Indonesian Journal of Applied Informatics
Vol. 3 No. 2 Tahun 2019 pISSN: 2548-3846 eISSN: 2598-5981

Pembuatan *Web interface snort* untuk Managemen Firewall dengan Operasi CRUD (Create, Read, Update, Delete) pada File System Snort dan Pengujian Web dengan Serangan serta Notifikasi pada Email dan Telegram

Rudi Hartono
Fakultas MIPA, Program Studi D3 Teknik Informatika
Universitas Sebelas Maret
rudi.hartono@staff.uns.ac.id

Muhamad Agung Sabekti
Fakultas MIPA, Program Studi D3 Teknik Informatika
Universitas Sebelas Maret
muhamadagung@student.uns.ac.id

Info Artikel

Kata Kunci :
Aplikasi firewall, alert, snort, web interface, rule snort, serangan, notifikasi.

Keywords :
Firewall application, alert, snort, web interfaces, snort rule, attack, notification.

Tanggal Artikel
Dikirim : 12 Februari 2019
Direvisi : 21 Februari 2019
Diterima : 10 Mei 2019

Abstrak

snort merupakan salah satu aplikasi *firewall* yang dikonfigurasi dalam terminal linux, meliputi konfigurasi *snort*, *input rule snort*, dan hasil alert *snort* pada terminal linux. Untuk mempermudah monitoring alert diterminal linux, maka alert diimplementasikan pada email dan telegram serta guna mempermudah dalam aktifitas dalam aktifitas input rule snort maka dibuatlah *web interface snort* . Metode untuk menangani Snort berjalan pada *mode inline* dengan menggunakan modul *daq_afpacket* dalam snort itu sendiri, dan untuk melakukan blok ketika terjadi serangan, snort menggunakan *firewall iptables*. Alert diimplementasikan pada email menggunakan protokol *ssmtp* dan pada telegram menggunakan id dan api telegram. Hasil dari penelitian menyatakan pembuatan web *interface* dapat dengan mudah mengelola *rule* dan alert *snort*, serta dapat diaplikasikan dalam beberapa serangan yang diujikan.

Abstarct

In general, snort is a firewall application that is configured in Linux terminals, including the implementation of snort, input snort rules, and snort warning results on Linux terminals. To monitor the linux warning alarm, the alerts are implemented on e-mail and telegram, as well as for input information in snort mode and then create a snort web interface. The method for handling Snort runs in inline mode by using the daq_afpacket module in the snort itself, and to block when an attack occurs, snort uses the iptables firewall. Alerts are implemented in e-mail using the ssmtp protocol and on telegrams using id and telegram fires. The results of the study state that making web interfaces can easily manage rules and snort alerts, and can be applied in several attacks that are tested.

1. PENDAHULUAN

Perkembangan internet yang semakin pesat membuat keamanan jaringan menjadi salah satu aspek yang wajib terpenuhi untuk menjaga stabilitas, kelancaran dan kehandalan sistem. Snort merupakan sebuah tools aplikasi keamanan yang berfungsi untuk mendeteksi intrusi-intrusi jaringan meliputi peyusupan, penyerangan, pemindaian [1] dan beragam bentuk ancaman lainnya yang dimanfaatkan untuk mempermudah menganalisa trafik data internet yang masuk guna meminimalisir adanya kejanggalan paket yang tidak diinginkan yang bisa membebani perangkat bahkan membuat perangkat jaringan menjadi berhenti berfungsi normal (*hang*) hingga arus data koneksi jaringan terganggu [2].

Snort yang sebelumnya telah berjalan dalam mode command line telah bekerja dengan baik untuk menjalankan fungsi dan rule nya, seiring perkembangan jaman, pengoperasion aplikasi snort dikembangkan untuk mempermudah merekam jejak log aktifitas jaringan sehingga tidak hanya dapat dikonfigurasi dalam mode *cli* (*command line interface*), namun log dan penambahan rule serangan disajikan dalam web interface dan notifikasi nya dalam email beserta telegram. Aplikasi snort ids akan melakukan pemberitahuan saat mendeteksi sesuatu yang dianggap sebagai suatu hal yang mencurigakan atau tindakan yang berbahaya dan tidak wajar [3].

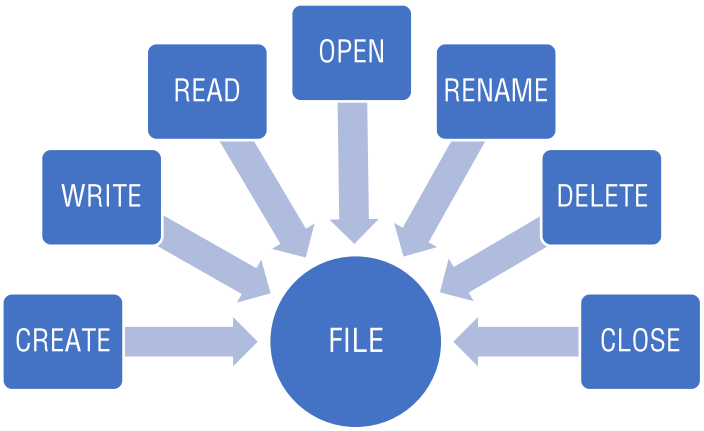
Dalam penelitian ini bertujuan untuk membuat *web interface snort* ini yang ditujukan agar aplikasi snort dapat lebih mudah untuk dioperasikan dengan rule yang dapat dimanajemen dari web dengan operasi *create, read, update* dan *delete (crud)* pada sistem rule snort.

2. METODE PENELITIAN

Dari penelitian yang dilakukan terdapat dua tahap pengembangan aplikasi snort yaitu tahap pembuatan *web interface snort* dan tahap pengujian rule dari *web interface snort* dengan melakukan percobaan serangan.

2.1 Tahap pembuatan *Web interface snort*

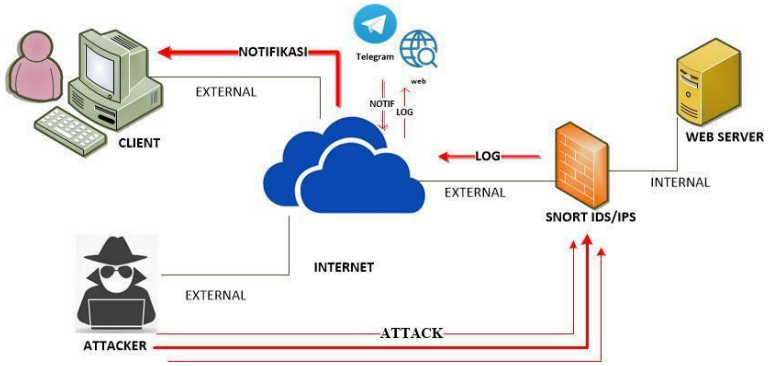
Pembuatan *web interface snort* menggunakan bahasa pemrograman php *file handling*, yaitu pemrograman php yang berisikan perintah-perintah untuk operasi dan managemen file php, meliput operasi membuat fille, mengganti nama file menghapus file, membuka dan menutup file, menulis dan menutup file php [5]. Dalam pengembangan lebih lanjut fungsi php juga menangani operasi *file handling system* yaitu fungsi untuk membuat, membaca, mengunggah, dan mengedit file. PHP menyediakan serangkaian fungsi yang dibangun untuk menangani file eksternal pada php. Dengan script diluar php tersebut, dapat mengontrol sistem server sehingga [5] dapat dipanggil sistem yang mengeksekusi aplikasi atau perintah tertentu yang didukung oleh lingkungan server. Diagram *file handling system* dengan bahasa pemrograman php dilihat pada Gambar 1.



Gambar 1. ***Diagram Operasi File Handling php***

2.2 Tahap Perancangan Topologi Jaringan

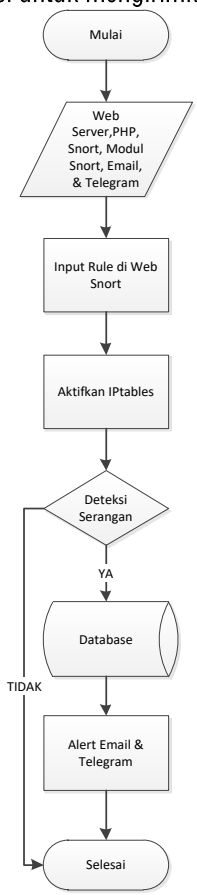
Pada tahap perancangan topologi terlihat ketika penyerang atau attacker yang ingin mengakses web server terhalang oleh firewall snort, kemudian ketika *action* serangan tersebut terjadi maka snort akan melakukan bloking akses pada attacker yang berusaha masuk untuk mengakses server dan pada saat itu alert notifikasi muncul pada email dan telegram yang memberitahukan bahwa telah ada indikasi serangan yang masuk sesuai dengan rule yang telah dimasukkan melaui *web interface snort*. Skema topologi jaringan ditunjukkan pada Gambar 2.



Gambar 2. ***Skema Skenario web dan aplikasi Snort***

2.3 Tahap Pengujian Rule dari *web snort* dengan melakukan percobaan serangan

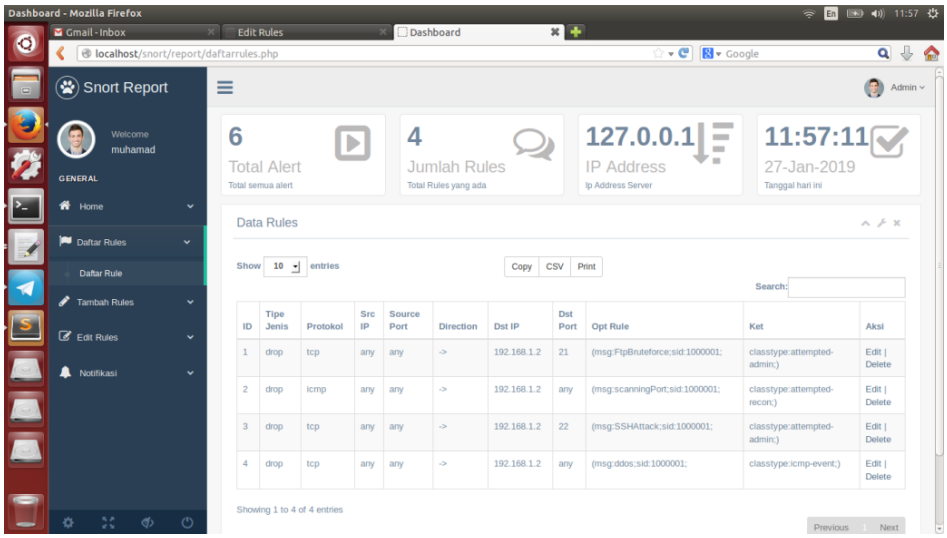
Tahap pengujian web interface dengan percobaan serangan *dengan rule* yang telah diinputkan pada web. Pengujian serangan terhadap sistem server Snort dilakukan untuk memastikan bahwa sistem monitoring yang di bangun berjalan optimal. Pengujian akan menggunakan beberapa tools yang dilakukan oleh PC Attacker seperti SSH Attack dan DDos Attack [4]. Pada Gambar 3, menggambarkan alur snort bekerja dalam mendeteksi serangan dan memberikan peringatan. Diawali dengan mempersiapkan dan mengaktifkan server, snort beserta modulnya, mengakatifkan protokol smtp utuk email, dan web api telegram. Setelah semua modul diaktifkan dan input rule dilakukan pada web snort, paket yang masuk akan ditangkap dan dianalisa oleh Snort berdasarkan aturan yang telah ditetapkan. Jika paket tersebut tidak terdeteksi sebagai sebuah intrusi atau serangan, maka paket tersebut akan ditolak dan proses berakhir, namun ketika paket tersebut terdeteksi sebagai sebuah intrusi, selanjutnya akan dilakukan pencatatan pada log file dan database. Setelah dicatat dan disimpan dalam database maka terjadi sebuah action untuk mengeksekusi file php yang berfungsi untuk mengirimkan notifikasi ke email dan telegram.



Gambar 3. ***Diagram Alur Snort***

3. HASIL DAN PEMBAHASAN

Pada Gambar 4. Menjelaskan bahwa bagian rule dari file sistem snort dapat di identifikasi menggunakan implementasi dari operasi file handling php. File dari rule snort dapat dimanajemen dengan operasi *create ,read, update,* dan *delete* (crud) sistem.



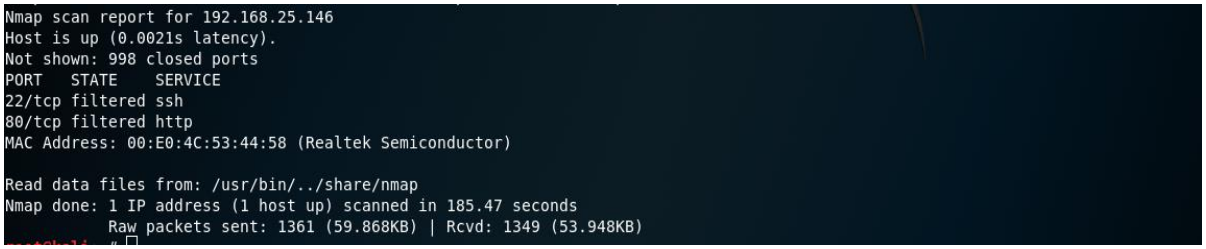
Gambar 4. *Dashboard Daftar Rule Snort.*

3.1. Tahap pengujian rule port scanning dengan nmap

Pada saat rule belum aktif ditunjukkan pada Gambar 5. dengan rule yang digunakan:
drop tcp any any -> 192.168.25.0/24 any (msg:"scanning port";)

Gambar 5. *Scanning port sebelum rule snort aktif*

Terlihat pada Gambar 5. bahwa ketika rule dan iptables belum aktif maka port ssh dan http masih berstatus open. Pada pengujian selanjutnya, penyerang akan melakukan scanning port dalam kondisi rule snort ips diaktifkan. Pada saat rule aktif ditunjukkan pada Gambar 6.



Gambar 6. Scanning port sesudah rule snort aktif .

Pada Gambar 6. terlihat bahwa ketika rule dan iptables aktif, port ssh dan http berstatus filtered, yang artinya telah berhasil terfilter oleh firewall iptables

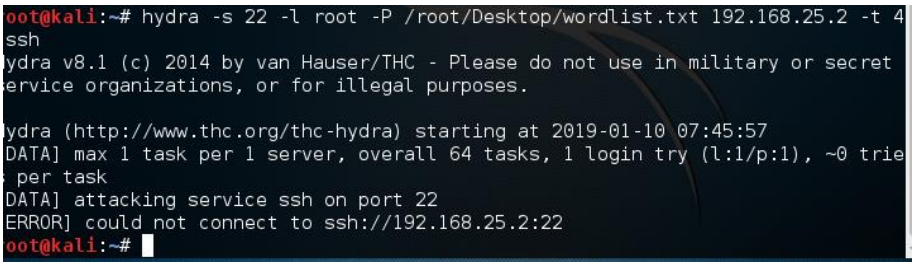
3.2. Tahap pengujian *rule ftp attack* dengan melakukan *login ftp*

Rule yang digunakan untuk mengidentifikasi ketika ada indikasi ftp attack di pc server. Pada Gambar 7. terlihat bahwa ketika rule snort aktif, akses untuk ftp ditutup dengan pesan connection refused.

Gambar 7. ***Pengujian ftp attack.***

3.3. Tahap pengujian *rule ssh attack* dengan melakukuan *login* dengan *ssh*

Rule yang digunakan digunakan untuk mengidentifikasi ssh brute force yaitu,
drop tcp any any -> 192.168.1.2 22 (msg:FTPBruteAttack;sid:1000002;classtype:attemot-recon;)

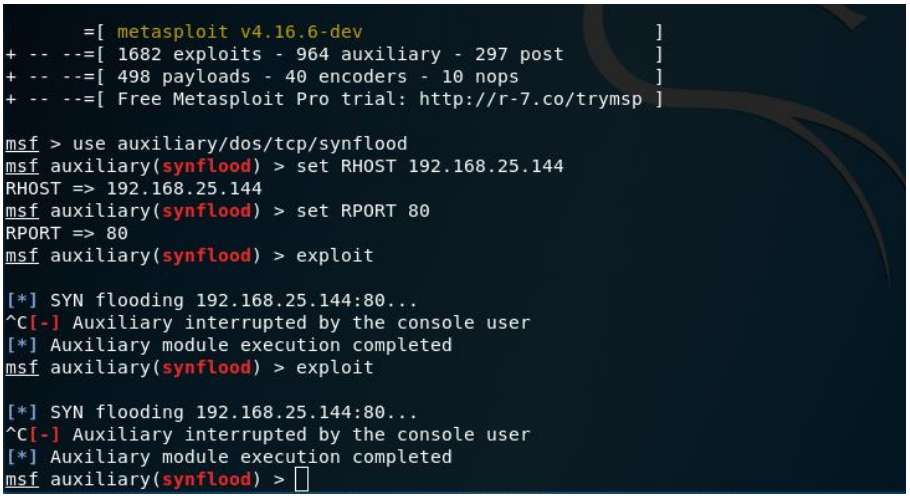


Gambar 8. ***Pengujian Ssh brute force .***

Pada Gambar 8. terlihat bahwa ketika rule snort aktif, akses untuk ftp ditutup dengan pesan *connection refused.*

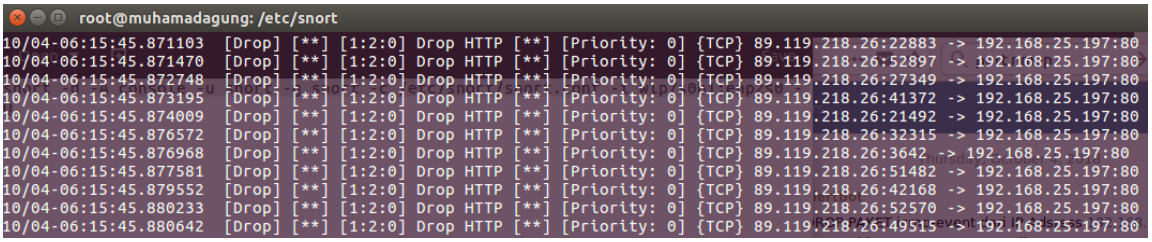
3.4. Tahap pengujian rule ddos menggunakan metasploit

Pada pengujian ini, penyerang akan melakukan jenis serangan *Denial of Service* (DoS) menggunakan *metasploit syn attack* di pc penyerang. Pengujian Ddos ditunjukkan pada Gambar 9.



Gambar 9. ***Pengujian DDos dengan Metasploit.***

Pc Tester menggunakan tools Metasploit dengan menggunakan *library syn flood* melalui remote host target dan remote port target, kemudian mengeksekusi dengan perintah exploit. Maka secara otomatis pengujian pada firewall yang berasda pada pc router snort akan dijalankan dan dimonitoring melalui terminal, serta dapat dihentikan dengan menekan shorchut Ctrl+C pada keybord. Log serangan ditunjukkan pada Gambar 10.

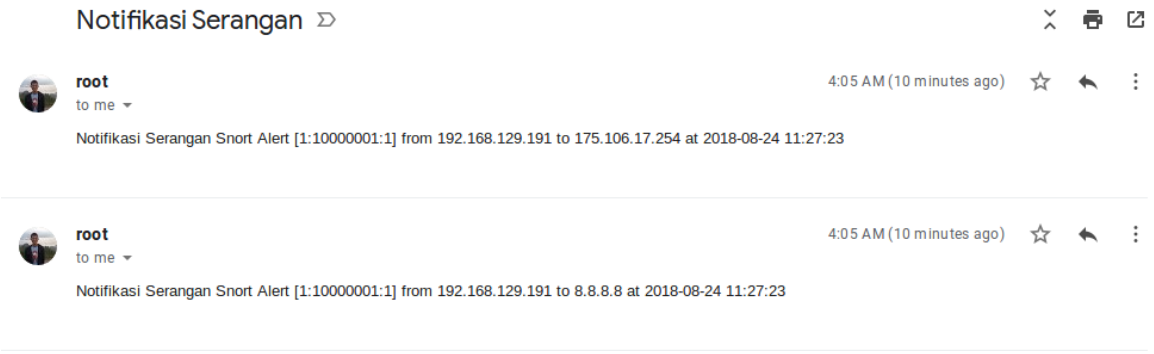


Gambar 10. ***Log serangan pada terminal linux.***

Pada Gambar 10. Terlihat dari hasil pengujian serangan yang ditampilkan pada konsole terminal linux Ubuntu ketika ada indikasi akses yang masuk.

3.5. Tahap pengujian notifikasi pada Email

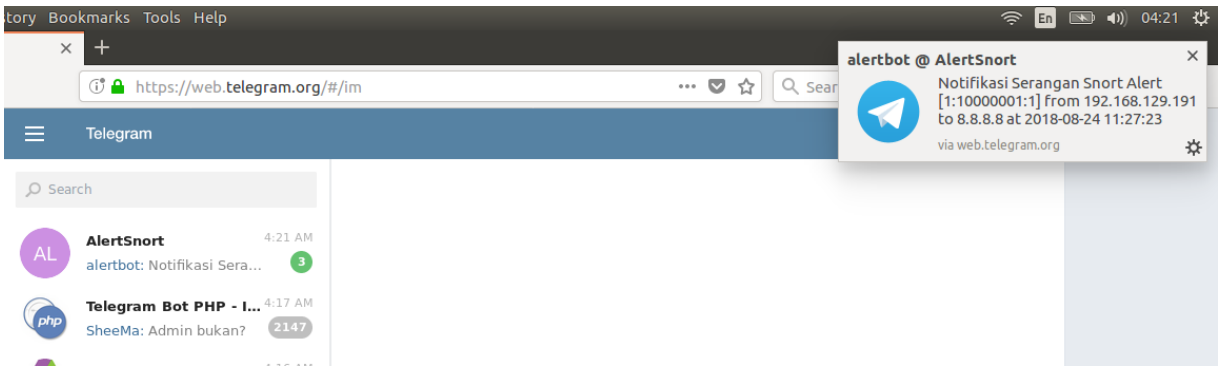
Notifikasi dari serangan yang terjadi pada email ditunjukkan pada Gambar 11. Notiifikasi tersebut berjalan dengan baik, ketika terjadi indikasi serangan maka notifikasi terkirim beberapa saat kemudian dan masuk ke dalam inbox email.



Gambar 11. ***Notifikasi Email.***

3.6. Tahap pengujian notifikasi pada Telegram

Notifikasi serangan melaui Telegram Web ditunjukkan pada Gambar 12, terlihat bahwa notifikasi tersebut berjalan dengan baik, ketika terjadi indikasi serangan maka log alert notifikasi terkirim beberapa saat kemudian dan masuk ke dalam telegram meskipun ada jeda beberapa saat.



Gambar 12. ***Notifikasi Telegram.***

4. KESIMPULAN

Pembuatan web aplikasi snort untuk manajemen rule snort berjalan dengan baik, rule yang di inputkan pada web dapat dijalankan untuk mengetahui indikasi serangan yang masuk ke dalam sistem, baik dari pengujian yang telah dilakukan yaitu port scanning, ftp attack, ssh attack, maupun ddos attack dapat berjalan baik sebagaimana mestinya seperti tersaji pada Tabel 1.

Tabel 1. Kesimpulan Pengujian.

Implementasi	*Pengujian*	*Hasil*	*Deskipsi*
Snort mode inline (Intrusion Preventive System) dengan rule yang telah dimasukkan dari web interfaces.	Percobaan serangan penyusupan dengan scan port, ftp attack, dan ddos attack	Sesuai	Menghasilkan analisa yang sesuai dengan metode yang digunakan dan konfigurasi yang dilakukan

DAFTAR PUSTAKA

[1] Muhammad, A. H. (2007). *Rahasia dan Trik Mengamankan Server Linux.* Yogyakarta: GAVA MEDIA.
[2] Rafiudin, R. (2010). *Menggayang Hacker dengan Snort.* Yogyakarta: Andi Publisher.
[3] Ervin Kusuma Dewi, P. (2017). ANALISIS LOG SNORT MENGGUNAKAN. *JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran)*, 72-79.
[4] Anugrah, I., & Rahmanto, R. (2017). SISTEM KEAMANAN JARINGAN LOCAL AREA NETWORK MENGGUNAKAN TEKNIK DE-MILITARIZED ZONE. *Jurnal Penelitian Ilmu Komputer, Sistem Embedded & Logic*, 91-106.
[5] Mubarok, M. H. (2011). *Sistem Kontrol via Web dengan CHI, PHP, dan AJax.* Jakarta: PT Elex Media Komputindo.