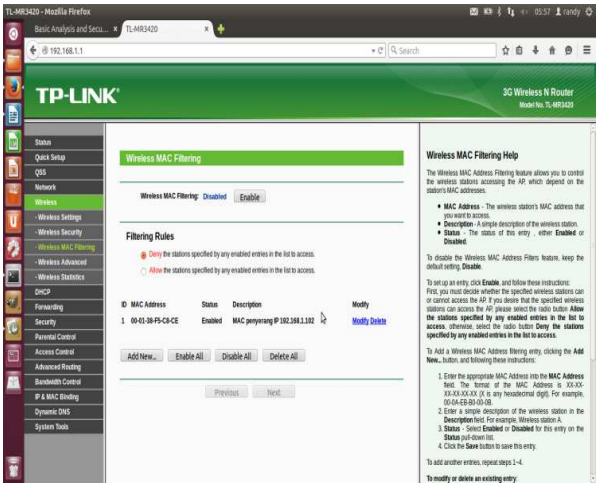


Gambar 12. Penanganan Serangan Dengan IP Tables

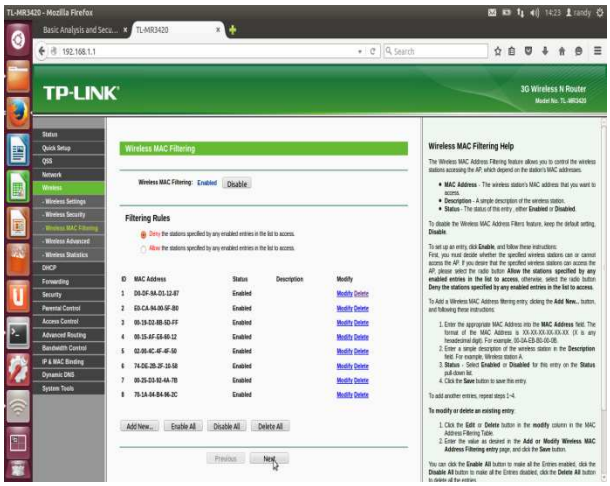
Setelah melakukan pengujian fungsi, pengujian serangan dan pengujian deteksi serangan, dengan melakukan pengujian penanganan serangan dengan menggunakan *Iptables* dan *MAC Filtering*. Pada gambar 14, gambar 15 dan gambar 16 berikut adalah tampilan hasil dari penanganan serangan yang dilakukan.

F. Hasil Analisis Menggunakan IDS

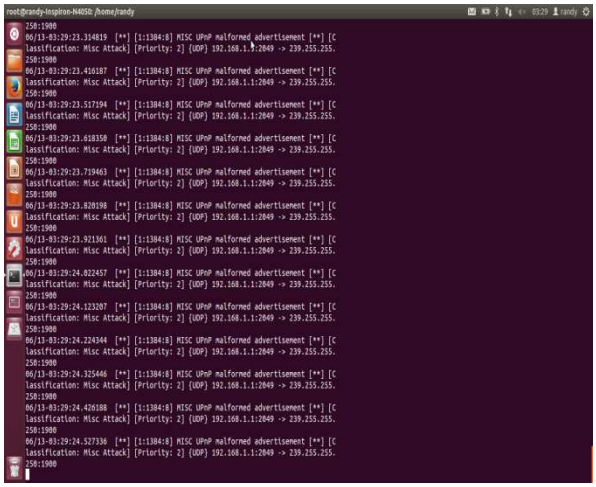
Analisisi pengujian data yang dilakukan berupa pengujian fungsionalitas interkoneksi IDS, dimulai dari proses serangan dan pendeteksian serta penanganan serangan. Tampilan dari data yang dianalisi, kondisi sebelum dan sesudah penyerangan. Setelah melakukan berbagai proses dalam penerapan IDS, terdapat kemudahan dalam penerapannya.



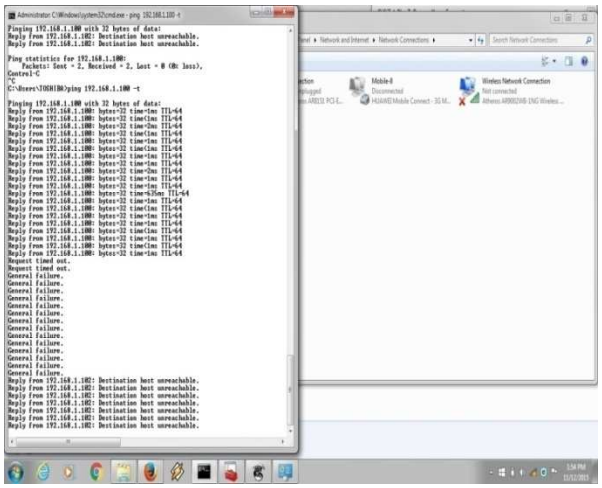
Gambar 13. Penanganan Serangan Dengan MAC Address



Gambar 15. Tampilan Hasil Penanganan Serangan



Gambar 14. Tampilan Hasil Penanganan Serangan



Gambar 16. Tampilan Hasil Penanganan Serangan

Analisis yang didapat dari hasil proses pengujian yang telah dilakukan adalah kondisi berjalan dengan baik pada saat sebelum terjadi penyerangan, kemudian terjadi gangguan saat serangan dilakukan, yang membuat pendeteksian menampilakan informasi dan rincian data dari penyerang, kemudian berhasil dilakukan penanganan dengan kondisi yang diperlukan sehinggan semua serangan berhasil diblokir dengan baik.

V. KESIMPULAN DAN SARAN

A. *Kesimpulan*

Rumusan kesimpulan dari keseluruhan proses penelitian yang telah dilakukan dari pembahasan yang sudah di uraikan dengan kesimpulan sebagai berikut :

Sistem IDS dalam mendeteksi serangan yang terjadi adalah dengan melakukan *scanning* terhadap sejumlah *source* dan lalu lintas yang terjadi didalam jaringan.

Mekanisme sistem kerja snort dan BASE yang telah berhasil di implementasikan dengan baik. Dalam pengujian sistem snort dan ACID yaitu dengan menggunakan *Ping attack* dan *Digital Blaster*.

Pencegahan yang dapat dilakukan terhadap penyerangan adalah dengan menggunakan *iptables*. Untuk mengatasi serangan dari *intruder* yaitu dengan cara *ping attack* ke sebuah server, maka dilakukan konfigurasi sebuah *rule* iptable, dimana *rule* tersebut untuk memblok berdasarkan alamat IP Address.

Analisi dari serangan, pendeteksian dan penanganan serangan dilakukan dengan menguraikan proses yang terjadi dalam pengujian interkoneksi antar aplikasi dan mesin sensor.

Setelah melakukan berbagai proses dalam penerapan IDS, terdapat kemudahan dalam penerapannya. Hasil yang diperoleh dari penerapan IDS ini yaitu suatu jaringan komputer dapat dipantau hanya melalui sebuah mesin atau komputer yang bertindak sebagai sensor dalam jaringan tersebut dan dapat melihat semua kejadian yang sedang terjadi di dalamnya.

B. Saran

Saran – saran yang dapat diberikan dari analisa kinerja sistem deteksi serangan dengan WIDS sebagai berikut diuraikan ini.

Dianjurkan melapisi sistem keamanan jaringan nirkabel dengan WIDS khususnya untuk deteksi dini terhadap serangan.

Menggunakan teknik pengujian lanjutan terhadap ancaman keamanan jaringan nirkabel.

DAFTAR PUSTAKA

[1] D. Ariyus, *Intrusion Detection System,* Andi Yogyakarta. Yogyakarta, 2007.

[2] D. Sopandi, *Instalasi dan Konfigurasi Jaringan Komputer*. Informatika Bandung, Bandung, 2008

[3] I. Dony, "*Intrusion Detection System*", ANDI, Yogyakarta, 2007.

[4] Junior, Dkk, *Perancangan Intrusion Detection System pada Jaringan Nirkabel BINUS Universitas,* Jakarta, 2009

[5] L. Putri, "*Implementasi Intrusion Detection System (IDS) menggunakan Snort pada jaringan Wireless", Skripsi* Program S1 Teknik Informatika Universitas Islam Negeri, Jakarta, 2011.

[6] Nugraha, M. Satria, "*Implementasi Intrusion Detection System untuk Filtering Paket Data", Skripsi* Program S1 Teknik Informatika Universitas Islam Negeri, Jakarta, 2010.

[7] S. Edhy, "*Komunikasi Data Dan Jaringan Komputer*", Graham Ilmu, Yogyakarta, 2005.

[8] Sukamaaji, Anjik dan Riant, *Konsep Dasar Pengembangan Jaringan dan Keamanan Jaringan*, Andi Yogyakarta, Yogyakarta, 2008.

[9] T. Lukas, "*Jaringan Computer"*, PT. Alex Media Komputindo, Jakarta, 1995.

[10] Wardhani, Helena, *Intrusion Detection System Snort*, Bandung, 2009