SISTEM PENDETEKSI DATA FLOODING MENGGUNAKAN SNORT (STUDI KASUS : JARINGAN IT TELKOM SURABAYA)

Reksa Aldian Rahmandy Putra[*1)], Oktavia Ayu Permata[2)], dan Rizky Fenaldo Maulana[3)]
[1)]Fakultas Teknologi Informasi & Bisnis, Institut Teknologi Telkom Surabaya, Jl. Ketintang No.156, Surabaya, 60231, Indonesia
reksa.aldian@student.ittelkom-sby.ac.id

Abstrak

Ancaman serangan jaringan, khususnya tipe serangan flooding, semakin meningkat dan dapat mengganggu kinerja jaringan dan layanan. Serangan ini berpotensi menyebabkan gangguan pada ketersediaan layanan jaringan, menghambat aktivitas layanan tersebut. Oleh karena itu, tujuan penelitian ini adalah mengembangkan dan menerapkan sistem pendeteksi serangan flooding menggunakan Snort pada jaringan IT Telkom Surabaya. Sistem pendeteksian ini bertumpu pada Snort, yang merupakan Host-based IDS. Penelitian melibatkan instalasi dan konfigurasi IDS menggunakan Snort pada server, dengan notifikasi serangan dikirimkan melalui Telegram Bot API. Serangan flooding seperti TCP Flood, UDP Flood, dan ICMP Flood diuji pada jaringan lokal. Hasil menunjukkan bahwa Snort efektif mendeteksi berbagai serangan flooding. Efisiensi pendeteksian berbeda, misalnya untuk SYN TCP Flood sebesar 48.19%, UDP Flood sebesar 35.95%, dan ICMP Flood sebesar 56.00%. Periode pengujian 10 menit meningkatkan efisiensi menjadi 72.68% untuk SYN TCP Flood, 67.73% untuk UDP Flood, dan 89.99% untuk ICMP Flood. Sebagai kesimpulan, Snort efektif melindungi jaringan dari serangan flooding dengan efisiensi beragam pada berbagai jenis serangan, sehingga dapat diandalkan mengatasi ancaman serangan jaringan.

Kata kunci: *Snort, Flooding Data, Snort Efisiensi*

1. Pendahuluan (Introduction)

Keamanan jaringan telah menjadi perhatian serius di dunia teknologi, terutama seiring dengan kemajuan pesat dalam komunikasi dan pertukaran data. Jaringan kampus, yang merupakan bagian penting dari lingkungan pendidikan, juga menghadapi risiko ancaman keamanan digital. Salah satu ancaman yang semakin mengkhawatirkan adalah serangan flooding data. Serangan ini bisa berdampak serius terhadap ketersediaan dan kehandalan layanan jaringan, yang akhirnya berpengaruh pada produktivitas dan operasional lembaga pendidikan.

Serangan data flooding, tergolong dalam Denial of Service (DoS) atau Distributed Denial of Service (DDoS), melibatkan pengiriman lalu lintas berlebihan ke infrastruktur jaringan yang menjadi target. Dampaknya adalah pembebanan berlebihan pada sumber daya jaringan dan layanan yang semestinya dapat diakses oleh pengguna menjadi tidak dapat dijangkau. Dalam konteks jaringan kampus, serangan data flooding dapat mengganggu aktivitas mahasiswa, dosen, dan staf dalam mengakses sumber daya online, sistem informasi akademik, serta layanan penting lainnya.

Kebutuhan akan akses internet semakin meningkat. Banyak juga Lembaga Pendidikan yang menggunakan internet sebagai sarana untuk membantu dalam kegiatan belajar mengajar. Internet memberikan kemudahan dalam komunikasi dan transmisi data. Selain memiliki banyak kelebihan, internet juga memiliki banyak kekurangan. Salah satu kelemahan internet adalah dalam hal keamanan siber. Serangan terhadap sistem keamanan siber merupakan hal yang umum akhir-akhir ini. Jenis serangan yang terjadi adalah DoS (*Denial of Service)* (Aprilianto, dkk, 2017)

Serangan flooding adalah tindakan menyerang yang dilakukan dengan mengirimkan permintaan berlebihan ke server, sehingga server menjadi tidak mampu untuk memproses permintaan tersebut, bahkan berpotensi menyebabkan sistem menjadi lambat atau crash. Efek dari situasi ini adalah jaringan

komputer tidak berjalan sebagaimana seharusnya, dan dampak dari serangan ini mencakup penggunaan sumber daya yang berlebihan, memakan banyak RAM, serta mengisi hardisk dengan data yang tidak relevan, sehingga mengganggu semua aktivitas yang berjalan, dan tidak lagi mampu menangani permintaan yang seharusnya dijalankan (Peniarsih dan Muhammadi, 2014).

Flooding data merupakan jenis serangan *Denial of Service (DoS)* di mana serangan ini mengincar komputer atau server dalam jaringnan lokal atau internet dengan tujuan menghabiskan sumber daya yang dimiliki oleh komputer tersebut hingga komputer tersebut tidak mampu menjalankan fungsinya secara efektif (A.H. Hambali dan Nurmiati, 2018)

Beberapa variasi pendeteksian terhadap serangan flooding data menggunakan berbagai metode, termasuk pemblokiran IP dan Port serta pemanfaatan *Wireshark* sebagai alat pemantauan (Nofriadi N, 2018). Penerapan filter *Firewall* pada router mikrotik juga memiliki efek positif dalam mengurangi jumlah paket data UDP yang dikirim oleh penyerang melalui port DNS, dapat mengurangi hingga 60% dari total paket yang masuk tanpa adanya firewall (Aprilianto, dkk, 2017) . Perlu ditekankan bahwa Snort dan *Firewall* memiliki peran yang berbeda dalam aspek keamanan jaringan. Snort lebih menitikberatkan pada deteksi serangan dan analisis yang mendalam, sedangkan firewall memberikan perlindungan umum melalui penghalangan akses dan isolasi jaringan.

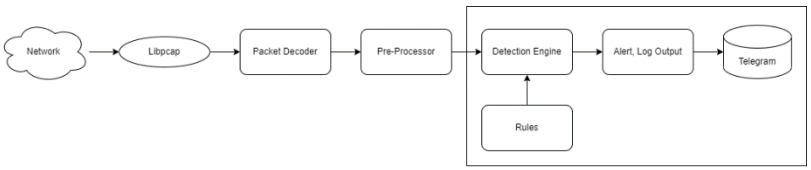
Dalam situasi ini, penelitian akan terfokus pada pengembangan dan pelaksanaan sistem pendeteksi flooding data menggunakan Snort dengan studi kasus yang terkait dengan jaringan kampus. Melalui studi kasus ini, akan diberikan pemahaman lebih mendalam mengenai operasi sistem pendeteksian dalam lingkungan real, serta akan diulas tantangan serta pencapaian dalam menjaga keamanan infrastruktur jaringan kampus dari serangan flooding data.

Tujuan dari penelitian ini adalah untuk mengenali sistem lalu lintas jaringan dan memantau adanya peringatan mengenai flooding data pada jaringan komputer secara terstruktur, untuk meningkatkan kenyamanan pengguna jaringan. Metode pendeteksian yang digunakan dalam penelitian ini adalah *Intrusion Detection System (IDS)* dengan menggunakan alat Snort. Komputer IDS akan berinteraksi dengan jaringan dan memonitor alur lalu lintas berdasarkan label data. Snort menjalankan fungsinya berdasarkan aturan yang ditetapkan. Snort adalah teknologi sensor yang berfokus pada pendeteksian penyalahgunaan dalam jaringan dan memiliki peraturan (*Rules)* yang membantu mengidentifikasi aktivitas jaringan. Ketika server komputer mencatat suatu insiden, informasi mengenai insiden tersebut akan direkam dalam log Snort. Lalu lintas ini kemudian dapat dipantau oleh *Administator* jaringan saat terjadi serangan flooding data, dengan notifikasi yang dikirim melalui Bot Telegram.

Dalam penerapan sistem monitoring flooding data ini, keluaran yang dihasilkan berupa notifikasi melalui Telegram. Telegram adalah sebuah platform pesan yang dapat diakses melalui ponsel dan komputer, dengan focus pada keamanan dan kecepatan dalam penyimpanan data. Oleh karena itu, dalam penelitian ini, diciptakan suatu sistem yang mampu mendeteksi dan memonitor lalu lintas yang terkena dampak dari serangan flooding data atau tidak, dengan notifikasi melalui Telegram.

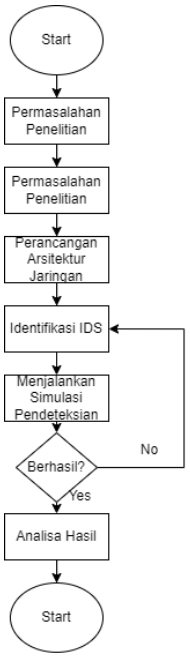
2. Metode Penelitian (Methods)

Metode merupakan suatu tahapan untuk melakukan proses penelitian agar tercapainya hasil yang diharapkan. Metode yang digunakan pada penelitian ini adalah IDS. IDS adalah singkatan dari *Intrusion Detection System*, yaitu sebuah sistem untuk mendeteksi aktivitas yang mencurigakan dalam sebuah sistem atau jaringan. IDS dapat melakukan inspeksi terhadap lalu lintas inbound dan outbound dalam sebuah sistem jaringan (Peniarsih dan Muhammadi, 2014). IDS bekerja dengan memonitor lalu lintas jaringan dan atau host serta dapat menganalisis paket-paket data yang masuk kedalam sistem.



Gambar 2.1 Metodologi Penelitian

Network memberikan sebuah informasi dan ditangkap oleh libpcap untuk memantau lalu lintas jaringan secara real time. Setelah masuk pada library jaringan, packet yang telah diterima akan dikirimkan ke *Detection Engine* agar dapat terdeteksi. Rules pada snort dibuat pada *Detection Engine*. Kemudian *Penetration Test* dengan semulasi penyerangan dilakukan terhadap masing masing port seperti *ICMP,UDP, dan SYN TCP*. Sebuah percobaan penyerangan ke *network* akan di*capture* pada library. *Detection Engine* akan mengidentifikasi apakah data yang dikirim tersebut merupakan serangan flooding atau tidak. Selanjutnya akan memberikan sebuah output berupa notifikasi rincian data pada Bot Telegram.



Gambar 2.2 Tahapan Penelitian

2.1 Permasalahan Penelitian

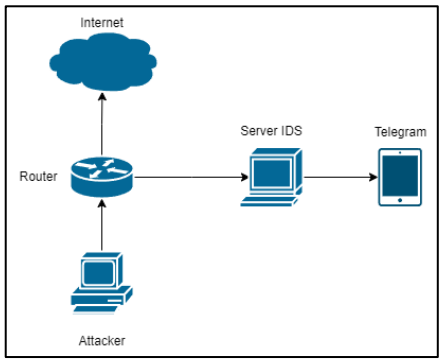
Permasalahan penelitian dilakukan untuk mendeteksi suatu masalah agar permasalahan yang diteliti dapat lebih terukur dan sebagai langkah awal dalam penelitian. Dimana dilakukannya identifikasi masalah pada studi kasus yang telah diterapkan. Sehingga akan memberikan hasil dari permasalahan penelitian ini dengan mengetahui jenis serangan yang terdeteksi oleh IDS pada jaringan di IT Telkom Surabaya.

2.2 Studi Literatur

Untuk memahami lebih lanjut dari suatu metode, maka yang akan perlu dilakukan adalah studi literatur. Studi literatur dilakukan untuk mencari penelitian-penelitian sebelumnya guna mendapatkan referensi. Referensi tersebut terdapat dibeberapa Jurnal nasional, Jurnal internasional maupun E-book. Hasil dari studi literatur penelitian ini adalah studi terkait sebuah metode untuk pendeteksian data flooding yaitu *Intruction Detection System* (IDS) yang berfungsi sebagai pendeteksi trafik lalu lintas jaringan, serta penggunaan tools Snort pada IDS untuk memantau sebuah trafik lalu lintas jaringan.

2.3 Perancangan Arsitektur

Pada perancangan arsitektur jaringan, peneliti menggunakan sebuah topologi guna memudahkan untuk menjalankan penelitian dan memberikan gambaran jelas bahwa suatu lalu lintas jaringan pada sebuah topologi.

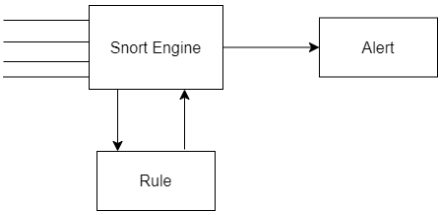


Gambar 2.3 Arsitektur Jaringan

Agar simulasi berjalan maka langkah awal yang dilakukan adalah menyusun arsitektur jaringannya. Pada penelitian ini akan menggunakan 2 PC sebagai Server dan sebagai Attacker. Kemudian terdapat router untuk menghubungkan Jaringan antara Internet, Server, dan Attacker. Nantinya kedua client akan menerima aliran data dari seorang attacker apakah mengalami flooding ataau tidak.

2.4 Identifikasi IDS

IDS terdiri dari Libpcap yang memiliki fungsi sama dengan tcpdump untuk mendeteksi sebuah paket. Fitur Packet Logger digunakan untuk men-debug sebuah lalu lintas jaringan, nantinya Snort akan memberikan sebuah peringatan yang sesuai dengan aturan yang didefinisikan dalam file konfigurasi. Snort Rule membantu dalam membedakan antara aktivitas internet yang normal dan aktivitas berbahaya. *Rules Snort* yang digunakan adalah rules yang sudah tersedia dari pengembang Snort itu sendiri.



Gambar 2.4 Bagan IDS

Bagan IDS (*Intrusion Detection System)* memiliki beberapa komponen utama yaitu pada *Snort Engine* adalah komponen utama sistem IDS yang bertugas untuk menganlisis lalu lintas jaringan dan mendeteksi aktivitas mencurigakan atau berbahaya. Pada Rule berisikan sebuah aturan yang digunakan oleh *Snort Engine* untuk mengidentifikasi sebuah intrusi. Aturan-aturan ini berisi pola-pola atau karakteristik dari serangan jaringan. Alert merupakan sebuah komponen yang bertanggung jawab untuk memberikan peringatan atau notifikasi kepada *administrator* jaringan jika terjadi intrusi

2.5 Menjalankan Simulasi

Simulasi Dilakukan dengan virtualisasi pada *Virtual Machine*. Setelah rancangan arsitektur jaringan telah dibangun, Selanjutnya membuat Rules. *Rules* dan *Detection Engine* memiliki sebuah alat pendukung yaitu Snort. Penedeteksian dilakukan dengan cara *attacker* mengirimkan berupa load data yang berlebih kepada internet. Kemudian data akan disimpan pada library libpcap dimana library yang dapat mengcapture semua paket data yang melewati interface. Lalu

dilakukannya Packet decoder setelah melalui Libpcap. Pada tahap Pre-Processor ini packet akan di defragmentasi dan dilakukannya deteksi scan port sehingga dapat memberikan hasil yang sesuai dengan parameter pengujian. Detection Engine memberikan sebuah rules dari tools Snort dimana packet yang dikirimkan apakah sudah sesuai dengan rules yang tersedia. *Detection Engine* akan memproses aturan yang telah dibuat. Setelah melalui Detection Engine packet akan memberikan output log berupa alert pada aplikasi Bot Telegram.

2.6 Analisa Hasil

Analisa hasil dilakukan terhaap hasil pengujian dengan meninjau besar paket, jenis serangan ketika penyerang melakukan penyerangan. Tingkat keefisiensian dari alat snort terhadap pendeteksian aktivitas lalu lintas jaringan sehingga mendapatkan sebuah tingkat efisiensi dari sistem tersebut.

3. Hasil dan Pembahasan (Results and Discussions)

Penelitian ini dilakukan untuk mendapatkan efisiensi dari pendeteksian snort. Dengan demikian, hasil pada pendeteksian ini memberikan hasil positif bahwa Snort mimiliki efisiensi yang baik dalam mendeteksi dan memfilter lalu lintas pada protokol-protokol. Percobaan *penetration test* dibagi menjadi dua tahap yaitu, tahap pertama dilakukan pada kurun waktu 5 menit, dan tahap kedua dilakukan pada kurun waktu 10 menit. Untuk memastikan apakah kurun waktu pengujian tersebut mendapatkan efisiensi yang maksimal dari Snort, didapatkan beberapa hasil percobaan pengujian ini dengan menganalisa Jenis serangan, Waktu Serangan, dan Snort Efisiensi tersebut.

Tabel 3.1 Parameter Penelitian

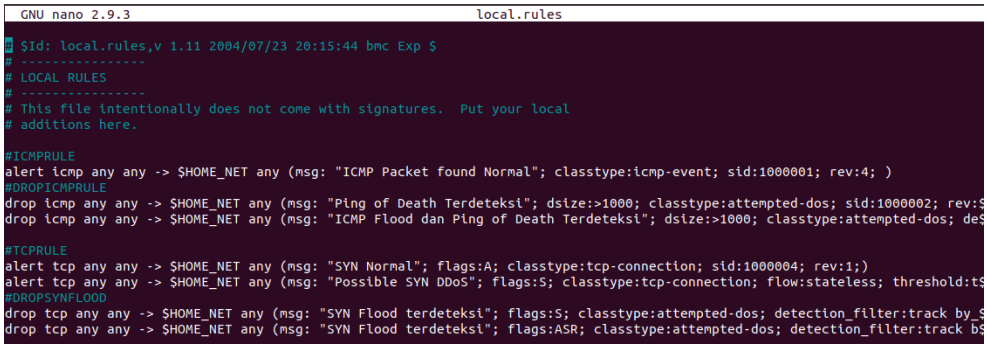
No	Parameter	Keterangan
1	Attack Type	Jenis serangan yang masuk kedalam rules
2	Periode Rules	Standart yang mengatur dari jumlah rules
3	Jumlah Paket	Jumlah paket yang datang sebelum dicek oleh rules
4	Waktu Paket	waktu yang terdeteksi

3.1. Pengujian

Pengujian serangan dilakukan pada jaringan lokal sesuai dengan rancangan yang telah dibuat. Dengan menggunakan bantuan alat penunjang bernama *hping3* sebagai pembantu untuk melakukan *Penetration Test*. Alamat IP dilakukan dengan pengalamatan secara *Dynamic Host Configuration Protocol (DHCP)* sehingga alamat IP yang didapatkan acak sesuai dengan gateway atau ISP.

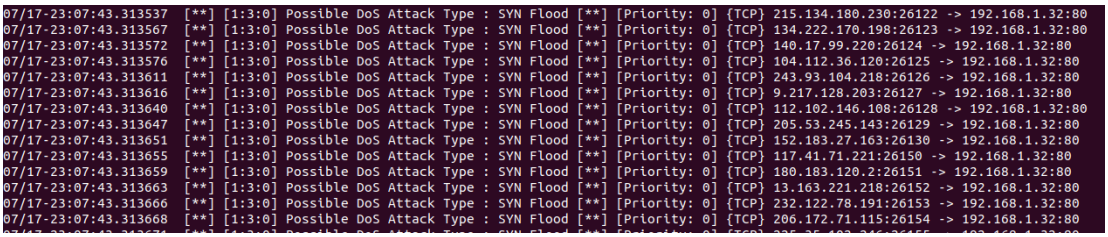
Tabel 3.2 Skenario Penyerangan

No	IP Address Penyerang	IP Address Target	Metode Serangan	Port
1.	192.168.1.20	192.168.1.32	TCP Flood	80,135,139,445
2.	192.168.1.20	192.168.1.32	UDP Flood	80,135,139
3.	192.168.1.20	192.168.1.32	ICMP Flood	22



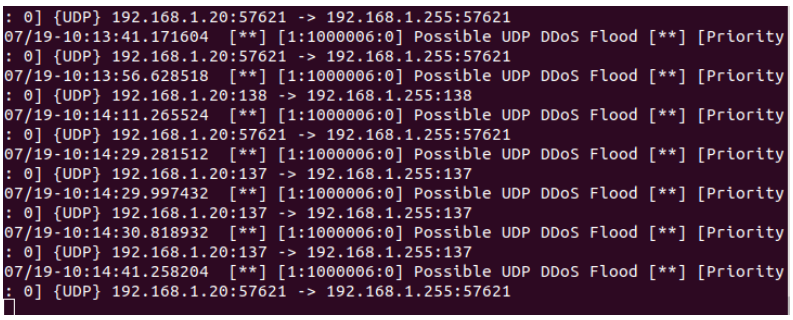
Gambar 3.1 Rules Snort

Pembuatan *rules* atau biasa disebut aturan aturan pada *Detection Engine* adalah hal yang dilakukan guna mendapatkan sebuah filter deteksi yang lebih mendalam dalam hal tersebut. Dengan menuliskan beberapa aturan pada *Rules*, Snort akan otomatis membaca bahwa adanya notifikasi dari sebuah serangan. Dengan menerapkan beberapa rules, akan mempermudah seorang *administrator* dalam melakukan deteksi kepada sistem sehingga snort dapat berjalan sesuai apa yang diharapkan.



Gambar 3.2 Hasil Deteksi terhadap TCP

Hasil deteksi sistem terhadap port SYN TCP terbukti bahwa adanya kemungkinan penyerangan terhadap port TCP. Pada jam tersebut. Sistem berhasil menganalisis bahwa terjadinya suatu kejadian yang berbahaya. Sistem membaca bahwa SYN Flood termasuk keadalam DoS Attack sehingga paket yang masuk dapat terbaca oleh sistem. Source IP yang didapatkan juga beragam dengan hal tersebut memungkinkan bahwa seorang penyerang menggunakan sebuah metode penyerangan *random source* atau biasa disebut dengan alamat ip acak.



Gambar 3.3 Hasil Deteksi terhadap UDP

Hasil deteksi pada UDP terdeteksi dengan keterangan *Possible UDP DDoS Flood* menandakan bahwa memungkinkan adanya suatu flooding yang dikirimkan oleh seorang penyerang terhadap port tersebut. Sehingga sistem membaca bahwa adanya kemungkinan Flooding terhadap konten UDP yang terkirimkan. Keterangan yang diberikan sistem terdiri dari tanggal kejadian, waktu kejadian, dan *Signature ID* (Kode unik), Asal IP, Tujuan IP dan keterangan kejadian tersebut.

3.2. Analisis Hasil

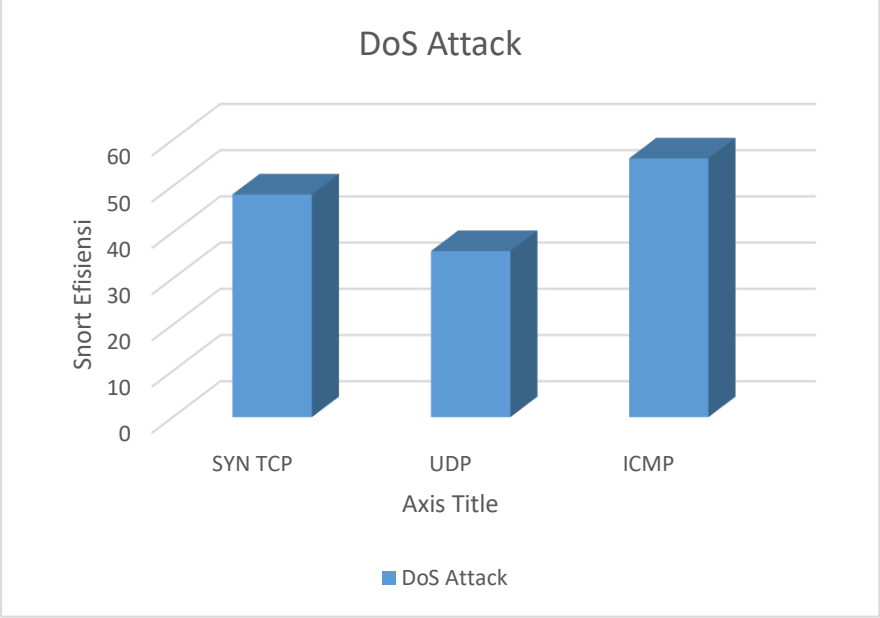
Hasil pengujian yang didapatkan pada dua periode waktu memiliki tingkat keakurasian dari sistem snort yang berbeda. Perbedaan itu bergantung pada banyaknya pengguna ataupun waktu sistem dalam mendeteksi adanya serangan. Snort mampu mendeteksi secara baik dengan periode waktu yang cukup besar. Pada periode yang telah diteliti, menghasilkan beberapa hasil yang berbeda pula, Peeneliti melakukan dengan dua periode waktu untuk percobaan mendapatkan snort efisiensi apakah berjalan dengan efektif atau tidak. Adapun untuk mencari efisiensi dari snort dapat dilakukan dengan cara perhitungan, guna untuk mendapatkan sebuah hasil dari kinerja sistem snort yang optimal dapat dilakukan dengan rumus sebagai berikut :

$$snort\ efficiency = \frac{total\ packet\ analyzed}{60 * 10000} \quad (1)$$

Tabel 3.3 Hasil snort efisiensi periode waktu 5 menit

No	Attack Type	Total Packet Received	Flooding Attack Detected Analyzed	Snort Efficiency
1.	SYN Flood	2915973	2891745	48.19%
2.	UDP Flood	2405364	2157474	35.95%
3.	ICMP Flood	3362640	3360281	56.00%

Adapun grafik batang untuk memvisualisasikan hasil pengambilan data. Pada table diatas menggunakan sumbu x menunjukan bahwa Snort Efisiensi pada sistem dan sumbu y menunjukan rentangan efisiensi dari 0% hingga 100%. Gambar visualisasi data dapat dilihat pada Gambar



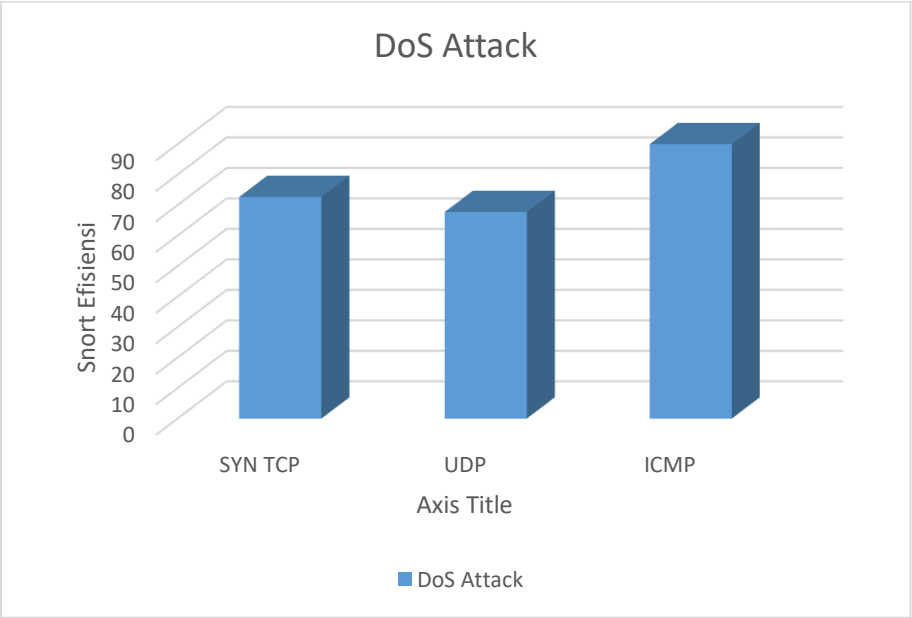
Gambar 3.4 Diagram batang periode waktu 5 menit

Berdasarkan gambar diatas, bahwa snort dapat mendeteksi aktivitas flooding pada jaringan kurang efektif, jumlah persen yang didapatkan oleh snort cenderung rendah dengan jumlah waktu yang singkat. Hal ini disebabkan karena waktu deteksi terlalu singkat sehingga snort kurang dapat melakukan pendeteksian secara efektif. Pengujian pada ICMP terbukti bahwa adanya snort mampu mendeteksi dengan cukup baik,

Tabel 3.4 Hasil snort periode waktu 10 menit

No	Attack Type	Total Packet Received	Flooding Attack Detected Analyzed	Snort Efficiency
1.	SYN Flood	4363548	4360836	72.68%
2.	UDP Flood	4544750	4063952	67.73%
3.	ICMP Flood	5401003	5399831	89.99%

Pada sebuah percobaan periode waktu 10 menit hasil yang didapatkan oleh snort memiliki efisiensi yang tinggi. Hal ini menunjukan bahwa snort dapat mendeteksi dengan kurun waktu yang lebih lama agar mendapatkan sebuah efisiensi dari snort. Besar paket yang datang juga cukup signifikan. Sehingga memiliki angka yang lebih banyak daripada percobaan periode sebelumnya.



Gambar 3.5 Diagram batang periode waktu 10 menit

Adapun Grafik batang untuk memvisualisasikan pengambilan data menunjukan bahwa protokol ICMP memiliki nilai efisiensi yang tinggi terhadap pendeteksian. Hal ini merupakan membutuhkan waktu yang cukup besar dalam pendeteksian dan percobaan dari Snort. Hasil pada ICMP memiliki hasil kenaikan yang cukup signifikan. Pada periode waktu 5 menit pertama mendapatkan sebanyak 56.00% dan periode waktu 10 menit kedua mendapatkan hasil sebanyak 89.99%. Dengan ini kemampuan kinerja snort bekerja dengan baik.

Tabel 3.5 Hasil Deteksi pada sistem

No	Skenario Pengujian	Hasil yang diharapkan	Hasil Pengujian sistem	Kesimpulan
1	SYN Flood	Terdeteksi	Terdeteksi	Berhasil
2	UDP Flood	Terdeteksi	Terdeteksi	Berhasil
3	ICMP Floo	Terdeteksi	Terdeteksi	Berhasil

Tabel 3.6 Hasil Deteksi pada Telegram

No	Jenis Serangan	Waktu		
		Awal Serangan	Waktu Terdeteksi	Terkirim
1.	SYN Flood	18:04:45	18:04:59	18:05:32
2.	UDP Flood	18:03:05	18:03:39	18:04:46
3.	ICMP Flood	17:58:54	18:01:03	18:01:01

Pada tabel diatas tercatat bahwa hasil deteksi pada pengujian berhasil dan terdeteksi oleh sistem, dimana pada Jenis serangan memiliki waktu terdeteksi yang berbeda beda namun, selisih waktu yang didapatkan ketika percobaan tidak memiliki perbedaan yang cukup jauh. Dengan terdeteksinya jenis serangan yang masuk, administrator dapat mengetahui tentang sebuah serangan yang masuk dan akan dilanjutkan dengan terkirimnya notifikasi pada telegram.

3.3. Kesimpulan (Conclusion)

Berdasarkan hasil pengujian, pendeteksian dibagi menjadi 3 yakni, pendeteksian terhadap *SYN Flood, UDP Flood,* dan *ICMP Flood* dengan memetakan port pendeteksian dan penyerangan dapat dilakukan secara terstruktur, sehingga tingkat keakuratan dari pendeteksian snort dapat maksimal. Rule yang telah dibuat dengan sistem IDS, Snort mampu mengenali beberapa macam penyerangan. *Penetration Test* dilakukan dengan dua periode waktu. Pada periode pertama dalam kurun waktu 5 menit terdeteksi *SYN TCP* sejumlah 2891745 Paket, *UDP* sejumlah 2157474 Paket, dan *ICMP* sejumlah 3360281 Paket yang menghasilkan snort efisiensi berbeda yakni, pada *SYN TCP* sejumlah 48.19%, pada *UDP* sejumlah 35.95%, dan pada *ICMP* sejumlah 56.00%. Pada periode kedua dalam kurun waktu 10 menit snort dapat mendeteksi beberapa paket yakni, *SYN TCP* sejumlah 43608336 Paket, *UDP* sejumlah 4063952 Paket, dan *ICMP* sejumlah 5399831 Paket. Hasil snort efisiensi dalam kurun waktu 10 menit yaitu, pada *SYN TCP* sejumlah 72.86%, *UDP* sejumlah 67.73%, dan *ICMP* sejumlah 89.99%.

Dari hasil analisis perbandingan Snort efisiensi, Pada periode pertama dan periode kedua terlihat mengalami peningkatan. Hal ini menunjukan bahwa snort mampu mengatasi lalu lintas jaringan yang semakin berat dalam periode waktu yang lebih lama, serta memberikan deteksi dan analisis yang efektif terhadap aktivitas jaringan yang beragam. Terbukti bahwa snort mampu melakukan pendeteksian berbagai macam serangan pada jaringan serta serangan DoS maupun DDoS lainnya sehingga serangan tersebut dapat berdampak kepada CPU target secara berlebihan.kepada CPU target secara berlebihan.

Ucapan Terima Kasih (Acknowledgement)

Saya ingin mengucapkan terimakasih kepada dosen pembimbing, Ibu Oktavia Ayu Permata dan Bapak Rizky Fenaldo Maulana yang telah membimbing dalam penulisan karya ilmian ini. Penulis juga ingin mengucapkan terima kasih kepada teman-teman dan keluarga yang telah memberikan dukungan moril dan materil selama proses penulisan karya ilmian ini.

Daftar Pustaka

D. Aprilianto, T. Fadila, and M. A. Muslim, “Sistem Pencegahan UDP DNS flood Dengan Filter Firewall Pada router Mikrotik,” Techno.Com, vol. 16, no. 2, pp. 114–119, 2017.

Peniarsih and N. Muhamadi “Sistem flooding data,” Jurnal sistem Informasi) Universitas Suryadarma, vol. 7, no. 1, 2014.

A. H. Hambali and S. Nurmiati, “Implementasi Intrusion Detection System (IDS) Pada Keamanan PC server TERHADAP serangan flooding data,” Sainstech: Jurnal Penelitian dan Pengkajian Sains dan Teknologi, vol. 28, no. 1, 2018.

Nofriadi, N. (2018). PERANCANGAN SISTEM PENCEGAHAN FLOODING PADA JARINGAN. *JURTEKSI (Jurnal Teknologi dan Sistem Informasi)*, *4*(2), 165-170.

Tumigolung, A. S. M., Lumenta, A. S. M., Rumangit, A. M. (2015). Perancangan Sistem Pencegahan *Flooding* Data Pada Jaringan Komputer. E-Journal Teknik Elektro dan Komputer. hal.8-22.

H. Saputra and N. Nofriadi, "Perancangan Sistem pencegahan flooding Pada Jaringan," JURTEKSI, vol. 4, no. 2, pp. 165–170, 2018.

B. Wijaya and A. Pratama, "Deteksi Penyusupan Pada server menggunakan metode intrusion detection system (IDS) berbasis snort," *Jurnal Sisfokom (Sistem Informasi dan Komputer)*, vol. 9, no. 1, pp. 97–101, 2020.

P. Panggabean, "Analisis network security snort METODE intrusion detection system untuk OPTIMASI Keamanan Jaringan Komputer," Jursima, vol. 6, no. 1, p. 1, 2018.
F. Wahyudi and L. T. Utomo, "Perancangan Security Network Intrusion Prevention System Pada PDTI Universitas Islam Raden Rahmat malang," Edumatic: Jurnal Pendidikan Informatika, vol. 5, no. 1, pp. 60–69, 2021.

A. Gupta and L. S. Sharma, "Mitigation of DOS and port scan attacks using Snort," International Journal of Computer Sciences and Engineering, vol. 7, no. 4, pp. 248–258, 2019

Y. P. Atmojo, 2018, "Bot Alert Snort dengan Telegram Bot API pada Intrusion Detection System : Studi Kasus IDS pada Server Web," Seminar Nasional Sistem Informasi dan Teknologi Informasi, 176-180, 2018.

N. Kunhare, R. Tiwari, and J. Dhar, "Network packet analysis in real time traffic and study of Snort ids during the variants of DOS attacks," *Hybrid Intelligent Systems*, pp. 362–375, 2020.

Parningotan, P. (2018). Analisis Network Security Snort Menggunakan Metode Intrusion Detection System (Ids) Untuk Optimasi Keamanan Jaringan Komputer. *JURSIMA Jurnal*, *6*(1).

I. Masud, K. Kusrini, and A. B. Prasetio, "Distributed Denial Of Service (DDOS) Attack Detection On Zigbee Protocol Using Naive Bayes Algoritm," *International Journal of Artificial Intelligence Research*, vol. 5, no. 2, Jun. 2021, doi: 10.29099/ijair.v5i2.214.