PROKASDADIK: Prosiding Kecerdasan Artifisial, Sains Data, dan Pendidikan Masa Depan
Vol. 3, No. 1, (2025) E-ISSN: 3063-5845

Analisis Efektivitas LLM dalam Deteksi Serangan DDoS pada Jaringan Komputer

Mursalat Asyidiq, Azhar Fathoni
Program Studi Teknik Informatika S-2, Universitas Pamulang
e-mail: mursalat@raharja.info, fathonimlg@unpam.ac.id

Abstrak--Serangan *Distributed Denial of Service (DDoS)* adalah salah satu ancaman siber yang paling merusak, di mana pelaku serangan membanjiri jaringan atau sistem dengan lalu lintas internet yang berlebihan untuk membuat layanan menjadi tidak tersedia. Serangan ini menggunakan *botnet* yang terdiri dari perangkat yang terinfeksi *malware* dan dapat merusak reputasi perusahaan serta menyebabkan kerugian finansial yang signifikan. Mengingat kompleksitas serangan *DDoS*, perusahaan perlu mengenali pola-pola serangan dan menggunakan alat deteksi yang tepat. Penelitian ini mengusulkan pemanfaatan model GPT-4o-mini dari *OpenAI* untuk mengidentifikasi serangan *DDoS* melalui analisis log yang dihasilkan oleh *Security Information and Event Management (SIEM)*. Dengan menggunakan *dataset* yang terdiri dari 300 data *training* dan 1125 data testing, model dilatih dengan berbagai fitur jaringan dan berhasil mencapai tingkat akurasi 0.99, presisi 0.98, *recall* 0.98, dan F1-Score 0.98. Hasil ini menunjukkan bahwa model GPT-4o-mini memiliki kinerja yang sangat baik dalam mendeteksi serangan *DDoS* dengan mengurangi *false positive* dan *false negative.* Meskipun demikian, penelitian ini memiliki keterbatasan, terutama terkait dengan ukuran *dataset* dan pemilihan model. Untuk penelitian selanjutnya, disarankan untuk memperluas ukuran *dataset* dan mengeksplorasi perbandingan dengan model lain seperti LSTM, CNN, atau *Transformer-based* untuk tugas deteksi serangan siber. Penelitian ini juga membuka kemungkinan untuk memperluas aplikasi deteksi serangan siber pada jenis serangan selain *DDoS*..

Kata kunci--Serangan *DDoS*, model *GPT-4o-mini, machine learning, deep learning, cybersecurity*.

I. PENDAHULUAN

Serangan *Distributed Denial of Service (DDoS)* adalah jenis serangan siber yang dilakukan dengan membanjiri jaringan, sistem, atau server dengan lalu lintas internet yang berlebihan. Tujuannya adalah untuk membuat layanan atau jaringan menjadi tidak tersedia bagi pengguna yang seharusnya berhak mengaksesnya. Dalam banyak kasus, serangan ini dilakukan dengan menggunakan *botnet*, yakni sekelompok perangkat yang terinfeksi *malware* dan dikendalikan oleh pelaku serangan. Ancaman *DDoS* tidak hanya berdampak pada *downtime* atau gangguan layanan, tetapi juga dapat merusak reputasi perusahaan dan merugikan pengguna. Serangan ini telah menghantui dunia daring dengan dampak yang merugikan, mengganggu layanan online yang vital dan menyebabkan kerugian finansial yang signifikan. Dalam konteks teknologi yang semakin canggih, para penyerang menggunakan daya komputasi yang terdistribusi untuk melancarkan serangan yang melumpuhkan sistem target, baik itu situs web, jaringan perusahaan, atau infrastruktur penting lainnya. Terdapat beberapa jenis serangan *DDoS* di antaranya serangan volumetrik, serangan lapisan aplikasi yang menargetkan kerentanan pada aplikasi web atau server dan serangan protokol yang mengeksploitasi kerentanan pada protokol jaringan. Serangan *DDoS* ini sangat berdampak serius bagi perusahaan karna akan berdampak pada ha-hal krusial perusahaan seperti keuangan, reputasi menjadi rendah, penurunan produktivitas dan ketidak nyamanan bagi pengguna layanan.

Untuk melindungi diri dari ancaman ini, perusahaan perlu memahami dan mengenali pola serangan *DDoS*. Mengingat serangan ini biasanya melibatkan peningkatan lalu lintas data yang drastis dan tiba-tiba, *monitoring* jaringan yang tepat dan alat deteksi serangan dapat menjadi bagian penting dari strategi pertahanan. Namun, deteksi saja tidak cukup – perusahaan juga harus memiliki rencana tindakan yang siap dilaksanakan saat serangan terjadi. Secara keseluruhan, serangan *DDoS* merupakan ancaman serius yang memerlukan perhatian dan tindakan proaktif. Pemahaman tentang bagaimana serangan ini berfungsi dan dampak yang dapat ditimbulkannya adalah langkah pertama menuju perlindungan yang efektif.

Large Language Models (LLM), seperti *GPT-4*, telah menunjukkan potensi besar dalam berbagai aplikasi kecerdasan buatan, termasuk dalam mendeteksi dan menganalisis ancaman siber. *LLM* adalah model pembelajaran mesin yang dilatih pada data teks dalam jumlah besar dan memiliki kemampuan memahami, menganalisis, serta menghasilkan teks yang menyerupai bahasa manusia. Dalam konteks deteksi serangan *DDoS*, *LLM* dapat digunakan untuk mengenali pola lalu lintas jaringan yang mencurigakan dengan menganalisis log jaringan secara *real-time* atau hampir *real-time*.

LLM dapat membantu dalam identifikasi anomali dengan mengklasifikasikan pola-pola data yang tidak sesuai dengan lalu lintas normal. Hal ini dilakukan melalui pembelajaran mendalam pada *dataset* lalu lintas jaringan yang telah dianotasi sebelumnya

PROKASDADIK: Prosiding Kecerdasan Artifisial, Sains Data, dan Pendidikan Masa Depan
Vol. 3, No. 1, (2025) E-ISSN: 3063-5845

untuk membedakan antara lalu lintas normal dan serangan. Selain itu, *LLM* dapat digunakan untuk menganalisis deskripsi serangan dari laporan kejadian atau mendeteksi pola yang belum diketahui sebelumnya berdasarkan pemahaman semantik dari data jaringan. Dengan kemampuan adaptasi dan pembelajaran yang terus berkembang, *LLM* dapat menjadi alat yang fleksibel dalam menghadapi serangan yang semakin kompleks.

Penelitian ini berfokus pada menganalisis efektivitas *LLM* dalam mendeteksi serangan *DDoS* pada jaringan komputer. Dengan memanfaatkan kemampuan pemrosesan bahasa alami yang canggih, *LLM* diharapkan mampu memberikan wawasan yang lebih mendalam dan akurat tentang pola serangan, serta membantu perusahaan merespons serangan dengan lebih efektif. Kombinasi antara pendekatan teknologi canggih seperti *LLM* dan strategi keamanan jaringan yang kuat dapat menjadi solusi inovatif untuk melindungi infrastruktur digital dari ancaman *DDoS* yang terus berkembang.

II. PENELITIAN TERKAIT

Penelitian terdahulu terkait deteksi serangan *DDoS* menunjukkan berbagai pendekatan dan metode yang telah digunakan untuk mengidentifikasi serta mencegah ancaman ini. Sebuah penelitian mengembangkan kerangka kerja adaptif berbasis *Transformer Encoder* dan *Gaussian Mixture Model (GMM)* untuk mendeteksi dan mengklasifikasikan serangan jaringan. Kerangka kerja ini berhasil mencapai akurasi klasifikasi sebesar 95,6% dengan kemampuan adaptasi terhadap ancaman baru melalui penyematan BERT, menjadikannya efektif untuk deteksi intrusi secara *real-time* [1].

Penelitian lain memanfaatkan kerentanan pada model bahasa besar (LLM) untuk mengembangkan strategi pertahanan terhadap ancaman siber yang digerakkan oleh LLM. Dengan tingkat keberhasilan pertahanan hingga 90%, pendekatan ini inovatif dalam menetralkan ancaman melalui eksploitasi bias dan keterbatasan memori pada LLM [2]. Selanjutnya, model GPT (GPT-3.5, GPT-4, dan Ada) digunakan dalam penelitian lain untuk mendeteksi serangan *DDoS* melalui *fine-tuning* dan pembelajaran *few-shot.* Model ini mencapai akurasi hingga 96% pada *dataset* Urban *IoT*, menunjukkan performa yang unggul dibandingkan dengan jaringan saraf tradisional [3].

Dalam studi lain, *Llama* 2 dioptimalkan untuk deteksi serangan *DDoS* secara waktu nyata dengan mengubah paket jaringan menjadi data teks. Pendekatan ini, meskipun menjanjikan, tidak memberikan detail metrik akurasi secara eksplisit, tetapi membuktikan efektivitasnya dalam skenario dunia nyata [4]. Penelitian juga dilakukan pada jaringan *Software-Defined Networking (SDN)* menggunakan *Snort* IDS dan *firewall* berbasis *Iptables* untuk mendeteksi dan mencegah serangan *DDoS*. Sistem ini efektif menggabungkan deteksi dan pencegahan, meskipun skalabilitasnya terbatas untuk serangan berskala besar [5].

Metode berbasis *Deep Neural Network (DNN)* digunakan dalam penelitian lain untuk mendeteksi serangan *DDoS*, mencapai akurasi 96,5% pada dataset tertentu, tetapi performanya menurun pada *dataset* lain dengan akurasi sedang sebesar 45,87% [6]. Selain itu, *Deep Q-Network (DQN)* juga diusulkan untuk mendeteksi serangan *DDoS* pada dataset CICDDoS2019, dengan akurasi mencapai 96%, mengungguli metode tradisional seperti *Logistic Regression* dan *Support Vector Regression* [7].

Studi lain membandingkan beberapa algoritme pembelajaran mesin, termasuk *Random Forest*, *Support Vector Machine* (SVM), *K-Nearest Neighbor (KNN)*, dan *Multi-layer Perceptron (MLP)*. Hasilnya menunjukkan *Random Forest* memiliki akurasi tertinggi sebesar 99,41%, menjadikannya metode yang paling efektif [8]. Penelitian lain mengkombinasikan seleksi fitur *Information Gain* dengan klasifikasi *Naïve Bayes* dan KNN untuk mendeteksi serangan *DDoS*. KNN terbukti lebih unggul dengan akurasi mencapai 99% dibandingkan *Naïve Bayes* [9]. Terakhir, algoritme Random Forest digunakan dalam penelitian lain untuk mendeteksi serangan DDoS pada jaringan SDN, mencapai akurasi sekitar 90% dengan waktu deteksi rata-rata 0,3 detik, menunjukkan efisiensinya untuk deteksi serangan secara cepat [10].

Penelitian-penelitian tersebut menunjukkan berbagai pendekatan dan model yang memiliki kelebihan dan kekurangan masing-masing, memberikan landasan yang kuat untuk pengembangan metode deteksi *DDoS* yang lebih baik di masa mendatang.

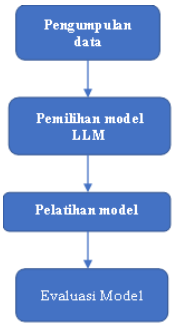
Table 1 : Analisis Pembanding Penelitian Terdahulu

Referensi	Metode Deteksi	Model	Akurasi	Kekurangan
[1]	Kerangka kerja adaptif menggunakan *Transformer Encoder* dan *Gaussian Mixture Model (GMM)*.	*Transformer Encoder* dan penyematan BERT	95,6%	Membutuhkan sumber daya komputasi untuk penyesuaian dinamis
[2]	Strategi pertahanan yang mengeksploitasi kerentanan pada model bahasa besar (LLM) yang menyerang	LLM	90%	Terbatas pada ancaman yang digerakkan oleh LLM tertentu
[3]	*Few-shot learning dan fine-tuning* dengan model bahasa besar (LLM)	Varian GPT (GPT-3.5, GPT-4)	95%	Ketergantungan pada *dataset* besar untuk *fine-tuning*

PROKASDADIK: Prosiding Kecerdasan Artifisial, Sains Data, dan Pendidikan Masa Depan
Vol. 3, No. 1, (2025) E-ISSN: 3063-5845

[4]	Deteksi *DDoS* waktu nyata melalui pemindaian paket jaringan dan konversi ke bentuk teks	*Llama* 2 dengan *QLoRA* dan pustaka TRL	Menjanjikan, tetapi tidak disebutkan secara eksplisit	Detail metrik performa terbatas
[5]	Sistem I*ntrusion Detection System (IDS)* menggunakan *Snort* dan *Iptables* untuk mitigasi *DDoS*.	*Snort* IDS dan firewall server	Tidak disebutkan	Skalabilitas terbatas untuk serangan skala besar
[6]	*Deep Neural Network (DNN)* dengan normalisasi min-max	DNN 4 lapis dengan aktivasi *ReLU* dan *Softmax*	96,5%	Akurasi sedang pada dataset yang lebih beragam
[7]	*Deep Q-Network (DQN)* untuk deteksi berbasis *reinforcement learning*	DQN dibandingkan dengan LR dan SVR	96%	Membutuhkan pelatihan intensif untuk hasil optimal
[8]	Pembelajaran mesin dengan beberapa algoritma	*Random Forest*, SVM, KNN, *Multi-layer Perceptron*	*Random Forest* : 99,41%; SVM : 98,37%; KNN : 99%; MLP : 93,97%	Detail mengenai skalabilitas terbatas
[9]	Seleksi fitur *Information Gain* dan klasifikasi *Naïve Bayes* dan KNN	*Naïve Bayes* dan KNN	99%	pada *dataset* tertentu; generalisasi belum dijelaskan
[10]	*Random Forest* untuk klasifikasi *DDoS* pada jaringan SDN	*Random Forest.*	90%	Tidak skala besar untuk jaringan kompleks

III. METODE PENELITIAN



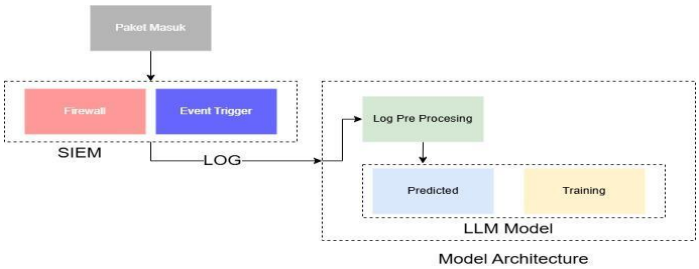
Gambar 1 : Metodologi Penelitian

Langkah yang digunakan dalam penelitian ini ditunjukkan, yang dijelaskan sebagai berikut. Pertama, dilakukan pengambilan *dataset* Pengumpulan data diambil dari *website* *Kaggle.com* (https://www.kaggle.com/datasets/jacobvs/ddos-attack-network-logs), *website* ini merupakan web yang cukup terkenal untuk pengambilan data dengan berbagai sumber dari dalam maupun luar negeri. paket data serangan *DDoS* dan aliran paket data normal UNSW-NB15 dalam bentuk *.pcap* Lalu dilakukan transformasi *dataset .pcap* menjadi format *.csv* untuk diekstraksi isinya. Data hasil ekstraksi selanjutnya dikuantifikasi untuk mendapatkan fitur lalu lintas jaringan. Selanjutnya, dilaksanakan proses normalisasi data hasil

IV. HASIL DAN PEMBAHASAN

Kebanyakan orang menggunakan LLM untuk menjawab tugas atau mencari tahu sesuatu berbasis teks. Namun kami melihat bahwa LLM dapat melakukan tugas yang lebih dari itu yaitu melakukan identifikasi serangan DDOS yang bersumber dari LOG yang didapat dari SIEM yang sudah di *pre-processing* sebelumnya. Gambaran arsitektur sistem dari penelitian yang kami buat adalah

PROKASDADIK: Prosiding Kecerdasan Artifisial, Sains Data, dan Pendidikan Masa Depan
Vol. 3, No. 1, (2025) E-ISSN: 3063-5845



Gambar 2 : Model Arsitektur

A. DATASET

Dari *dataset* yang sudah ada, untuk melakukan *training* kami menggunakan kolom PKT_IN, PKT_OUT, PKT_R, PKT_DELAY_NODE, PKT_RATE, BYTE_RATE, PKT_AVG_SIZE, UTILIZATION, PKT_DELAY, PKT_SEND_TIME, PKT_RESERVED_TIME, FIRST_PKT_SEND, dan LAST_PKT_SEND untuk melatih model dengan kolom PKT_CLASS sebagai target atau label. Dari banyak data yang ada di dataset tersebut kami menggunakan 300 data *training* dan 1125 data untuk data testing.

B. TRAINING

Kami menggunakan model gpt-4o-mini dari *openai* untuk melakukan identifikasi serangan DDOS tersebut, di mana model tersebut kami latih dengan menggunakan *prompt message* berikut :

```
Data : \n
        PKT_IN : [PKT_IN]
        PKT_OUT : [PKT_OUT]
        PKT_R : [PKT_R]
        PKT_DELAY_NODE : [PKT_DELAY_NODE]
        PKT_RATE : [PKT_RATE]
        BYTE_RATE : [BYTE_RATE]
        PKT_AVG_SIZE : [PKT_AVG_SIZE]
        UTILIZATION : [UTILIZATION]
        PKT_DELAY : [PKT_DELAY]
        PKT_SEND_TIME : [PKT_SEND_TIME]
        PKT_RESEVED_TIME : [PKT_RESEVED_TIME]
        FIRST_PKT_SENT : [FIRST_PKT_SENT]
        LAST_PKT_RESEVED : [LAST_PKT_RESEVED]
Label : [LABEL] \n
```

Gambar 3 : *Prompt Message* Latih

Setelah model dilatih dengan *prompt* di atas kami melakukan data testing dengan meminta model untuk memprediksi apa label dari data yang kami berikan. Untuk melakukan data testing kami menggunakan *prompt message* sebagai berikut

```
Data : \n
        PKT_IN : [PKT_IN]
        PKT_OUT : [PKT_OUT]
        PKT_R : [PKT_R]
        PKT_DELAY_NODE : [PKT_DELAY_NODE]
        PKT_RATE : [PKT_RATE]
        BYTE_RATE : [BYTE_RATE]
        PKT_AVG_SIZE : [PKT_AVG_SIZE]
        UTILIZATION : [UTILIZATION]
        PKT_DELAY : [PKT_DELAY]
        PKT_SEND_TIME : [PKT_SEND_TIME]
        PKT_RESEVED_TIME : [PKT_RESEVED_TIME]
        FIRST_PKT_SENT : [FIRST_PKT_SENT]
        LAST_PKT_RESEVED : [LAST_PKT_RESEVED]
\n
Apa prediksi label dari log tersebut? Kembalikan dengan format 'Predicted
: Label' contoh : Predicted : Normal
```

Gambar 4 : *Promp Message Testing*

Bisa diperhatikan pada baris terakhir dari prompt message tersebut kami melakukan standarisasi *response* di mana kami ingin respon yang diberikan oleh model hanya memberikan hasil prediksi dengan format '*Predicted* : [LABEL]'. Hal ini bertujuan agar data yang diberikan bersifat seragam sehingga mempermudah proses pencatatan dan evaluasi model di tahap berikutnya.

C. EVALUASI

Hasil dari data *training* dan data testing sebelumnya kami evaluasi menggunakan *confusion matrix*, dari *confusion matrix* itu kami dapat melakukan perhitungan lanjutan untuk menentukan *Accuracy*, *Precision*, *Recall*, dan *F1-Score* dari model tersebut.

- *Accuracy* digunakan untuk mengukur proporsi prediksi yang benar yang dibuat oleh model, dapat dihitung dengan rumus

$$Acc = \frac{TP + TN}{TP + TN + FP + FN}$$

Gambar 5 : Rumus *Accuracy*

PROKASDADIK: Prosiding Kecerdasan Artifisial, Sains Data, dan Pendidikan Masa Depan
Vol. 3, No. 1, (2025) E-ISSN: 3063-5845

- *Precision* digunakan untuk menghitung proporsi contoh positif yang ditentukan dengan benar dari semua contoh yang diprediksi sebagai positif, dapat dihitung dengan rumus

$$Presisi = \frac{TP}{TP + FP}$$

Gambar 6 : Rumus *Precision*

- *Recall* digunakan untuk mengukur proporsi positif aktual yang diidentifikasi dengan benar oleh model, dapat dihitung dengan rumus

$$Recall = \frac{TP}{TP + FN}$$

Gambar 7 : Rumus *Recall*

- *F1-Score* digunakan untuk mencari rata-rata dari precision dan recall untuk memberikan keseimbangan dari model, dapat dihitung dengan rumus

$$F1\text{-}Score = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall}$$

Gambar 8 : Rumus *F1-Score*

Dari hasil evaluasi menggunakan *confusion matrix* didapati hasil *Accuracy, Percision, Recal* dan *F1-Score* sebagai berikut :

Tabel 2 : *Confusion Matrix*

	Normal	UDP-Flood	Smurf	SIDDOS
Normal	1008	7	3	0
UDP-Flood	9	84	0	0
Smurf	5	0	4	0
SIDDOS	3	0	0	2

Tabel 3 : Tabel Akurasi Model

ACC	Precision	Recal	F1
0.99	0.98	0.98	0.98

Hasil evaluasi model menunjukkan performa yang sangat baik dengan metrik yang hampir sempurna. Tingkat akurasi (*accuracy*) mencapai 0.99, menandakan bahwa 99% prediksi yang dibuat oleh model sudah sesuai dengan data sebenarnya. Nilai presisi (*precision*) sebesar 0.98 menunjukkan bahwa dari semua prediksi positif yang dihasilkan model, 98% benar-benar merupakan kelas positif, mencerminkan kemampuan model dalam menghindari *false positive.* Selain itu, nilai *recall* sebesar 0.98 mengindikasikan bahwa model berhasil mengidentifikasi 98% dari total kasus positif yang ada, menandakan kemampuannya dalam menangani *false negative*. Kombinasi antara presisi dan *recall* menghasilkan nilai F1-Score sebesar 0.98, yang mencerminkan keseimbangan optimal antara kedua metrik tersebut. Secara keseluruhan, model ini dapat diandalkan untuk digunakan pada tugas yang membutuhkan tingkat akurasi dan ketelitian tinggi

V. KESIMPULAN

Penelitian ini menunjukkan bahwa model GPT-4o-mini dari *OpenAI* dapat dioptimalkan untuk tugas identifikasi serangan *DDoS* berdasarkan log yang dihasilkan oleh SIEM. Dengan menggunakan *dataset* yang terdiri dari 300 data *training* dan 1125 data testing, model dilatih menggunakan fitur-fitur seperti *PKT_IN, PKT_OUT, PKT_R, hingga LAST_PKT_SEND,* dengan target kolom PKT_CLASS. Evaluasi model dilakukan menggunakan *confusion matrix* yang menghasilkan tingkat akurasi sebesar 0.99, presisi 0.98, *recall* 0.98, dan F1-Score 0.98. Hasil ini mengindikasikan performa model yang sangat baik dalam mengidentifikasi serangan *DDoS*, baik dari segi menghindari *false positive* maupun menangani *false negative.*

Namun, penelitian ini memiliki beberapa keterbatasan. Pertama, jumlah data *training* yang digunakan relatif kecil, sehingga performa model pada *dataset* yang lebih besar atau kompleks perlu divalidasi lebih lanjut. Kedua, penelitian ini hanya menggunakan satu jenis model (GPT-4o-mini) tanpa melakukan perbandingan dengan model lain yang mungkin lebih cocok untuk tugas ini. Selain itu, proses *pre-processing* data juga dapat mempengaruhi hasil, sehingga pendekatan *pre-processing* yang lebih variatif perlu dieksplorasi.

Untuk penelitian selanjutnya, disarankan untuk meningkatkan ukuran dan keragaman *dataset* guna menguji performa model dalam skenario yang lebih kompleks. Selain itu, membandingkan performa model ini dengan model lain seperti LSTM, CNN, atau *Transformer-based* khusus untuk *time-series* data dapat memberikan wawasan tambahan. Penelitian ini juga dapat diperluas

PROKASDADIK: Prosiding Kecerdasan Artifisial, Sains Data, dan Pendidikan Masa Depan
Vol. 3, No. 1, (2025) E-ISSN: 3063-5845

dengan mengevaluasi kemampuan model dalam mendeteksi jenis serangan lain selain *DDoS*, sehingga aplikasinya menjadi lebih komprehensif.

DAFTAR PUSTAKA

[1] Adjewa, F., Esseghir, M., & Merghem-Boulahia, L. (2024). LLM-based Continuous Intrusion Detection Framework for Next Gen Networks. *arXiv*.
[2] Ayzenshteyn, D., Weiss, R., & Mirsky, Y. (2024). The Best Defense is a Good Offense: Countering LLM-Powered Cyberattacks. *arXiv preprint arXiv:2410.15396*.
[3] Guastalla, M., Zong, M., Hekmati, A., Li, Y., Krishnamachari, B., & Cui, Y. (2024). Application of Large Language Models to DDoS Attack Detection. *Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, *497*, 65–75.
[4] Mahmoodi, M., & Jameii, S. M. (2024). Utilizing Large Language Models for DDoS Attack Detection. In *2024 OPJU International Technology Conference (OTCON) on Smart Computing for Innovation and Advancement in Industry 4.0* (pp. 1–5). IEEE.
[5] Pratiwi, D. Y. D., & Adrian, R. (2023). Deteksi dan Mitigasi Serangan DDoS pada Software Defined Network Menggunakan Snort dan Packet Filtering Iptables (Tugas Akhir, D4 Teknologi Jaringan). Universitas Gajahmada.
[6] Simarmata, P., Saragih, N. F., & Jaya, I. K. (2023). Deteksi Serangan DDoS Pada VPS Menggunakan Metode Deep Neural Network. *METHOTIKA: Jurnal Ilmiah Teknik Informatika*, *3*(1), 1–10.
[7] Purba, R., Lestari, W. S., & Ulina, M. (2022). Deteksi Serangan DDoS Menggunakan Deep Q-Network. *Jurnal Teknik Informatika dan Sistem Informasi*, *9*(1), 648–658.
[8] Maulana, I., & Alamsyah, A. (2023). Optimalisasi Deteksi Serangan DDoS Menggunakan Algoritma Random Forest, SVM, KNN, dan MLP pada Jaringan Komputer. *Indonesian Journal of Mathematics and Natural Sciences*, *46*(2), 83–92.
[9] Jatmika, M. O. (2022). Implementasi Naive Bayes dan KNN pada Deteksi Serangan DDoS pada Jaringan Metro (Tugas Akhir S1). Universitas Mercu Buana Bekasi.
[10] Harto, M. K., & Basuki, A. (2021). Deteksi Serangan DDoS Pada Jaringan Berbasis SDN Dengan Klasifikasi Random Forest. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, *5*(4), 1329–1333.