Web App Pendeteksi Jenis Serangan Jaringan Komputer dengan Memanfaatkan *Snort* Dan *Log Honeypot*

Asmah Akhriana*[1], Andi Irmayana[2]
[1,2]Program Studi Teknik Informatika STMIK Dipanegara Makassar
E-mail: *[1]**rhyna.akhriana@gmail.com**, [2]irmayana180985@gmail.com

Abstrak

Seiring dengan perkembangan Teknologi Informasi saat ini yang selalu berubah menjadikan keamanan suatu informasi sangatlah penting, terlebih lagi pada suatu jaringan yang terkoneksi dengan internet. Namun yang disayangkan adalah ketidakseimbangan antara setiap perkembangan suatu teknologi tidak diiringi dengan perkembangan pada sistem keamanan itu sendiri, sehingga cukup banyak sistem – sistem yang masih lemah dan harus ditingkatkan dinding keamanannya. Penelitian ini bertujuan untuk merancang interface berbasis Web App untuk memudahkan pengguna atau administrator dalam mengamankan komputer jaringan dari berbagai jenis serangan. Metode Instrusion detection system (IDS) digunakan untuk mendeteksi aktivitas yang mencurigakan dalam sebuah sistem atau jaringan dengan memanfaatkan snort dan honeypot. Honeypot dibangun pada sebuah komputer bersama dengan apache, mysql, dan snort. Honeypot akan bertindak sebagai target untuk menarik penyerang dan membuat log informasi dari penyerang tersebut dan snort untuk mengaplikasikan rule yang dibuat dari web. Fungsional sistem kemudian akan diuji menggunakan metode pengujian black box. Hasil dari penelitian ini disimpulkan bahwa Interface berbasis Web App yang dibuat dapat digunakan untuk membantu pengguna maupun administrator dalam menjaga data dan informasi pada komputer server dari berbagai jenis serangan pada jaringan komputer.

Kata Kunci—*Web App*, serangan jaringan, *snort*, *honeypot*, *Intrusion Detection System* (IDS).

Abstract

Along with the current development of Information Technology is always changing to make the security of an information is very important, especially on a network connected to the internet. But what is unfortunate is that the imbalance between each development of a technology is not accompanied by developments in the security system itself, so that there are quite a lot of systems that are still weak and have to be increased by the security wall. This study aims to design a Web-based App interface to facilitate users or administrators in securing network computers from various types of attacks. The Instrusion detection system (IDS) method is used to detect suspicious activity in a system or network using snort and honeypot. Honeypot is built on a computer along with Apache, MySQL, and Snort. Honeypot will act as a target to attract attackers and log information from the attacker and snort to apply the rules made from the web. The functional system will then be tested using the black box testing method. The results of this study concluded that Web App-based interfaces that are created can be used to help users and administrators in maintaining data and information on server computers from various types of attacks on computer networks.

Keywords—*Web App, Network Attacks, snort, honeypot, Intrusion Detection System (IDS).*

1. PENDAHULUAN

Keamanan teknologi informasi (IT) merupakan sebuah hal mendasar yang penting untuk diperhatikan dalam sebuah lingkungan organisasi maupun individu. Serangan terhadap *server* dan *service* pada organisasi, sampai pembajakan akun sosial pada individu, apapun bentuknya, tindakan ini hanya mendatangkan kerugian. Keamanan jaringan komputer sebagai bagian dari sebuah sistem sangat penting untuk menjaga validitas dan integritas atau serta menjamin ketersediaan layanan bagi penggunanya. Sistem harus dilindungi dari segala macam serangan dan usaha-usaha penyusupan oleh pihak yang tidak berhak.

Beberapa aplikasi atau sistem telah dikembangkan dan diterapkan untuk mengatasi serangan yang terjadi. Contohnya teknik pengamatan dengan *firewall* atau *Intrusion Prevention System* (IPS) untuk mencegah serangan, atau pendeteksian pada saat mulai terjadi serangan dengan *Intrusion Detection System* (IDS). *Snort* merupakan salah satu aplikasi yang dapat berfungsi sebagai IDS ataupun IPS. *Snort* merupakan suatu perangkat lunak untuk mendeteksi penyusup dan mampu menganalisis paket yang melintasi jaringan secara *real time*. *Traffic* dan *logging* ke dalam *database* serta mampu mendeteksi berbagai serangan yang berasal dari luar jaringan. *Snort* merupakan sebuah *software* yang bersifat *open source* yang dikembangkan oleh Marty Roesch dan tersedia secara gratis di *www.snort.org. Snort* dapat digunakan pada sistem operasi Linux, Windows, BSD, Solaris dan sistem operasi lainnya. *Snort* merupakan *network based* IDS yang menggunakan metode *Signature Based Detection*, menganalisis paket data apakah sesuai dengan jenis serangan yang sudah diketahui olehnya.

Honeypot merupakan sebuah sistem atau komputer yang sengaja "dikorbankan" untuk menjadi target serangan dari *hacker. Honeypot* adalah sebuah sumber daya yang bersifat seakan-akan target yang sebenarnya, yang dengan sengaja disediakan untuk diserang atau diambil alih. Oleh karena itu, *Honeypot* akan diamati, diserang bahkan dieksploitasi oleh penyerang atau penyusup. Tujuan utama dari *Honeypot* ini adalah untuk mengumpulkan informasi dari suatu serangan dan penyerang yang melakukannya. *Intruder* atau penyerang merupakan istilah umum yang diberikan untuk menggambarkan seseorang yang berusaha untuk masuk kedalam sistem dalam arti berusaha menggunakan sistem dimana mereka tidak memiliki autorisasi atau menggunakan sistem untuk maksud yang menyimpang diluar hak-hak yang mereka miliki.

Sistem *honeypot* biasanya hanya sebuah sistem yang dihubungkan dengan jaringan produktif, atau sistem yang asli, dengan tujuan untuk menjebak penyerang. Sistem tersebut kemudian dapat mengemulasikan berbagai variasi sistem atau lubang-lubang dari sistem yang mudah untuk diserang. Komputer tersebut melayani setiap serangan yang dilakukan oleh *hacker* dalam melakukan penetrasi terhadap *server* tersebut. Metode ini ditujukan agar *administrator* dari server yang akan diserang dapat mengetahui trik penetrasi yang dilakukan *hacker* serta bisa melakukan antisipasi dalam melindungi *server* yang sesungguhnya. Setiap tindakan yang dilakukan oleh penyusup yang mencoba melakukan koneksi ke *honeypot* tersebut, maka *honeypot* akan mendeteksi dan mencatatnya.

Dari pemaparan diatas, rumusan masalah yang ada pada penelitian ini adalah bagaimana merancang sistem pendeteksi serangan jaringan komputer berbasis web dengan memanfaatkan *snort* dan *honeypot* untuk memudahkan pengguna atau *administrator* dalam mengamankan jaringan.

Tujuan penelitian ini adalah mengimplementasikan *snort* dan *honeypot* dalam meningkatkan keamanan jaringan dalam bentuk sistem pendeteksi serangan jaringan komputer berbasis web sehingga dapat memberikan kemudahan bagi pengguna maupun *administrator* dalam menjaga data dan informasi pada komputer server dari berbagai jenis serangan pada jaringan komputer.

Ada beberapa penelitian terkait yang telah dilakukan sebelumnya. Dalam upaya mengembangkan dan menyempurnakan pemanfaatan *snort* dan *honeypot* ini perlu dilakukan studi pustaka (*literature review*) sebagai salah satu dari penerapan metode penelitian yang dilakukan, diantaranya sebagai berikut :

1. Penelitian yang telah dilakukan oleh Arya Ervan Loeresta (2014) dengan judul

"Impelementasi *honeypot* Sebagai Pendeteksi *Malware* Pada Layanan *Cloud Computing*". Pada penelitian ini dirancang sistem keamanan menggunakan *honeypot* untuk mendeteksi *malware* pada layanan *cloud computing*. Karena banyaknya ancaman-ancaman keamanan jaringan yang dapat menyebabkan *server* mati dan tidak dapat beroperasi lagi. Hal inilah yang harus ditangani untuk meminimalkan celah keamanan sedini mungkin.

2. Penelitian yang telah dilakukan oleh Fuadielah Danok Eka Putra (2014) dengan judul "Analisa Perbandingan Performa *Intrusion Detection System Snort, Low Interaction Honeypot,* dan *High Interaction Honeypot*". Pada penelitian ini, dirancang sebuah sistem dengan pada lingkungan Ubuntu yang diletakkan pada jaringan PT. Citra Media Solusindo yang kemudian dianalisa dengan membandingkan kinerja sistem saat dilakukan ujicoba serangan, perbandingan ini dilihat dari fungsi sistem mampukah sistem menangkap serangan dengan baik, kinerja server (CPU, memori, dan *bandwidth*) saat terjadi serangan dengan berbagai intensitas serangan dan bagaimana *respon time* sistem terhadap serangan.
3. Penelitian yang telah dilakukan oleh Muh. Masruri Mustofa (2013) yang berjudul "Penerapan Sistem Keamanan *Honeypot* dan IDS Pada Jaringan Nirkabel (Hotspot)". Pada penelitian ini dilakukan pengamatan secara lagsung terhadap jaringan *hotspot* di UAD. Analisis dilakukan untuk mendapatkan hasil serta data yang bisa dijadikan sebagai acuan guna menerapkan suatu sistem keamanan jaringan *hotspot* berbasis *honeypot* dan *snort*. Sistem hasil implementasi diuji dengan dua metode yaitu *Alpha Test* dan *Beta test.*
4. Penelitian yang telah dilakukan oleh Syaumi Husnan (2013) yang berjudul "Implementasi *Honeypot* Untuk Meningkatkan Sistem Keamanan Server Dari Aktivitas Serangan". Pada penelitian ini penulis membangun sistem keamanan jaringan untuk mengamankan *server* pada STIKes Kusuma Husada. Dalam mengamankan *server* di STIKes Kusuma Husada dari serangan, maka diperlukan adannya implementasi *honeypot* untuk meningkatkan keamanan *server*. *Honeypot* diimplemetasikan menggunakan *honeypot* jenis *low interaction* yaitu *honeyd* dan *software* pnedukung lainnya seperti *portsentry, FARPD, Honeyd-wiz, apache.* Uji coba dengan melakukan *ping* dan *scanning* terhadap *IP host honeyd* menggunakan *nessus* untuk mengetahui *honeypot* berhasil menjebak penyerang.
5. Penelitian yang telah dilakukan oleh Lidia Putri (2011) berjudul "Implementasi *Intrusion Detection System* (IDS) Menggunakan *Snort* Pada Jaringan *Wireless*". Pada penelitian ini dibangun kemanan jaringan pada jaringan *wireless* di SMK Triguna yang berfungsi untuk mengawsi *traffic* jaringan dan kegiatan-kegiatan mencurigakan didalam sistem jaringan tersebut. Penulis menggunakan *Snort, ACID, Ntop* pada mesin sensor IDS yang berbasis Ubuntu.

Dari beberapa sumber *literature review* maka peneliti dapat mengetahui bahwa penelitian tentang pemanfaatan *Snort* dan *Honeypot* berbasis Web dalam pendeteksian serangan jaringan komputer belum pernah dilakukan.

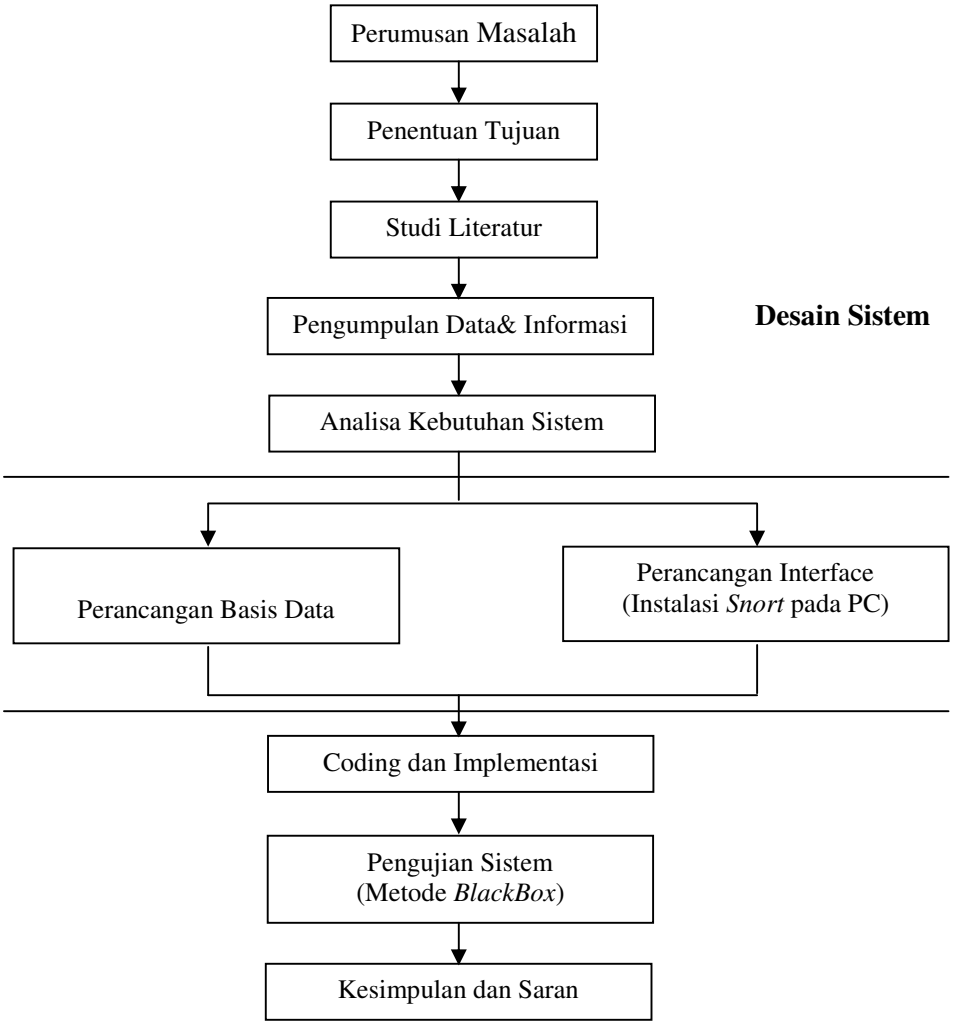
2. METODE PENELITIAN

2.1 Tahapan Penelitian

Untuk mencapai hasil penelitian yang ditargetkan, terdapat beberapa tahapan yang harus dilakukan secara berkala antara lain :

1. Studi Literatur, yaitu dengan melakukan studi dari buku-buku pustaka yang berkaitan dengan masalah yang dibahas, juga melalui artikel-artikel dari internet baik itu jurnal, prosiding ataupun artikel yang lain yang ada kaitannya dengan penelitian yang akan dibuat.
2. Metode Pengambilan Data, yaitu observasi ke lokasi penelitian dan mengambil data-data yang dibutuhkan yang berkaitan dengan keamanan data dan informasi.
3. Analisis Sistem, dimana penulis melakukan analisa spesifikasi yang dibutuhkan terkait penelitian ini seperti perangkat lunak (*software*), perangkat keras (*hardware*) dan perangkat jaringan yang dibutuhkan untuk sistem IDS.
4. Desain Sistem, yaitu merancang interface sistem sesuai dengan tujuan penelitian yang ditargetkan.

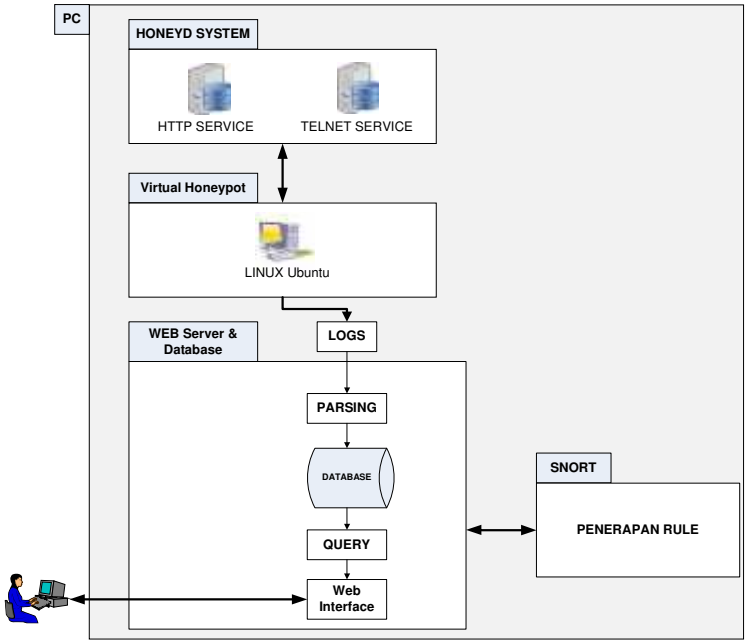
5. Instalasi *Snort* pada perangkat computer.
6. Membuat rancangan database *snort*.
7. Coding dan Implementasi Sistem, pembuatan sistem dengan penerapan beberapa *rule snort* IDS ke bahasa pemrograman.
8. Pengujian dan pemeliharaan, sistem selanjutnya akan diuji dimana testing dilakukan dengan melakukan pengujian pada halaman-halaman sistem yang dirancang menggunakan metode pengujian *Blackbox.*
9. Menarik kesimpulan dari hasil penelitian yang diperoleh dan memberikan saran-saran yang dianggap perlu.



Gambar 1. Alur Penelitian

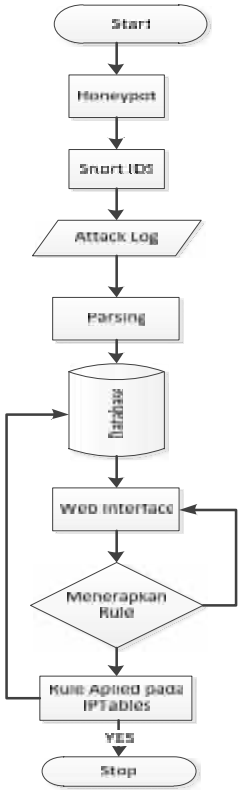
2.2 Model Arsitektur Kerja Sistem

Snort melakukan pengawasan terhadap *traffic* jaringan dan pengawasan terhadap kegiatan-kegiatan yang mencurigakan didalam sebuah sistem jaringan dimana yang menjadi target dari serangan adalah *honeypot*. Pada gambar 1 ditunjukkan *honeypot* yang dibangun pada sebuah computer bersama dengan *apache, MySQL*, dan *snort. Honeypot* akan bertindak sebagai target untuk menarik penyerang dan membuat log informasi dari penyerang tersebut. *Apache*dan MySQL untuk aplikasi Web, dan *snort* untuk mengaplikasikan rule yang dibuat dari web.
Berikut gambar arsitektur kerja dari sistem yang akan dibangun :



Gambar 2 . Arsitektur Kerja Sistem

File log yang berisi informasi akan diparsing ke dalam database. Informasi ini yang akan digunakan oleh web dan membuat rule dari informasi tersebut. Untuk lebih jelasnya, Gambar 3 menjelaskan tentang alur sistem lebih lengkap.



Gambar 3. Alur Kerja Sistem

2.3 Alat Dan Bahan Penelitian

Dalam proses perancangan ini, diperlukan alat dan bahan yang dapat mendukung keberhasilan perancangan. Alat dan bahan perancangan yang digunakan adalah :

Alat Penelitian

a. PC sebagai *node* yang dilindungi. Dikonfigurasi untuk menjalani *service*layanan *web* dengan spesifikasi sebagai berikut:
   a) Prosesor Intel Core 2 Duo 1.7 GHz
   b) Ram 2 GB
   c) 40 GB HDD
b. Sistem Operasi yang digunakan untuk menjalankan *webserver* adalah GNU/Linux Ubuntu Server Edition versi 9.04 sedangkan Sistem operasi yang digunakan untuk menjalankan Honey adalah GNU/LinuxSlackware64 versi 13.0
c. Perangkat lunak yang digunakan adalah:
   a) Libnids-1.21
   b) Snort-2.8.4.1-3
   c) IPTables-1.4.1.1
   d) MySQL-5.0
   e) Honeyweb-0.4
   f) Perl
   g) Apache2
   h) PHP-5.12
   i) Web browser
   j) Xampp

Bahan Penelitian

Bahan Penelitian yang digunakan sebagai berikut:

1. Nmap
2. Telnet
3. Ping
4. Worm
5. HTTP *Normal Request*(dari *web browser*)

2.4 Teknik Pengumpulan Data

Adapun teknik pengumpulan data yang digunakan dalam penelitian ini adalah:

1. Studi Literatur yaitu dengan melakukan studi dari buku-buku pustaka, artikel ataupun jurnal ilmiah yang relevan yang digunakan sebagai acuan dalam penelitian ini.
2. Teknik Observasi yaitu suatu cara yang dilakukan untuk memperoleh data dengan mengamati secara langsung terhadap aktifitas Jaringan Komputer yang ada.
3. Teknik Wawancara yaitu suatu cara yang dilakukan untuk memperoleh jawaban atas pertanyaan yang berkaitan dengan masalah penelitian kepada bagian-bagian yang terkait di dalamnya.

2.5 Teknik Pengujian Sistem

Fungsional system secara keselurukan akan diuji menggunakan metode pengujian *black box*. Uji coba *blackbox* berusaha untuk menemukan kesalahan dalam beberapa kategori seperti Fungsi-fungsi yang salah atau hilang, Kesalahan interface, Kesalahan dalam struktur data atau akses database eksternal, Kesalahan performa, Kesalahan inisialisasi dan terminasi. Adapun analisis kebutuhan sistem fungsional dalam sistem monitoring penyusup jaringan komputer meliputi :

1. Sistem dapat mengidentifikasi adanya usaha-usaha penyusupan pada suatu jaringan komputer.
2. Sistem dapat memberikan notifikasi melalui web interface jika ada usaha-usaha penyusupan pada suatu jaringan komputer.

3. HASIL DAN PEMBAHASAN

3.1 Tampilan Aplikasi

Berikut ini akan dijelaskan Tampilan dari isi pada aplikasi Web App yang telah dibuat.

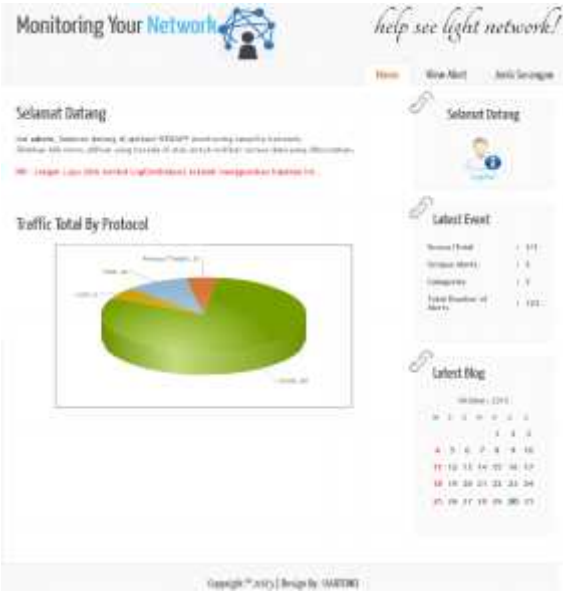
a. Tampilan Halaman Login

Gambar 4. Tampilan Halaman Login

Pada gambar 4 di perlihatkan tampilan halaman *login* admin. Admin hanya perlu menginputkan *username* dan *password* yang telah tersimpan dalam database yang benar setelah itu mengclick tombol login.

b. Tampilan Halaman Utama Web APP

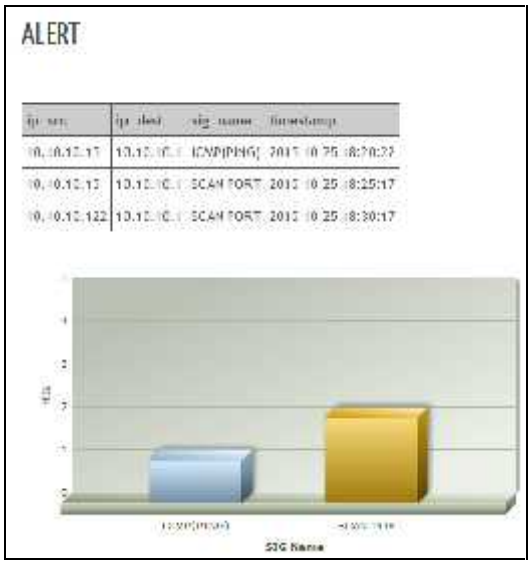
Halaman ini merupakan tampilan utama untuk *admin* ketika telah sukses melakukan *login*. Pada halaman ini terdapat menu home, menu *view alert*, view jenis seragan dan *logout*.



Gambar 5. Tampilan Halaman Utama Web APP

c. Tampilan Halaman *View Alert*

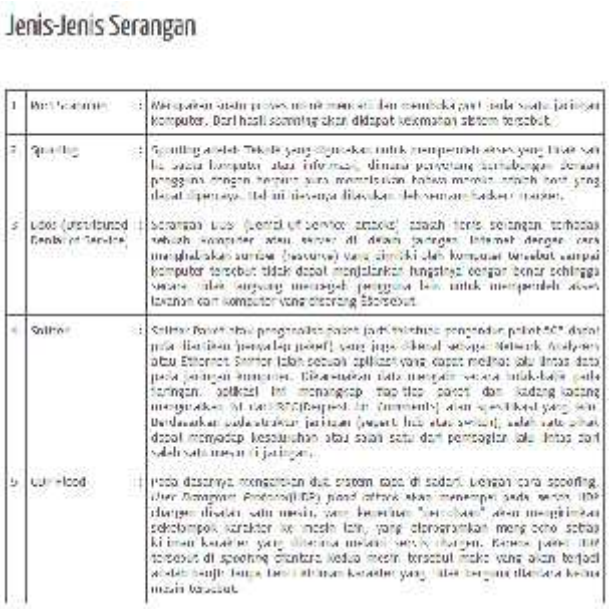
Halaman ini merupakan halaman *view alert*. Dimana halaman ini digunakan oleh *admin* untuk menampilkansemua data serangan yang masuk atau yang menyerang *honeypot*. Tampilan serangan berbentuk tabel dan *View Alert* seperti yang terlihat pada gambar 6.



Gambar 6. Tampilan Halaman View Alert

d. Tampilan Halaman Jenis Serangan

Halaman ini merupakan halaman *view* jenis serangan. Dimana halaman ini digunakan oleh *admin* untuk menampilkansemua jenis-jenis serangan yang terdaftar pada aplikasi *snort* seperti yang terlihat pada gambar 7.



Gambar 7. Tampilan Halaman Jenis Serangan

3.2 Pengujian Sistem

Pengujian sistem dilakukan dengan menggunakan metode *Black Box Testing*, dimana metode ini merupakan *testing* pengujian yang dilakukan hanya mengamati hasil eksekusi melalui data uji dan memeriksa fungsional dari perangkat lunak. Dengan menggunakan metode pengujian *black box*, perekayasa sistem dapat menemukan kesalahan dalam sistem.

Berikut adalah hasil pengujian sistem pada halaman-halaman sistem yang telah dirancang :

Tabel 1. Hasil Pengujian Sistem

No	Komponen	Butir Uji	Hasil Yang Diharapkan	Hasil Pengujian
1	Menu Utama Halaman Aplikasi	Menekan tombol aplikasi (Home, View Alert, View Rule, dan Tombol Keluar)	Ketika usermenekan tombol-tombol menu yang tersedia maka akan tampil halaman yang diinginkan sesuai nama menu	[✓] Diterima [ ] Ditolak
2	Menu Home	Tampil data	Ketika user mengakses atau membuka halaman home maka akan tampil data awal web app.	[✓] Diterima [ ] Ditolak
3	Menu view Alert	Tampil data	Ketika user mengakses atau membuka halaman view alert maka akan tampil semua data serangan dan jenis serta ip penyerang.	[✓] Diterima [ ] Ditolak
4	Menu View Rule	Tampil data	Ketika user mengakses atau membuka halaman view rule maka akan tampil rule-rule serangan yang digunakan snort.	[✓] Diterima [ ] Ditolak
5	Tombol Keluar	Keluar dari aplikasi	Ketika user meng clicktombol keluar maka system akan tertutup dan kembali ke halaman login.	[✓] Diterima [ ] Ditolak

3.3 Implementasi Sistem

Contoh Kasus : Nmap *Port Scanning Attack*

Pada kasus ini, penulis akan mensimulasikan dan menganalisis jenis aktivitas *port scanning* dengan menggunakan Nmap, yang dilakukan dari kedua mesin penyerang.

Langkah pertama adalah membuat *rule/signature* untuk mendefenisikan jenis aktivitas ini. Berdasarkan hasil analisis *traffic,* penulisan *rules* untuk mendeteksi Nmap ping sebagai berikut:

Alert icmp any any -> any any (msg:"ICMP PING NMAP attack";dzise:0;itype:8:rev:1;sid:100003;)

Signature atau *rule* diatas akan meng-generate *alert Snort* jika mendeteksi akses protocol ICMP yang berasal dari segmen jaringan eksternal maupun internal, melalui *port* - berapapun ke 10.10.10.1 (mesin server) *port* berapapun: keterangan *rules*: "ICMP PING NMAP attack"; berukuran paket 0 byte; menggunakan tipe icmp 8; revisi *rules* pertama: nomor identitas *rules* 10003.

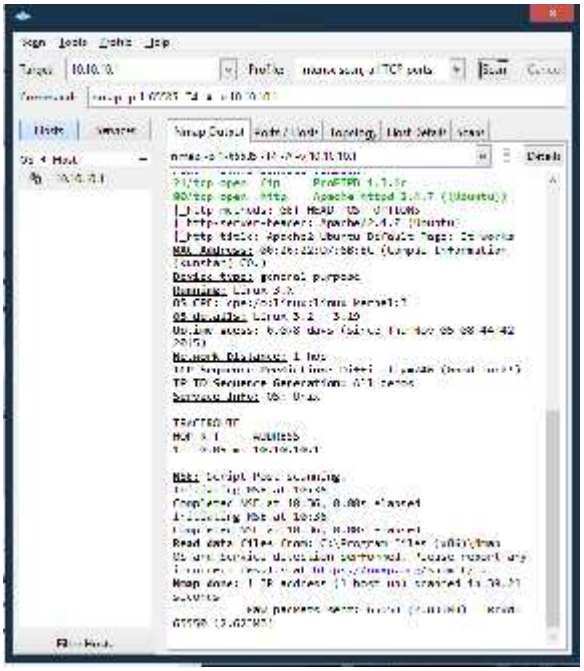
Langkah kedua adalah menerapkan *rules/signature* baru ini dengan menempatkan pada direktori *rules* Snort (/etc/snort/rules). Pada penelitian ini, penulis menyimpan *signature* ini dengan nama local.rules. setelah itu, proses Snort harus di *restart*, agar Snort dapat mendeteksi, membaca, dan menerapkan *rules* baru tersebut pada kode intinya. Proses untu me-*restart Snort*

adalah (/etc/init.d/snort restart). Langkah ketiga adalah melancarkan serangan dengan menggunakan Nmap seperti pada gambar 8.

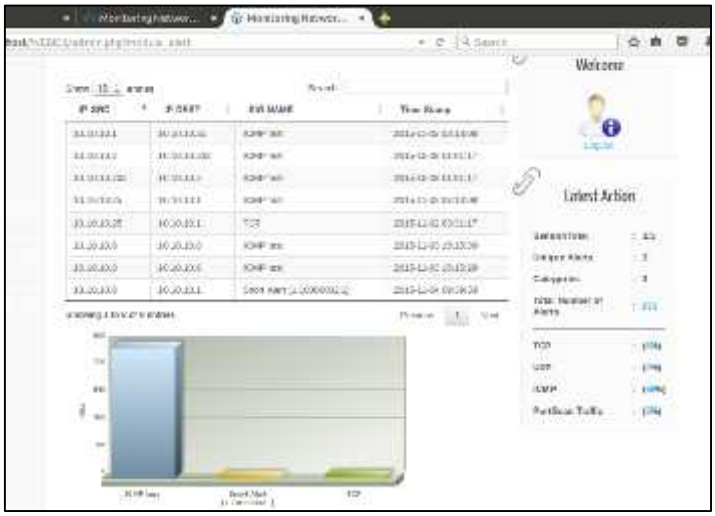
Tampilan Web App IDS (Intrusion Detection System) setelah diserang menggunakan Ping *attack* (ICMP *Traffic*), dan Nmap.

Ketika dilakukan penyerangan ke mesin server dengan IP 10.10.10.1 menggunakan ping *attack,* dan Nmap pada Web App IDS menanggkap *Traffic profiles by protocol* berjenis ICMP sebesar 89%, TCP sebesar 6% dan Port Scan sebesar 5%. Pada kuadran sisi kanan terdapat informasi banyaknya serangan yang masuk yang dibagi menjadi 4 bagian TCP, UDP, ICMP dan *Port scan Traffic* seperti tampilan gambar 9.

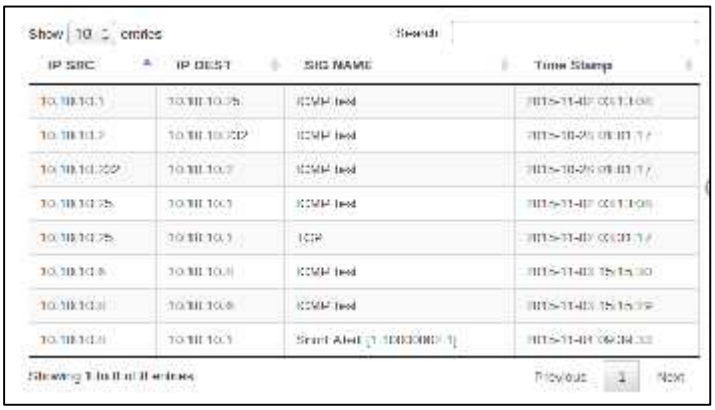
Pada gambar 10 dapat dilihat beberapa informasi serangan yang masuk. Terlihat berturut-turut dari kiri ke kanan adalah IP SRC (alamat IP sumber), IP DEST (alamat IP tujuan), SIG NAME, *time stamp* (waktu terjadinya *alert*).



Gambar 8. Uji Coba Nmap dari Penyerang ke Server



Gambar 9 . Web App IDS ketika dilakukan penyerangan



Gambar 10. Tampilan tabel informasi serangan

Dari data yang diperoleh, maka penulis dapat melakukan pencegahan terhadap penyerangan tersebut. Dalam melakukan pencegahan ini, penulis melakukannya dengan IPTables. Untuk mengatasi serangan dari *intruder* yaitu dengan cara *ping attack* ke sebuah mesin *server.* Penulis menuliskan sebuah *rule* iptables. Dimana *rule* tersebut untuk memblok berdasarkan alamat IP address.

root@pegasus-Lenovo-G450:#iptables –I FORWARD 10.10.10.2 –j DROP

Penulis memasukkan sebuah perintah untuk melakukan pemblokiran terhadap komputer penyerang. Penulis menggunakan perintah "iptables –I FORWARD –s 10.10.10.2 –j DROP" yang berjalan pada konsole. "-I" ataus *Insert* digunakan untuk memasukan perintah pada baris *chain,* perintah akan berada pada posisi *rules* teratas sehingga proses dapat dijalankan lebih awal. Dapat dilihat dengan menggunakan perintah iptables –L. pada tabel *chain FORWARD* perintah yang dimasukan tadi terdapat ada awal baris *rules.* "FORWARD" pada iptables digunakan untuk meneruskan paket dari jaringan eksternal ke dalam jaringan internal melalui mesin *firewall.* Perintah ini digunakan karena serangan ini berasal dari luar jaringan yang masuk kedalam jaringan internal melalui mesin *firewall.* "-s" untuk mencocokan paket berdasarkan alamat IP sumber. "10.10.10.1" merupakan *source* dari komputer penyerang yang akan diblokir. "-j DROP" mend-*drop* paket dan menolah untuk diproses lebih lanjut.

4. KESIMPULAN

Implementasi aplikasi telah selesai dan dapat digunakan untuk membantu pengguna maupun *administrator* dalam menjaga data dan informasi pada komputer server dari berbagai jenis serangan pada jaringan komputer.

5. SARAN

Untuk pengembangan selanjutnya, *snort* sebagai salah satu sistem keamanan jaringan hendaknya dapat dikembangkan tidak hanya sebagai sistem pendeteksi gangguan keamanan jaringan tetapi juga sebagai sistem pencegahan gangguan keamanan serta bisa dilakukan penambahan modul-modul tambahan yang mendukung kinerja *Intrusion Detection System* akan membantu efisiensi kerja sistem, seperti pengaturan *rule-rule* dan juga penambahan *frond end.*

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada pihak-pihak yang telah membantu sehingga penelitian ini dapat diselesaikan dengan baik serta dapat di implementasikan walaupun masih jauh dari kesempurnaan.

DAFTAR PUSTAKA

Husnan, S. (2013). *Implementasi Honeypot untuk Meningkatkan Sistem Keamanan Server dari Aktivitas Serangan* (Doctoral dissertation, Universitas Muhammadiyah Surakarta).

Leoresta, Arya Ervan,(2014),"Implementasi honeypot sebagai pendeteksi malware pada layanan cloud computing." *Program Studi Teknik Informatika Fakultas Sains dan Teknologi. Yogyakarta: Universitas Islam Negeri Sultan Kalijaga*.

Mustofa, M. M.,Aribowo, E. (2013). Penerapan Sistem Keamanan Honeypot dan IDS pada Jaringan Nirkabel (Hotspot). *Jurnal Sarjana Teknik Informatika*.

Putra, Fuadielah Danok Eka (2014) *Analisa Perbandingan Performa Intrusion Detection System Snort, Low Interaction Honeypot Dan High Interaction Honeypot.* Skripsi thesis, Universitas Muhammadiyah Surakarta.

Putri, L. (2011). Implementasi intrusion detection system (IDS) menggunakan snort pada jaringan wireless (studi kasus: SMK Triguna Ciputat).