DETEKSI SERANGAN JARINGAN KOMPUTER BERBASIS SNORT DENGAN INTEGRASI NOTIFIKASI REAL-TIME MELALUI TELEGRAM

Ardy Januantoro [1], Supangat [2]
[1,2] Universitas 17 Agustus 1945 Surabaya
ardyjanuantoro@untag-sby.ac.id

ABSTRAK

Serangan jaringan komputer yang semakin kompleks memerlukan solusi andal untuk melindungi sistem dari ancaman serius seperti kebocoran data dan kerusakan sistem. Penelitian ini menggunakan Snort sebagai *Intrusion Detection System* (IDS) untuk mendeteksi serangan *DDoS, SYN Flood, SQL Injection, Port Scanning*, dan *Brute Force*, yang diintegrasikan dengan notifikasi real-time melalui Telegram. Dalam pengujian penelitian ini, Snort dikonfigurasi khusus dan diuji dengan simulasi serangan oleh 10 komputer secara bersamaan pada jaringan target. Hasilnya, Snort berhasil mendeteksi semua jenis serangan dengan akurat, membuktikan keandalannya sebagai alat deteksi intrusi. Hasil pengujian menunjukkan rata – rata waktu deteksi keberhasilan sebesar 0,67 detik. Serangan *DDoS* menjadi serangan yang cepat untuk terdeteksi dengan waktu 0,4 detik sedangkan waktu terlama untuk serangan yang terdeteksi adalah serangan *brute force* dengan waktu tercepat 1,2 detik. Integrasi notifikasi *real-time* memungkinkan respons cepat terhadap ancaman siber, meningkatkan efisiensi pengelolaan keamanan jaringan. Studi ini menunjukkan bahwa kombinasi Snort dan notifikasi real-time adalah solusi efektif untuk meningkatkan perlindungan jaringan dan mempercepat mitigasi ancaman secara proaktif.

Keyword : *Cybersecurity, Network Security, Intrusion Detection System (IDS), Snort*

1. PENDAHULUAN

Serangan pada jaringan komputer terus meningkat seiring berkembangnya teknologi, memberikan dampak signifikan bagi berbagai sektor, termasuk pemerintahan, perusahaan, hingga individu. Dampak dari serangan ini meliputi kebocoran data sensitif, kerusakan infrastruktur sistem, dan terganggunya layanan operasional. Salah satu ancaman utama adalah serangan DDoS, SYN Flood, SQL Injection, Port Scanning, dan Brute Force, yang dirancang untuk mengganggu atau mengeksploitasi sistem. Kondisi ini menuntut pengembangan solusi keamanan yang lebih tangguh dan responsif.

Berbagai penelitian sebelumnya telah mengeksplorasi penggunaan sistem deteksi intrusi untuk mendeteksi serangan jaringan. Misalnya melakukan deteksi dengan menggunakan *firewall*[1], analisis implementasi deteksi intrusi [2] , implementasi deteksi intrusi berbasis tanda tangan dan anomali untuk pemantauan ancaman *real-time* [3]. Melakukan analisis hasil dari deteksi intrusi menggunakan model teratas termasuk KNeighbours, XGBoost, dan AdaBoost dengan akurasi tinggi dalam klasifikasi biner dan multi-kelas[4], sistem deteksi intrusi untuk memantau jaringan untuk aktivitas mencurigakan[5][6], implementasi sistem deteksi intrusi menggunakan metode deteksi tanda tangan atau anomali untuk mengidentifikasi potensi ancaman keamanan[7], serta memberikan peringatan untuk perilaku yang mencurigakan[8].

Snort adalah salah satu Intrusion Detection System (IDS) berbasis open-source yang populer, digunakan untuk mendeteksi dan menganalisis pola lalu lintas jaringan. Cara kerja Snort melibatkan inspeksi paket secara real-time menggunakan aturan (rules) yang telah didefinisikan untuk mendeteksi aktivitas mencurigakan. Ketika Snort menemukan pola yang sesuai dengan aturan tersebut, sistem akan menghasilkan peringatan atau log untuk ditindaklanjuti.

Penelitian ini memanfaatkan Snort yang dikonfigurasi secara khusus untuk mendeteksi lima jenis serangan: DDoS, SYN Flood, SQL Injection, Port Scanning, dan Brute Force. Selain itu, Snort diintegrasikan dengan notifikasi real-time melalui Telegram untuk memberikan pemberitahuan langsung ketika serangan terdeteksi. Proses pengujian melibatkan simulasi serangan oleh 10 komputer terhadap jaringan target untuk mengukur keefektifan deteksi.

Hasil yang diharapkan dari penelitian ini adalah membuktikan keandalan Snort dalam mendeteksi berbagai jenis serangan secara akurat serta meningkatkan kecepatan respons melalui notifikasi real-time. Dengan demikian, solusi ini dapat menjadi langkah strategis dalam memperkuat keamanan jaringan dan mitigasi ancaman siber secara proaktif.

2. TINJAUAN PUSTAKA

2.1. *Network Attack*

Serangan jaringan merupakan upaya oleh entitas yang tidak sah untuk mengganggu, mengambil alih, atau menimbulkan kerusakan pada infrastruktur jaringan komputer. Serangan ini dapat dikategorikan secara sistematis menjadi dua jenis utama: serangan pasif, yang dirancang untuk mengawasi lalu lintas data tanpa memodifikasinya, dan serangan aktif, yang memerlukan perubahan data atau gangguan layanan jaringan. Contoh ilustratif serangan aktif mencakup DDoS

(Distributed Denial of Service), di mana penyerang berusaha untuk membanjiri server target dengan lalu lintas penipuan, dan SYN Flood, yang memanfaatkan protokol TCP untuk menghabiskan sumber daya server[9], metode seperti SQL Injection dan Port Scanning semakin digunakan untuk mengidentifikasi kerentanan dalam arsitektur sistem[10]. Mengingat kerumitan ancaman jaringan yang meningkat, sangat penting untuk membangun mekanisme pertahanan yang kuat, termasuk Sistem Deteksi Intrusion (IDS), untuk mendeteksi dan mengurangi serangan dengan cepat.

2.2. *Intrussion Detection System* (IDS)

Sistem Deteksi Intrusion (IDS) merupakan perangkat canggih atau aplikasi perangkat lunak yang digunakan untuk pengawasan lalu lintas jaringan atau sistem, dengan tujuan mengidentifikasi perilaku anomali yang mungkin menandakan serangan cyber[11]. Paradigma operasional IDS menyerupai alat pemantauan keamanan, karena ia membedakan potensi ancaman yang didasarkan pada pola serangan yang mapan (berbasis tanda tangan) atau penyimpangan dari perilaku normatif (berbasis anomali). IDS dapat secara sistematis diklasifikasikan menjadi dua kategori utama: IDS Berbasis Jaringan (NIDS), yang meneliti lalu lintas jaringan, dan IDS Berbasis Host (HIDS), yang berkonsentrasi pada aktivitas yang terjadi di tingkat sistem. Selain itu, IDS tidak hanya membantu dalam identifikasi ancaman cyber tetapi juga menghasilkan log atau laporan yang dapat digunakan untuk keperluan investigasi forensik.

2.3. *Snort*

Snort mewakili Sistem Deteksi Intrusion (IDS) open-source terkemuka yang banyak digunakan dalam skala global. Awalnya dikonseptualisasikan oleh Martin Roesch pada tahun 1998, Snort memiliki kapasitas untuk mengidentifikasi berbagai ancaman, termasuk SQL Injection, serangan DDoS, Port Scanning, dan berbagai serangan berorientasi protokol lainnya. Menurut penelitian yang dilakukan oleh Ebrima Jaw dan Xueqing Wang[12], Snort menggunakan metodologi deteksi berbasis aturan di mana aturan yang diartikulasikan dengan cermat memfasilitasi pengenalan sistem terhadap pola serangan yang berbeda. Selain itu, Snort dilengkapi dengan kemampuan untuk pencatatan aktivitas jaringan dan pemberitahuan peringatan real-time, menjadikannya pilihan yang disukai dalam infrastruktur keamanan. Integrasi sinergis Snort dengan platform alternatif, seperti Telegram agar semakin meningkatkan kemanjuran dan efisiensi operasionalnya dalam domain mitigasi ancaman.

3. METODE PENELITIAN

Penelitian ini menggunakan pendekatan eksperimental untuk menguji keefektifan Snort sebagai Intrusion Detection System (IDS) dalam mendeteksi serangan jaringan dengan simulasi yang dilakukan dalam lingkungan terkendali. Metode penelitian ini mencakup tahapan-tahapan sebagai berikut

3.1. Konfigurasi Snort

Konfigurasi snort dimulai dengan instalasi snort pada *Operating System*. Setelah dilakukan instalasi langkah selanjutnya adalah melakukan konfigurasi *ip address* pada jaringan yang akan dipindai oleh snort apakah ada serangan atau tidak.

3.2. Konfigurasi Rule pada Snort

Pada Snort, rule (aturan) adalah inti dari cara kerja sistem untuk mendeteksi atau mencegah serangan. Rule mendefinisikan pola-pola (signatures) yang digunakan oleh Snort untuk memeriksa lalu lintas jaringan dan mendeteksi aktivitas mencurigakan atau serangan yang dikenal. Rule ini menentukan apa yang harus diperhatikan dan tindakan yang akan dilakukan jika pola tersebut ditemukan. Adapun rule yang akan dikonfigurasi yaitu *local.rule*. Perbedaan *local.rule* dan rule yang lain yaitu *local.rule* adalah tempat untuk membuat konfigurasi rule secara *custom.* Peneliti dapat melakukan konfigurasi secara bebas tetapi masih dalam kaidah penulisan rule pada snort. Jenis serangan yang akan di implementasikan pada rule adalah *DDoS, SYN Flood, SQL Injection, Port Scanning, dan Brute Force*. Adapun bentuk *rules* dapat dilihat pada gambar 1.

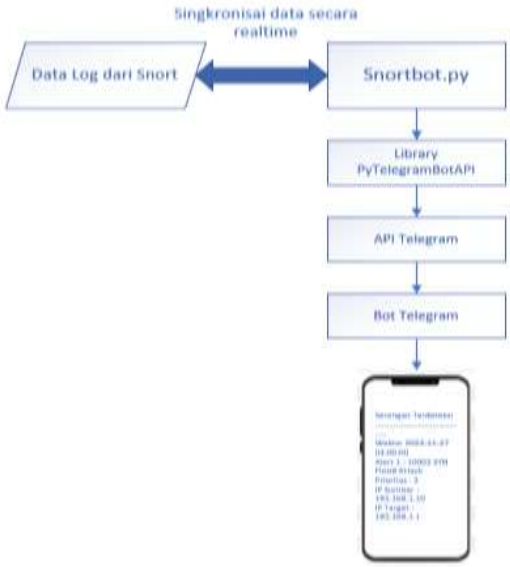
Gambar 1. Konfigurasi *local.rule*

Pada gambar 1 dapat dilihat bahwa terdapat kriteria pada aturan penulisan *rules snort*. Contoh pada *rule* untuk deteksi *SYN Flood*, pertama terdapat fungsi *alert* yang berarti adanya peringatan, *TCP* adalah protocol yang dilakukan pemindaian, *any any* adalah lalu lintas yang masuk/keluar dari sumber mana saja baik *source IP* maupun *port*, *msg* adalah pesan yang akan di kirimkan ke log jika terdeteksi, *count 100* adalah jumlah packet yang ter-*capture*, *second 1* adalah jumlah batas waktu packet ter-*capture, sid* adalah *identifier* unik untuk rule.

3.3. Notifikasi Telegram

Untuk konfigurasi notifikasi telegram, peneliti membuat program dengan menggunakan Bahasa pemrograman python. Langkah pertama peneliti menmbuat bot telegram terlebih dahulu kemudian

menyimpan nomor token dan ChatID telegram. Setelah mendapatkan nomor token dan ChatID, data tersebut akan dimasukkan kedalam program untuk dapat mengirimkan pesan lewat telegram kepada admin. Untuk berkomnukasi dengan bot telegram, peneliti menggunakan library PyTelegramBotAPI. Proses identifikasi alert pada snort kemudian dikirimkan menggunakan bot telegram dapat didlihat pada Gambar 2.



Gambar 2. Proses pengiriman notifikasi lewat telegram

Pada gambar 2 dapat dilihat terdapat 5 tahapan bagaimana program ini dapat mengirimkan notifikasi via telegram ke admin jaringan, Pertama mengatur konfigurasi agar program dapat mensingkronkan data dengan log snort secara realtime, hal ini berfungsi untuk memberikan informasi secara cepat ke admin jaringan jika ada tindakan penyerangan. Langkah kedua adalah melakukan koneksi dengan menggunakan library PyTelegramBotAPI ke API bot telegram yang sudah dibuat sebelumnya. Kemudan Bot akan mengiriman informasi yang diberikan oleh program ke admin jaringan via bot telegram.

3.4. Simulasi Serangan

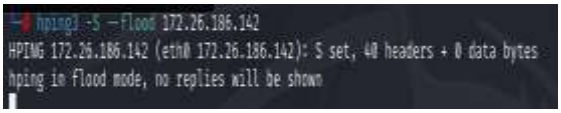
Dalam penelitian ini, simulasi serangan dilakukan untuk menguji kemampuan Snort sebagai Intrusion Detection System (IDS) dalam mendeteksi lima jenis serangan jaringan yang sering terjadi. Sepuluh komputer klien digunakan untuk melancarkan serangan secara bersamaan ke server target. Setiap komputer menjalankan jenis serangan yang berbeda, menciptakan skenario serangan simultan yang menyerupai kondisi nyata. Sistem operasi yang digunakan yaitu OS Kali Linux, ParrotOS, BlackBox linux

Setiap serangan dilakukan dalam interval waktu yang bervariasi, dengan durasi masing-masing serangan berkisar antara 5 hingga 15 detik. Jenis serangan yang disimulasikan meliputi

a. DDoS (Distributed Denial of Service): Komputer klien membanjiri server target dengan paket data dalam jumlah besar untuk mengganggu operasional layanan. Alat yang digunakan *slowloris*

Gambar 3. Proses serangan DdoS

b. SYN Flood: Serangan ini mengirimkan permintaan SYN palsu secara berulang, memanfaatkan celah dalam protokol TCP untuk menguras sumber daya server. Alat yang digunakan adalah *Hping3*



Gambar 4. Proses serangan SYN Flood

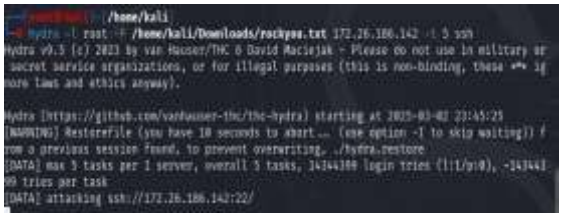
c. SQL Injection: Klien mengeksploitasi kerentanan pada input SQL dengan menyisipkan perintah berbahaya untuk mengakses data sensitif atau merusak basis data. Alat yang digunakan adalah *sqlmap*

Gambar 5. Proses serangan SQL Injection

d. Port Scanning: Komputer klien memindai port-port terbuka pada server untuk mengidentifikasi celah keamanan yang dapat dieksploitasi. Alat yang digunakan adalah *nmap*

Gambar 6. Proses serangan Port Scanning

e. Brute Force: Serangan ini mencoba menebak kredensial login melalui kombinasi password yang dihasilkan secara sistematis. Alat yang digunakan adalah *Hydra*



Gambar 7. Proses serangan Brute Force

Seluruh jenis serangan ini dirancang untuk menguji keandalan konfigurasi Snort dalam mendeteksi aktivitas mencurigakan. Snort diharapkan dapat mengenali pola serangan tersebut melalui aturan deteksi (rules) yang telah ditentukan sebelumnya. Data hasil deteksi kemudian dicatat dalam log sistem dan dikirimkan secara real-time ke Telegram sebagai notifikasi peringatan. Pendekatan ini bertujuan untuk memastikan bahwa sistem tidak hanya mampu mendeteksi serangan tetapi juga memberikan respons yang cepat dan efektif terhadap ancaman jaringan.

3.5. Pengumpulan Data

Dalam penelitian ini, data deteksi serangan dikumpulkan melalui log yang dihasilkan oleh Snort selama simulasi. Log tersebut mencatat berbagai parameter penting, seperti waktu terjadinya serangan, jenis serangan yang dilancarkan, dan status deteksi oleh sistem. Informasi ini memungkinkan evaluasi akurasi dan keandalan Snort dalam mengenali berbagai jenis ancaman jaringan. Berikut adalah data komputer beserta simulasi serangan yang dilakukan

Tabel 1. Data Komputer dan Jenis Serangan

No.	Komputer	Jenis Serangan
1	PC-1	DDoS
2	PC-2	SYN Flood
3	PC-3	SQL Injection
4	PC-4	Port Scanning
5	PC-5	Brute Force
6	PC-6	DDoS
7	PC-7	SYN Flood
8	PC-8	SQL Injection
9	PC-9	Port Scanning
10	PC-10	Brute Force

Selain mencatat deteksi serangan, penelitian ini juga mengukur kecepatan respons sistem dengan memanfaatkan fitur notifikasi real-time melalui Telegram. Setiap kali Snort mendeteksi serangan, notifikasi langsung dikirimkan ke saluran Telegram yang telah dikonfigurasi. Waktu pengiriman notifikasi dari momen deteksi dicatat untuk mengevaluasi efektivitas integrasi tersebut dalam memberikan peringatan dini.

Hasil pengumpulan data ini menjadi landasan untuk menganalisis performa Snort, baik dari sisi akurasi deteksi maupun kecepatan respons terhadap ancaman. Pengukuran waktu pengiriman notifikasi juga memberikan gambaran konkret mengenai kemampuan sistem untuk mendukung tindakan mitigasi ancaman secara cepat dan proaktif.

3.6. Analisis Keberhasilan

Analisis keberhasilan menggunakan penghitungan akurasi deteksi serangan oleh Snort untuk mengevaluasi keefektifannya sebagai Intrusion Detection System (IDS). Akurasi dihitung berdasarkan jumlah serangan yang berhasil terdeteksi dibandingkan dengan total serangan yang dilancarkan selama simulasi. Setiap jenis serangan, seperti DDoS, SYN Flood, SQL Injection, Port Scanning, dan Brute Force, dianalisis untuk memastikan kemampuan Snort dalam mengenali pola ancaman yang berbeda.

Selain itu, rata-rata waktu deteksi dan pengiriman notifikasi real-time melalui Telegram diukur untuk menilai kecepatan respons sistem. Waktu deteksi dihitung dari momen serangan dimulai hingga Snort mencatat log deteksi, sedangkan waktu pengiriman notifikasi dihitung dari momen log dibuat hingga notifikasi diterima di Telegram. Data ini memberikan wawasan mengenai efisiensi sistem dalam memberikan peringatan dini terhadap serangan.

Hasil penelitian ini kemudian dibandingkan dengan literatur terkait untuk validasi. Studi sebelumnya yang membahas efektivitas IDS seperti Snort digunakan sebagai pembanding untuk menilai apakah hasil penelitian ini sejalan atau lebih unggul dari temuan sebelumnya. Pendekatan ini memastikan bahwa hasil yang diperoleh tidak hanya mencerminkan performa sistem dalam lingkungan simulasi, tetapi juga relevan dengan aplikasi nyata di dunia keamanan jaringan.

4. HASIL DAN PEMBAHASAN

4.1. Notifikasi Telegram

Gambar 8. Telegram Bot

Pada gambar 8 dapat dilihat hasil dari notifikasi telegram yang dikirim oleh program bot telegram menggunaakan Bahasa pemrograman python, terdapat beberapa informasi yang berhasil terdeteksi oleh snort yaitu waktu serangan, jenis seranganm klasifikasi serangan, prioritas, ip penyerang dan ip target. Dengan hasil tersebut maka data berhasil terkirim secara realtime

4.2. Hasil simulasi serangan

Tabel 2. Hasil Simulasi Serangan

No.	Komputer	Jenis Serangan	Waktu serangan (detik)	Status Deteksi	Waktu deteksi
1	PC-1	DDos	10	Berhasil	0,4
2	PC-2	SYN Flood	12	Berhasil	0,7
3	PC-3	SQL Injection	8	Berhasil	0,8
4	PC-4	Port Scanning	6	Berhasil	0,6
5	PC-5	Brute Force	15	Berhasil	1,2
6	PC-6	DDos	11	Berhasil	0,4
7	PC-7	SYN Flood	9	Berhasil	0,5
8	PC-8	SQL Injection	7	Berhasil	0,8
9	PC-9	Port Scanning	5	Berhasil	0,6
10	PC-10	Brute Force	13	Berhasil	1,1

4.3. Analisis Keberhasilan

Hasil penelitian menunjukkan bahwa Snort berhasil mendeteksi seluruh jenis serangan yang disimulasikan dengan tingkat akurasi 100%. Keberhasilan ini mencerminkan kemampuan Snort dalam mengenali pola ancaman yang beragam, termasuk DDoS, SYN Flood, SQL Injection, Port Scanning, dan Brute Force, tanpa ada satu pun serangan yang terlewatkan.

Dari segi waktu deteksi, Snort menunjukkan performa yang sangat responsif dengan rata-rata waktu deteksi sebesar 0,68 detik sejak serangan dimulai. Hal ini menandakan bahwa Snort mampu secara cepat mengidentifikasi aktivitas mencurigakan dalam jaringan, memberikan waktu yang cukup untuk mengambil tindakan mitigasi.

Selain itu, integrasi sistem dengan Telegram memungkinkan pengiriman notifikasi real-time dengan kecepatan tinggi. Notifikasi dikirimkan dalam waktu kurang dari 1 detik setelah serangan terdeteksi oleh Snort. Kecepatan ini memastikan bahwa operator jaringan dapat segera menerima peringatan dan merespons ancaman dengan cepat, sehingga potensi kerusakan dapat diminimalkan. Hasil ini menegaskan bahwa kombinasi Snort dan notifikasi real-time adalah solusi efektif dalam meningkatkan keamanan jaringan.

5. KESIMPULAN DAN SARAN

Penelitian ini bertujuan untuk menunjukkan bahwa Snort mampu mendeteksi berbagai jenis serangan jaringan dengan tingkat akurasi yang tinggi. Lima jenis serangan utama yang disimulasikan, termasuk DDoS, SYN Flood, SQL Injection, Port Scanning, dan Brute Force, diidentifikasi oleh Snort dengan efisiensi yang menjanjikan. Hasil pengujian menunjukkan rata – rata waktu deteksi keberhasilan sebesar **0,67 detik**. Serangan *DDoS* menjadi serangan yang cepat untuk terdeteksi dengan waktu 0,4 detik sedangkan waktu terlama untuk serangan yang terdeteksi adalah serangan *brute force* dengan waktu tercepat 1,2 detik. Selain itu, integrasi dengan sistem notifikasi real-time melalui Telegram diharapkan dapat meningkatkan responsivitas dalam menghadapi ancaman siber, memungkinkan operator jaringan untuk segera mengambil langkah mitigasi setelah serangan terdeteksi. Dengan memanfaatkan kombinasi ini, sistem keamanan yang lebih tangguh dan cepat dapat diwujudkan.

Ke depannya, penelitian akan diperluas untuk menguji kemampuan Snort dalam mendeteksi ancaman yang lebih kompleks, seperti serangan berbasis kecerdasan buatan dan malware polymorphic, serta mengevaluasi integrasinya dengan teknologi lain, seperti machine learning untuk meningkatkan akurasi deteksi dan prediksi ancaman secara proaktif..

DAFTAR PUSTAKA

[1] Q. Zhang, Z. Luo, and J. Zhang, “Analysis of the Application of Firewall and Intrusion Detection Technology in Network Security in the Era of Big Data,” in *Proceedings - 2023 2nd International Joint Conference on Information and Communication Engineering, JCICE 2023*, 2023. doi: 10.1109/JCICE59059.2023.00042.

[2] Aishwarya Londhe, Sahil Gawathe, Prathamesh Pandey, Gajanan Date, and Sameer Meshram, “Intrusion Detection System,” *International Journal of Advanced Research in Science, Communication and Technology*, pp. 31–35, Apr. 2024, doi: 10.48175/IJARSCT-17606.

[3] , Mr. Sahil Bhelkar, “Network Intrusion Detection System,” *INTERANTIONAL JOURNAL OF SCIENTIFIC RESEARCH IN ENGINEERING AND MANAGEMENT*, vol. 08, no. 04, pp. 1–5, Apr. 2024, doi: 10.55041/IJSREM31278.

[4] A. Singh, J. Prakash, G. Kumar, P. K. Jain, and L. S. Ambati, “Intrusion Detection System,”

*Journal of a*pp. 1–25, Feb. 2024, doi: 10.4018/JDM.338276.

[5] S. Kumar and M. Kumar, "INTRUSION DETECTION SYSTEMS – A LEGACY FROM SECURITY PERSPECTIVE IN ALL SORTS OF NETWORK COMPUTING," in *Futuristic Trends in Information Technology Volume 2 Book 20*, Iterative International Publishers, Selfypage Developers Pvt Ltd, 2023, pp. 41–57. doi: 10.58532/V2BS20P1CH2.

[6] S. Devede and S. Soni, "A review intrusion protection system on the internet/network," *International Journal of Computing and Artificial Intelligence*, vol. 5, no. 1, pp. 96–101, Jan. 2024, doi: 10.33545/27076571.2024.v5.i1b.87.

[7] Safana Hyder Abbas, Wedad Abdul Khuder Naser, and Amal Abbas Kadhim, "Subject review: Intrusion Detection System (IDS) and Intrusion Prevention System (IPS)," *Global Journal of Engineering and Technology Advances*, vol. 14, no. 2, pp. 155–158, Feb. 2023, doi: 10.30574/gjeta.2023.14.2.0031.

[8] S. Mirlekar and K. P. Kanojia, "Role of Intrusion Detection System in Network Security and Types of Cyber Attacks-A Review," *International Journal of Innovations in Engineering and Science*, vol. 7, no. 9, pp. 28–32, Aug. 2022, doi: 10.46335/IJIES.2022.7.9.6.

[9] J. Liu, "Enhancing Network Security Through Router-Based Firewalls: An Investigation into Design, Effectiveness, and Human Factors," 2024.

[10] A. D. Tudosi, D. G. Balan, and A. D. Potorac, "Secure network architecture based on distributed firewalls," in *2022 16th International Conference on Development and Application Systems, DAS 2022 - Proceedings*, 2022. doi: 10.1109/DAS54948.2022.9786092.

[11] O. Coates, "Intrusion Detection Systems: A Survey and Taxonomy," Jun. 15, 2022. doi: 10.20944/preprints202206.0218.v1.

[12] E. Jaw and X. Wang, "A novel hybrid-based approach of snort automatic rule generator and security event correlation (SARG-SEC)," *PeerJ Comput Sci*, vol. 8, p. e900, Mar. 2022, doi: 10.7717/peerj-cs.900