IMPLEMENTASI SISTEM KEAMANAN JARINGAN MENGGUNAKAN SURICATA DAN NTOPNG

IMPLEMENTATION OF NETWORK SECURITY SYSTEM USING SURICATA AND NTOPNG

Fahmi Bagaskara Perdana[1], Dr. Ir. Rendy Munadi, M.T.[2], Arif Indra Irawan, S.T., M.T.[3]
[1,2,3] Prodi S1 Teknik Telekomunikasi, Fakultas Teknik Elektro, Universitas Telkom
[1]**fahmibagaskara@student.telkomuniversity.ac.id**, [2]**rendymunadi@telkomuniversity.co.id**,
[3]**arifirawan@telkomuniversity.ac.id**

Abstrak

Masalah keamanan jaringan semakin menjadi perhatian dikarenakan perkembangan teknologi informasi yang semakin cepat. Hal ini membuat seseorang secara ilegal untuk masuk ke dalam sistem dan membuat lumpuh sistem tersebut. Selain itu, adanya celah dan tidak adanya sistem keamanan yang melindungi sistem menjadikan sistem rentan terhadap serangan.

Oleh karena itu, pada Tugas Akhir ini dibuatlah sebuah sistem kemanan dengan menggunakan Suricata sebagai *Network Intrusion Detection System* (NIDS) dan Ntopng sebagai alat untuk me-*monitoring* jaringan hingga ke layer-7. Dengan fokus pada serangan *Denial of Services* (DoS), maka akan dilihat perbandingan antara kedua aplikasi tersebut dalam menangani serangan DoS.

Dari hasil penelitian ini, berdasarkan *rule* Suricata yang penulis buat, penulis berhasil mendeteksi semua serangan yang diujicobakan. Sedangkan pada *rule default* pada Ntopng, penulis hanya mampu mengidentifikasi jenis serangan DoS berupa SYN *flood*. Untuk serangan DoS dengan tujuan *website server*, pada bagian akurasinya, *rule* Suricata yang penulis buat lebih unggul daripada *rule default* pada Ntopng untuk aplikasi LOIC sebesar 52,70%, sedangkan untuk aplikasi Hping3 sebesar 48,80%, dan aplikasi GoldenEye sebesar 52,84%. Sedangkan untuk serangan DoS dengan tujuan FTP *server*, pada bagian akurasinya, *rule* Suricata yang penulis buat juga lebih unggul daripada *rule default* pada Ntopng untuk aplikasi LOIC sebesar 52,30%, sedangkan untuk aplikasi Hping3 sebesar 59,97%. Sehingga ada perbedaan jauh antara persentase akurasi, *precision rate*, dan *recall rate* dari Suricata dan Ntopng yaitu Suricata lebih unggul dalam ketepatan akurasi *rule*-nya dalam mendeteksi serangan DoS.

Kata kunci : *suricata, ntopng, rule,* DoS.

Abstract

Network security issues are increasingly a concern due to the rapid development of information technology. This makes someone illegally enter the system and paralyze the system. In addition, there are loopholes and no security system that protects the system making the system vulnerable to attack.

Therefore, in this final project a security system is made by using Suricata as a Network Intrusion Detection System (NIDS) and Ntopng as a tool to monitor the network up to layer 7. With a focus on Denial of Services (DoS) attacks, it will be seen a comparison between the two applications in dealing with DoS attacks.

In this research, based on the rules of Suricata that the author made, the author managed to detect all attacks that were tested. While the default rule on Ntopng, the author is only able to identify the type of DoS attack in the form of SYN flood. For DoS attacks with the purpose of a website server, the accuracy of the Suricata rule created by the author is superior to the default rule on Ntopng for LOIC applications at 52.70%, while for Hping3 applications at 48.80%, and GoldenEye applications at 52.84% . Whereas for DoS attacks with the aim of FTP server, the accuracy of the Suricata rule that I made was also superior to the default rule on Ntopng for LOIC applications by 52.30%, while for Hping3 applications it was 59.97%. So there is a big difference between the percentage of accuracy, precision rate, and recall rate of Suricata and Ntopng, where Suricata is superior in the accuracy of its rules in detecting DoS attacks.

Keywords : *suricata, ntopng, rule,* DoS.

1. Pendahuluan

Intrusion Detection and Prevention Systems (IDPS) adalah sistem yang dapat me-*monitor host* atau jaringan untuk mencurigai aktivitas atau perilaku anomali kemudian mengambil tindakan yang tepat untuk melawan mereka. Sistem IDS dapat berada di mana saja dalam jaringan. IDS hanya mendeteksi aktivitas berbahaya dan memperingatkan administrator sehingga administrator yang harus memutuskan cara mengatasi peringatan itu. Di sisi lain, sistem IPS berada sebagai sistem *inline* dan selain menghasilkan alarm, IPS bisa secara otomatis bereaksi terhadap aktivitas abnormal. IPS bisa memblokir sumber serangan atau dapat mengatur ulang koneksi [1].

Menurut Kumar dalam jurnalnya disimpulkan bahwa *rules* Snort sangat mudah diimplementasikan. Walaupun terbilang cukup mudah, *rules* Snort sendiri sangatlah ampuh untuk mendeteksi segala jenis paket yang mencurigakan. Sedangkan *rules* Suricata juga tidak jauh berbeda dengan Snort. Bahkan *rules* Snort dapat diimplementasikan ke dalam Suricata [2]. Sama seperti Snort, menurut Brian Cusack dalam jurnalnya disimpulkan bahwa Suricata juga mampu mendeteksi aktivitas mencurigakan dan mengumpulkannya menjadi *evidence* yang dapat digunakan sebagai antisipasi serangan berikutnya [3]. Perancangan Suricata yang baik harus meperhatikan apakah *rule* yang dibuat berhasil dalam mendeteksi sebuah penyusupan yang terjadi, karena banyak *rule default* pada Suricata yang tidak dapat mendeteksi sebuah ancaman. Disisi lain kita tidak boleh melupakan kemampuan dari mesin yang digunakan dalam pengimplementasian Suricata.

Dalam meningkatkan sistem keamanan jaringan dapat juga menggunakan Ntopng (*ntop next generation*) [4]. Ntopng merupakan *open source network traffic monitor* yang berfungsi untuk menampilkan trafik penggunaan jaringan hingga ke *layer*-7 [5]. Ntopng bisa menampilkan informasi tentang trafik dan daftar pengguna (*host*) yang menggunakan sebuah jaringan secara detail.

Melihat permasalahan di atas, dalam penelitian ini mencoba untuk membuat sebuah *rule* pada Suricata dan melakukan percobaan pengaplikasiaan *rule* tersebut. Kemudian dari hasil percobaan ini kita dapat melihat efektifitas *rule* yang telah dibuat apakah dapat menangkap tindakan penyusupan dengan tepat, serta akan dibandingkan pula dengan *rule default* pada Ntopng. Di samping itu, percobaan dilakukan dengan milihat tingkat pemakaian CPU (*Centra Prossesing Unit*) dan RAM (*Random Access Memory*) pada setiap pengaplikasian *rule* agar dapat menetukan kemampuan mesin yang akan digunakan yang pada akhirnya mempengaruhi kecepatan alur data yang terjadi pada jaringan komputer.

2. Dasar Teori

2.1 *Intrusion Detection System* (IDS)

IDS merupakan sebuah teknik atau metode yang digunakan unutk mendeteksi aktifitas mencurigakan yang terjadi pada *network* dan *host level*. IDS mengumpulkan data-data dari sensor yang ada dan akan menganalisis data tersebut, kemudian akan memberikan peringatan apakah terjadi penyusupan pada jaringan komputer [1] [6].

2.2 Suricata

Suricata merupakan *network based intrusion detection and prevention system* yaitu suatu perangkat lunak yang dapat digunakan untuk mendeteksi dan mencegah (*Detection System dan Prevention System*) terhadap lalu lintas sebuah jaringan. Suricata adalah IDS *open source* yang dikembangkan oleh *Open Information Security Foundation* (OISF) [7].

2.2.1 *Rule*

Rule dapat diartikan secara langsung yang berarti peraturan. Peran *rule* sangat penting dalam proses pendeteksian dikarenakan Suricata menggunakan *rule* untuk menentukan apakah lalu lintas jaringan yang melaluinya merupakan ancaman atau bukan [2]. Secara umum *rule* terdiri dari dua bagian yaitu *rule header* dan *rule option* [8].

2.2.2 *Signature*

Signature adalah sebuah *pattern* yang kita lihat dalam sebuah paket data. *Pattern* digunakan untuk mendeteksi satu atau beberapa serangan. *Signature* dapat dilihat dibeberapa bagian dari paket data tergantung jenis serangan yang dilakukan [2].

2.2.3 *Alert*

Alert adalah sebuah peringatan yang diberikan kepada seorang administrator ketika terjadinya sebuah penyusupan atau sebuah aktivitas mencurigakan. Ketika IDS mendeteksi adanya penyusupan, Suricata akan memberikan peringatan kepada administrator [2].

2.2.4 *Logging System*

Logging tergantung pada *detection engine* menangkap sebuah aktivitas mencurigakan. Ketika sebuah aktivitas mencurigakan terjadi maka sistem akan melakukan pencatatan pada aktivitas tersebut atau memberikan sebuah peringatan kepada administrator bahwa telah terjadi sebuah tindakan yang mencurigakan. Pada umumnya *log* akan tersimpan pada tempat penyimpanan /var/log/Suricata [8]. Berikut *command* yang sering digunakan : tail -f/var/log/suricata/fast.log.

2.3 Ntopng

Ntopng (*ntop next generation*) merupakan *open source network traffic monitor* yang berfungsi untuk menampilkan trafik penggunaan jaringan hingga ke *layer-7* [4]. Ntopng bisa menampilkan informasi tentang trafik dan daftar pengguna (*host*) yang menggunakan sebuah jaringan secara detail.

2.4 *Server*

Server merupakan induk dari segala komputer yang terhubung pada sebuah jaringan yang berfungsi sebagai pengatur sistem jaringan. *Server* juga menjalankan perangkat lunak administratif yang mengontrol akses terhadap jaringan dan sumber daya yang terdapat di dalamnya [9].

2.4.1 FTP *Server*

FTP *server* adalah suatu *server* yang berfungsi untuk memberikan layanan tukar menukar *file* di mana *server* tersebut selalu siap memberikan layanan FTP apabila mendapat permintaan (*request*) dari FTP *client* [9].

2.4.2 *Web Server*

Web server merupakan *software* yang memberikan layanan data. Berfungsi menerima permintaan HTTP atau HTTPS dari klien yang dikenal dengan *browser web* dan mengirimkan kembali hasilnya dalam bentuk halaman-halaman *website.* Di mana *website* tersebut umumnya berbentuk dokumen HTML. *Web server* merupakan *host* yang paling banyak menjadi sasaran dan diserang di lingkup jaringan suatu organisasi [9].

2.5 Linux

Linux adalah sebuah sistem operasi yang mengatur semua sumber daya *hardware* yang berkaitan dengan *computer desktop* atau laptop. Pengertian mudahnya, sistem operasi mengelola komunikasi antara *software* dan *hardware* [10].

2.6 DoS (*Denial of Services*)

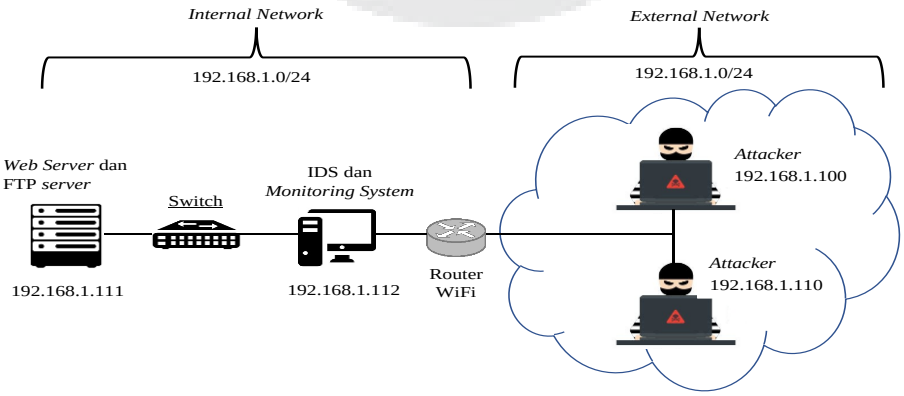
DoS merupakan singkatan dari *Denial of Services*, sebuah teknik penyerangan terhadap sebuah sistem dengan jalan menghabiskan sumber data sistem tersebut sehingga tidak dapat diakses lagi [11]. Sumber daya dapat berupa CPU, RAM, *Swap*, *cache*, maupun *bandwidth*. Berikut jenis serangan DoS : UDP *Flood Attack*, ICMP *Flood*, *Ping Flood*, *Ping of Death*, SYN *Flood*, dan HTTP *Flood* [11]. Berikut adalah *tools* DoS : LOIC, Hping3, dan GoldeEye [11].

3. Model Sistem dan Perancangan

3.1 Desain Sistem

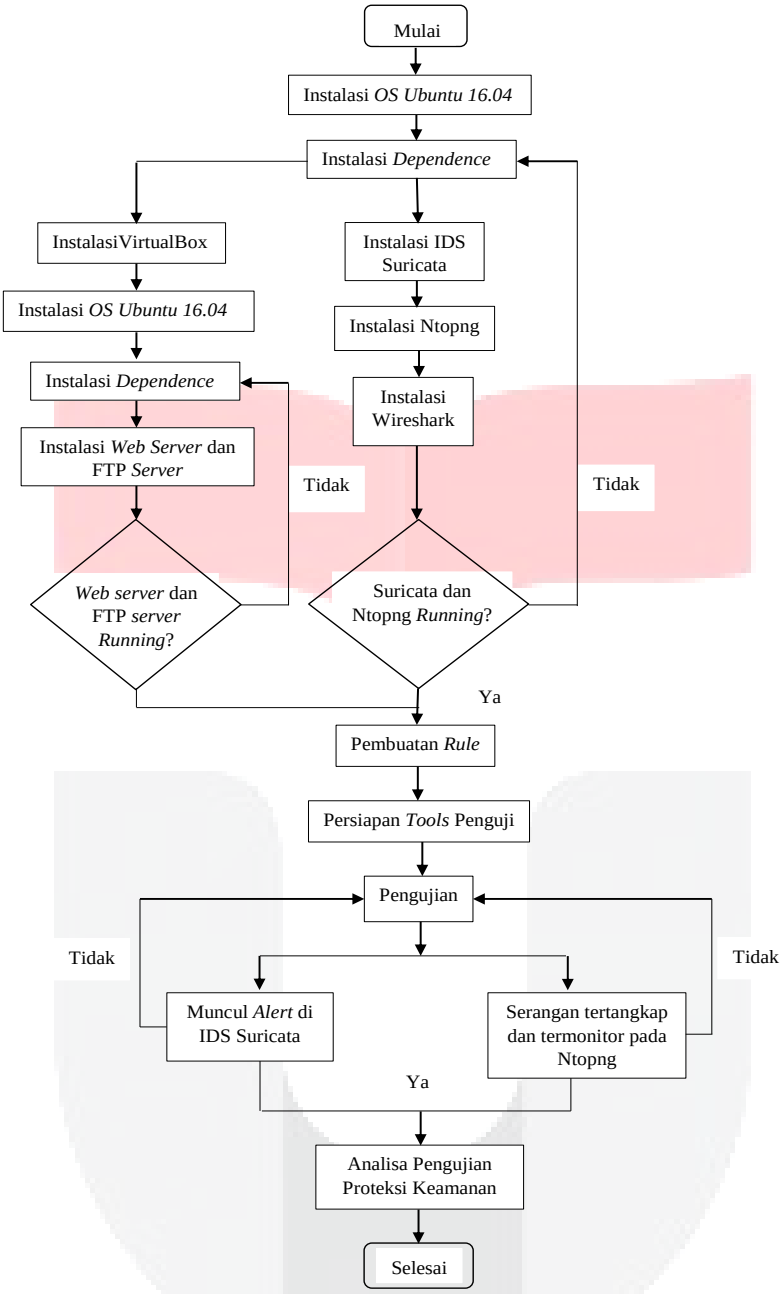
Desain sistem dapat dilihat pada gambar 1, di mana ada 2 *server* yaitu Apache2 sebagai *Web server* dan ProFTPD sebagai FTP *server*. Kedua *server* tersebut ada pada *Basic Pentesting* 1 yang merupakan *virtual machine* yang khusus ditujukan untuk pengujian penetrasi. Pada *virtual machine* virtualbox, penulis menggunakan mode *bridge* yang mana *guest* terhubung ke jaringan fisik dari *host.*

IDS pada sistem ini menggunakan Suricata dan untuk *monitoring* sistemnya menggunakan Ntopng. Sistem yang dijalankan IDS telah dikonfigurasi dengan dua *interface*, salah satunya terhubung ke jaringan *internal* dan yang lainnya terhubung ke jaringan *external* [1]. Skenario yang akan digunakan adalah pertama tidak ada serangan, lalu yang kedua penulis menjalankan Suricata serta Ntopng dan akan memulai beberapa serangan dari sistem penyerang ke *Web server* dan FTP *server*. Setelah itu akan dilakukan perbandingan untuk menentukan *tools* yang lebih bagus dalam mendeteksi sebuah ancaman DoS.



Gambar 1 Desain Sistem Keamanan [1].

3.2 Diagram Alir Pengerjaan



Gambar 2 Diagram Alir Pengerjaan.

Pada proses pertama yaitu diawali dengan *install* OS Ubuntu 16.04 (*Host*) lalu dilanjutan dengan *install dependence.* Selanjutnya *download* dan *install* Virtualbox, Wireshark, Suricata, dan Ntopng. Setelah selesai, lakukan konfigurasi Virtualbox. Pada Virtualbox lakukan instalasi OS Ubuntu 16.04 (*Guest*) lalu lakukan instalasi *dependence* dan dilanjutkan dengan instalasi *Web server* dan FTP *server*. Setelah selesai, dilanjutkan ke konfigurasi Suricata, dan Ntopng. Lakukan percobaan untuk *Web server*, FTP *server*, Suricata, dan Ntopng. Jika berhasil maka lanjut ke tahap selanjutnya, jika tidak maka kembali ke instalan *dependence* untuk ditinjau ulang. Jika berhasil, lanjut ke pembuatan *rule* untuk sistem yang dibuat. Setelah semua proses selesai, lakukan pengujian pada sistem. Jika berhasil muncul *alert* pada Suricata dan tertangkapnya serangan yang ter-*monitor* pada Ntopng, maka lanjut ke tahap berikutnya. Jika tidak berhasil, maka akan dilakukan pengujian ulang. Tahap berikutnya yaitu pengumpulan data serta menganalisa serangan. Tahap terakhir yaitu mengambil kesimpulan dari sistem keamanan yang dibuat.

3.3 Pembuatan *Rule*

3.3.1 *Rule* Pada Suricata

1. LOIC

Jenis serangan DoS pada LOIC adalah HTTP *flood*, UDP *flood*, dan ICMP *flood*. *Rule* untuk LOIC yang digunakan dalam mendeteksi adanya tindakan DoS pada jaringan komputer adalah sebagai berikut :

alert tcp any any -> any 80 (msg: "LOIC"; ttl:128; flow:to_server; flags: PA; threshold: type threshold, track by_dst, count 500, seconds 60; classtype:attempted-dos; sid:2000005; rev:1; metadata:created_at 2019_07_03, updated_at 2019_07_03;)

alert tcp any any -> any 21 (msg: "LOIC"; ttl:128; flow:to_server; flags: PA; threshold: type threshold, track by_dst, count 500, seconds 60; classtype:attempted-dos; sid:2000006; rev:1; metadata:created_at 2019_07_03, updated_at 2019_07_03;)

2. Hping3

Jenis serangan DoS pada hping3 adalah *SYN Flood*. *Rule* untuk hping3 yang digunakan dalam mendeteksi adanya tindakan DoS pada jaringan komputer adalah sebagai berikut :

alert tcp any any -> any 80 (msg: "HPING3"; ttl:64; flow:to_server; flags: S; threshold: type threshold, track by_dst, count 100, seconds 5; classtype:attempted-dos; sid:2000001; rev:1; metadata:created_at 2019_07_03, updated_at 2019_07_03;)

alert tcp any any -> any 21 (msg: "HPING3"; ttl:64; flow:to_server; flags: S; threshold: type threshold, track by_dst, count 100, seconds 5; classtype:attempted-dos; sid:2000002; rev:1; metadata:created_at 2019_07_03, updated_at 2019_07_03;)

3. GoldenEye

Jenis serangan DoS pada GoldenEye adalah *SYN Flood*. *Rule* untuk GoldenEye yang digunakan dalam mendeteksi adanya tindakan DoS pada jaringan komputer adalah sebagai berikut :

alert tcp any any -> any 80 (msg: "GOLDENEYE"; ttl:64; flow:to_server,established; flags: S; ack:0, seq:0; classtype:denial-of-service; sid:2000003; rev:1; metadata:created_at 2019_07_03, updated_at 2019_07_03;)

3.3.2 *Rule* Pada Ntopng

Untuk *rule* pada Ntopng, penulis menggunakan *rule* bawaan dari Ntopng itu sendiri yang mana *rule* tersebut telah diatur jika jumlah SYN yang dikirm oleh suatu *host* melebihi jumlah tertentu maka akan dianggap sebagai ancaman dan Ntopng akan memunculkan sebuah *alert*.

3.4 Parameter Performansi Sistem

Untuk mengevaluasi kinerja dari sistem keamanan yang dibuat, penulis menggunakan 3 ukuran, yaitu *precision rate, recall rate*, dan akurasi. *Precision rate* adalah tingkat ketepatan antara informasi yang diminta oleh pengguna dengan jawaban yang diberikan oleh sistem. *Precision rate* menjelaskan apakah terdapat *false positive* (FP) dari *rule* yang diimplementasikan. Sebuah tes degan kekhususan tinggi memiliki tingkat *false positive* yang rendah [12].

Recall rate adalah tingkat keberhasilan sistem dalam menemukan kembali sebuah informasi. *Recall rate* mejelaskan apakah terdapat *false negative* (FN) dari *rule* yang diimplementasikan. Sebuah tes dengan sensitivitas tinggi memiliki tingkat *false negative* yang rendah [12].

Akurasi adalah kedekatan hasil pengukuran dengan nilai sesungguhnya. Atau bisa disebut proporsi kelas klasifikasi yang benar yaitu *True Positive* (TP) dan *True Negativ*e (TN) di atas jumlah total klasifikasi [12]. Dengan tingkat akurasi yang tinggi yaitu 100%, maka sistem kemanan yang dibuat sudah memiliki tingkat keamanan yang sangat baik.

Penulis menghitung 3 ukuran tersebut dengan menggunakan rumus sebagai berikut [17] :

$$Precision\ rate\ = \frac{\mathrm{TP}}{\mathrm{TP+FP}} \text{ x } 100\ \% \quad (3.1)$$

$$Recall\ rate = \frac{\mathrm{TP}}{\mathrm{TP+FN}} \text{ x } 100\ \% \quad (3.2)$$

$$Accuracy = \frac{(\mathrm{TP+TN})}{(\mathrm{TP+TN+FP+FN})} \text{ x } 100\ \% \quad (3.3)$$

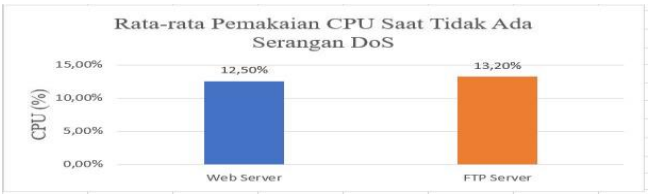
Selain itu, digunakan juga perbandingan *resource* pemakaian pada RAM dan CPU yang digunakan sebelum menjalankan Suricata dan sesudah menjalankan Suricata dengan *set rules* yang diimplementasikan.

4. Hasil dan Analisis

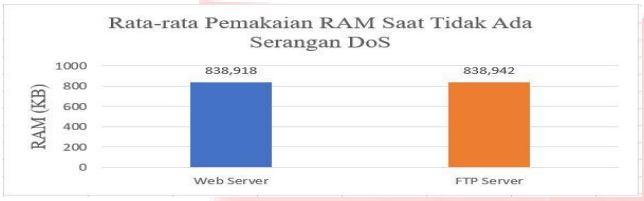
Pengujian pada penelitian ini dilakukan berdasarkan parameter pengujian guna meningkatkan keamanan pada sistem yang dibuat. Pengujian yang dilakukan terbagi menjadi lima. Pengujian dilakukan meliputi performansi dari sistem yang dibuat dengan pengukuran CPU, RAM, *precision rate, recall rate,* dan akurasi.

4.1 Pengujian Serta Analisa Serangan Pada CPU dan RAM

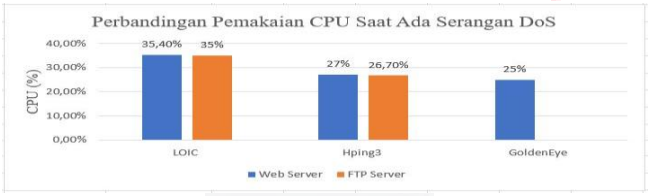
Untuk pengujian *resource* akan dilakukan 3 kali percobaan dalam waktu 5 menit pada saat ada serangan masuk dan saat tidak ada serangan masuk. Percobaan 1 dilakukan pada jam 12.00, percobaan 2 pada jam 15.00, dan percobaan 3 pada jam 16.00.



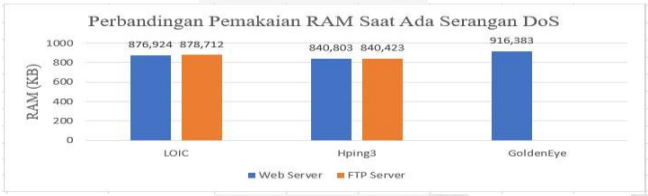
Gambar 3 Perbandingan pemakaian CPU saat tidak ada serangan DoS.



Gambar 4 Perbandingan pemakaian RAM saat tidak ada serangan DoS.



Gambar 5 Perbandingan pemakaian CPU saat ada serangan DoS.



Gambar 6 Perbandingan pemakaian RAM saat ada serangan DoS.

4.2 Hasil Pengujian Sistem

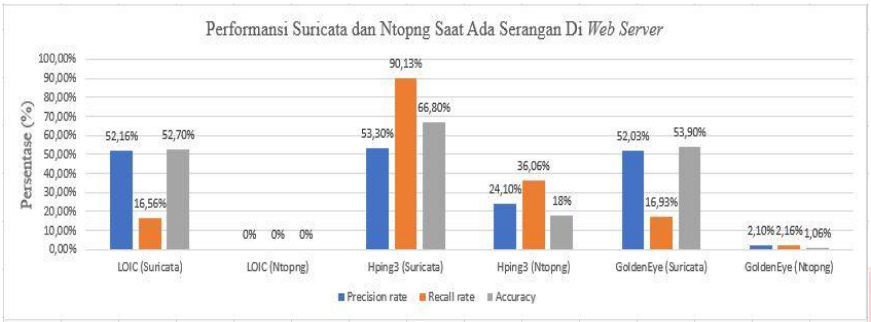
Pada sub bab ini akan menjelaskan tentang hasil dari percobaan *rules* berdasarkan *precision rate* yang berarti tingkat ketepatan antara informasi yang diminta oleh pengguna dengan jawaban yang diberikan oleh sistem, *recall rate* yang berarti tingkat keberhasilan sistem dalam menemukan kembali sebuah informasi, dan berdasarkan akurasi yang berarti kedekatan hasil pengukuran dengan nilai sesungguhnya. Untuk pengujian ini, akan dilakukan 3 kali percobaan serangan DoS dengan 3 aplikasi DoS *Attack* yang berbeda dan kemudian akan dianalisa bagaimana performansi pada Suricata dan Ntopng dalam menangani serangan DoS tersebut.

4.2.1 Performansi Suricata dan Ntopng Saat Tidak Ada Serangan

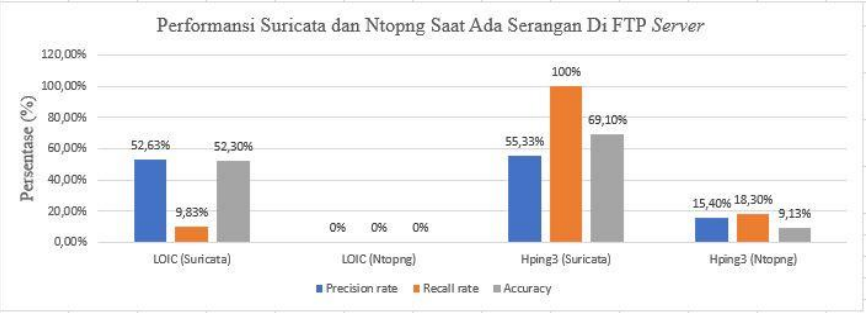
Gambar 7 Performansi Suricata dan Ntopng saat tidak ada serangan.

Suricata dan Ntopng sama-sama memperoleh akurasi 100% saat tidak ada serangan yang berarti kedua IDS tersebut tidak mendeteksi *false positif* pada setiap percobaan.

4.2.2 Performansi Suricata dan Ntopng Saat Ada Serangan



Gambar 8 Persentase performansi Suricata dan Ntopng saat ada serangan di *website server*.



Gambar 9 Persentase performansi Suricata dan Ntopng saat ada serangan di FTP *server*.

5. Kesimpulan dan Saran

5.1 Kesimpulan

1. Pengujian dan implementasi *rule* Suricata yang penulis buat dapat mendeteksi semua jenis serangan yang diujikan. Berbeda dengan *rule default* pada Suricata untuk aplikasi LOIC, Hping3, dan GoldenEye tidak mampu mengidentifikasi jenis serangan DoS. Sedangkan untuk *rule default* pada Ntopng hanya mampu mengidentifikasi jenis serangan DoS berupa SYN *flood*.
2. Untuk IDS pada Suricata, saat mendeteksi adanya serangan maka Suricata hanya akan akan menampilkan *alert*. Sedangkan untuk IDS Ntopng, saat mendeteksi adanya serangan maka Ntopng akan menampilkan *alert* namun disajikan pula beberapa grafik seperti yang menunjukkan *traffic, packets, ports, peers, protocols*, DNS, *flows*, dan *talkers*.
3. Untuk pengujian penggunaan *resource* CPU dan RAM, diketahui bahwa saat awal *server* IDS baru saja hidup dan Suricata belum mendeteksi adanya ancaman maka terlihat kondisi CPU dan RAM masih dalam keadaan normal. Namun ketika Suricata mendeteksi adanya serangan yang diujikan terdapat kenaikan penggunaan CPU dan RAM. Hasil kenaikan ini dan sisa *resource* yang ada, *server* IDS masih dapat bekerja dengan baik pada pengujian yang dilakukan.
4. Aplikasi DoS pada LOIC lebih mendominasi serangan DoS yang mempengaruhi CPU dengan persentase tertinggi diantara 2 aplikasi lainnya yaitu dengan memperoleh rata-rata CPU 35,2%. Sedangkan aplikasi DoS pada GoldenEye lebih mendominasi serangan DoS yang mempengaruhi RAM yaitu dengan rata-rata sebesar 916.383 KB.
5. Untuk serangan DoS dengan tujuan *website server*, pada bagian akurasinya, *rule* Suricata yang penulis buat lebih unggul daripada *rule default* pada Ntopng untuk aplikasi LOIC sebesar 52,70%, sedangkan untuk aplikasi Hping3 sebesar 48,80%, dan aplikasi GoldenEye sebesar 52,84%.
6. Untuk serangan DoS dengan tujuan FTP *server*, pada bagian akurasinya, *rule* Suricata yang penulis buat juga lebih unggul daripada *rule default* pada Ntopng untuk aplikasi LOIC sebesar 52,30%, sedangkan untuk aplikasi Hping3 sebesar 59,97%.
7. Ada perbedaan jauh juga antara nilai performansi dari *rule* Suricata yang penulis buat dengan *rule default* pada Ntopng yaitu *rule* Suricata lebih unggul pada ketepatan *precision rate*, *recall rate*, maupun akurasi dalam mendeteksi serangan DoS. Penyebabnya adalah Ntopng mencatat banyaknya *false* dan rendahnya akurasi

paket yang dianalisis yaitu karena *rule default* pada Ntopng hanya mampu mengidentifikasi jenis serangan DoS berupa SYN *flood*.

5.2 Saran

1. Pengujian *rule* yang telah diimplementasikan dengan menggunakan metode teknik atau *tools* yang lain sehinga dapat meningkatkan akurasi *rule* dalam mendeteksi serangan.
2. Menggabungkan Suricata dengan Ntopng dengan diharapkannya mewujudkan sistem keamanan yang lebih handal.
3. Penelitian selanjutnya agar dapat mengabungkan penerapan IDS dan IPS agar ancaman yang terjadi dapat langsung ditangani oleh sistem.

Daftar Pustaka:

[1] R. M. Yousufi, P. Lalwani, and M. B. Potdar, " A Network-Based Intrusion Detection and Prevention System with Multi-Mode Counteractions," International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS), IEEE, 2017.

[2] A. Kumar, S. Chandak, and R. Dewanjee, "Recent Advances in Intrusion Detection Systems: An Analytical Evaluation and Comparative Study," International Journal of Computer Applications, vol. 86, no. 4, pp. 32-37, Januari 2014.

[3] B. Cusack and M. Alqahtani, "Acquisition Of Evidence From Network Intrusion Detection Systems," in Proceedings of the 11th Australian Digital Forensics Conference, pp. 36-43, Desember 2013.

[4] L. Deri, M. Martinelli, and A. Cardigliano, "Realtime high-speed network traffic monitoring using ntopng," in Proceedings of the 28th USENIX Conference on Large Installation System Administration, ser. LISA'14, 2014.

[5] Anonim, "Ntopng,". 6 Juni, 2018. Available at: website, https://www.ntop.org. [Diakses 27 September 2018, 19:08:08 WIB].

[6] M. Albayati, and B. Issac, "Analysis of Intelligent Classifiers and Enhancing the Detection Accuracy for Intrusion Detection System," International Journal of Computational Intelligence, pp. 841-853, September 2015.

[7] D. A. Bhosale and V. M. Mane, "Comparative Study and Analysis of Network Intrusion Detection Tools," International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT), pp. 312-315, 2015.

[8] S. Patel, A. Sonker, "Rule-Based Network Intrusion Detection System for Port Scanning with Efficient Port Scan Detection Rules Using Snort," International Journal of Future Generation Communication and Networking, vol. 9, no. 6, pp. 339-350, 2016.

[9] R. H. Putra and W. Sugeng, "Implementasi Cluster Server pada Raspberry Pi dengan Menggunakan Metode Load Balancing," Jurnal Edukasi dan Penelitian Informatika (JEPIN), vol. 2, no. 1, pp. 41-45, Juni 2016.

[10] Anonim, "Linux,". 7 Juni, 2018. Available at: website, https://www.ntop.org. [Diakses 27 September 2018, 19:15:08 WIB].

[11] Anonim, "Mengenal Berbagai Jenis Serangan Pada Jaringan Komputer,". 20 Januari, 2017. Available at: website, https://netsec.id. [Diakses 27 September 2018, 20:12:17 WIB].

[12] T. Aldwairi, D. Perera, and M. Novotny, "An Evaluation of the Performance of Restricted Boltzmann Machines as a Model for Anomaly Network Intrusion Detection," Analytics and Security Institute, High Performance Computing Collaboratory, Juli 2018.