IMPLEMENTASI INTRUSION PREVENTION SYSTEM (IPS) MENGGUNAKAN SNORT DAN IP TABLES BERBASIS LINUX

Yoga Widya Pradipta
D3 Manajemen Informatika, Fakultas Teknik, Universitas Negeri Surabaya, yogapradipta26@gmail.com

Asmunin
Jurusan Teknik Informatika, Fakultas Teknik, Universitas Negeri Surabaya, asmunin@unesa.ac.id

Abstrak

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

IPS (*Network Intrusion Prevention System*) mampu mendeteksi serangan dan melakukan drop pada serangan. Melakukan penerapan pada sistem operasi Linux menggunakan Snort dalam mode inline dan mampu mencegah dari serangan yang dapat mengancam..

Kata Kunci: *Keamanan Jaringan, Intrusion Prevention System, Denial Of Service, Linux, Snort*

Abstract

Network security system becomes very important in maintaining a network, attacks that can disturb and even damage the connection system between devices connected will be very harmful. To gain security in a network sometimes we have to feel the discomfort in its use, this is often a consideration in the application of a network security system.

IPS (Network Intrusion Prevention System) is capable of detecting attacks and dropping attacks. Performing on a Linux operating system using Snort in inline mode and able to prevent from threatening attacks

Keywords: *Network Security, Intrusion Prevention System, Denial Of Service, Linux, Snort*

PENDAHULUAN

Perkembangan teknologi yang sangat pesat menuntut meningkatnya kualitas keamanan jaringan. Terutama dengan semakin terbukanya pengetahuan tentang *hacking* dan *cracking* yang didukung oleh *tools* yang bisa didapatkan dengan mudah dan gratis. Selain itu ancaman keamanan jaringan komputer juga datang dari virus, *malicious*, trojan, *worm*, *DOS*, *spoofing*, *sniffing*, *spamming*, dan lainnya. Hal – hal inilah yang akan mengancam keamanan sebuah sistem jaringan dimana data dapat dengan mudah diambil bahkan dirusak oleh *intruder* atau *attacker*.

Banyak metode yang bisa dilakukan untuk dapat mengamankan sebuah sistem jaringan. Salah satunya adalah dengan menggunakan *Intrusion Prevention System* (IPS). IPS sendiri merupakan kombinasi antara fasilitas *blocking capabilities* dari Firewall dan kedalaman inspeksi paket data dari *Intrusion Detection System* (IDS). Pada saat bekerja, IPS akan membuat akses kontrol dengan cara melihat konten aplikasi sehingga IPS mampu mencegah serangan yang datang dengan bantuan administrator dan akan menghalangi suatu serangan sebelum terjadi eksekusi dalam memori. Diperlukannya implementasi seperti judul:

"IMPLEMENTASI INTRUSION PREVENTION SYSTEM (IPS) MENGGUNAKAN SNORT DAN IP TABLES BERBASIS LINUX".

Tujuan dari penelitian tugas akhir ini adalah untuk mengetahui, melakukan *drop* dan menerapkan cara kerja IPS (*Intrusion Prevention System*) dalam mendeteksi serangan pada sistem jaringan komputer.

Manfaat yang didapat dari pembuatan tugas akhir ini adalah untuk dapat mendeteksi serangan DOS, dapat melakukan *block* otomatis pada *attacker* dan mampu menangkal *attacker* untuk mecegah terjadinya akses masuk pada server.

Implementasi aplikasi hanya sebatas pada pembuktian bahwa aplikasi dapat berjalan di atas sistem yang dibangun.

KAJIAN PUSTAKA

Keamanan Jaringan

Keamanan jaringan ialah sistem perlindungan terhadap jaringan serta aplikasi-aplikasinya dari serangan-serangan atau kegiatan yang dapat mengancam validitas dan integritas data. (Binanto, 2007)

Jenis jenis serangan yang biasa terdeteksi oleh IDS :

1. Spoofing
   Spoofing adalah Teknik yang digunakan untuk memperoleh akses yang tidak sah ke suatu computer atau informasi, dimana penyerang berhubungan

dengan pengguna dengan berpurapura memalsukan bahwa mereka adalah *host* yang dapat dipercaya. Hal ini biasanya dilakukan oleh seorang *hacker/cracker*.

2. Ddos (Distributed Denial of Service)
   Serangan DOS (*Denial Of Service attacks*) adalah jenis serangan terhadap sebuah komputer atau server di dalam jaringan internet dengan cara menghabiskan sumber (*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 diserang tersebut.
3. Sniffer
   Sniffer Paket atau penganalisa paket (arti tekstual: pengendus paket — dapat pula diartikan 'penyadap paket' yang juga dikenal sebagai *Network Analyzers* atau *Ethernet Sniffer* ialah sebuah aplikasi yang dapat melihat lalu lintas data pada jaringan komputer. Dikarenakan data mengalir secara bolak balik pada jaringan, aplikasi ini menangkap tiap tiap paket dan kadang kadang menguraikan isi dari RFC (*Request for Comments*) atau spesifikasi yang lain.
4. SQL Injection
   Injeksi SQL atau SQL Injection memiliki makna dan arti yaitu sebuah teknik yang menyalahgunakan sebuah celah keamanan yang terjadi dalam lapisan basis data sebuah aplikasi.Celah ini terjadi ketika masukan pengguna tidak disaring secara benar dari karakter karakter pelolos bentukan string yang diimbuhkan dalam pernyataan SQL atau masukan pengguna tidak bertipe kuat dan karenanya dijalankan tidak sesuai harapan. Ini sebenarnya adalah sebuah contoh dari sebuah kategori celah keamanan yang lebih umum yang dapat terjadi setiap kali sebuah bahasa pemrograman atau skrip diimbuhkan di dalam bahasa yang lain. SQL injection adalah jenis aksi hacking pada keamanan komputer di mana seorang penyerang bisa mendapatkan akses ke basis data di dalam sistem. S. SQL injection exploits dan sejenisnya adalah hasil *interfacing* sebuah bahasa lewat informasi melalui bahasa lain. Dalam hal SQL injection, sebuah bahasa pemrograman seperti PHP atau Perl mengakses database melalui SQL query. Jika data yang diterima dari pengguna akhir yang dikirim langsung ke database dan tidak disaring dengan benar, maka yang penyerang dapat menyisipkan perintah SQL nya sebagai bagian dari input.
5. *Man-in-The-Middle* (MiTM) *attacking*
   Serangan ini terjadi saat *attacker* bertindak sebagai perantara diantara dua *node*yang saling berkomunikasi. *Attacker* tidak akan tampak pada kedua sisi *node* tersebut dan dapat melihat atau mengubah isi dari *traffic*.

IPS (*Intrusion Prevention System*)

Intrusion Prevention System (IPS) adalah sebuah perangkat lunak atau perangkat keras yang bekerja untuk *monitoring* trafik jaringan, mendeteksi aktivitas yang mencurigakan dan melakukan pencegahan dini terhadap penyusupan atau kejadian yang dapat membuat jaringan menjadi berjalan tidak seperti sebagaimana mestinya. IPS merupakan pendekatan yang sering digunakan untuk membangun sistem keamanan komputer, IPS mengombinasikan teknik *firewall* dan metode *intrusion detection system* (IDS) dengan sangat baik. Teknologi ini dapat digunakan untuk mencegah serangan yang akan masuk ke jaringan lokal dengan memeriksa dan mencatat semua paket data serta mengenali paket dengan sensor saat seragan teridentifikasi. Jadi IPS bertindak seperti layaknya *firewall* yang akan mengizinkan atau menghalang paket data. (Raven Alder, 2007)

Secara khusus, IPS memiliki empat komponen utama, yaitu:

1. *Normalisasi Traffic*: menginterpretasikan *traffic* jaringan dan melakukan analisa terhadap paket yang disusun kembali, seperti halnya fungsi *block* sederhana.
2. *Detection Engine*: mendeteksi *traffic* jaringan dan melakukan *patternmatching*terhadap tabel acuan dan respon yang sesuai.
3. *Service Scanner*: membangun suatu tabel acuan untuk mengelompokkan informasi.
4. *Traffic Shaper*: membentuk dan mengatur *traffic* jaringan.

Ada 2 jenis IPS, yaitu *Host Based Intrusion Prevention System* (HIPS) dan *Network Based Intrusion Prevention System*(NIPS).

***1. Host Intrusion Prevention System* (HIPS)**

Host-based Intrusion Prevention System (HIPS) sama seperti halnya *Host Based Intrusion Detection System* (HIDS). Program agent HIPS diinstall secara langsung di sistem yang diproteksi untuk dimonitor aktifitas sistem internalnya. HIPS di binding dengan kernel sistem operasi dan *services* sistem operasi sehingga HIPS bisa memantau dan menghadang *system call* yang dicurigai dalam rangka mencegah terjadinya intrusi terhadap *host*. HIPS juga bisa memantau aliran data dan aktivitas pada aplikasi

tertentu. Sebagai contoh HIPS untuk mencegah *intrusion* pada *webserver* misalnya. Dari sisi *security* mungkin solusi HIPS bisa mencegah datangnya ancaman terhadap *host*. Tetapi dari sisi *performance*, harus diperhatikan apakah HIPS memberikan dampak negative terhadap *performance host*. Karena mengi*nstall* dan binding HIPS pada system operasi mengakibatkan penggunaan *resource* komputer *host* menjadi semakin besar.

***2. Network Intrusion Prevention System* (NIPS)**

Network-based Intrusion Prevention System (NIPS) tidak melakukanpantauan secara khusus di satu host saja. Tetapi melakukan pantauan dan proteksi II-16 dalam satu jaringan secara *global*. NIPS menggabungkan fitur IPS dengan *firewall*dan kadang disebut sebagai *In-Line* IDS atau *Gateway Intrusion Detection System* (GIDS). Sistem kerja IPS yang populer yaitu pendeteksian berbasis *signature*, pendeteksian berbasis anomali, dan monitoring *file* pada sistem operasi *host*.

a. Sistematika IPS yang berbasis *signature* adalah dengan cara mencocokkan lalu lintas jaringan dengan *signature database* milik IPS yang berisi *attacking rule* atau cara-cara serangan dan penyusupan yang sering dilakukan oleh penyerang. Sama halnya dengan antivirus, IPS berbasis *signature* membutuhkan *update* terhadap *signature database* untuk metode-metode penyerangan terbaru. IPS berbasis *signature* juga melakukan pencegahan terhadap ancaman intrusi sesuai dengan *signature database* yang bersangkutan.

b. Sistematika IPS yang berbasis anomali adalah dengan cara melibatkan pola-pola lalu lintas jaringan yang pernah terjadi. Umumnya, dilakukan dengan menggunakan teknik statistik. Statistik tersebut mencakup perbandingan antara lalu lintas jaringan yang sedang di *monitor* dengan lalu lintas jaringan yang biasa terjadi (*normal state*). Metode ini dapat dikatakan lebih kaya dibandingkan *signature-based* IPS. Karena *anomalybased* IPS dapat mendeteksi gangguan terhadap jaringan yang terbaru yang belum terdapat di *database* IPS. Tetapi kelemahannya adalah potensi timbulnya *false positive*, yaitu pesan/log yang belum semestinya dilaporkan. Sehingga tugas *Network Administrator* menjadi lebih rumit, dengan harus memilah-milah mana yang merupakan serangan yang sebenarnya dari banyaknya laporan *false positive* yang muncul.

Teknik lain yang digunakan adalah dengan cara melakukan monitoring berkas-berkas sistem operasi pada *host*. IPS akan melihat apakah ada percobaan untuk mengubah beberapa berkas sistem operasi, utamanya berkas log. Teknik ini diimplementasikan dalam IPS jenis *Host Based Intrusion Prevention System* (HIPS). Teknik yang digunakan IPS untuk mencegah serangan ada dua, yaitu *sniping* dan *shunning*.

a. *Sniping*: memungkinkan IPS untuk menterminasi serangan yang dicurigai melalui penggunaan paket TCP RST atau pesan ICMP *Unreachable*.

b. *Shunning*: memungkinkan IPS mengkonfigurasi secara otomatis *firewall* untuk melakukan *drop traffic* berdasarkan apa yang dideteksi oleh IPS. Untuk kemudian melakukan *prevention* atau pencegahan terhadap koneksi tertentu.

Inline dan Passive Mode

IPS merupakan pengembangan dari IDS Sebagai IPS, Snort hanya menganalisa paket yang ada dan memberikan peringatan bila terjadi serangan dari *hacker*. Jika seperti ini kasusnya, IDS dikatakan bekerja dalam modus pasif. Bila ingin Snort memblokir upaya serangan dan memberikan respon atas serangan *attacker* maka Snort harus berkerja sebagai IPS, Snort akan berfungsi sebagai IPS bila berjalan dalam modus *inline*. Dari ujicoba serangan *backdoor* dan *synflood* yang telah dilakukan terbukti bahwa Snort Inline dapat melakukan drop terhadap serangan *backdoor* dan *synflood* dapat disimpulkan bahwa metode IPS lebih handal daripada Metode IDS yang hanya menganalisa packet yang ada. Sehingga disarankan untuk meningkatkan kemampuan sistem pada masa yang akan dating.

Snort IDS

Snort merupakan sebuah produk terbuka yang dikembangkan oleh Marty Roesch dan tersedia gratis di www.snort.org. Snort bisa digunakan pada sistem operasi Linux, Windows, BSD, Solaris dan sistem operasi lainnya. Snort merupakan IDS berbasis jaringan yang menggunakan metode deteksi *rule based*, menganalisis paket data apakah sesuai dengan jenis serangan yang sudah diketahui olehnya. Snort digunakan karena memiliki beberapa kelebihan berikut: mudah dalam konfigurasi dan penambahan aturan-aturan, gratis, dapat berjalan pada sistem operasi yang berbeda-beda. (Cox, Kerry, 2004.)

DOS (*Denial Of Service*)

Denial of Service (DOS) adalah jenis serangan terhadap sebuah komputer atau server di dalam jaringan internet dengan cara menghabiskan sumber (*resource*)

yang dimiliki. Bentuk umum dari serangan DoS ini adalah dengan cara mengirim paket data dalam jumlah yang sangat bersar terhadap suatu *server* dimana *server* tersebut tidak bisa memproses semuanya. Bentuk lain dari serangan DoS ini adalah memanfaatkan *port-port* yang rentan dari sistem operasi. Tidak semua DoS merupakan akibat dari serangan keamanan jaringan. Kesalahan dalam *coding* suatu program juga bisa mengakibatkan kondisi seperti serangan DoS. Ada beberapa jenis dari DoS, antara lain:

1. *Distributed Denial of Service* (DDoS)
   Terjadi saat penyerang berhasil menggabungkan beberapa layanan system dan menggunakannya sebagai pusat untuk menyebarkan serangan terhadap korban.
2. *Distributed Reflective Denial of Service* (DRDoS)
   Memanfaatkan operasi normal dari layanan internet seperti *protocol protocol update* DNS dan router. DRDoS ini menyerang fungsi dengan mengirim update dalam jumlah yang sangat besar kepada berbagai macam layanan server atau router dengan *menggunakan address spoofing* kepada target korban.
3. *SYN flooding*
   Upaya untuk membanjiri sinyal SYN kepada sistem yang menggunakan protocol TCP/IP dalam melakukan inisiasi sesi komunikasi.
4. *Smurf Attack*
   Server digunakan untuk membanjiri korban dengan data sampah yangtidak berguna. Server atau jaringan yang dipakai menghasilkan respon paket yang banyak seperti ICMP ECHO paket atau UDP paket dari satu paket yang dikirim.
5. *Ping of Death*
   Dengan menggunakan tool khusus, penyerang dapat mengirimkan paket ping yang *oversize* yang banyak kepada korban. *Ping of death* tidak lebih dari semacam serangan *buffer overflow*. Serangan ini dapat menyebabkan *crash* sistem, f*reeze* atau *reboot*.
6. *Stream Attack*
   Serangan ini terjadi saat banyak jumlah paket yang besar dikirim menuju ke *port* pada sistem korban menggunakan sumber nomor yang random.

Fail2Ban

Fail2ban merupakan salah satu aplikasi yang membantu administrator dalam mengamankan jaringan. Fail2ban beroperasi dengan memblokir IP yang mencoba melanggar keamanan sistem. Alamat IP yang diblokir dapat dilihat pada file log (misalnya: `/var/log/pwdfail`, `/var/log/auth.log`, dan lain-lain) dan melarang setiap IP yang berupaya *login* terlalu banyak atau melakukan tindakan yang tidak diinginkan lainnya dalam jangka waktu yang ditetapkan oleh administrator.

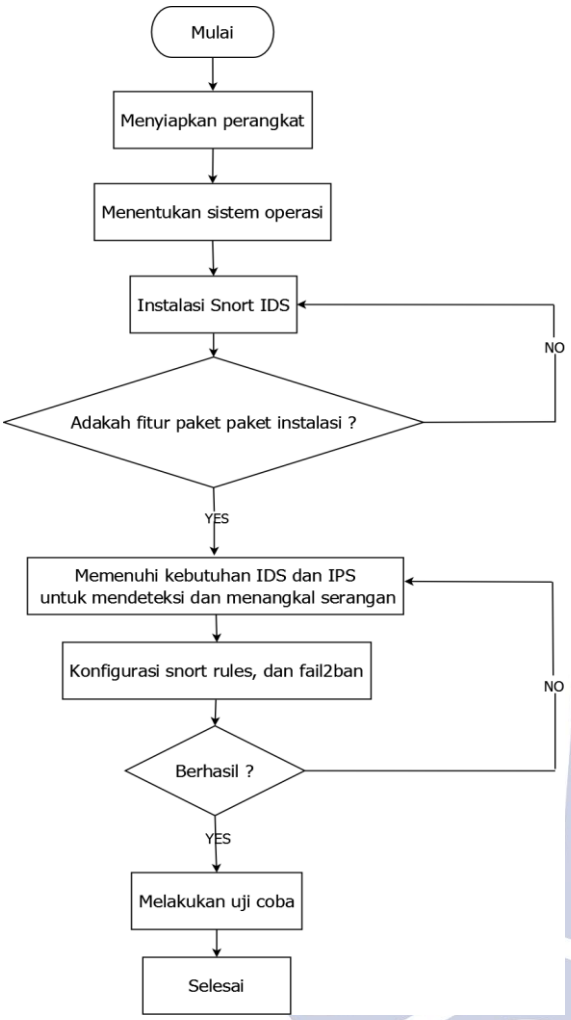
Fail2ban biasanya dirancang untuk membuka atau membolehkan host yang diblokir dalam jangka waktu tertentu, sehingga tidak mengunci setiap koneksi yang mungkin telah terkonfigurasi sementara. Namun, mengunci IP dalam waktu beberapa menit biasanya cukup untuk menghentikan koneksi jaringan yang sedang melakukan *flood* atau membanjiri system jaringan, serta mengurangi kemungkinan suksesnya untuk melakukan *dictionary attack*. (Daranto, Michael, 2007)

NMAP (*NETWORK MAPPER*)

Merupakan salah satu *tools* eksplorasi jaringan, dan secara eksklusif menjadi salah satu andalan yang sering digunakan oleh administrator jaringan. Dengan Nmap kita dapat melakukan penelusuran ke seluruh jaringan dan mencari tahu *service* apa yang aktif pada port yang lebih spesifik. Nmap merupakan salah satu *tools* yang paling banyak digunkan untuk melakukan scanning jaringan dan terkenal sebagai tool yang multi platform, cepat dan ringan. Nmap berjalan pada semua jenis OS, baik mode console maupun grafis.

METODE

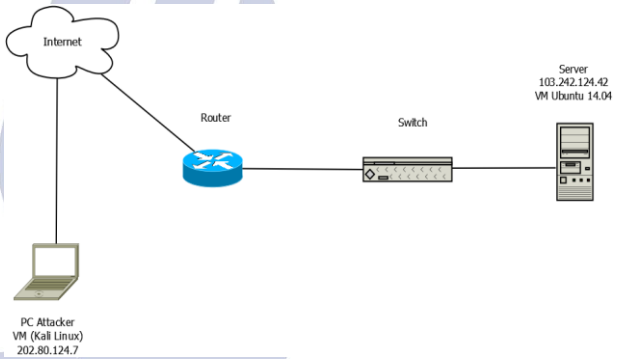
Analisis Sistem



Gambar 1. Flowchart alur instalasi Snort

Pada gambar 2, dijelaskan bahwa Snort umumnya pada dasarnya memang banyak digunakan sebagai IDS namun ada kalanya Snort juga dapat difungsikan sebagai IPS untuk langkah pendeteksian dan penanganan masalah pada serangan-serangan jaringan. Dalam konsep topologi SNORT IDS dan IPS memiliki perbedaan pada sisi topologi jaringan yang dibuat, yakni pada peletakan posisi server dimana Snort berada. Pada IDS topologi yang dibuat umumnya server yang difungsikan sebagai SNORT IDS berada pada posisi yang sama dan satu network dengan network yang ingin dipantau aktivitasnya guna pendeteksian dini akan serangan-serangan yang masuk, sehingga snort dapat melihat segala aktivitas yang berjalan menuju network yang dilindungi.

Berbeda pada topologi IPS Snort, pada konsep IPS, server snort harus menjembatani antara network yang dilindungi dan network luar, jadi server snort harus memiliki 2 ethernet sebagai penghubung antara network arah keluar dan satu ethernet lagi sebagai penghubung ke arah network yang ingin dilindungi. Jadi konsep IPS ini semua koneksi yang masuk ke arah server yang dilindungi, sebelumnya semua koneksi juga sudah melewati server snort terlebih dahulu, jadi jika ada serangan maka akan lebih cepat dalam mendapatkan alert atau peringatan dari adanya paket-paket yang mencurigakan, serta snort dapat dapat langsung menghentikan serangan yang masuk, baik berupa drop paket, menolak (reject) paket dan dengan cara pemblokiran yang lain melalui rule. Adapun untuk melakukan drop paket dapat menambahkan perintah pada rule-rule yang dibuat pada snort. Untuk menerapkan metode IPS snort, konfigurasi pada snort.conf harus dalam mode inline, karena hanya dengan mode inline maka memungkinkan SNORT untuk melihat setiap paket dan menangani paket yang mencurigakan secara langsung.



Gambar 2. Topologi.

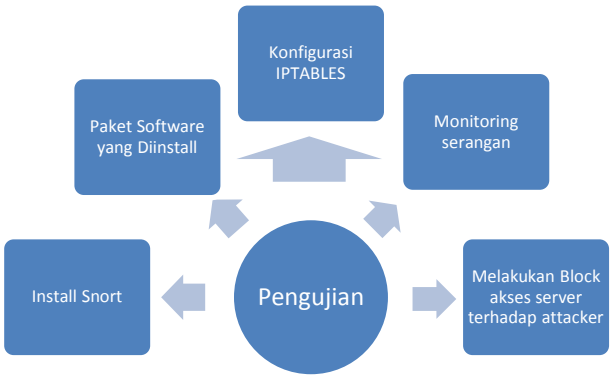
Kebutuhan Perangkat

Berikut ini beberapa kebutuhan perangkat dalam pengujian.

Tabel 1. Kebutuhan perangkat

CPU	Memory (RAM)	Harddisk	Network
Intel Core I5-2500 @3.30 GHz	2 Gigabyte	SATA 465 GB	Intel GigaBit NICs

Skenario Pengujian



Gambar 3. Skenario Pengujian

Penjelasan dari gambar 3 adalah sebagai berikut :

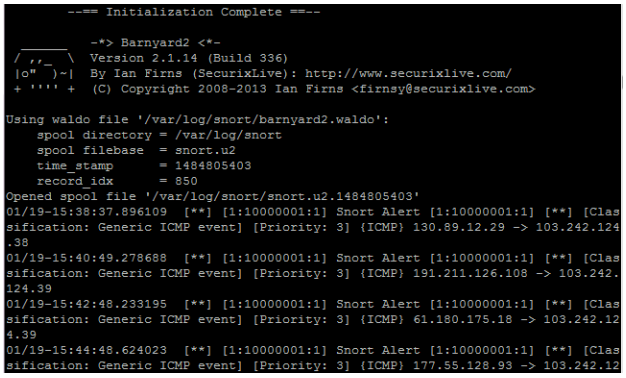
1. Install Snort
   Disini penulis akan memberikan langkah-langkah untuk menginstall Snort dengan benar, karena jika ada satu packet yang tidak terinstall, maka file biasanya akan *corrupt*.
2. Paket software yang dinstall
   Disini penulis memberitahukan bahwa setiap paket software yang diperlukan dalam tugas akhir ini sangat banyak, dan penulis ingin pembaca benar-benar memahami setiap paket yang diinstall oleh penulis.
3. Konfigurasi IPTABLES
   Disini penulis memberikan *rule* atau aturan untuk konfigurasi IPTABLES
4. *Monitoring* serangan
   Disini penulis memberikan tahapan dari judul yang dibuat, tujuan penulis membuat tugas akhir ini adalah untuk melakukan *block* atau melakukan *filter* paket serangan. Tetapi untuk bisa melakukan *block*. Penulis terlebih dahulu harus memonitoring paket serangan yang berjalan dengan menggunakan IDS, setelah diketahui serangan yang ada, baru penulis menggunakan mode IPS untuk melakukan *block* dan *filter*.
5. Melakukan *block* akses server terhadap attacker
   Disini penulis membuat langkah terakhir yaitu melakukan *block* terhadap *attacker* agar tidak dapat melakukan hak akses masuk pada server.

HASIL DAN PEMBAHASAN

1. Monitoring serangan

Mengirim serangan pada barnyard

Setelah melakukan *installasi* barnyard dengan sukses, penulis akan memaparkan hasil serangan yang lewat pada barnyard, DOS akan mengirimkan alert detected berupa ICMP. Sehingga serangan dapat ter-*monitoring* pada barnyard.

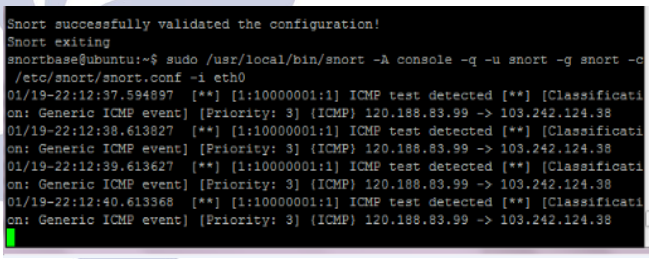


Gambar 4 Monitoring serangan pada barnyard

Dijelaskan bahwa Snort alert sudah bisa bekerja mengirimkan paket serangan DDOS berupa PING, pada gambar 4.69 sudah terdapat IP Address 130.89.12.29 melakukan PING terhadap IP server dan fungsinya juga untuk melemahkan server.

2. Mendeteksi serangan pada Snort

Setelah melakukan *installasi* dan konfigurasi pada Snort, selanjutnya penulis akan melakukan deteksi serangan pada Snort. Sama seperti pada barnyard, attacker akan mencoba melakukan PING pada IP server 103.242.124.42 untuk melemahkan kinerja server.

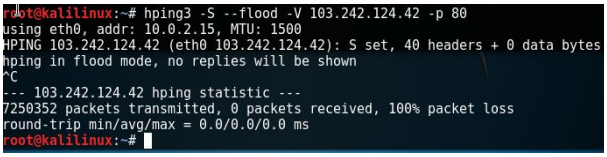


Gambar 5 Snort deteksi serangan

3. Mendeteksi serangan *SyNFlooding*

a. Penulis memberikan informasi tentang serangan DOS SynFlooding pada server, biasanya jika server tiba-tiba *down*, sedangkan penggunaan *traffic* web rendah. Hal ini bisa terjadi karena adanya serangan DOS, mulai dari yang ringan hingga yang berat. DOS singkatan dari *Denial Of Service*. Model serangannya mirip, tapi jumlah pelakunya yang beda. Pertama yang dilakukan terlebih dahulu yaitu install paket hping3.

b. Setelah itu, lakukan dengan perintah berikut: `hping3 -S --flood -V 103.242.124.42 -p 80` perintah tersebut di ketik di PC attacker (KaliLinux).



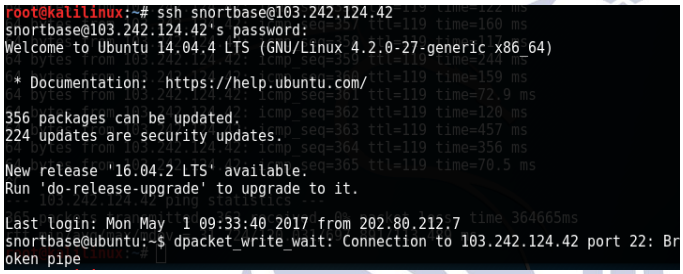
Gambar 6. Proses attacker menyerang server

4. Melakukan *block* akses server terhadap *attacker*

Mengatasi hak akses agar attacker tidak bisa masuk pada server

a. Pertama kita lakukan percobaan sebelum dilakukannya blocking pada attacker, lakukan dengan perintah berikut:

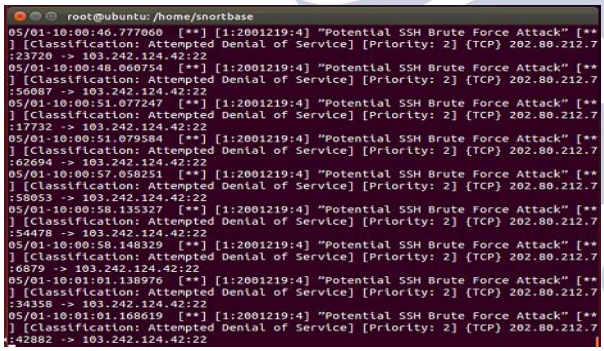
```
ssh snortbase@103.242.124.42
```



Gambar 7. Login SSH

b. Setelah itu mulai melakukan hak akses terhadap server. Maka digunkan perintah berikut:

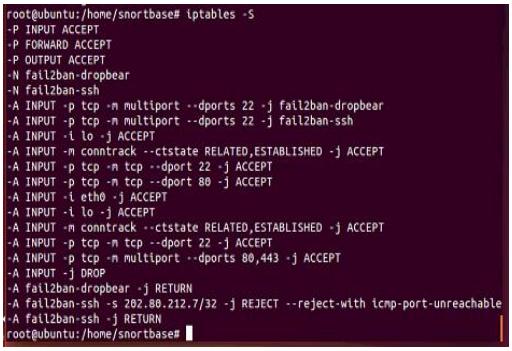
```
hydra -l root -P
/home/yoga/passlist.txt
103.242.124.42 ssh
```



Gambar 8. Melakukan running tes terdeteksi

c. Lalu jika ingin melihat hasil atau record IP attacker yang terkena *block*. Setelah itu lakukan dengan perintah berikut:

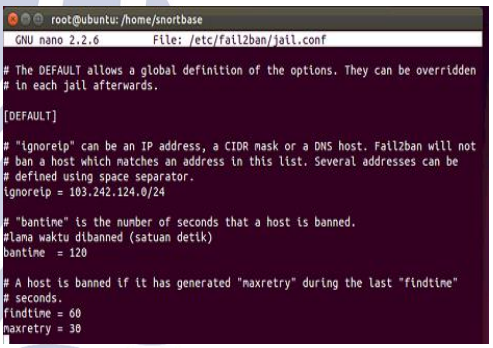
```
Iptables -S
```



Gambar 9. Record block alamat IP attacker

d. Jika penulis ingin mengubah lama ban time, maka dapat diatur untuk berapa detik. Maka digunakan perintah berikut:

```
etc/fail2ban/jail.conf
```



Gambar 10. Konfigurasi Fail2Ban

e. Yang terakhir, penulis akan melakukan tes *attacker* apakah bisa melakukan akses terhadap server.

Pada gambar 11 bahwa attacker sudah tidak dapat melakukan hak akses terhadap server

Gambar 11. Block akses attacker

PENUTUP

Simpulan

Dari hasil implementasi dan pengujian dapat diambil kesimpulansebagai berikut :

1. Sistem operasi Linux bisa berjalan di vmware
2. Konfigurasi Snort menggunakan 2 cara, pertama melalui konfigurasi IDS, setelah itu konfigurasi IPTABLES
3. Paket-paket software yang diinstall adalah pelengkap agar proses dalam pengerjaanya berjalan

dengan baik

4. Jenis serangan akan muncul di tampilan monitoring snort dan barnyard, dan juga dapat mengetahui paket TCP, IP server dan IP *intruder* serangan.
5. Metode IPS dapat memonitoring hingga melakukan *drop* dan melakukan filter jenis serangan

Saran

Konfigurasi *Intrusion Prevention System* (IPS) yang dibangun dalam penyelesaian tugas akhir ini masih jauh dari sempurna, untuk itu konfigurasi IPS ini diharapkan dapat menjadi bahan atau salah satu referensi bagi pembaca dan pengembang lainnya agar dapat terciptanya teknologi awan yang lebih baik lagi. Beberapa saran yang diperlukan antara lain:

1. Menambah rules pada Snort
2. Melakukan aplikasi ke dalam system operasi windows dan lain-lain
3. Dalam proses virtualisasi diharapkan mempunyai RAM laptop yang besar

DAFTAR PUSTAKA

Alder, Raven. Snort 2.1 Intrusion Detection, Second Edition. Rockland, MA 02370: Syngress Publishing, Inc. 2004

Ardiyanto, Yudhi, 2010. *System Instrusion Detection System*. Sourcefire Inc. United States of America.

Ashari, Ahmad, dkk. 2010. *Linux Sistem Administrasi.* Bandung: Informatika.

Cox, Kerry, 2004. Managing Security With Snort and IDS Tools. O'Reilly Media Inc. United States of America

Daranto, Michael. (2007). Fail2ban di Slackware v12.x | Slackerbox. Diambil 26 Februari 2017. http://www.slackerbox.com/node/552

Endorf, Schultz dan Mellander, 2005 IDS, BASE NIDS. *International Journal of Information & Network Security (IJINS) (ONLINE)*. Vol 2,2. Dari *http://www.iaesjournal.com/online/index.php/IJINS/article/download/1753/685* Diakses 22 November 2016

Iwan, Binanto, (2007), "Membangun Jaringan Komputer Praktis Sehari-hari", Graha Ilmu, Candi Gebang Permai, Yogyakarta.

Jason Weir, Snort 2.9.6.x on Debian 7.6, https://s3.amazonaws.com/snort-org-site/production/document_files/files/000/000/049/ Diakses 25 November 2106

P. O'Rourke and M. Keefe. 2001. *Performance Evaluation of Linux Virtual Server*.

Verma K, Hasbullah H, Kumar.A "An Efficient Defense Method agains UDP-Spoofed Flooding Traffic of Denial Of Service (DOS) attacks in VANET" IACC, 2013