Indonesian Journal of Computer Science
ISSN 2302-4364 (*print*) dan 2549-7286 (*online*)
Jln. Khatib Sulaiman Dalam, No. 1, Padang, Indonesia, Telp. (0751) 7056199, 7058325
Website: ijcs.stmikindonesia.ac.id | E-mail: ijcs@stmikindonesia.ac.id

Analisis Keamanan Jaringan Komputer Menggunakan Metode IDS dan IPS dengan Notifikasi *Telegram*

Taufiq Syaiful Huda, Subektiningsih[2]
taufiqsyaifulhuda@students.amikom.ac.id, subektiningsih@amikom.ac.id
[1,2] Universitas Amikom Yogyakarta

Informasi Artikel

Diterima : 16 Nov 2024
Direview : 18 Jan 2024
Disetujui : 15 Feb 2024

Kata Kunci

keamanan jaringan komputer, IDS, IPS, *Telegram*.

Abstrak

Penelitian ini bertujuan untuk menganalisis keamanan jaringan komputer dengan memanfaatkan metode *Intrusion Detection System* (IDS) dan *Intrusion Prevention System* (IPS) yang dilengkapi dengan notifikasi melalui platform *Telegram*. IDS dan IPS merupakan teknologi yang penting dalam menjaga integritas dan keamanan jaringan komputer dari ancaman serangan jaringan komputer. Penelitian ini mencoba mengintegrasikan kedua sistem ini untuk mendeteksi potensi intrusi dan secara aktif mencegah serangan sambil memberikan notifikasi real-time melalui *Telegram*, memungkinkan administrator untuk segera mengambil tindakan responsif. Metodologi yang digunakan mencakup implementasi dan konfigurasi IDS/IPS, serta pengujian terhadap skenario intrusi yang mungkin terjadi. Hasil penelitian ini diharapkan dapat memberikan wawasan yang lebih baik tentang tingkat keamanan jaringan komputer dan mengidentifikasi cara-cara yang lebih efektif dalam menghadapi ancaman serangan jaringan komputer.

Keywords

computer network security, IDS, IPS, Telegram.

Abstract

This research aims to analyze computer network security using the Intrusion Detection System (IDS) and Intrusion Prevention System (IPS) methods equipped with notifications via the Telegram platform. IDS and IPS are important technologies in maintaining the integrity and security of computer networks from the threat of computer network attacks. This research attempts to integrate these two systems to detect potential intrusions and actively prevent attacks while providing real-time notifications via Telegram, allowing administrators to take immediate responsive action. The methodology includes implementing and configuring IDS/IPS and testing possible intrusion scenarios. The results of this research will provide better insight into the level of computer network security and identify more effective ways of dealing with the threat of computer network attacks.

A. Pendahuluan

Pengamanan dalam jaringan komputer menjadi semakin penting seiring dengan kemajuan teknologi dan internet [1]. Pentingnya menjaga keamanan jaringan komputer sebagai komponen sistem utama adalah untuk menjamin data tetap valid dan utuh serta layanan selalu tersedia bagi pengguna [2]. Dalam perkembangan teknologi dan internet yang cepat, tersedia banyak alat yang dapat digunakan untuk melancarkan aktivitas yang dapat merugikan, seperti serangan *brute force* TCP, *port scanning*, dan *Internet Control Message protocol* (ICMP) [3]. Inilah faktor-faktor yang berpotensi mengancam keamanan sistem jaringan, dimana data dapat dengan mudah diambil dan bahkan rusak oleh pihak yang berusaha merusak, menyusup, atau menyalahgunakan data dan sistem [4].

Perlu adanya metode dan sistem yang dapat mengidentifikasi serangan pada jaringan komputer serta mencegah serangan pada jaringan komputer. Metode yang digunakan untuk mengidentifikasi serta mencegah serangan jaringan komputer, yaitu *Intrusion Detection System* (IDS) dan *Intrusion Prevention System* (IPS) [5]. yang bekerja bersama untuk melindungi jaringan dari serangan yang berpotensi merusak. *Intrusion Detection System* (IDS) merupakan sistem yang memiliki kemampuan untuk mengidentifikasi serangan dan ancaman yang timbul di dalam jaringan komputer, dan akan memberikan notifikasi awal kepada administrator jaringan saat aktivitas yang mencurigakan terdeteksi dalam jaringan komputer [6]. *Intrusion Prevention System* (IPS) merupakan sistem yang bertugas untuk memantau lalu lintas jaringan, mengidentifikasi aktivitas yang mencurigakan, dan mencegah serangan yang bisa mengganggu kinerja jaringan [7].

Metode *Intrusion Detection System* (IDS) dengan tools *Snort* Berfungsi untuk mendeteksi pihak yang dapat menganalisis paket data yang sedang berjalan melalui jaringan secara *real time*, mencatatnya dalam database, dan mengenali berbagai bentuk serangan yang berasal dari luar jaringan [8]. Beberapa keunggulan *Snort* jika dibandingkan dengan perangkat lunak IDS lainnya meliputi ukuran kode sumber yang kompak, kompatibilitas dengan berbagai sistem operasi, kemampuan mendeteksi serangan jaringan dengan cepat, kemudahan dalam mengonfigurasi, dan yang terutama, *Snort* ini dapat digunakan tanpa biaya [9]. Dengan menggunakan metode *Intrusion Prevention System* (IPS) dengan tools *Iptables* berfungsi untuk mengatasi serangan dengan memperkuat keamanan jaringan melalui tindakan penyaringan terhadap aliran data. Dengan *Iptables*, aturan-aturan dapat dibuat untuk mengatur aliran data, termasuk membatasi ukuran data yang diizinkan, jenis paket yang diterima, mengelola aliran data berdasarkan asal dan tujuan, *forwarding*, NAT, pengalihan, pengaturan port, dan pembentukan *firewall* [10], [11].

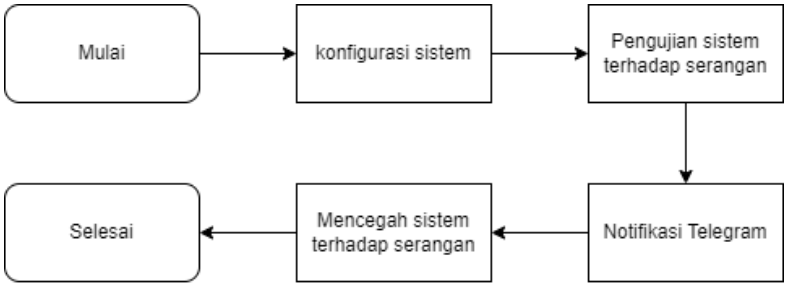
Penggunaan tools *Snort* sebagai pendeteksi serangan secara *realtime* dilakukan dengan memanfaatkan *Telegram* Bot untuk notifikasi. *Telegram* adalah aplikasi yang memungkinkan pengguna untuk mengirim pesan dengan cepat dan dengan keamanan yang terjamin. Selain itu, salah satu fitur yang kerap dimanfaatkan oleh pengembang adalah *Telegram* Bot [12]. Bot *Telegram* adalah akun khusus yang bisa didaftarkan ke Server *Telegram* tanpa perlu nomor telepon tambahan. Fungsinya sebagai perantara antara kode program dan server *Telegram* [13]. Notifikasi menjadikan Administrator Jaringan dapat segera mengetahui ancaman atau aktivitas mencurigakan dalam jaringan, sehingga tindakan dapat diambil dengan

cepat [14].*Telegram* dipilih karena keamanan, kecepatan pengirimannya, ketersediaan aplikasi di berbagai platform, serta dapat diakses dimana saja [15].

Penelitian ini akan menjelaskan tahapan yang diperlukan untuk menerapkan IDS dan IPS di lingkungan jaringan komputer, termasuk pemilihan perangkat keras dan perangkat lunak yang cocok. Selain itu, kami akan menggambarkan proses pengintegrasian notifikasi *Telegram* ke dalam sistem keamanan ini. Melalui analisis keamanan jaringan komputer, tujuan kami adalah meningkatkan pemahaman tentang kolaborasi teknologi IDS dan IPS untuk melindungi jaringan, serta bagaimana penggunaan notifikasi *Telegram* bisa meningkatkan tanggapan terhadap ancaman potensial. Harapannya, penelitian ini akan memberikan panduan berguna bagi profesional keamanan jaringan dan administrator sistem dalam usaha menjaga keamanan dan integritas jaringan komputer.

B. Metode Penelitian

Berdasarkan analisis sistem keamanan yang ingin dibuat, maka disusunlah tahapan-tahapan penelitian yang disajikan pada Gambar 1.



Gambar 1. Alur Penelitian

Tahapan analisisnya sebagai berikut :

1. Mulai
2. Mengkonfigurasi *Snort* dengan mode IDS untuk mendeteksi serangan jaringan.
3. Menguji sistem IDS dengan serangan pada port ICMP (ICMP *Flooding*) dan TCP (Telnet dan SSH).
4. Bot *Telegram* akan memberikan notifikasi serangan yang masuk kepada administrator untuk melakukan tindakan pencegahan.
5. Mencegah serangan yang masuk menggunakan *Iptables* dengan mode IPS
6. Selesai

Dalam menunjang penelitian ini, komponen penting yang digunakan terdiri dari perangkat keras dan perangkat lunak. Perangkat keras yang digunakan adalah laptop, sedangkan perangkat lunak yang digunakan adalah sistem operasi Linux. Berikut spesifikasi perangkat lunak yang digunakan pada penelitian ini dijelaskan pada Tabel 1.

Tabel 1. Spesifikasi Perangkat Lunak

Perangkat Lunak	Spesifikasi
Virtual Machine	Virtualbox
Sistem operasi (Target)	Ubuntu 20.04 TLS
Sistem operasi (Attacker)	Kali Linux
Intrusion Detection System (IDS)	Snort
Intrusion Prevention System (IPS)	Iptables

C. Hasil dan Pembahasan

1. Konfigurasi Sistem

Bagian ini menjelaskan tentang proses instalasi serta konfigurasi yang harus dilakukan terlebih dahulu untuk membentuk sistem monitoring sebelum dilakukan pengujian.

a. Konfigurasi *Snort* IDS

Melakukan instalasi paket *snort* dengan *script* # apt install snort yang terdapat pada Gambar 2.

```
root@taufiqsyaifulhuda-VirtualBox:/home/taufiqsyaifulhuda# apt install snort
Reading package lists... Done
Building dependency tree
Reading state information... Done
snort sudah versi terbaru (2.9.7.0-5build1).
Paket-paket berikut dipasang secara otomatis dan tak diperlukan lagi:
  gir1.2-goa-1.0 libfwupdplugin1 libllvm11 libxmlb1
Gunakan 'apt autoremove' untuk menghapus itu.
0 dimutakhirkan, 0 baru terinstal, 0 akan dihapus dan 31 tidak akan dimutakhirk
an.
```

Gambar 2. Instalasi *Snort*

b. Konfigurasi *Rule Snort* IDS

Melakukan penambahan *rules* serangan yang berfungsi untuk memunculkan notifikasi serangan yang akan dideteksi oleh *snort* dan bot-*Telegram* dengan *scrip*t seperti dibawah ini:

alert icmp any any -> $HOME_NET any (msg:”Ada Percobaan Ping”;sid:1000001;rev:1)
alert tcp any 23 -> $HOME_NET any (msg:”Ada Serangan dari port Telnet”;sid:1000002;rev:2)
alert tcp any 22 -> $HOME_NET any (msg:”Ada Serangan dari port Telnet”;sid:1000008;rev:3)

Pada Gambar 3 merupakan konfigurasi dari aturan serangan yang ditetapkan.

```
alert icmp any any -> $HOME_NET any (msg:"Ada Percobaan Ping";sid:1000001;rev:>
alert tcp any any -> $HOME_NET 23 (msg:"Ada Serangan dari port Telnet";sid:100>
alert tcp any any -> $HOME_NET 22 (msg:"Ada Serangan dari port SSH";sid:100000>
```

Gambar 3. Konfigurasi *Rules Snort*

c. Konfigurasi Bot

Melakukan konfigurasi bot-*Telegram* dengan cara memasukkan chat_id dan token yang akan menjadi *command* informasi ketika adanya serangan. *Script* konfigurasi
chat_id="-988918184"
token="6465298431:AAGid0QVvfQJJOPrj_y2Y0_MQC1I0BNA20"
Penerapan konfigurasi bot *Telegram* ditampilkan pada Gambar 4.

```
#Chat ID dan bot token Telegram
chat_id="-988918184"
token="6465298431:AAGid0QVvfQJJOPfrj_y2Y0_MQC1I0BNA20"
```

Gambar 4. Konfigurasi *Chat id* dan Token

2. Pengujian Sistem Terhadap Serangan

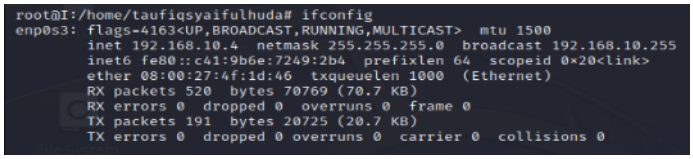
Pengujian terhadap serangan ini menggunakan jenis serangan ICMP *Flooding*, TCP Telnet, dan TCP SSH, dengan tujuan untuk secara komprehensif menguji dan

mengevaluasi keamanan sistem. Pengujian dengan serangan ICMP *Flooding* digunakan untuk menguji daya tahan sistem terhadap serangan. Pengujian dengan serangan TCP Telnet dan TCP SSH relevan dalam pengujian karena sering digunakan untuk mengakses sistem dari jarak jauh.

1. Pengujian Serangan ICMP *Flooding*

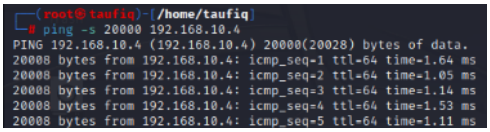
Sebelum melakukan pengujian serangan ICMP *Flooding*, yang pertama adalah melihat IP dari target dan kemudian baru melakukan pengujian penyerangan.

Pengecekan ip target dilakukan dengan perintah = *ifconfig* pada Gambar 5.



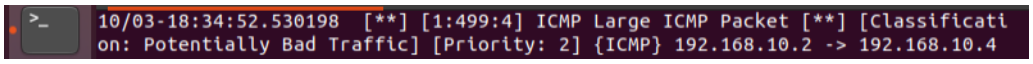
Gambar 5. Menampilkan IP 192.168.10.4 (Target)

Pengujian ICMP *flooding* dengan perintah = Ping -s 20000 192.168.10.4
Serangan ICMP dengan Kali linux pada protokol ping bertujuan untuk membanjiri lalu lintas jaringan komputer. Berikut adalah bentuk serangan ICMP *flooding* :



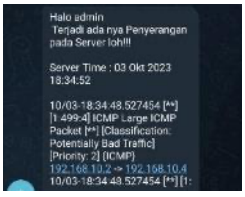
Gambar 6. Serangan ICMP *flooding*

S*nort* IDS akan mendeteksi serangan ICMP *flooding* yang berisikan tanggal, jam, SID, ip target, ip penyerangan, serta jenis serangan yang masuk melalui *snort* seperti pada sajian Gambar 7.



Gambar 7. Snort mendeteksi Serangan ICMP *flooding*

Serangan yang terdeteksi pada tanggal 03 Oktober 2023 jam 18.34.52 "ICMP Large ICMP Packet" memiliki SID 499 sumber serangan dari IP 192.168.10.2 dengan target IP 192.168.10.4. kemudian mengirimkan notifikasi serangan melalui aplikasi *Telegram* yang ditunjukkan pada Gambar 8.



Gambar 8. Notifikasi serangan ICMP *Flooding*

Tabel 2 menyajikan hasil pengujian serangan ICMP *Flooding* dengan berbagai packet.

Tabel 2. Tabel pengujian packet ICMP *Flooding*

Packet ICMP Flooding	Respon SNORT	Respon Telegram
20000	Berhasil	Berhasil
40000	Berhasil	Berhasil
50000	Berhasil	Berhasil
60000	Berhasil	Berhasil

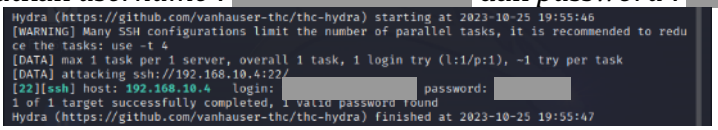
2. Pengujian Serangan TCP

Sebelum melakukan pengujian TCP SSH dan Telnet hal yang utama adalah melihat port manakah yang terbuka di protokol TCP kemudian mendapatkan informasi *username* dan *password* dengan target IP 192.168.10.4. *Port scanning* adalah untuk mendapatkan informasi mengenai port yang terbuka pada komputer dilakukan dengan perintah = nmap

Gambar 9. Menampilkan port yang terbuka yang diserang oleh attacker

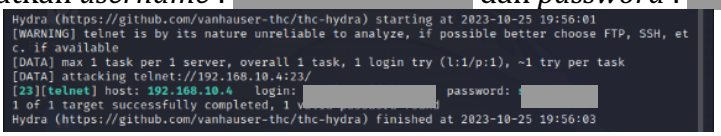
Mendapatkan informasi username dan password dengan target IP 192.168.10.4 dengan perintah = hydra

Mendapatkan *username* : [redacted] dan *password* : [redacted]



Gambar 10. *Username* dan *password* port ssh

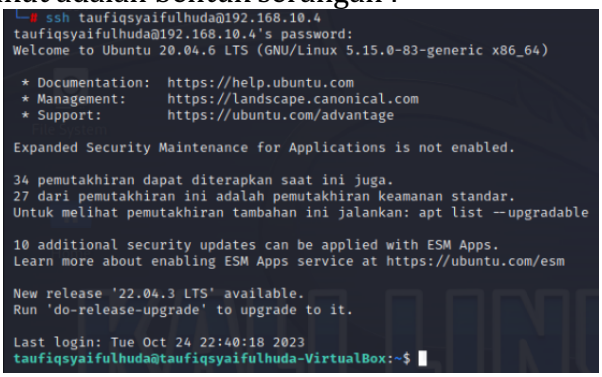
Mendapatkan *username* : [redacted] dan *password* : [redacted]



Gambar 11. *Username* dan *password* port Telnet

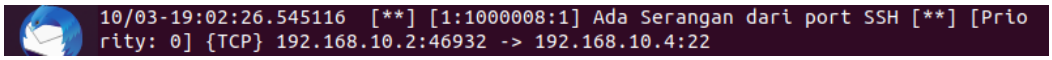
a. Pengujian TCP SSH

Pengujian TCP SSH dengan perintah = ssh @taufiqsyaifulhuda 192.168.10.4
Serangan TCP SSH dengan Kali linux pada protokol TCP bertujuan untuk masuk ke jaringan komputer. Berikut adalah bentuk serangan :



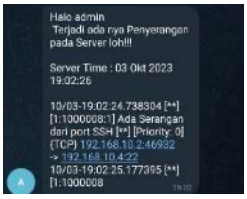
Gambar 12. Serangan TCP SSH

S*nort* IDS akan mendeteksi serangan TCP yang berisikan tanggal, jam, SID, ip target, ip penyerangan, serta jenis serangan yang masuk melalui *snort* seperti berikut:



Gambar 13. *Snort* mendeteksi Serangan TCP SSH

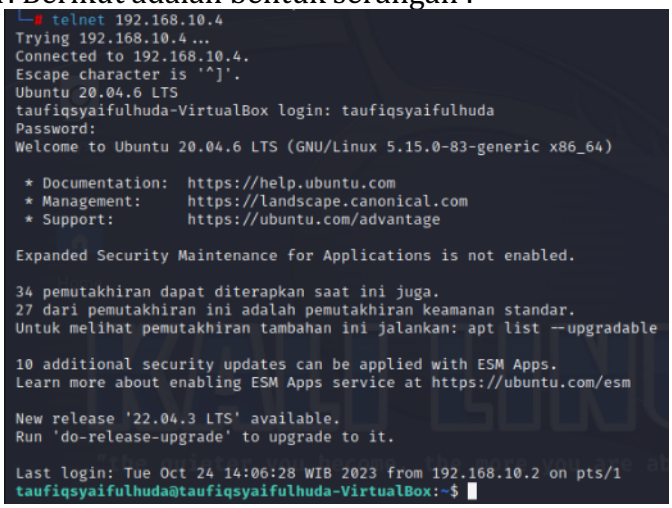
Serangan yang terdeteksi pada tanggal 03 Oktober 2023 jam 19.02.26 "Ada serangan dari port SSH" memiliki SID 1000008 sumber serangan dari IP 192.168.10.2 dengan target IP 192.168.10.4 dengan port 22. kemudian mengirimkan notifikasi serangan melalui aplikasi *Telegram*.



Gambar 14. Notifikasi serangan TCP SSH

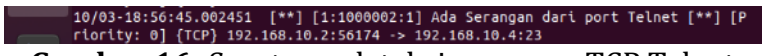
b. Pengujian TCP Telnet

Pengujian TCP TELNET dengan perintah = telnet 192.168.10.4
Serangan TCP TELNET dengan Kali linux pada protokol TCP bertujuan untuk masuk ke jaringan komputer. Berikut adalah bentuk serangan :



Gambar 15. serangan TCP Telnet

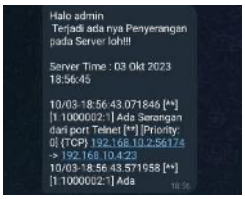
S*nort* IDS akan mendeteksi serangan TCP Telnet yang berisikan tanggal, jam, SID, ip target, ip penyerangan, serta jenis serangan yang masuk melalui *snort* seperti berikut:



Gambar 16. *Snort* mendeteksi serangan TCP Telnet

Serangan yang terdeteksi pada tanggal 03 Oktober 2023 jam 18.56.45 "Ada serangan dari port Telnet" memiliki SID 1000002 sumber serangan dari IP

192.168.10.2 dengan target IP 192.168.10.4 dengan port 23. kemudian mengirimkan notifikasi serangan melalui aplikasi *Telegram*.

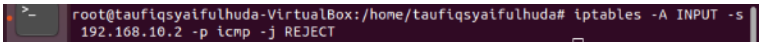


Gambar 17. Notifikasi serangan TCP Telnet

3. Mencegah Sistem Terhadap Serangan

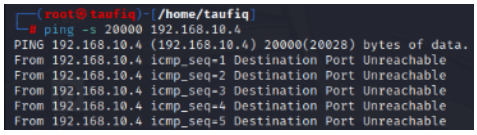
a. Mencegah Serangan ICMP *Flooding*

Serangan ICMP *Flooding* pada gambar 7 dan muncul notifikasi serangan pada gambar 8. Kemudian melakukan pencegahan serangan ICMP *Flooding* menggunakan *Iptables* dengan perintah : iptables -A INPUT -s 192.168.10.2 -p icmp -j REJECT



Gambar 18. Mencegah serangan ICMP *Flooding* menggunakan *Iptables*

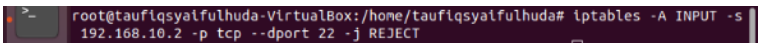
Lalu, melakukan penyerangan lagi dan hasilnya *Destination Port Unreachable*



Gambar 19. Hasil pencegahan serangan ICMP *Flooding* berhasil

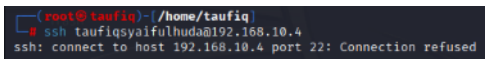
b. Mencegah Serangan TCP SSH

Telah terjadi serangan TCP SSH pada gambar 13 dan muncul notifikasi serangan pada gambar 14. Kemudian melakukan Pencegahan serangan TCP SSH menggunakan I*ptables* dengan perintah :iptables -A INPUT -s 192.168.10.2 -p tcp –dport 22 -j REJECT



Gambar 20. Mencegah serangan TCP SSH menggunakan *Iptables*

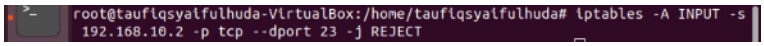
Lalu diserang lagi dan hasilnya *Connection refused*



Gambar 21. Hasil pencegahan serangan TCP SSH berhasil

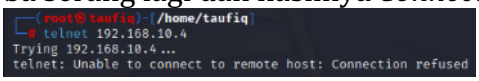
c. Mencegah Serangan TCP Telnet

Telah terjadi serangan TCP Telnet pada gambar 16 dan muncul notifikasi serangan pada gambar 17. Kemudian melakukan pencegahan serangan TCP Telnet menggunakan *Iptables* dengan perintah : iptables -A INPUT -s 192.168.10.2 -p tcp –dport 23 -j REJECT



Gambar 22. Mencegah serangan TCP Telnet menggunakan *Iptables*

Lalu kita coba serang lagi dan hasilnya *Connection refused*



Gambar 23. Hasil pencegahan serangan TCP Telnet berhasil

Tabel 3. Hasil Pengujian Sistem Terhadap Semua Jenis Serangan

No	Jenis Serangan	Respon IDS (Snort)	Respon Time Notifikasi *Telegram*	Respon IPS (IPtables)
1	Serangan ICMP Flooding	18.34.52	18.34.52	Destination Port Unreachable
2	Serangan TCP SSH	19.02.26	19.02.26	Connection refused
3	Serangan TCP Telnet	18.56.45	18.56.45	Connection refused

Berdasarkan ketiga jenis serangan tersebut IDS dan IPS mampu bekerja baik dalam mendeteksi dan mencegah serangan, dan respon *Telegram* memberikan notifikasi secara *real-time*. Diantara ketiga jenis serangan tersebut IDS dan IPS adalah solusi yang sangat efektif dalam menghadapi serangan ICMP *Flooding*, karena serangan ICMP *Flooding* hanya mengirimkan serangan dalam paket banyak dan menyerang melalui protokol bukan masuk ke port.

D. Simpulan

Berikut adalah kesimpulan dari penelitian ini berdasarkan hasil pengujian. Pertama, sistem Snort IDS terbukti mampu bekerja efektif sebagai pemberi peringatan serangan ICMP *Flooding*, TCP SSH, dan TCP Telnet. Selanjutnya, penerapan bot-tele.sh terbukti mampu merekam serangan ICMP *Flooding*, TCP SSH, dan TCP Telnet, Serta memberikan notifikasi secara *real-time* melalui bot *Telegram* kepada administrator. Terakhir, sistem Iptables IPS juga terbukti efektif sebagai pencegah serangan ICMP *Flooding*, TCP SSH, dan TCP Telnet. Dengan demikian, hasil pengujian menunjukkan bahwa kedua sistem tersebut mampu memberikan perlindungan yang solid terhadap serangan-serangan tersebut. Diantara ketiga jenis serangan tersebut IDS dan IPS adalah solusi yang sangat efektif dalam menghadapi serangan ICMP *Flooding*, karena serangan ICMP *Flooding* hanya mengirimkan serangan dalam paket banyak dan menyerang melalui protokol bukan masuk ke port. Penelitian selanjutnya dapat berfokus pada analisis serangan dari protokol UDP dan cara yang dapat diterapkan untuk mencegah serangan tersebut.

E. Referensi

[1] S. Dwiyatno, "Analisis Monitoring Sistem Jaringan Komputer Menggunakan Software Nmap," *PROSISKO J. Pengemb. Ris. dan Obs. Sist. Komput.*, vol. 7, no. 2, pp. 108–115, 2020, doi: 10.30656/prosisko.v7i2.2522.

[2] H. Alamsyah, R. -, and A. Al Akbar, "Analisa Keamanan Jaringan Menggunakan Network Intrusion Detection and Prevention System," *JOINTECS (Journal Inf. Technol. Comput. Sci.*, vol. 5, no. 1, p. 17, 2020, doi: 10.31328/jointecs.v5i1.1240.

[3] A. Putra Dwi and M. Alghozy Bey Ridho Thorriq, "Analisis dan Implementasi Keamanan Jaringan File Transfer Protocol (FTP) Menggunakan Intrusion Prevention System (IPS) pada Mikrotik," *Smart Comp Jurnalnya Orang Pint. Komput.*, vol. 11, no. 4, 2022, doi: 10.30591/smartcomp.v11i4.4263.

[4] R. Kurniawan and F. Prakoso, "Implementasi Metode IPS (Intrusion Prevention System) dan IDS (Intrusion Detection System) untuk Meningkatkan Keamanan Jaringan Program Studi Teknik Informatika 1)2) Sekolah Tinggi Manajemen Informatika dan Komputer Indo Daya Suvana," vol. 2, no. 02, 2020.

[5] G. Tambunan and M. IGN, "Implementasi Keamanan Ids / Ips Dengan Snort Dan IP Tables pada Server," *Semin. Nas. Mhs. Ilmu Komput. dan Apl. Jakarta-Indonesia, 28 Januari 2020 IMPLEMENTASI*, pp. 10–16, 2020.

[6] K. F. I. Ilham, E. I. Alwi, and F. Fattah, "Penerapan dan Analisis Network Security Snort Menggunakan Intrusion Detection System pada Serangan UDP Flood," *INFORMAL Informatics J.*, vol. 8, no. 1, p. 94, 2023, doi: 10.19184/isj.v8i1.34003.

[7] M. Hardjianto, "Sistem Monitoring Serangan Ssh Dengan Metode Intrusion Prevention System (IPS) Fail2ban Menggunakan Python Pada Sistem Operasi Linux," *J. TICOM Technol. Inf. Commun.*, vol. 11, no. 1, pp. 33–38, 2022.

[8] M. Aplikasi and I. D. S. Snort, "JITEKH Mengoptimalkan Pencegahan Serangan Brute Force pada Linux JITEKH," vol. 11, no. 2, pp. 57–61, 2023.

[9] I. Putu, A. E. Pratama, N. Kade, and M. Handayani, "Implementasi Ids Menggunakan Snort Pada Sistem Operasi Ubuntu," *J. Mantik Penusa*, vol. 3, no. 1, pp. 176–181, 2019, [Online]. Available: www.snort.org

[10] A. Z. Mardiansyah, Y. M. Abdussyakur, and A. H. Jatmika, "OPTIMASI PORT KNOCKING DAN HONEYPOT MENGGUNAKAN IPTABLES SEBAGAI KEAMANAN JARINGAN PADA SERVER (Port Knocking and Honeypot Optimization using IPTables for Serverrs Network Security)," vol. 3, no. 2, 2021, [Online]. Available: http://jtika.if.unram.ac.id/index.php/JTIKA/

[11] J. K. Barends, F. Dewanta, and N. B. A. Karna, "Perancangan dan Analisis Intrusion Prevention System Berbasis SNORT dan IPTABLES dengan Integrasi Honeypot pada Arsitektur Software Defined Network," *Multinetics*, vol. 7, no. 2, pp. 163–176, 2022, doi: 10.32722/multinetics.v7i2.4276.

[12] K. Nalakhudin, M. Imron, and M. A. Wiedanto Prasetyo, "Pemanfaatan Notifikasi *Telegram* Untuk Monitoring Perangkat CCTV Rumah Sakit Orthopaedi Purwokerto," *Technomedia J.*, vol. 6, no. 1, pp. 56–65, 2021, doi: 10.33050/tmj.v6i1.1564.

[13] M. Mahmud, Purnawansyah, and M. Hasnawi, "Implementasi Bot *Telegram* Untuk Monitoring Jaringan Dengan Pendekatan Security Policy Development Life Cycle Pada Kementerian Kelautan dan Perikanan Untia," *Bul. Sist. Inf. dan Teknol. Islam* , vol. 3, no. 2, pp. 127–133, 2022.

[14] P. Febriyanti and S. Rusmin, "Pemanfaatan Notifikasi *Telegram* Untuk Monitoring Jaringan," *J. SIMETRIS*, vol. 10, no. 2, pp. 725–732, 2019.

[15] M. Sari, T. H. Putri, Carly Dosen, and M. D. Batubara, "Implementasi Monitoring Jaringan Berbasis Web Terintegrasi Cacti Menggunakan Notifikasi *Telegram*," *Query J. Inf. Syst.*, vol. 4, no. 1, pp. 25–34, 2020.