IMPLEMENTASI INTRUSION DETECTION SYSTEM MENGGUNAKAN SNORT, BARNYARD DAN BASE PADA SISTEM OPERASI LINUX

[1]Rizki Triandini, [2]Yeffry Handoko Putra
[1,2]Jurusan Teknik Komputer, Fakultas Teknik dan Ilmu Komputer
Universitas Komputer Indonesia, Jl. Dipati Ukur No.112-116, Bandung 40132
Email :[1]rizkitriandini@gmail.com, [2]yeffryhp@unikom.ac.id

ABSTRAK

Intrusion Detection System (IDS) dapat didefinisikan sebagai kegiatan yang bersifat anomaly, incorrect, inappropriate yang terjadi di jaringan atau host. Dan IDS sendiri adalah sistem keamanan yang bekerja bersama Firewall untuk mengatasi Intrusion. IDS mampu mendeteksi penyusup dan memberikan respon secara real time. Terdapat dua teknik yang digunakan dalam IDS yaitu, NIDS (Network Based Intrusion Detection System) dan HIDS (Host Based Intrusion Detection System). Pada percobaan kali ini IDS dibagun menggunakan perangkat lunak Snort. Snort merupakan Open Source Intrusion detection System (IDS) yang digunakan untuk pemantauan dan pencegahan terhadap gangguan pada jaringan komputer. Agar mempermudah administrator dalam melihat dan membaca hasil log dari setiap paket data yang masuk atau keluar maka menggunakan Basic Analysis and Security Engine (BASE). Pada percobaan ini PC Server menggunakan sistem operasi Linux Ubuntu 14.04 LTS. Pengujian dilakukan pada Local Area Network dengan topologi Star. Dimana hasil gangguan yang disebabkan DoS dan Port Scanner dapat dikenali oleh Snort IDS dan menampilkan log secara lengkap, baik dari waktu, tanggal kejadian dan sumber IP Address dari pengganggu.

Kata kunci: IDS, Snort, NIDS, HIDS, BASE

1. PENDAHULUAN

Dalam jaringan komputer, khususnya yang berkaitan dengan keamanan jaringan merupakan hal yang sangat penting untuk diperhatikan, karena akan banyak terjadi hal-hal yang dapat menggangu keaman atau kestabilan koneksi dari jaringan komputer tersebut, baik yang berkaitan dengan *hardware* maupun yang berkaitan dengan *software*.

Beberapa contoh masalah yang sudah umum dalam keamanan jaringan adalah adanya gangguan pada sistem yang dapat terjadi karena faktor ketidak sengajaan yang dilakukan oleh pengelola (*human error*), akan tetapi tidak sedikit pula yang disebabkan oleh pihak ketiga yang mencoba menggagu pengguna jaringan tersebut yang berupaya melakukan perusakan, penyusupan atau penyalahgunaan data maupun sistem. Maka untuk membantu mempermudah administrator jaringan dalam mengelola jaringan tersebut dibutuhkan satu sistem yang dapat membaca dan mengenali setiap paket data yang masuk atau yang keluar secara cepat dan tepat.

Salah satu solusi yang dapat digunakan adalah dengan membangun *intrusion Detection System* (IDS). IDS adalah sebuah aplikasi perangkat lunak atau perangkat keras yang dapat mendeteksi aktifitas yang mencurigakan dalam sebuah sistem atau jaringan. IDS sendiri dapat membaca paket-paket data yang masuk maupun yang keluar secara otomatis yang nantinya akan memberikan sebuah laporan atau *log* kepada administrator jaringan.

Dengan memanfaatkan sistem IDS pada suatu jaringan maka kemungkinan adanya gangguan yang disebabkan oleh pihak ketiga terhadap suatu jaringan dapat di minimalisir, karena nantinya setiap gangguan yang disebabkan akan tederteksi oleh IDS, yang nantinya IDS akan memberikan peringatan atau *alert* kepada admininistrator jaringan.

2. DASAR TEORI

2.1. Jaringan Komputer

Jaringan komputer merupakan hubungan dari sejumlah perangkat yang dapat saling berkomunikasi satu sama lain. Pada jaringan komputer terdapat dua buah komputer atau perangkat yang saling berhubungan satu sama lain dan terjadi proses komunikasi dan transfer paket data di dalamnya.

Berdasarkan klasifikasinya, jaringan komputer dibagi menjadi:

1. Berdasarkan cakupan Area Geografis
2. Berdasarkan Topologi jaringan
3. Arsitektur Komputer

2.2. Topologi Jaringan Komputer

Topologi jaringan dibagi menjadi dua macam yaitu topolgi fisik dan topologi logika :

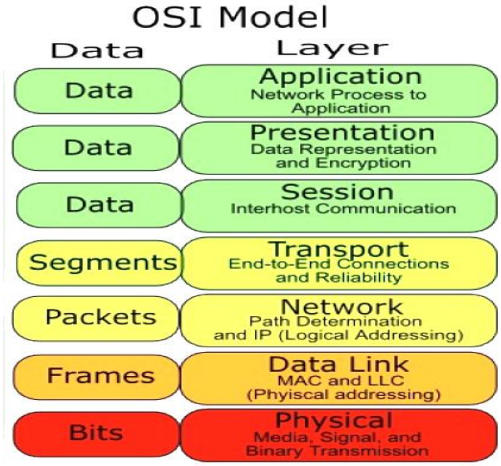
1. Topologi fisik adalah topologi yang mengambarkan secara fisik bagaimana suatu perangkat terhubung dengan perangkat lain sehingga menjadi suatu jaringan.
2. Topologi logika adalah topologi yang fungsinya untuk menampilkan bagaimana pengaturan perangkat jaringan agar bisa terhubung satu samalainnya sehingga dapat melakukan komunikasi

2.3. Protokol Jaringan

Protokol adalah kumpulan aturan yang mengatur komunikasi data. Protokol mengindentifikasi apa itu komunikasi, bagaimana cara berkomunikasi, serta kapan komunikasi dapat dilakukan.

2.3.1. Model OSI

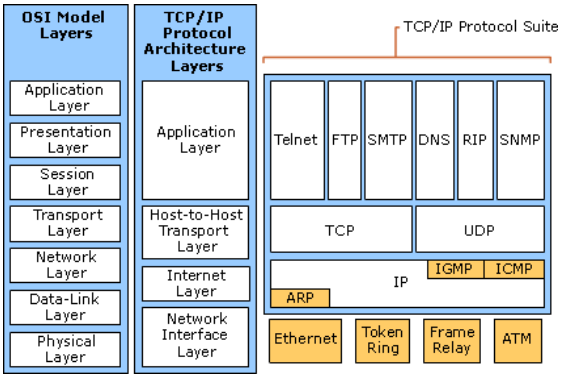
Operation System Interconnection (OSI) adalah kumpulan dua sistem yang berbeda dan dapat berkomunikasi tanpa memperhatikan arsitektur yang mendasari sistem tersebut. Tujuan OSI model adalah untuk menunjukan bagai mana cara untuk memfasilitasi dua sistem yang berbeda tanpa memerlukan perubahan logika pada *hardware* dan *software.*[1]



Gambar 1. Layer pada Model OSI

2.3.2. TCP/IP Protocol Suite

TCP/IP protocol suite dikembangkan sebelum model OSI. Namun lapisan-lapisan TCP/IP tidak sama dengan lapisan-lapisan yang dimiliki oleh OSI. Protocol TCP/IP hanya dibuat atas lima lapisan, yaitu : *Pyshical, Data Link, Network, Transport* dan *application*.



Gambar 2. Lapisan Pada Model OSI

2.4. Intrusion Detection System (IDS)

Intrusion Detection System (IDS) merupakan sistem untuk mendeteksi adanya *Intrusion* yang dilakukan oleh *Intruder* atau penyusup didalam jaringan. IDS sendri sanga mirip seperti alarm, apabila IDS mencatat adanya suatu serangan dalam jaringan, maka IDS akan memperinati administrator jaringan tersebut. IDS dapat dideinisikan sebagai kegiatan yang bersifat *anomaly, incorrect, inappropriate* yang terjadi di jaringan atau host. Dan IDS sendiri adalah sistem keamanan yang bekerja bersama *Firewall* untuk mengatasi *Intrusion*[2].

Berikut beberapa komponen yang terdapat pada IDS [4] :

1. Sensor : yang dapat mengenali adanya *securty event.*
2. *Console* : yang dapat memonitor event, alert dan mengontol sensor
3. *Central Enggine* : yang mana berguna untuk menyimpan *event logged* yang dilakukan oleh sensor kedalam database dan menggunakan aturan-aturan keamana yang beguna untuk menangani event yang terjadi.

Diliat dari kemampuan mendeteksi serangan dalam jaringan, maka IDS dibagai menjadi dua, yaitu [3] :

1. Network-Based IDS (NIDS)
2. Host-Based IDS (HIDS)

2.4.1. Network-Based IDS (NIDS)

Network-Based Intrusion Detection System (NIDS) akan melakukan *capture paket data* yang masuk pada jaringan LAN atau Wireless dan mengumpulkan paket-paket data tersebut, serta melakukan analisa terhadap paket-paket tersebut apah paket normal tau paket serangan.

2.4.2. Host-Based IDS (HIDS)

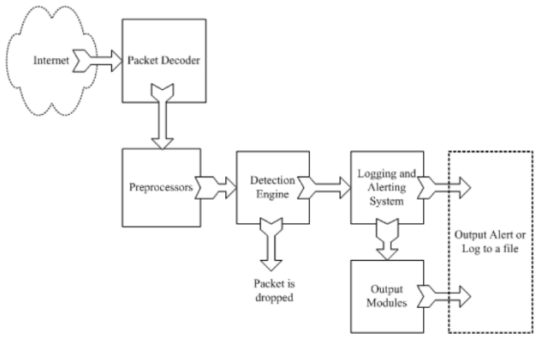
Host-based Intrusinon detection system (HIDS) hanya melakukan pemantauan pada perangkat komputer tertentu dalam jaringan. IDS dapat melihat kedalam sistem dan aplikasi file log untuk mendeteksi aktivitas penyusup. HIDS hanya akan berkerja apabila terjadi serangan dan akan memberikan laporan atau peringakatan kepada administrator secara *real time*.

2.5. SNORT

Snort merupakan *Open Source Intrusion detection System* (IDS) yang bisa di konfigurasi sebagai *Intrusion Detection System* (IPS) untuk pemantauan dan pencegahan pada jaringan komputer.

Ketika bekerja sebagai IDS Snort dapat mendeteksi serangan atau paket data yang masuk kedalam jaringan komputer dan memperingatkan administrator jaringan untuk mengambil tindakan pencegahan.

Snort secara logis dibagi menjadi beberapa komponen. Komponen-komponen ini bekerja sama untuk mendeteksi serangan tertentu dan untuk menghasilkan *output* dengan format yang dibutuhkan pada *System Detection*[4].



Gambar 3. Komponen Snort

2.6. Firewall

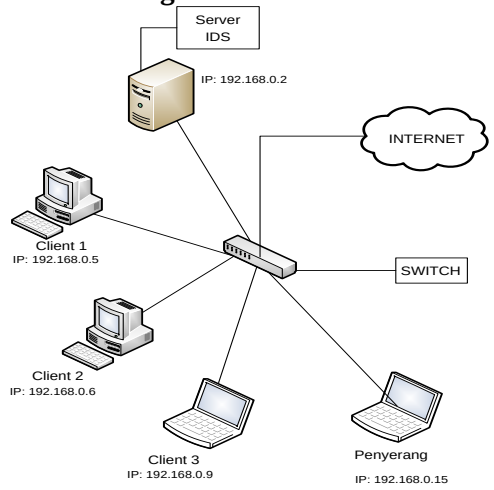
Defenisi dasar dari *Firewall* sendiri adalah komponen atau suatu kelompok komponen yang mengatur akses antara internal network (network yang akan dilindungi) dengan eksternal network (Internet) atau anta beberapa kumpulan network. *Firewall* dapat berupa solusi *hardware* dan *software* yang membatasi akses dari jaringan internet ke Internet atau sebaliknya.

Firewall dapat juga digunakan untuk memisahkan dua atau lebih bagian jaringan lokal. *Firewall* merupakan suatu cara yang efektif untuk melindungi jaringan dari ancaman gangguan lewat Internet. Selain itu F*irewall* dalam jaringan komputer membatasi dan menjaga kerusakan pada suatu bagian jaringan agar tidak menyebar ke bagian lain jaringan

2.7. Basic Analysis and Security Engine (BASE)

BASE merupakan *interface* yang akan digunakan untuk menganalisa *output* dari snort. BASE bekerja dengan menggunakan *query* pada *log database* snort yang akan disimpan kedala *database mysql server*. Kemudian akan menampilkan kedalam bentuk web sehingga *log database* tersebut mudah untuk di analisa.

3. Perancangan Sistem

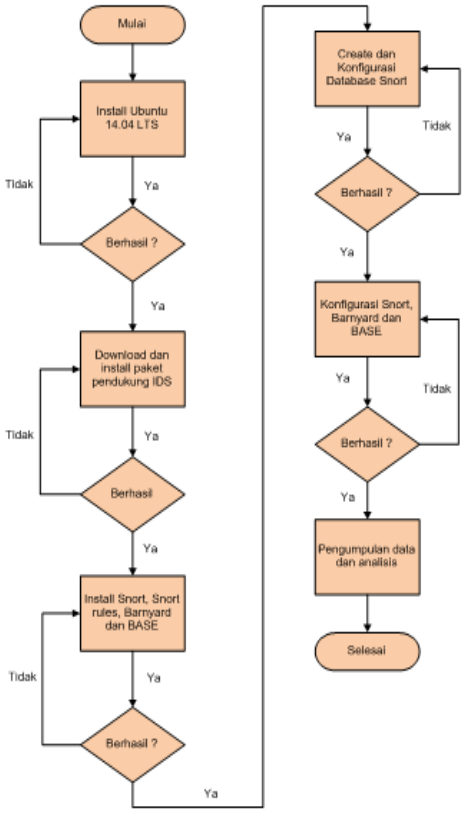


Gambar 4. Topologi Sistem Secara Umum

Dari gambar 3 di atas, dapat dijelaskan bagaimana topologi perancangan IDS:

1. Pada PC *Server* IDS dipasang satu *ethernet card* dengan alamat IP 192.168.0.
2. PC Server IDS dan client dihubungkan oleh sebuah *switch* agar semua perangkat dapat berkomunikas*i.*
3. untuk PC penyerang terdapat pada satu jaringan lokal dengan alamat IP 192.168.0.15
4. Untuk monitoring dilakukan pada PC client atau PC Server IDS.

Berikut ini diagram alir perancangan sistem.



Gambar 5.Diagram Alir Perancangan Sistem

Pada diagram alir diatas dijelaskan bahwa proses awal untuk merancang sistem ini diawali dengan instalasi sistem operasi, dan menginstall paket-paket pendukung dari IDS, konfigurasi Snort, Barnyard dan BASE.

3.1. Instalasi Paket Pendukung IDS

Setelah tahap instalasi sistem operasi selanjutnya adalah tahap instalasi paket-paket pendukung IDS tersebut.

```
# sudo apt-get install apache2
libapache2-mod-php5 mysql-server
mysql-common mysql-client php5-mysql
libmysqlclient-dev php5-gd php-pear
libphp-adodb php5-cli libwww-perl
libnet1 libnet1-dev libpcre3 libpcre3-
dev autoconf libcrypt-ssleay-perl
```

```
libtool libssl-dev build-essential
automake gcc make flex bison git
```

3.2. Instalasi Snort, Barnyard dan BASE

Tahap senjutnya adalah instalasi Snort.

```
# sudo apt-get install snort
```

Instalasi Barnyard.

```
# git clone
   http://github.com/firnsy/barnyard2.
   git barnyard2
# cd barnyard2
# autoreconf -fvi -I ./m4
# ./configure -with-mysql -with-mysql-
   libraries=/usr/lib/i386-linux-gnu
# make
# make install
# cp etc/barnyard2.conf
   /usr/local/snort/etc
# mkdir /var/log/barnyard2
# chmod 666 /var/log/barnyard2
# touch /var/log/snort/barnyard2.waldo
# chown snort.snort
   /var/log/snort/barnyard2.waldo
```

Instalasi BASE.

wget
http://sourceforget.net/projects/secureideas/files/BASE/base-1.4.5/base-1.4.5.tar.gz

```
# tar zxvf base-1.4.5.tar.gz
# mv base-1.4.5
```

3.3. Membuat dan konfigurasi Database Snort

Selanjutnya dilakukan tahap pembuatan database untuk Snort.

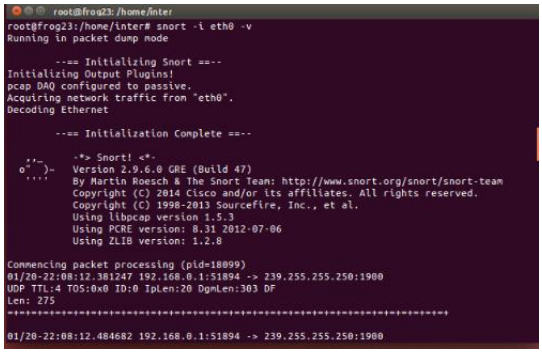
```
# mysql -u root -p (isi password
mysql)

Mysql > create database snortdb;
Mysql > grant create, insert, select,
delete, update on snortdb.* to
snortuser@localhost identified
'password';
Mysql > flush privileges;
Mysql > exit;
```

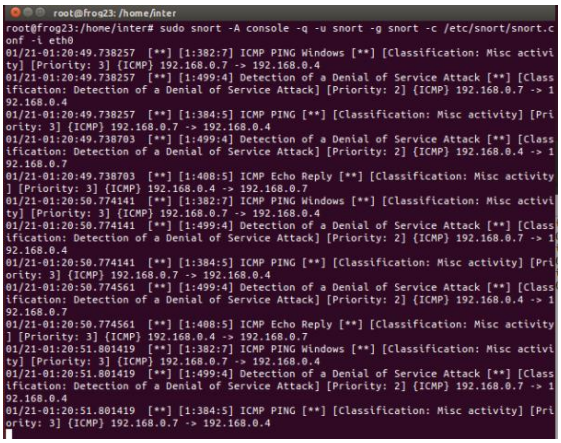
4. PENGUJIAN DAN ANALISA

4.1. Pengujian Fungsional Snort IDS

Pengujian ini dilakukan dengan cara mengecek fungsi-fungsi pada Snort Server dan memeriksa semua koneksi antara server IDS dan client.



Gambar 6. Status Snort

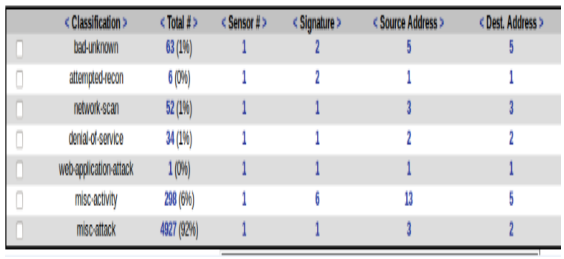


Gambar 7. Status snort.conf

4.2. Pengujian Snort Server Terhadap Gangguan

Pada pengujian ini dilakukan penyerangan dengan teknik gangguan yang bisa dilakukan para *hacker* terhadap sebuah jaringan yaitu DoS (Denial of Services) dan *Scanning Port* menggunakan Nmap.

Dari pengujian tersebut didapatkan beberapa katagori *alert* yang di baca oleh Snort IDS Server yang di tampilakan ke BASE.



Classification	Total #	Sensor #	Signature	Source Address	Dest. Address
bad-unknown	63 (1%)	1	2	5	5
attempted-recon	6 (0%)	1	2	1	1
network-scan	52 (1%)	1	1	3	3
denial-of-service	34 (1%)	1	1	2	2
web-application-attack	1 (0%)	1	1	1	1
misc-activity	288 (6%)	1	6	13	5
misc-attack	4927 (92%)	1	1	3	2

Gambar 8. Katagori Alert Snort IDS

Dari gambar 8 tersebut, terdapat *Classification* dari paket-paket data yang di baca oleh Snort IDS.

Lalu berikut ini adalah hasil *alert log* pada server Snort IDS saat terjadi gangguan DoS .

01/21-01:20:49.738257 [**] [1:382:7] ICMP PING Windows [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.0.7 -> 192.168.0.4

01/21-01:20:49.738257 [**] [1:499:4] Detection of a Denial of Service Attack [**] [Classification: Detection of a Denial of Service Attack] [Priority: 2] {ICMP} 192.168.0.7 -> 192.168.0.4
01/21-01:20:49.738257 [**] [1:384:5] ICMP PING [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.0.7 -> 192.168.0.4
01/21-01:20:49.738703 [**] [1:499:4] Detection of a Denial of Service Attack [**] [Classification: Detection of a Denial of Service Attack] [Priority: 2] {ICMP} 192.168.0.4 -> 192.168.0.7
01/21-01:20:49.738703 [**] [1:408:5] ICMP Echo Reply [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.0.4 -> 192.168.0.7
01/21-01:20:50.774141 [**] [1:382:7] ICMP PING Windows [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.0.7 -> 192.168.0.4
01/21-01:20:50.774141 [**] [1:499:4] Detection of a Denial of Service Attack [**] [Classification: Detection of a Denial of Service Attack] [Priority: 2] {ICMP} 192.168.0.7 -> 192.168.0.4

Log gangguan pada Terminal *Console* diatas mendeteksi ada nya gangguan DoS, dengan menunjukan waktu dan tanggal kejadian, jenis gangguan, klasifikasi dari gangguan dan alamat IP dari komputer penggangu ke komputer tujuan.



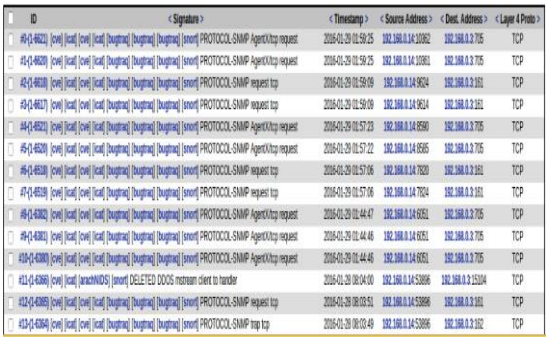
Gambar 9. Tampilan Alert DoS pada BASE

Pada gambar 9 dapat dilihat log gangguang yang disebabkan oleh DoS ditampilkan ke *interface* BASE dengan signature Snort Alert dan TimeStamp atau waktu kejadian yang lengkap dengan tanggal dan waktu kejadian, BASE juga dapat menampilkan Source IP Address penggangu dan IP Address mana yang dituju.

Berikut hasil dari log gangguan menggunakan Nmap yang ditampilkan pada terminal *console* :

01/21-01:18:18.091453 [**] [1:1421:11] SNMP AgentX/tcp request [**] [Classification: Attempted Information Leak] [Priority: 2] {TCP} 192.168.0.7:53984 -> 192.168.0.4:705
01/21-01:18:18.593977 [**] [1:1421:11] SNMP AgentX/tcp request [**] [Classification: Attempted Information Leak] [Priority: 2] {TCP} 192.168.0.7:53984 -> 192.168.0.4:705
01/21-01:18:19.102199 [**] [1:1421:11] SNMP AgentX/tcp request [**] [Classification: Attempted Information Leak] [Priority: 2] {TCP} 192.168.0.7:53984 -> 192.168.0.4:705
01/21-01:18:19.102199 [**] [1:1421:11] SNMP AgentX/tcp request [**] [Classification: Attempted Information Leak] [Priority: 2] {TCP} 192.168.0.7:53984 -> 192.168.0.4:705

Log diatas adalah tampilan log gangguan pada terminal *console* dimana juga terdapat waktu dan tanggal kejadian dan jenis gangguan yang di munculkan oleh Nmap, beserta klasifikasi dari gangguan tersebut dan alamat IP sumber ke tujuan.



Gambar 10. Tampilan Alert Nmap pada BASE

Pada gambar 10 dapat dilihat tampilan log gangguan pada BASE mencatat hasil dari *Port Scanner* yang dilakukan Nmap dimana pada *Signature* BASE menujukan adanya PROTOCOL-SNMP AgentX/tcp request, PROTOCOL-SNMP request tcp dan DELETED DDOS mnstream client to handler. Dan *Timestamp* pada BASE akan menunjukan tanggal dan waktu dari kejadian yang di baca oleh Snort IDS, sedangkan untuk *Source Address* dan *Dest. Address* merupakan tampilan dari IP penggagu (Source Address) ke IP komputer tujuan (Dest Address), bahkan BASE juga menampilkan pada layer mana gangguan tersebut masuk.

4.3. ANALISA

Dari dua hasil gangguan yang dibaca oleh Server IDS, dapat dilihat gangguan DoS dan *port scanner* yang dikirimkan kepada server IDS dapat di baca oleh Snort IDS dengan sangat baik, bahkan Snort IDS bisa mencatat kapan waktu gangguan dan alamat dari IP Address tersebut dan menampilkannya pada terminal *console* dan pada *interface* BASE.

Pada pengujian Server *Snort* IDS terhadap gangguan DoS dan *Prot Scanning* menggunakan Nmap, server Snort IDS juga bekerja dengan baik. Dari pengujian didapat beberapa *classification* gangguan berdasarkan level gangguan tersebut adalah :

1. *Attempted-recon* log gangguan ini termasuk dalam katagori (medium).
2. *denial-of-service* log gangguan ini termasuk dalam kategori (medium).
3. *misc-attack log* gangguan ini termasuk dalam kategori (medium).
4. *bad-uniknown* log gangguan ini termasuk dalam kategori (medium).
5. *network-scan* log gangguan ini termasuk dalam kategori (low).

5. KESIMPULAN

Berdasarkan pembahasan dan pengujian yang telah dilakukan terhadap Server Snort IDS maka dapat diambil kesimpulan terhadap rancangan Server IDS adalah sebagai berikut:

1. IDS yang dirancang untuk mendeteksi adanya gangguan terhadap suatu jaringan dapat dibaca dengan baik oleh Snort IDS .
2. Snort IDS yang diinstal pada sistem operasi Linux Ubuntu Desktop 14.04 LTS dapat berjalan dengan baik dan dapat digunaka dalam pemantauan jaringan.
3. Dari hasil pengujian gangguan terhadap Server Snort IDS. Snort IDS dapat mengenali jenis gangguan yang ditimbulkan dan dapat menampilkan secara cepat dan tepat kapan terjadinya gangguan dan dari mana asal gangguan tersebut.

6. Saran

Saran yang diajukan untuk pengembangan selanjutnya dan melengkapi kekurangan-kekurangan adalah sebagai berikut :

1. Adanya penanggulangan secara langsung terhadap gangguan yang masuk atau yang terbaca oleh Snort IDS.
2. Dilakukannya pengujian gangguan terhadap jaringan yang berbeda, tidak hanya pada dalam satu jaringan saja.

7. REFERENSI

[1] Behrouz A. Forouzan, Data Communication and Networking, Mc-Graw Hill, Fourth Edition, 2007.

[2] Rando Veizi. IDS/IPS: Intrusion Detection/Prevention System

[3] Abraham Nethanel Setiawan Junior, Agus Harianto, Alexander. Perancangan dan Implementasi Intrusion Detection System pada Jaringan Nirkabel BINUS University

[4] Rehman, Rafeeq Ur. Intrusion Detection Systems with Snort, (Online), diakses pada tanggal 26 Maret 2015 dari word wide web : http://ptgmedia.pearsoncmg.com/images/0131407333/downloads/0131407333.pdf

[5] www.snort.org