Analisis Efektivitas Rule Snort dalam Mendeteksi Serangan Jaringan

by Suci Sekar Sari

Submission date: 27-Aug-2024 08:52PM (UTC+0700)
Submission ID: 2439059528
File name: REPEATER_-_Vol._2_No._4_OKTOBER_2024_hal_1-15.docx (1.34M)
Word count: 3409
Character count: 21156

Repeater : Publikasi Teknik Informatika dan Jaringan
Volume. 2 No. 4 Oktober 2024
E-ISSN : 3046-7276, dan P-ISSN : 3046-7284, Hal. 01-15
DOI: https://doi.org/10.62951/repeater.v2i4.194
Available online at: https://journal.aptii.or.id/index.php/Repeater

Analisis Efektivitas *Rule Snort* dalam Mendeteksi Serangan Jaringan

Suci Sekar Sari [1*], Agus Tedyyana [2]
[1,2] Teknik Informatika, Keamanan Sistem Informasi, Politeknik Negeri Bengkalis, Indonesia
sucisuci2001@gmail.com [1*], agustedyyana@polbeng.ac.id [2]
Alamat: Jl. Bathin Alam, Sungai Alam. Bengkalis Riau - 28711
Korespondensi penulis: sucisuci2001@gmail.com

Abstract. *The effectiveness of snort rules in detecting network attacks is still a question that needs further research. Although snort rules have been developed and updated regularly, network attacks also continue to evolve and may have patterns that have not been detected by existing snort rules. The main objective is to develop and analyze the effectiveness of snort rules in detecting attack patterns on the network. Benefits Provides a better understanding of the effectiveness of snort rules in detecting network attacks. The result of this research is that the development of snort rules that have been carried out is able to filter attack patterns that were previously undetectable such as DDoS attacks and is able to provide telegram notifications.*

Kata kunci *: IDS, Snort, Telegram, and DdoS*

Abstrak. Keefektifan rule snort dalam mendeteksi serangan jaringan masih menjadi pertanyaan yang perlu diteliti lebih lanjut. Meskipun rule snort telah dikembangkan dan diperbarui secara teratur, serangan jaringan juga terus berkembang dan mungkin memiliki pola yang belum terdeteksi oleh rule snort yang ada. Tujuan utamanya yaitu mengembangkan dan menganalisis efektivitas rule snort dalam mendeteksi pola serangan pada jaringan. Manfaatnya Menyediakan pemahaman yang lebih baik tentang efektivitas rule snort dalam mendeteksi serangan jaringan. Hasil penelitian ini yaitu pengembangan rule snort yang telah dilakukan mampu memfilter pola serangan yang sebelumnya tidak terdeteksi seperti serangan DDoS dan mampu memberikan notifikasi telegram.

Kata kunci : IDS, *Snort,* Telegram, dan DDoS

1. LATAR BELAKANG

Keamanan jaringan komputer dan server menjadi poin utama yang harus di rawat dan dijaga, bagi seorang administrator jaringan sangat penting untuk bisa melakukan pencegahan dan identifikasi pengguna yang tidak berhak untuk mengakses jaringan komputer. Keamanan jaringan komputer bertujuan untuk menjaga agar data di dalamnya tetap aman, utuh, dan valid. Dengan menjaga keamanan jaringan, kita bisa melindungi informasi, data, dan memastikan bahwa infrastrukturnya berjalan dengan baik. Ini membantu mencegah risiko seperti penyusupan atau ancaman yang dapat merusak fungsi jaringan. Jika keamanan jaringan tidak dijaga, bisa muncul masalah seperti gangguan, pengintipan, perubahan, atau pemalsuan pada jaringan komputer [2].

Penting sekali untuk memperhatikan keamanan jaringan, karena banyak hal yang dapat mengganggu keamanan dan stabilitas koneksi komputer. Beberapa contoh masalah yang umum terjadi dalam keamanan jaringan termasuk gangguan sistem yang bisa disebabkan oleh kesalahan tak disengaja dari pengelola. Namun, tidak sedikit pula masalah yang timbul akibat upaya merusak, menyusup, atau menyalahgunakan data dan sistem oleh pihak ketiga [4]. Salah satu solusinya adalah dengan memanfaatkan sistem pendeteksi

Received: Juli 02, 2024; Revised: Juli 30, 2024; Accepted: Agustus 26, 2024; ; Online Available : Agustus 27 2024;

intrusi (Intrusion Detection System/IDS) [5]. Intrusion Detection System (IDS) adalah sebuah sistem yang dirancang untuk mendeteksi serangan dan ancaman yang terjadi pada jaringan komputer, baik itu terkait dengan jaringan lokal maupun internet. IDS mampu melakukan pengawasan terhadap aktivitas jaringan dengan tujuan untuk mengidentifikasi tindakan yang mencurigakan atau tidak sah yang dapat merusak keamanan sistem jaringan [6]. Aspek keamanan jaringan meliputi stabilitas, integritas, dan validasi data yang sangat penting. Program Intrusion Detection System (IDS) berbasis jaringan yang dapat mendeteksi upaya penyusupan pada sistem jaringan computer salah satunya yaitu snort.

Snort adalah sebuah perangkat lunak yang digunakan untuk mendeteksi penyusup dan dapat menganalisis lalu lintas jaringan secara real-time. Snort memiliki kemampuan untuk mendeteksi berbagai jenis serangan [7]. Kelebihan Snort adalah mendukung berbagai platform dan sistem operasi, termasuk Linux dan Windows. Snort juga termasuk perangkat lunak open source dengan dukungan komunitas yang luas di internet. Hal ini memungkinkan pengguna Snort untuk dengan mudah memperbarui aturan (rule) Snort dibandingkan dengan perangkat lunak IDS lainnya [8]. Metode Signatures bekerja dengan mencocokkan aturan-aturan dengan lalu lintas yang sedang dideteksi, jika ada kecocokan, itu menandakan adanya serangan yang terjadi. Sementara itu, metode Anomaly Detection menggunakan aturan-aturan yang mengidentifikasi lalu lintas yang tidak biasa atau anomali dengan lalu lintas yang sedang dideteksi [6].

Namun, keefektifan rule snort dalam mendeteksi serangan jaringan masih menjadi pertanyaan yang perlu diteliti lebih lanjut. Meskipun rule snort telah dikembangkan dan diperbarui secara teratur, serangan jaringan juga terus berkembang dan mungkin memiliki pola yang belum terdeteksi oleh rule snort yang ada. Keberhasilan pendeteksian dan efektivitasnya sangat tergantung pada kemampuannya untuk mengenali dan merespons serangan dengan cepat. Beberapa organisasi telah mengintegrasikan notifikasi melalui platform komunikasi instan seperti Telegram ke dalam sistem Snort mereka. Tujuannya adalah untuk meningkatkan kemampuan Snort dalam mendeteksi dan merespons serangan dengan lebih baik.

Dengan menggunakan notifikasi Telegram, respons terhadap serangan menjadi lebih efisien karena pesan notifikasi dapat segera diteruskan ke pihak yang berwenang untuk mengatasi serangan secara tepat waktu. Penggunaan Telegram sebagai saluran notifikasi dipilih karena popularitas nya sebagai aplikasi pesan instan yang cepat dan andal dengan dukungan lintas platform, sehingga memudahkan administrator dan tim keamanan untuk berkoordinasi dan merespons serangan dari perangkat apa pun. Dalam konteks ini, analisis

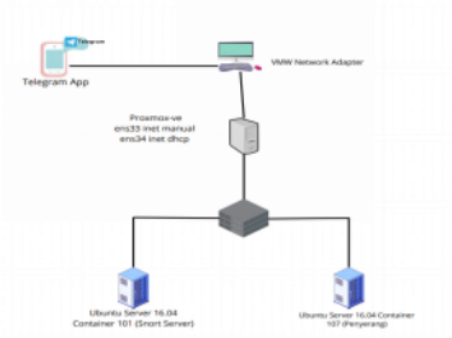
efektivitas rule snort mengacu pada penilaian sejauh mana rule yang diterapkan pada Snort mampu mendeteksi serangan yang dihadapi. Pada analisis tersebut, penting untuk memastikan bahwa rule yang diterapkan sesuai dengan jenis serangan yang paling mungkin terjadi dalam lingkungan jaringan tertentu. Selain itu, evaluasi rule snort juga harus mempertimbangkan tingkat keakuratan deteksi (minimal false positive dan false negative) serta kemampuan sistem untuk menangani ancaman dengan respons yang tepat[5].

Selain menyediakan aplikasi Telegram juga menyediakan API bagi pengguna untuk membuat bot yang dapat digunakan dan dikembangkan untuk sistem informasi. Telegram dapat digunakan untuk melakukan kegiatan pemantauan jaringan sebagai penerima pemberitahuan jika terjadi serangan dari luar. Salah satu penggunaan Telegram untuk kegiatan pemantauan ini dengan menerima pesan dari IDS yang langsung terkirim menuju ke akun pengguna sehingga dapat mengetahui serangan yang terjadi walaupun sedang tidak di depan komputer server. Bot API yang terus berkembang sehingga dapat membuat bot yang dinamis dan dapat merespon pesan dari administrator jaringan. Implementasi bot mulai banyak digunakan karena mempunyai keunggulan yaitu dapat menyediakan data ke pengguna yang tidak terbatas oleh waktu dan dapat dikembangkan oleh siapa saja. Pada penelitian ini menghasilkan sistem pemantauan sebuah server yang lebih fleksibel karena dapat dipantau dari mana saja, sehingga seorang administrator tidak perlu selalu di depan komputer server untuk mengawasi server. Penggunaan model pemantauan seperti ini dapat mengefisienkan dari aspek waktu dan tenaga bagi seorang administrator server. Pengiriman pemberitahuan yang cepat dari server menuju Telegram juga akan membantu administrator dalam melakukan tindakan ketika server sedang terjadi sesuatu.[9]

2. METODE PENELITIAN

Kolaborasi mekanisme IDS dan telegram dalam penelitian ini diusulkan melalui skema pengintegrasian Snort sebagai perangkat lunak deteksi serangan dan Telegram-API sebagai perangkat lunak untuk notifikasi serangan. Mekanisme kerja dari sistem yang diusulkan adalah sebagai berikut: Snort akan memantau aktivitas lalu lintas dalam jaringan dan merekamnya ke dalam sebuah log file. Ketika ditemukan aktivitas yang mencurigakan (berpotensi sebagai sebuah serangan), maka snort akan memberikan notifikasi kepada administrator melalui Telegram-API. Ketika trafik yang dipantau terbukti merupakan upaya serangan dari hacker, maka administrator dapat melakukan perubahan firewall rules pada server secara langsung dengan cara mengirimkan pesan melalui bot Telegram.

Keseluruhan sistem akan dijalankan secara simulasi dengan menggunakan network simulator VMWare.



Gambar 1. Keseluruhan sistem akan dijalankan secara simulasi dengan menggunakan network simulator VMWare

Mekanisme Deteksi Serangan Berbasis Snort

Snort merupakan jenis IDS singlethrading yang menggunakan rules dalam bentuk teks untuk menjalankan perintah, melakukan penanganan serangan dan memberikan aksi atau eksekusi terhadap setiap event yang terdeteksi. Snort tidak hanya diterapkan pada aliran trafik masuk namun juga setiap trafik yang keluar dari jaringan. Snort sendiri dapat bekerja dalam 3 mode, yaitu sebagai penyadap (sniffer), penyimpan data (packet logger), dan sebagai pendeteksi serangan (network intrusion detection)

Pada penelitian ini, Snort ditempatkan pada server utama (IP: 192.168.64.137) yang bertanggung jawab dalam merekam setiap aktivitas lalu lintas pada jaringan. Beberapa konfigurasi yang dilakukan pada server Snort antara lain konfigurasi dasar snort sebagai IDS, pembuatan rules dan konfigurasi file bpf sebagai database IP yang dianggap aman (trusted IP=bukan IP penyerang)

Mekanisme Pengendalian Serangan via Telegram

Telegram Bot-API merupakan sebuah perangkat lunak yang memungkinkan adanya interaksi antara bot dengan pengguna (administrator). Bot ini secara khusus dalam kepentingan keamanan jaringan memiliki kemampuan untuk mengirimkan perintah jarak jauh serta memberikan peringatan terhadap serangan.

Dalam proses penelitian ini terdapat Langkah kerja yang dilakukan. Seperti pada Gambar 2 Tahapan penelitian.

Gambar 2. Langkah Kerja

Persiapan dan Analisis Kebutuhan Penelitian

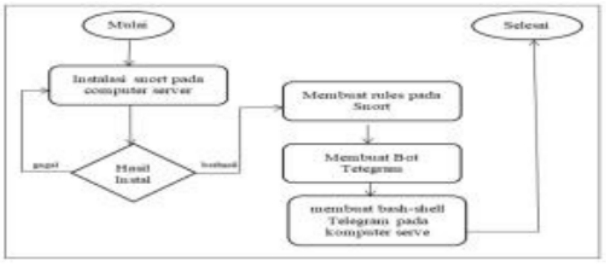
Persiapan adalah tahapan yang pertama dilakukan dimana untuk mempersiapkan kebutuhan yang diperlukan untuk melakukan penelitian. Perangkat lunak yang digunakan untuk melakukan penelitian ini dapat dilihat pada Tabel 3 di bawah ini:

Tabel 1. Langkah Kerja

No	Spesifikasi	Fungsi
1	Ubuntu Server	Sistem Operasi pada komputer server
2	Snort	Software Intrusion Detection System(IDS)
3	Kali linux	Sistem operasi pada computer penyerang
4	Telegram Api	Sebagai penerima pemeberitahunan serangan dari server
5	VM Ware Workstation ubuntu server	Sebagai virtual machine untuk menjalan kan sistem operasi ubuntu server

Konfigurasi Ubuntu Server, Snort, Telegram API

Setelah tahapan persiapan dan analisis kebutuhan penelitian selanjutnya adalah tahapan konfigurasi Ubuntu server, Snort dan Telegram. Dalam melakukan tahapan ini dilakukan sesuai diagram pada gambar 4 berikutini:



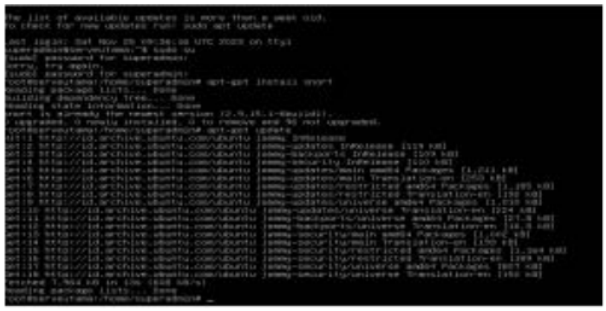
Gambar 3. Tahapan Konfigurasi Ubuntu Server

3. HASIL DAN PEMBAHASAN

Prosedur Eksperimen

Melihat pentingnya fungsi server sebagai penyedia informasi dan layanan dalam sebuah jaringan, maka penting untuk memastikan bahwa server selalu dalam keadaan aman dan dapat diakses dengan lancar. Server dituntut untuk memiliki tingkat realibilitas dan keamanan yang baik, karena banyak ancaman yang mungkin saja terjadi untuk

mengganggu kinerja server, seperti adanya virus, serangan brute force, distributed denial of service (DDoS) dan sebagainya. Melakukan pemantauan secara manual terhadap kinerja server tentunya menjadi hal yang tidak memungkinkan, sehingga dibutuhkan sebuah sistem yang dapat menggantikan posisi manusia untuk melakukan pemantauan secara kontinu. Penelitian ini mengusulkan mekanisme sistem pemantauan dan pengendalian terhadap serangan DDoS melalui metode *Intrusion Detection System* (IDS) Pengintegrasian Snort sebagai IDS dan Telegram-API diusulkan untuk meningkatkan keamanan pada lingkungan server.



Gambar 4. Server

```
$sudo apt-get install Snort
```

Skrip 1 berisi instruksi untuk menginstal Snort di server Ubuntu. Setelah berhasil menginstal Snort, langkah selanjutnya adalah membuat aturan (rules) di Snort agar dapat mendeteksi serangan yang mungkin terjadi di server. Untuk menulis aturan tersebut, kita perlu masuk ke direktori "snort" dan kemudian masuk ke dalam subdirektori "rules".

Gambar 5. Snort

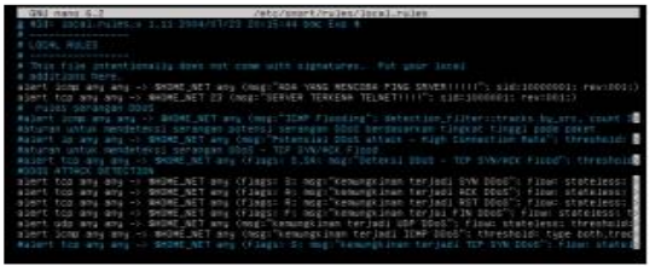
```
cd /etc/Snort/rules
```

Skrip 2 setelah berhasil masuk ke dalam direktori "rules", selanjutnya kita perlu masuk ke dalam file bernama "*local.rules*". File ini merupakan tempat di mana kita dapat

menuliskan aturan khusus atau modifikasi aturan yang sudah ada untuk deteksi serangan pada Snort.

```
nano local.rules
```

Skrip 3. Skrip untuk masuk ke files local rules, Skrip ini terdapat kode program yang berfungsi untuk masuk ke dalam file *local.rules* dengan menggunakan perintah *nano local.rules* untuk pertama kali masih belum terdapat rules jadi harus memasukan rules secara manual sesuai kebutuhan. Untuk menyimpan perubahan isi file tekan Ctrl + X lalu tekan Y dan terakhir Enter, Aturan baru akan disimpan di file *local.rules*. Jika tidak ingin menyimpan perubahan, cukup tekan Ctrl + C untuk keluar tanpa menyimpan.



Gambar 6. Isi file local rules

Gambar 6 menunjukkan apa yang ada di dalam file aturan lokal (local.rules). Penjelasan elemen-elemen aturan:

a. alert tcp any any -> $HOME_NET any: Aturan ini berlaku untuk semua paket TCP yang menuju ke $HOME_NET pada semua port sumber ke semua port tujuan.
b. flags: S, A, R, F: Aturan ini memeriksa paket TCP yang memiliki flag SYN, ACK, RST, FIN (permintaan sinkronisasi) yang diatur.
c. msg:"kemungkinan SYN DDoS": Ini adalah pesan yang akan dicetak jika aturan terpenuhi. Pesan ini menunjukkan bahwa aturan ini didesain untuk mendeteksi kemungkinan serangan SYN DDoS.
d. flow: stateless: Menandakan bahwa aturan ini beroperasi dalam mode stateless, yang berarti setiap paket diperlakukan secara terpisah dan tidak ada pelacakan keadaan koneksi.
e. threshold: type both, track by_dst, count 100000, seconds 10: Aturan ini memiliki ambang batas (threshold) untuk mendeteksi potensi serangan. Jika lebih dari 100,000 paket SYN terdeteksi menuju ke $HOME_NET dalam waktu 10 detik, aturan akan memicu.

f. sid:100002; rev:1;: Ini adalah ID tanda tangan unik (SID) untuk aturan ini, dan rev menunjukkan revisi aturan.

Setelah konfigurasi Snort selesai, langkah selanjutnya adalah mengonfigurasi aplikasi Telegram untuk menerima informasi serangan. Untuk membuat bot pada Telegram, buka aplikasi Telegram, cari "*BotFather*" di kolom pencarian, lalu mulai membuat bot dengan mengetik "/*start*" di kolom obrolan.

Gambar 7. Memulai Membuat Bot

Gambar 7 menunjukkan layar awal pada bot Telegram ketika pertama kali membuat bot. Selanjutnya, perlu mengetikkan "/*newbot*" di kolom obrolan untuk membuat bot baru di Telegram. Setelah itu, berikan nama "ubuntuserver" untuk bot yang akan di gunakan.

Gambar 8. Pemberian nama bot dan nama pengguna bot

Pada gambar 8, akan melihat tampilan pada bot Telegram saat memberikan nama pada bot. Selain itu, juga dapat melihat toke yang di berikan diberikan untuk mengakses API Telegram, yaitu 6889059461:AAF9G376NiHUw7Uf2kuyI__sxy69g4n3gn0. Token ini nantinya akan digunakan pada server Ubuntu untuk menghubungkan server dengan Telegram. Untuk menguji token bot telegram, masukkan kode program berikut pada halaman browser.

```
https://api.telegram.org/bot6889059461:AAF9G376NiHUw7Uf2kuyI__sxy69g4n3gn0/get
updates
```

Skrip untuk menguji Token Telegram Bot

Gambar 9. Hasil Pengujian Token Telegram Bot

Gambar 9 menunjukkan bahwa token bot Telegram yang sudah diuji berhasil digunakan. Selanjutnya, untuk mendapatkan chat ID bot, ketikkan pada kolom pencarian aplikasi Telegram, lalu klik tombol start. Reload link token Telegram bot yang masih tersedia di halaman browser.

Gambar 10. Untuk Mengetahui Chat Id bot Pada

Dari Gambar 10, terlihat bahwa ID obrolan (chat ID) bot Telegram sudah ada, yaitu 5272554956. Setelah berhasil membuat bot Telegram, langkah selanjutnya adalah menghubungkan Ubuntu Server dengan Telegram dengan membuat skrip bash shell di Ubuntu Server. Artinya, setelah bot Telegram dibuat, langkah berikutnya adalah membuat skrip atau perintah dalam bahasa Bash (*bash-shell*) di Ubuntu Server agar bisa berkomunikasi dengan bot Telegram tersebut.

Gambar 11. Isi perintah pada bash-shell Pada

Pada gambar 11 dapat dijelaskan isi pesan pemberitahuan yang akan dikirimkan ke aplikasi Telegram diambil dari isi file log-tele.txt yang terdapat pada direktori ubuntu, kemudian mengirimkan pesan melalui chat id dan token yang sudah ditentukan. Setelah

bash-shell berhasil dibuat langkah selanjutnya adalah pengujian serangan ke ubuntu server sekaligus menguji ubuntu server apakah berhasil dalam mengirimkan pesan ke aplikasi Telegram.

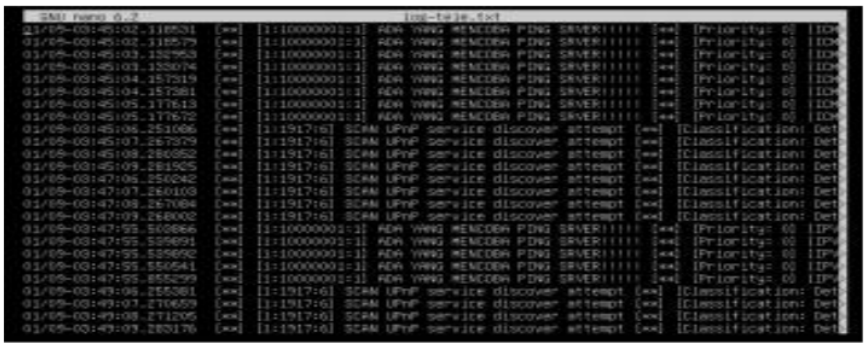
```
Snort -A console > /home/superadmin/log-tele.txt -c /etc/Snort/Snort.conf -l /var/log/Snort/
```

Skrip 6. Perintah untuk menjalankan Snort

Dalam "skrip 6," terlihat bahwa ini adalah kode program untuk menjalankan Snort pada server Ubuntu. Snort adalah sebuah sistem deteksi intrusi (Intrusion Detection System) yang berfungsi untuk mendeteksi serangan pada jaringan komputer.

Ketika ada serangan masuk, Snort akan membaca paket serangan yang masuk dan mencocokan dengan aturan (*rule*) yang telah ditentukan. Jika paket serangan cocok dengan aturan yang ada, Snort akan memberitahukan tentang serangan yang terjadi sesuai dengan aturan yang cocok tersebut. Selain memberikan peringatan atau pemberitahuan, Snort juga akan menyimpan log pemberitahuan ke dalam file /home/superadmin/log-tele.txt.

Artinya, Snort bertindak sebagai deteksi dini terhadap serangan yang masuk ke server, memberikan pemberitahuan sesuai dengan aturan yang dilanggar, dan mencatat informasi terkait ke dalam file log-tele.txt untuk analisis lebih lanjut atau tindakan lebih lanjut.



Gambar 12. Hasil Penyimpanan Pemberitahuan di dalam file /home/superadmin/log-tele.txt

Dalam percobaan ini, terdapat gambar ke-9 yang menunjukkan log dari notifikasi. Setelah notifikasi berhasil disimpan dalam file bernama log-tele.txt, langkah berikutnya pada program bash-shell Telegram akan mengambil isi dari log tersebut untuk kemudian dikirimkan melalui aplikasi Telegram. Proses dianggap selesai saat notifikasi berhasil terkirim ke aplikasi Telegram.

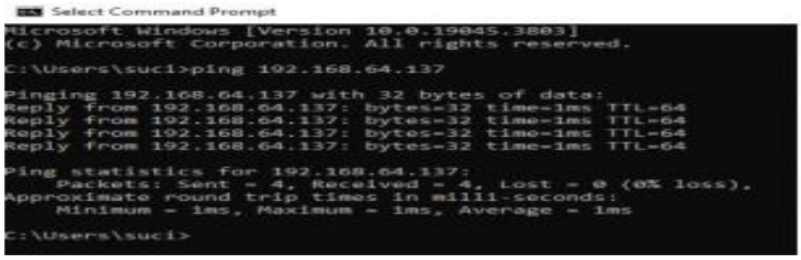
Dalam penelitian ini, digunakan aplikasi Command Prompt pada Windows 10 sebagai komputer penyerang. Penyerangan yang dilakukan mencakup mencoba

mengirimkan ping ke alamat IP komputer server dan mencoba mengakses komputer server menggunakan telnet.

```
Ping 192.168.64.137
```

Skrip 7. Perintah untuk mengirimkan Ping

Pada skrip 7, terdapat perintah untuk mengirimkan ping dari komputer penyerang. Alamat IP komputer penyerang dalam kasus ini adalah 192.168.64.137



Gambar 13. Hasil perintah Ping IP ICMP Server pada Command Promp

Gambar 14. Hasil log pemberitahuan untuk ping

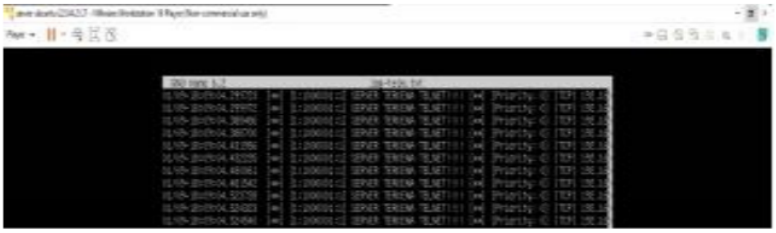
Pada Gambar 14, log menunjukkan adanya notifikasi bahwa ping tersebut telah diterima oleh komuter server.

```
telnet 192.168.64.137
```

Skrip 8. Perintah untuk menjalankan telnet pada command prompt

Pada skrip 8 perintah Telnet dari Command Prompt di Windows untuk mengakses komputer server pastikan bahwa Telnet sudah terpasang sebelumnya.

Pada Gambar 3.12 terlihat bahwa komputer penyerang berhasil mengakses komputer server melalui alamat IP komputer server.



Gambar 15. Hasil log pemberitahuan untuk telnet

Gambar 15 juga menunjukkan log mencatat adanya percobaan akses menggunakan Telnet dari komputer penyerang. Setelah berhasil melakukan serangan yang dikirimkan oleh komputer penyerang dan diterima oleh komputer server.

```
hping3 –S –lood –p 22 192.168.64.xxx
```

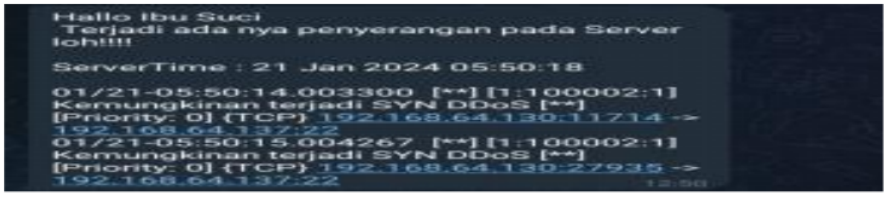
Skrip 9. DDOS Hping3 terhadap target

Dalam serangan DDoS menggunakan Hping3, kita menggunakan opsi -s sebagai "socket" yang penting. Opsi --flood digunakan untuk membuat serangan lebih kuat dengan mengirimkan banyak data ke target. -p digunakan untuk memilih port target, seperti port 22, dan kita menyasar alamat IP 192.168.64.xxx Serangan ini bertujuan membuat kekacauan dan memberatkan jaringan target dengan cara yang agresif.

Gambar 16. perintah hping3 pada kali

Gambar 17. Hasil log pemberitahuan serangan

Pada gambar di komputer yang sudah diterapkan snort dapat mendeteksi saat ada serangan DDOS Hping3 ke jaringan komputer dengan alert waktu, jenis serangan dan ip penyerang.



Pada Gambar 16 adalah tampilan pemberitahuan serangan yang muncul pada aplikasi Telegram.

4. Hasil Eksperimen

Hasil dari eksperimen dalam pengembangan *rules snort* yaitu mampu mendeteksi serangan yang sudah di kebangkan dalam *rule snort* salah satunya yaitu serangan DDoS dan mampu memberikan notifikasi ke dalam aplikasi telegram.

Tabel 2. *rule snort* salah satunya yaitu serangan DDoS dan mampu memberikan notifikasi ke dalam aplikasi telegram

No	Serangan	Pola Seranga	Keberhasilan	
			Berhasil	Tidak
1.	**DDoS**	Flags S	Berhasil	-
		Flags A	Berhasil	-
		Flags R	Berhasil	-
		Flags F	Berhasil	-
Total			4	

Hasil = 4 X 100% = 1 x 100 = 100%

$$\frac{Hasil}{Jumlah} x100\% = \frac{4}{4} x\ 100\% = 100\%$$

Dari hasil analisis eksperimen yang dilakukan bahwa rules snort yang telah di kembangkan efektif untuk mengenali pola-pola serangan DDoS.

5. KESIMPULAN

Berdasarkan hasil dari Eksperimen dan Analisa yang telah dilakukan terhadap Server Snort IDS dengan efektifan rule snort maka dapat diambil kesimpulan sebagai berikut:

a. hasil pengembangan rule snort yang telah dilakukan mampu memfilter pola serangan yang sebelumnya tidak terdeteksi seperti seranga DDoS
b. Penerapan bash shell Snort yang mampu mengirimkan Alert serangan DDoS dengan notifikasi melalui bot telegram kepada admin.

DAFTAR REFERENSI

J. Lirama *et al*., "IMPLEMENTASI *INTRUSION DETECTION SYSTEM* (IDS) UNTUK MENDETEKSI SERANGAN METASPLOIT EXPLOIT," no. April, pp. 41–50, 2023.

K. Politeknik and N. Bengkalis, "11 th Applied Business and Engineering Conference 11 th Applied Business and Engineering Conference," no. September, pp. 217–224, 2023.

B. Fachri and F. H. Harahap, "Simulasi Penggunaan *Intrusion Detection System* (IDS) Sebagai Keamanan Jaringan dan Komputer," vol. 4, no. April, pp. 413–420, 2020, doi: 10.30865/mib.v4i2.2037.

K. Saleh, "IMPLEMENTASI *INTRUSION DETECTION SYSTEM* (IDS) PADA SERVER WEB PT . XYZ MENGGUNAKAN SNORT IMPLEMENTASI *INTRUSION DETECTION SYSTEM* (IDS) PADA SERVER WEB PT . XYZ MENGGUNAKAN SNORT," no. April, pp. 1–5, 2020.

S. Kasus, L. Vi, J. Kampus, and I. S. T. Akprind, "Jurnal JARKOM Vol . 8 No . 1 Juni 2020 Jurnal JARKOM Vol . 8 No . 1 Juni 2020," vol. 8, no. 1, pp. 10–19, 2020.

T. Komputer, F. Vokasi, and U. B. Darma, "PENERAPAN SISTEM KEAMANAN *INTRUSION DETECTION SYSTEM* SNORT PADA JARINGAN DISKOMINFO KABUPATEN OKI".

B. Wijaya and A. Pratama, "Deteksi Penyusupan Pada Server Menggunakan Metode *Intrusion Detection System* (IDS) Berbasis Snort," vol. 09, pp. 97–101, 2020.

I. P. Gede, A. Sudiatmika, I. P. Yesha, A. Ariwanta, I. G. Ayu, and S. Melati, "Mengoptimalkan Keamanan Jaringan Komputer Menggunakan Snort dan Telegram Bot yang Terintegrasi dengan Mikrotik," vol. 3, no. 4, pp. 247–256, 2022, doi: 10.47065/josyc.v3i4.2037.

R. Artikel, N. Christianto, and W. Sulistyo, "Model Pemantauan Keamanan Jaringan Melalui Aplikasi Telegram Dengan Snort," vol. 7, pp. 702–714, 2021.

D. D. Mahendra and F. S. Mukti, "Sistem Deteksi dan Pengendalian Serangan Denial of Service pada Server Berbasis Snort dan Telegram-API," vol. 21, no. 3, pp. 511–522, 2022.

D. Untuk, M. Salah, S. Syarat, and U. Memperoleh, "SNORT DENGAN METODE PENETRATION TEST DI LABOR TEKNIK INFORMATIKA UNIVERSITAS ISLAM RIAU Skripsi UNIVERSITAS ISLAM RIAU," 2021.

D. R. Arrasy and A. Noertjahyana, "RESOURCES DARI TOOLS PENDETEKSI SERANGAN SNORT DAN SURICATA YANG DI PASANG DI WEB".

S. Adam and A. Suryadi, "BULLETIN OF COMPUTER SCIENCE RESEARCH Monitoring Notifikasi Status Services Pada Os Linux Menggunakan Bot Telegram," vol. 3, no. 1, pp. 103–108, 2022, doi: 10.47065/bulletincsr.v3i1.219.

"No Title," no. 2, 2022.

R. N. Dasmen, C. Ariyanto, M. H. Surya, and H. Ramadhan, "Penerapan Snort Sebagai Sistem Pendeteksi Serangan Keamanan Jaringan," vol. 7, pp. 8–12, 2022.

"PEMBANGUNAN SISTEM MONITORING NETWORK SECURITY MENGGUNAKAN *INTRUSION DETECTION SYSTEM* SNORT DENGAN LOG ANALISIS SPLUNK (Studi Kasus : PT . H-One Kogi Prima Auto Technologies Indonesia) TUGAS AKHIR ' Pembangunan Sistem Monitoring Network Security Menggunakan *Intrusion Detection System* Snort Dengan Log Analisis Splunk (Studi Kasus : PT . H-One Kogi Prima Auto Technologies Indonesia),'" 2022.

H. Yanto, "Intruder Detection Monitoring System in Computer Networks Using Snort Based Sms Alert (Sistem Monitoring Deteksi Penyusup Dalam Jaringan Komputer Menggunakan Snort Berbasis Sms Alert)," vol. 7, no. 2, pp. 159–170, 2020.

L. F. Nainggolan, N. F. Saragih, and F. G. N. Larosa, "Monitoring Keamanan Jaringan Pada Server Ubuntu Dari Serangan DDoS Menggunakan Snort IDS," vol. 2, no. 2, pp. 1–10, 2022.

Analisis Efektivitas Rule Snort dalam Mendeteksi Serangan Jaringan

ORIGINALITY REPORT

8%	7%	4%	0%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Yohanes Priyo Atmojo. "Analisa Performa Raspberry Pi sebagai Intrusion Detection System: Studi Kasus IDS Pada Server Web", Eksplora Informatika, 2018 Publication	1%
2	www.researchgate.net Internet Source	1%
3	"Trust, Privacy and Security in Digital Business", Springer Science and Business Media LLC, 2008 Publication	<1%
4	Submitted to Universitas Muslim Indonesia Student Paper	<1%
5	repository.uir.ac.id Internet Source	<1%
6	journal.unimar-amni.ac.id Internet Source	<1%
7	M. Al Ikhsan M. Al Ikhsan, Muhammad Idham. "PERBANDINGAN BIAYA TEBAL	<1%

	PERKERASAN JALAN PADA WILAYAH PRIORITAS (Studi Kasus Desa Kuala Penaso, Kecamatan Talang Muandau, Bengkalis, Riau)", Jurnal TeKLA, 2020 Publication	
8	portuguese.abacademies.org Internet Source	<1 %
9	www.bu.univ-rennes2.fr Internet Source	<1 %
10	Mansur Mansur, Kasmawi Kasmawi, Riau Datin Azura, Suci Sekar Sari. "WORKSHOP PEMANFAATAN TEKNOLOGI WEB UNTUK PENGGUNAAN SISTEM BUKU INDUK SISWA SEKOLAH DASAR BERBASIS ONLINE", Tanjak: Jurnal Pengabdian Kepada Masyarakat, 2021 Publication	<1 %
11	dwikidinaldi.blogspot.com Internet Source	<1 %
12	ejurnal.methodist.ac.id Internet Source	<1 %
13	imanagustrian.blogspot.co.id Internet Source	<1 %
14	journal.uniga.ac.id Internet Source	<1 %
15	repository.mercubuana.ac.id Internet Source	<1 %

16	repository.universitasbumigora.ac.id Internet Source	<1 %
17	ejournal.instiki.ac.id Internet Source	<1 %
18	id.123dok.com Internet Source	<1 %
19	journal.widyakarya.ac.id Internet Source	<1 %
20	jurnal.untan.ac.id Internet Source	<1 %
21	jutif.if.unsoed.ac.id Internet Source	<1 %
22	repositori.usu.ac.id Internet Source	<1 %
23	widuri.raharja.info Internet Source	<1 %
24	doku.pub Internet Source	<1 %

Exclude quotes	On	Exclude matches	Off
Exclude bibliography	On		

Analisis Efektivitas Rule Snort dalam Mendeteksi Serangan Jaringan

GRADEMARK REPORT

FINAL GRADE

/0

GENERAL COMMENTS

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

PAGE 7

PAGE 8

PAGE 9

PAGE 10

PAGE 11

PAGE 12

PAGE 13

PAGE 14

PAGE 15