ANALISIS PERFORMANSI IDS MENGGUNAKAN METODE DETEKSI *ANOMALY-BASED* TERHADAP SERANGAN DOS

IDS PERFORMANCE ANALYSIS USING ANOMALY-BASED DETECTION METHODS TO DOS ATTACK

Aghnia Shahibah Fadhlillah[1], DR. Nyoman Bogi A K,ST.,MSEE[2], Arif Indra Irawan, ST.,MT.[3]
[1,2,3] Prodi S1 Teknik Telekomunikasi, Fakultas Teknik Elektro, Universitas Telkom
[1]**aghniasf@student.telkomuniversity.ac.id**, [2]**aditya@telkomuniversity.co.id**,
[3]**arifirawan@telkomuniversity.ac.id**

Abstrak

***Intrusion Detection System* (IDS) merupakan sistem yang dapat mendeteksi aktivitas yang mencurigakan dalam sebuah jaringan. Metode deteksi *Anomaly-Based* dipilih agar dapat mendeteksi aktivitas yang mencurigakan dan tidak normal bagi sistem yang tidak dapat dilakukan oleh metode *Signatured-based.* Pada penelitian ini dilakukan pengujian serangan menggunakan tiga tools DoS yaitu tools LOIC, Torshammer dan Xerxes dengan scenario pengujian yaitu menggunakan IDS serta tanpa IDS.**

Dari hasil pengujian yang telah dilakukan IDS berhasil mendeteksi serangan yang dikirim, untuk pengiriman paket serangan terbanyak berurutan yaitu Torshammer, Xerxes dan LOIC. Pada pendeteksian tools serangan Torshammer kepada target FTP Server didapatkan sebanyak 9421 paket, untuk tools Xerxes yaitu sebanyak 10618 paket dan tools LOIC sebanyak 6115 paket. Sedangkan serangan kepada target Web Server untuk tools torhsammer sebanyak 299 paket, untuk tools Xerxes sebanyak 530 paket dan untuk tools LOIC sebanyak 103 paket. Akurasi dari hasil performansi IDS yaitu sebesar 88,66%, presisi sebesar 88,58% serta false positive rate sebesar 63,17%.

Kata kunci : ***Intrusion Detection System, Anomaly-Based,* Keamanan Jaringan, *Denial of Service***

Abstract

Intrusion Detection System (IDS) is a system that can detect suspicious activity in a network. Anomaly-Based detection method is chosen to be able to detect suspicious and abnormal activities for the system that cannot be done by Signatured-based methods. In this study, attack testing was carried out using three DoS tools, namely LOIC, Torshammer and Xerxes tools with the testing scenario of using IDS and without IDS.

From the results of testing that has been done, IDS has successfully detected the attack sent, for sending the most consecutive attack packages, namely Torshammer, Xerxes and LOIC. In the detection of tools, Torshammer's attack on the FTP Server target was 9421 packages, for Xerxes tools, there were 10618 packages and LOIC tools as many as 6115 packages. While attacks on the target Web Server for torhsammer tools as many as 299 packages, for Xerxes tools as many as 530 packages and for LOIC tools as many as 103 packages. The accuracy of the IDS performance results is 88.66%, precision is 88.58% and the false positive rate is 63.17%.

Keywords: ***Intrusion Detection System, Anomaly-Based, Network Security, Denial of Service***

1. Pendahuluan

Keamanan jaringan kini menjadi hal utama yang dibutuhkan untuk mengamankan setiap data. Karena semakin berkembangnya teknologi informasi maka serangan yang dilakukan oleh penyerang juga sangat bermacam-macam. Salah satu serangan yang sering dilancarkan serta tergolong mudah untuk diimplementasikan adalah DoS. Serangan DoS memiliki berbagai macam metode dan biasanya memanfaatkan sumber daya target dengan cara menghabiskan sumber daya servernya sehingga tidak bisa diakses atau mengalami *down*.

Untuk mencegah resiko terkena serangan, maka dibutuhkan langkah dalam pencegahan dengan suatu sistem untuk mengamankan jaringan agar tidak menjadi target seorang penyerang yang bisa merugikan. Salah satu sistem yang dapat digunakan untuk mencegah resiko terkena serangan yaitu *Intrusion Detection System* (IDS) yang dapat mendeteksi aktivitas yang mencurigakan dalam sebuah jaringan dan memberikan sebuah peringatan kepada administrator[1][2][3][4].

Metode deteksi dari IDS terbagi menjadi 2, yaitu *Signature-Based Detection* dan *Anomaly-Based Detection*[1]. Pada penelitian sebelumnya [4][5] telah dilakukan analisis terhadap metode deteksi *signatured-based* yang mampu mendeteksi bermacam-macam *malware* dengan waktu pendeteksian yang relatif cepat serta hasil yang cukup akurat, tetapi kelemahan dalam deteksi *signature-based* yaitu tidak dapat mendeteksi pola serangan baru dan selalu memerlukan pembaruan *rules database* secara manual, sehingga dalam penelitian ini mengusulkan untuk membangun sebuah sistem IDS dengan melakukan analisis menggunakan metode deteksi serangan *Anomaly-based* agar dapat mendeteksi serangan yang mencurigakan dan tidak normal bagi sistem tanpa harus melakukan *update rules* secara manual.

Tujuan dilakukan penelitian ini adalah untuk melakukan simulasi untuk melindungi keamanan jaringan dengan IDS menggunakan metode deteksi *Anomaly-Based,* melakukan analisis pengaruh dari serangan dengan IDS dan tanpa IDS serta melakukan analisis performansi baik dalam tingkat akurasi, presisi dan False Positive Rate.

2. Metodologi

2.1 IDS

Intrusion Detection system (IDS) merupakan sebuah perangkat lunak maupun perangkat keras keamanan yang melakukan monitoring terhadap trafik jaringan dan memberikan peringatan kepada administrator jika terjadi suatu aktivitas yang dianggap berbahaya ataupun mencurigakan [2][6][3][7]. IDS bekerja dengan cara memonitor aktivitas sistem dan memeriksa kerentanan di dalam sistem, integritas *file* dan melakukan analisis pola berdasarkan serangan.[1] [8]. Pada dasarnya IDS melengkapi sistem keamanan jaringan jika diletakkan bersama dengan firewall, hanya saja IDS bersifat pasif yang hanya dapat melakukan deteksi saja tanpa adanya tindakan yang dilakukan.[2]

2.1.1 Metode Deteksi

a. *Signature-Based:* Paket yang masuk ke dalam sistem akan dimonitor dan disimpan kedalam log file, lalu setiap paket yang masuk akan diidentifikasi dan dibandingkan dengan pola yang telah ditetapkan. Jika terdapat paket yang sesuai dengan pola yang ada maka paket yang masuk akan di *drop* dan akan ada peringatan kepada administrator[1][8][9][10]. Pada umumnya *signatured-based* lebih mudah diimplementasikan karena *rule* yang digunakan menggunakan algoritma *pattern matching* yang dapat mendeteksi serangan dengan cepat serta memiliki *false negative* atau kesalahan deteksi yang rendah.[8]

b. *Anomaly-based:* Melakukan monitoring lalu lintas paket dan membandingkannya dengan pola lalu lintas normal sistem (*baseline*)[10] terhadap intrusi. Jika ada lalu lintas yang mencurigakan dan tidak sesuai *baseline* maka akan diidentifikasi dan memberikan peringatan kepada administrator serta melakukan pencegahan. [1] [8][11] Karena adanya *baseline*, metode deteksi ini dapat mendeteksi serangan jenis baru yang masuk kedalam sistem.

2.1.2 Performansi IDS

[12]Untuk mengetahui tingkat kinerja dari sistem yang telah dibangun, sebuah IDS harus dapat melakukan klasifikasi yang sesuai dengan keadaan yang terjadi. IDS harus dapat membedakan keadaan intrusi dan kondisi normal. Berikut ini merupakan tabel matrix yang mewakili hasil dari klasifikasi IDS[12]:

Tabel 2. 1 Matrix Evaluasi Deteksi Sistem

Kondisi Asli	Hasil Deteksi	
	Intrusi	Normal
Intrusi	True Positive (TP)	False Negative (FN)
Normal	False Positive (FP)	True Negative (TN)

- True Positive (TP): Intrusi yang berhasil dideteksi oleh IDS
- True Negative (TN): Kondisi normal yang berhasil dideteksi sebagai keadaan normal oleh IDS
- False Positive (FP): Kondisi normal yang diklasifikasikan intrusi oleh IDS
- False Negative (FN): Intrusi yang tidak terdeteksi oleh IDS dan diklasifikasikan debagai keadaan normal.

Untuk dapat menghitung performansi IDS yang telah dibangun dibutuhkan beberapa parameter yaitu sebagai berikut:

1. Akurasi (CR): Perbandingan dari klasifikasi yang benar dengan jumlah total pada dataset[12].

$$CR = \frac{TP + TN}{TP + TN + FP + FN} \quad (2\text{-}1)$$

2. Presisi (PR): Sebagian dari data yang dideteksi sebagai positif yang sebenarnya posistif[12].

$$PR = \frac{TP}{TP + FP} \quad (2\text{-}2)$$

3. False Positive Rate (FPR): Perbandingan antara jumlah normal yang terdeteksi intrusi dan total jumlah yang normal[12].

$$FPR = \frac{FP}{TN + FP} \quad (2\text{-}3)$$

2.2 Denial of Service (DoS)

Denial of Service (DoS) merupakan sebuah serangan yang bertujuan untuk merusak atau mengacaukan layanan. Serangan DoS bekerja dengan cara menghabiskan sumber daya yang dimiliki server sehingga server tersebut tidak bisa diakses dan tidak dapat menjalankan fungsinya. Biasanya penyerang sering melakukan serangan DoS dengan cara membanjiri trafik dengan banyak paket yang dikirim ke server. Selain itu teknik yang digunakan untuk melakukan serangan DoS sangat bermacam-macam diantaranya adalah menghabiskan *bandwidth*, serangan paket SYN, serangan paket ICMP, menyerang keamanan aplikasi, mengirimkan *request* layanan yang banyak, serangan *peer-to-peer* atau permanen DoS. [11]

2.3 LOIC

Low Orbit Ion Cannon (LOIC) adalah salah satu tools yang bersifat open source dan dapat digunakan untuk melakukan serangan DoS atau DdoS serta dapat dipasang didalam sistem operasi Linux maupun Windows[13].

2.4 Xerxes

Xerxes merupakan salah satu tools sederhana yang dapat digunakan untuk melakukan serangan DoS dengan cara menyerang server secara langsung dan dapat diluncurkan dari satu sistem. Xerxes bekerja dengan cara melakukan serangan *flooding* dengan protokol TCP ke target yang mengakibatkan sumber daya target habis. Xerxes mampu mencatat web server tanpa perlu menghasilkan sejumlah besar lalu lintas yang memungkinkan tidak diperhatikan oleh pertahanan jaringan[14].

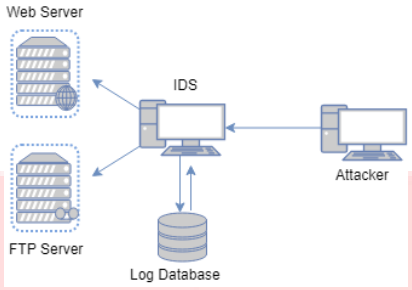
2.5 Torshammer

Torshammer adalah tools serangan DoS slow post HTTP yang ditulis dengan bahasa python yang sangat efisien dan memiliki dampak yang besar pada server apache. Sama seperti serangan DoS biasa, slow post DoS mengakibatkan sumber daya target menjadi habis dengan cara menghasilkan sejumlah besar koneksi HTTP POST selama mungkin yang dapat bertahan sekitar 1000-30000 detik[15].

3. Perancangan Sistem

3.1. Desain Sistem

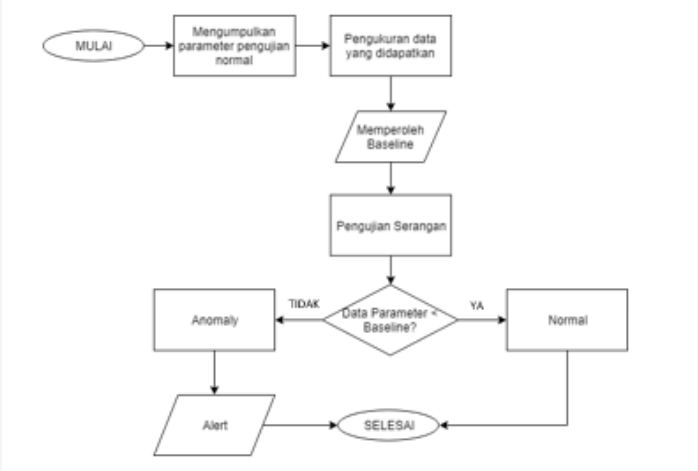
Gambaran umum desain sistem dapat dilihat pada gambar 1, target serangan adalah Web Server dan FTP Server dengan melewati IDS. Semua trafik yang masuk melewati sistem IDS akan dimasukkan dan disimpan didalam *log database*. IDS akan disimulasikan berupa Network IDS yang tidak hanya mendeteksi satu client tetapi dapat mendeteksi pada satu jaringan. Dibangun pada system operasi Ubuntu 16.04 dengan salah satu tools IDS yaitu Zeek yang dapat mendeteksi sesuatu yang dianggap anomaly serta berbahaya didalam jaringan, salah satunya adalah serangan DoS. Selain itu, didalam OS utama akan dipasang sebuah mesin virtual target yang berisi layanan Web serta FTP. Attacker akan berada pada jaringan yang sama tetapi dari luar PC dan melakukan berbagai metode serangan DoS kepada masing-masing target



Gambar 1.1 Blok Sistem

Berikut merupakan alur proses penelitian sistem IDS berdasarkan deteksi anomali dapat dilihat pada gambar 3.3. Tahapan dari proses serangan berdasarkan deteksi anomali adalah sebagai berikut:

1. Sistem akan memonitor dan mencatat paket yang masuk maupun keluar jaringan.
2. Semua informasi yang didapatkan dari memonitor jaringan diukur untuk menentukan rata-rata trafik jaringan normal seperti penggunaan CPU, *memory* dan *network* pada server target.
3. Hasil dari pengumpulan trafik akan dianalisis untuk dijadikan rata-rata normal sistem target.
4. Melakukan pengujian berupa serangan DoS dengan beberapa tools yaitu LOIC, Torshammer dan Xerxes untuk melihat apakah ketiga *tools* serangan DoS ini efektif untuk menyerang sebuah target yang telah dipasang system IDS.
5. Ketika sistem mendeteksi adanya perbedaan antara trafik normal maka sistem IDS akan memberikan peringatan kepada administrator.



Gambar 1.2 Diagram Alir Sistem

3.2 Spesifikasi Komponen

3.2.1 Spesifikasi Perangkat Keras

Untuk dapat menjalankan sistem ini, dibutuhkan beberapa spesifikasi perangkat keras yaitu sebagai berikut:

Tabel 3.1 Spesifikasi Perangkat Keras

Perangkat	**Target**	**Attacker**
Sistem Operasi	Ubuntu 16.04	Kali Linux
RAM	4GB	4GB
CPU	AMD A8-7410 APU with AMD Radeon R5 Graphics 2.20Gz	Intel Core i3
Hardisk	60GB	50GB

3.2.2 Spesifikasi Perangkat Lunak

1. Target:
   - Ubuntu 16.04, merupakan salah satu operation system yang digunakan sebagai OS utama yang akan dipasangi NIDS.
   - Zeek merupakan salah satu tools IDS yang akan digunakan untuk mendeteksi anomali.
   - VMWare merupakan *Virtual Machine* yang digunakan untuk menjalankan Server target.
   - Apache Web Server merupakan *tools* web server yang akan menjadi salah satu target.
   - VSFTPD FTP Server merupakan *tools* yang akan menjadi salah satu target.
2. Attacker:
   - Nmap merupakan *tools* yang akan digunakan untuk melakukan scanning service yang berjalan pada server target.
   - LOIC, Torshammer, Xerxes merupakan *tools* DoS *attack* yang menyerang kepada server target.

4. Hasil dan Analisis

4.1 Hasil Pengujian dan Analisis Serangan

Berikut ini merupakan kondisi normal target sebelum terkena serangan dalam tiga waktu pengujian dengan rata-rata total CPU 16,37%, Memory 44,33%, dan Network 111,43 bytes/s.

Tabel 4. 1 Keadaan Normal Server Target

	Percobaan 1	Percobaan 2	Percobaan 3
CPU	16,70%	13,23%	18,96%
Memory	44,48%	40,28%	41,95%
Network	23,32 bytes/s	133,16bytes/s	177,82 bytes/s

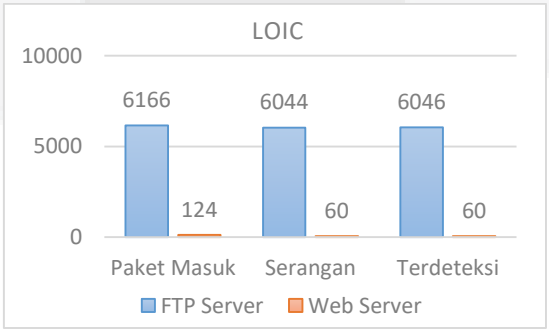
4.1.1 Pengujian dan Analisis Serangan tools LOIC

Dari hasil pengujian yang telah dilakukan, dapat dilihat pada tabel 4.2 rata-rata CPU web server selama penyerangan tanpa memasang IDS mengalami kenaikan dari kondisi normal yaitu sebanyak 30% sedangkan untuk FTP server mengalami kenaikan sekitar 49% dan setelah dipasang IDS masing-masing target mengalami penurunan. Pada parameter memory web server maupun FTP server memiliki hasil yang hampir sama ketika sebelum dipasang IDS mengalami sedikit kenaikan dan sedikit penurunan sesudah dipasang IDS.

Tabel 4. 2 Hasil Serangan Tools LOIC

		Tanpa IDS		Dengan IDS	
		Web Server	FTP Server	Web Server	FTP Server
CPU %		46,29	65,64	42,07	61,75
Memory %		44,27	44,84	44	44,65
Network (Kbps)	Receive	500,65	466,14	497,73	469,41
	Send	143,5	149,21	140,44	148,89

Pada gambar 4.5 merupakan hasil dari pendeteksian IDS yang berhasil melakukan pengecekan paket serta melakukan pemberitahuan kepada administrator.



Gambar 4. 1 Serangan yang Masuk dan terdeteksi IDS

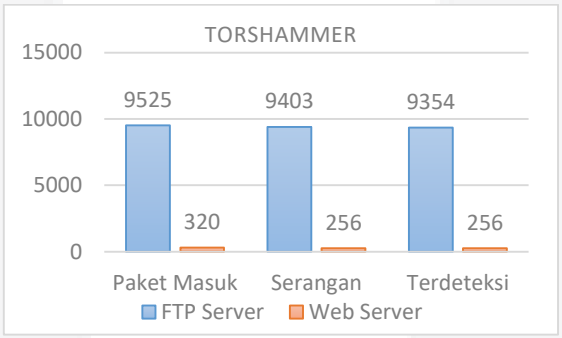
Serangan yang masuk kepada target FTP server jauh lebih banyak dibandingkan dengan target web server, hal ini terjadi karena pada penyerangan tools LOIC sesi HTTP terjadi setelah flag SYN, SYN+ACK serta ACK dengan jangka waktu yang relatif sebentar hanya saja ukuran data lebih besar dibandingkan dengan dua tools serangan lain. Dan pada penyerangan FTP Server pola serangan hampir sama hanya saja ukuran data yang dikirimkan oleh LOIC kepada target FTP server lebih besar dibandingkan dengan ukuran data yang dikirimkan kepada target web server sehingga beban yang masuk kepada CPU target FTP server dan web server sedikit berbeda.

4.1.2 Pengujian dan Analisis Serangan Tools Torshammer

Pada tabel hasil dari pengujian serangan menggunakan tools torshammer pada target web server sebelum memasang IDS naik sekitar 7% sedangkan pada target FTP server mengalami kenaikan hingga 80% tetapi setelah dipasang IDS masing-masing target mengalami sedikit penurunan. Untuk parameter memory baik pada target web server maupun FTP server setelah terkena serangan mengalami kenaikan tetapi setelah dipasang IDS mengalami penurunan.

Tabel 4. 3 Hasil Serangan Tools Torshammer

		Tanpa IDS		Dengan IDS	
		Web Server	FTP Server	Web Server	FTP Server
CPU %		24,74	96,44	23,01	95,57
Memory %		62,53	84,18	36,14	67,37
Network (Kbps)	Receive	10,88	38,54	10,73	34,45
	Send	10,71	34,15	10,57	27,84



Gambar 4. 2 Serangan yang Masuk dan Terdeteksi IDS

Serangan yang masuk kedalam target FTP server jauh lebih banyak dibandingkan dengan web server. Hal ini terjadi karena pada penyerangan terhadap FTP Server pola kerja TCP selalu mengirimkan flag SYN, SYN+ACK serta out of order karena permintaan yang terlalu banyak dan tanpa adanya ACK untuk beberapa waktu tertentu sehingga melakukan retransmission atau request pengiriman kembali. Dapat dilihat dari network usage receiving pada target FTP server lebih tinggi dibandingkan dengan web server. Sehingga CPU serta memory pada target FTP relative tinggi dikarenakan selalu banyak sesi yang terjadi sebelum mendapatkan flag ACK dan koneksi FTP terhubung. Sedangkan pada web server penyerangan protokol TCP pada torshammer selalu mengirim koneksi dengan flag SYN, SYN+ACK, ACK lalu sesi HTTP terjadi. Request selanjutnya selalu dilayani ketika request sebelumnya berhasil sehingga paket yang masuk kepada target web server tidak terlalu banyak dan CPU serta memory tidak terlalu terpengaruh.

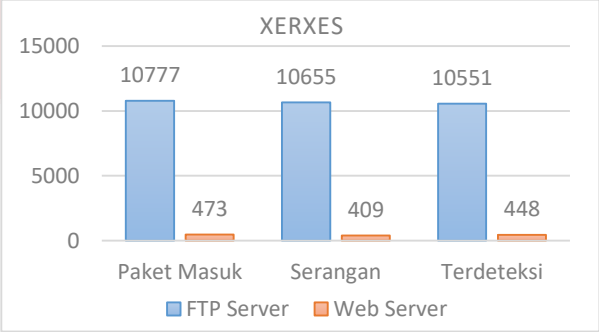
4.1.3 Pengujian dan Analisis Serangan Tools Xerxes

Dari hasil pengujian yang telah dilakukan, dapat dilihat pada table 4.6 bahwa perbandingan CPU yang dihasilkan cukup berbeda, pada Web Server tanpa memasang IDS rata-rata CPU hanya naik sekitar 5% sedangankan pada FTP server rata-rata CPU naik hingga 72% dan setelah dipasang IDS masing-masing target sedikit mengalami penurunan. Untuk memory pada Web Server sebelum dipasang IDS naik hingga 22% dan sedikit menurun setelah dipasang IDS. Sedangkan pada FTP server, memory naik hingga 40% sebelum dipasang IDS dan sedikit mengalami penurunan setelah dipasang IDS.

Tabel 4. 8 Hasil Serangan Tools Xerxes Tanpa IDS

		Tanpa IDS		Dengan IDS	
		Web Server	FTP Server	Web Server	FTP Server
CPU %		30,67	90,81	20,69	89,73
Memory %		66,19	86,89	35,32	81,57
Network (Kbps)	Receive	63,01	87,26	61,54	86,63
	Send	61,87	90,23	60,17	90,18

Lalu Sistem IDS mencatat semua log yang masuk kedalam jaringan dan memberikan peringatan ketika serangan dari tools Xerxes masuk.



Gambar 4. 3 Serangan yang Masuk dan terdeteksi IDS

Pada target FTP server serangan yang masuk terhitung lebih banyak dari pada serangan yang masuk kedalam web server. Hal ini terjadi karena pada penyerangan terhadap FTP Server, sesi FTP Selalu terjadi setelah flag SYN, SYN+ACK dan ACK lalu koneksi FTP terhubung dengan request yang sangat banyak sehingga CPU serta Memory meningkat ketika diserang. Sedangkan pada penyerangan pada web server, protokol TCP mengirimkan flag yang berurutan sesuai three way handshake yaitu SYN, SYN+ACK, ACK lalu pengiriman data. Sesi ini selalu berlangsung setiap suatu data selesai dikirimkan kepada target. Seperti yang terlihat pada table 4.6 bahwa network usage pada FTP server selalu lebih banyak dibandingkan dengan web server.

4.2 Hasil dan Analisis Permormansi IDS

Pada pengujian menggunakan IDS dilakukan perhitungan untuk melihat hasil dari performansi IDS, parameter yang digunakan yaitu akurasi dan presisi serta false positive rate.

Tabel 4. 11 Hasil Performansi IDS

Tools	LOIC		TORSHAMMER		XERXES	
Target	FTP Server	Web Server	FTP Server	Web Server	FTP Server	Web Server
Akurasi	98,88%	65,32%	98,41%	83,98%	98,79%	86,56%
Presisi	98,87%	65,04%	99,37%	83,29%	99,29%	85,61%
False Positive Rate	55,64%	67,18%	54,91%	79,61%	54,51%	67,18%

Dapat dilihat pada table 4.11 performansi dari pendeteksian serangan terhadap FTP server rata-rata memiliki hasil yang hampir sama yaitu untuk akurasi sekitar 98%, presisi sekitar 99% dan untuk false positive rate sekitar 54%. Sedangkan dari hasil pendeteksian terhadap web server terhadap serangan LOIC berbeda dibandingkan dengan tools Torshammer dan Xerxes karena false positive yang dihasilkan cukup tinggi sedangkan paket serangan yang masuk sedikit sehingga mempengaruhi hasil akurasi dan presisi. Sedangkan pada false positive rate torshammer terhadap target web server lebih tinggi dibandingkan yang lain karena cara kerja serangan torshammer terhadap web server dengan cara melakukan koneksi HTTP POST seperti paket normal namun dengan koneksi yang banyak sehingga IDS keliru membedakan antara serangan dan paket normal. Dari keseluruhan hasil akurasi dan presisi, pendeteksian pada web server selalu lebih rendah dibandingkan dengan FTP server karena false positive yang dihasilkan pada web server selalu lebih tinggi. Semakin tinggi false positive yang dihasilkan maka akan semakin rendah akurasi dan presisi yang didapatkan.

5. Kesimpulan

1. IDS yang telah dibangun dapat mengurangi efek dari rata-rata tiga tools serangan DoS.
2. Rata-rata akurasi pada system yang telah dibangun adalah sebesar 88,66%, rata-rata presisi sebesar 88,58% serta rata-rata false positive rate sebesar 63,17%.
3. Berdasarkan hasil performansi yang didapat, hasil akurasi dan presisi dari web server selalu lebih rendah dibandingkan dengan FTP server dikarenakan false positive web server yang lebih tinggi dibandingkan dengan FTP server.
4. Pada pengujian serangan menggunakan tiga tools DoS, target FTP Server selalu terserang lebih banyak paket yang mempengaruhi kepada CPU dan Memory yang menyebabkan kenaikan tinggi dibandingkan kondisi normal daripada target Web Server karena cara kerja serangan tools yang berbeda memperngaruhi target yang dituju.
6. Secara keseluruhan target web server dan FTP server dapat dilumpuhkan oleh tools LOIC. Sedangkan untuk tools serangan Xerxes dan Torshammer hanya efektif melumpuhkan target FTP server saja.
7. Meskipun pada system IDS dengan metode anomaly-based dapat mendeteksi serangan secara otomatis tanpa melakukan update rules, tetapi system ini memiliki kekurangan pada false positive yang terhitung tinggi.
8. Untuk akurasi pendeteksian pada sistem yang telah dibangun sudah cukup baik karena rata-rata serangan diklasifikasikan dengan benar oleh IDS.

Daftar Pustaka:

[1] T. Thomas, “Network Security First Step,” 2005. .

[2] R. T. Gaddam and M. Nandhini, “An analysis of various snort based techniques to detect and prevent intrusions in networks: Proposal with code refactoring snort tool in Kali Linux environment,” *Proc. Int. Conf. Inven. Commun. Comput. Technol. ICICCT 2017*, no. Icicct, pp. 10–15, 2017.

[3] S. P. Zuhri *et al.*, “PENYELEKSIAN NOTIFIKASI SERANGAN PADA JARINGAN KOMPUTER BERBASIS IDS SNORT MENGGUNAKAN METODE K-,” vol. 2015, pp. 2–5, 2015.

[4] M. Program and S. D. Manajemen, “Sistem Monitoring Pendeteksian Penyusup Menggunakan Snort pada Jaringan Komputer Fakultas Ekonomi Universitas Methodist Indonesia Sistem Monitoring Pendeteksian Penyusup Menggunakan Snort pada Jaringan Komputer Fakultas Ekonomi Universitas Methodist Indonesia Dosen Program Studi D-III Manajemen Informatika Universitas Methodist Indonesia Abstrak,” no. September 2015, 2018.

[5] A. Garg, “Performance Analysis of Snort-based Intrusion Detection System,” pp. 0–4, 2016.

[6] M. Program and S. D. Manajemen, “Sistem Monitoring Pendeteksian Penyusup Menggunakan Snort pada Jaringan Komputer Fakultas Ekonomi Universitas Methodist Indonesia Sistem Monitoring Pendeteksian Penyusup Menggunakan Snort pada Jaringan Komputer Fakultas Ekonomi Universitas Methodist Indon,” no. September 2015, 2018.

[7] M. M. K. Means, “Seleksi Notifikasi Serangan Berbasis IDS Snort,” vol. 3, no. 2, pp. 31–38, 2017.

[8] S. Potteti and N. Parati, “Intrusion detection system using hybrid Fuzzy Genetic algorithm,” *Proc. - Int. Conf. Trends Electron. Informatics, ICEI 2017*, vol. 2018-Janua, pp. 613–618, 2018.

[9] D. Silalahi, Y. Asnar, and R. S. Perdana, “Rule generator for IPS by using honeypot to fight polymorphic worm,” *Proc. 2017 Int. Conf. Data Softw. Eng. ICoDSE 2017*, vol. 2018-Janua, pp. 1–5, 2018.

[10] R. M. Yousufi, P. Lalwani, and M. B. Potdar, “A network-based intrusion detection and prevention system with multi-mode counteractions,” *Proc. 2017 Int. Conf. Innov. Information, Embed. Commun. Syst. ICIIECS 2017*, vol. 2018-Janua, pp. 1–6, 2018.

[11] J. Hidayat, *Certified Ethical Hacker 500% Illegal*. Jakarta: Jasakom, 2014.

[12] G. Kumar, “Evaluation Metrics for Intrusion Detection Systems - A Study,” vol. 2, pp. 11–17, 2014.

[13] Sleshdot Media, “LOIC,” 2019. [Online]. Available: https://sourceforge.net/projects/loic/. [Accessed: 25-Feb-2019].

[14] sepehrdaddev, “Xerxes,” 2018. [Online]. Available: https://github.com/sepehrdaddev/Xerxes. [Accessed: 25-Feb-2019].

[15] S. Media, “Torshammer,” 2019. [Online]. Available: https://sourceforge.net/projects/torshammer/. [Accessed: 25-Feb-2019].