BAB II
TINJAUAN PUSTAKA

2.1 Pencapaian Terdahulu

Penelitian ini bertujuan untuk mengembangkan *Security Information & Event Management (*SIEM*)* untuk mencegah akses ke URL *berbahaya* dengan menggunakan algoritma *Large Language Model*. Untuk mendukung penelitian ini, berbagai pencapaian terdahulu telah ditinjau untuk memahami pendekatan dan metode yang telah digunakan untuk mendeteksi dan mencegah akses pada URL *berbahaya*. Beberapa penelitian yang relevan adalah sebagai berikut:

Tabel 2.1 Penelusuran Literatur

Penelitian ke-1	
Nama Penulis	Nguyen et al. (2020)
Judul	Detecting abnormal DNS traffic using unsupervised *machine learning*
Hasil	Membandingkan kinerja empat algoritma pembelajaran mesin yang tidak diawasi: K-means, Gaussian Mixture Model (GMM), Density-Based Spatial *Clustering* of Applications with Noise (DBSCAN), dan Local Outlier Factor (LOF) pada Boss dari SOC Dataset Versi 1 (Botsv1) dataset dari proyek Splunk untuk mendeteksi *Malicious* DNS Traffic.
Penelitian ke-2	
Nama Penulis	Rafi et al. (2023)
Judul	Analisis *Malicious* URL pada file menggunakan metode K-Means Clustering berbasis Host-Based Feature Extraction
Hasil	Menghasilkan sebuah dataset URL yang memiliki fitur DNS Record dari URL yang akan digunakan sebagai data untuk

	melakukan clustering dengan K-Mean. Dengan menggunakan nilai k=2, kluster benign dan *malicious* URL dihasilkan sebagai visualisasi dari data hasil clustering dengan K-Mean.
Penelitian ke-3	
Nama Penulis	(Do Xuan et al., 2020)
Judul	*Malicious* URL Detection based on *Machine learning*
Hasil/Bukti	Melakukan deteksi *Malicious* URL menggunakan algoritma Random Forest dan Support Vector Machine untuk memprediksi adanya *Malicious* URL pada sistem dan dikelompokkan berdasarkan tingkat prediksi URL yang aman atau yang tidak aman.
Penelitian ke-4	
Nama Penulis	(Singh & Roy, 2020)
Judul	Detecting *Malicious* DNS over HTTPS Traffic Using *Machine learning*
Hasil/Bukti	Analisa terhadap deteksi *Malicious* DNS pada trafik jaringan menggunakan *machine learning* dengan algoritma Random Forest. Dimana data diklasifikasikan menjadi DNS yang aman dan DNS yang tidak aman berdasarkan hasil confusion matrix dari dataset yang diambil.
Penelitian ke-5	
Nama Penulis	(Tiwari, 2019)
Judul	Suspicious URL Detection using Dynamic Learning Model with *Machine learning*
Hasil/Bukti	Pembuatan aplikasi sederhana berbasis python untuk mendeteksi adanya URL yang mencurigakan dengan

	menggunakan algoritma Linear SVM Classifier, K Nearest Neighbors Classifier, Random Forest Classifier dan menunjukkan hasil akhir dalam bentuk persentase terhadap URL tujuan yang dicari.
Penelitian ke-6	
Nama Penulis	(Hilal et al., 2023)
Judul	*Malicious* URL Classification Using Artificial Fish Swarm Optimization and Deep Learning
Hasil	Mengembangkan model Artificial Fish Swarm Algorithm dengan Deteksi dan Klasifikasi URL Berbahaya dengan menggunakan Deep Learning (AFSADL-MURLC). Model AFSADL-MURLC yang disajikan bertujuan untuk membedakan URL berbahaya dari URL asli. Hasil simulasi menegaskan keunggulan model AFSADL-MURLC yang diusulkan dibandingkan dengan pendekatan terbaru berdasarkan berbagai ukuran.
Penelitian ke-7	
Nama Penulis	(Sim & Kim, 2023a)
Judul	*Malicious* URL Detection based on Supervised and Unsupervised Learning using MobileBERT Embedding
Hasil/Bukti	Kinerja yang lebih tinggi dicapai ketika dimensi vektor embedding dikurangi menggunakan PCA daripada autoencoder, dan URL berbahaya dapat dideteksi dengan probabilitas tinggi hanya dengan menggunakan vektor yang di-embedding melalui MobileBERT tanpa pengurangan dimensi. Dalam Unsupervised Learning, recall keseluruhan lebih tinggi daripada presisi, dan peningkatan jumlah sampel data normal meningkatkan kinerja deteksi.

Penelitian ke-8	
Nama Penulis	(Shaheed & Kurdy, 2022)
Judul	Web Application Firewall Using *Machine learning* and Features Engineering
Hasil	Mengembangkan model firewall aplikasi web yang menggunakan Feature Extracting menggunakan Dataset CSIC 2010, HTTPParams 2015, Hybrid dataset (CSIC 2010 and HTTPParams), *machine learning* Naïve Bayes, Logistc Regression, Decision Tree, Support Vector Machine. dan rekayasa fitur untuk mendeteksi serangan web umum. Model yang diusulkan menganalisis permintaan yang masuk ke server web, mem-parsing permintaan tersebut untuk mengekstrak empat fitur yang menggambarkan bagian permintaan HTTP (URL, payload, dan header), dan mengklasifikasikan apakah permintaan tersebut normal atau anomali. Hasilnya menunjukkan bahwa model yang diusulkan mencapai akurasi klasifikasi sebesar 99,6% dengan dataset yang digunakan dalam penelitian ini dan 98,8% dengan dataset dari server web nyata.
Penelitian ke-9	
Nama Penulis	(Muzaki et al., 2020)
Judul	Improving Security of Web-Based Application Using ModSecurity and Reverse Proxy in Web Application Firewall
Hasil/Bukti	Melakukan pengujian, pemantauan dan pemblokiran menggunakan ModSecurity sebagai WAF dengan metode Reverse Proxy untuk mencegah serangan Cross-site scripting, SQL Injection dan web vulnerability scanning yang tidak sah. Dari hasil yang dilakukan, seperti cross-site scripting, SQL

	injection, dan pemindaian kerentanan web yang tidak sah, semua ancaman berhasil digagalkan oleh ModSecurity dan metode reverse proxy yang diimplementasikan dalam WAF.
Penelitian ke-10	
Nama Penulis	Almasri, M. N., et al. (2024)
Judul	Detecting Phishing URLs using the BERT Transformer Model
Hasil/Bukti	*Encoder* yang diusulkan mengungguli model *deep learning* berbasis karakter *state-of-the-art* dan model BERT yang berfokus pada keamanan siber di berbagai tugas dan *dataset*. Klasifikasi yang dihasilkan mencapai akurasi 95-99% dalam mendeteksi situs berbahaya dari URL mereka dengan *false positive* yang sederhana.
Penelitian ke-11	
Nama Penulis	Li, Y., et al. (2024)
Judul	Continuous Multi-Task Pre-training for Malicious URL Detection and Webpage Classification
Hasil/Bukti	urlBERT mengungguli model yang dilatih awal standar dan mode *multi-task*-nya mampu memenuhi kebutuhan dunia nyata. Menunjukkan potensi untuk deteksi URL berbahaya dan klasifikasi halaman web secara bersamaan.

2.2 Tinjauan Teoritis

Dalam konteks penelitian ini, tinjauan teori digunakan sebagai dasar untuk penyesuaian dengan topik penelitian dan sebagai pedoman untuk melakukan penelitian yang baik.

2.2.1 URL

Menurut Do Xuan et al (2020), Uniform Resource Locator (URL) digunakan untuk merujuk pada sumber daya di Internet. Karakteristik dari URL adalah dua komponen dasar seperti ID protokol, yang menunjukkan protokol mana yang harus digunakan, dan nama sumber daya, yang menentukan alamat IP atau nama domain di mana sumber daya berada. Anda dapat melihat bahwa setiap URL memiliki struktur dan format yang spesifik.

2.2.2 *Security Information & Event Management (*SIEM*)*

Menurut Bezas & Filippidou (2023) dan Horng et al., (2023), *Security Information & Event Management (*SIEM*)* adalah alat keamanan siber yang menggabungkan *Security Information Management (SIM)* dan *Security Event Management (SEM)* untuk menawarkan pendekatan komprehensif untuk deteksi dan respons ancaman siber. Sistem SIEM mengumpulkan, menganalisis, menormalkan, dan mengkorelasikan data dari berbagai sumber untuk mengidentifikasi potensi ancaman siber secara langsung dan menawarkan pandangan terpusat tentang posisi keamanan organisasi.

2.2.2.1 Wazuh

Wazuh adalah platform keamanan siber open-source yang terkenal dan lengkap yang dapat memantau keamanan, mendeteksi ancaman, dan menanggapi insiden secara real-time. Ini berfungsi sebagai solusi yang kuat untuk sistem deteksi intrusi berbasis host (HIDS) dan manajemen informasi dan peristiwa keamanan (SIEM). Ini memungkinkan organisasi untuk mengumpulkan, menganalisis, dan mengkorelasikan data keamanan dari berbagai sumber, seperti log, kejadian, dan lalu lintas jaringan. Wazuh beroperasi dalam arsitektur client-server. Agent Wazuh

dipasang pada endpoint atau perangkat yang dipantau, seperti laptop, server, atau kontainer. Tugasnya adalah mengumpulkan data sistem, log, dan informasi keamanan lainnya. Selanjutnya, data dikirimkan ke Manajer Wazuh Pusat untuk dianalisis, diindeks, dan disimpan. Manajer Wazuh menganalisis data sesuai dengan beberapa aturan yang telah dikonfigurasi, dan akan memberikan peringatan (pesan) ketika ada kejadian yang sesuai dengan aturan tertentu. Wazuh Dashboard yang terintegrasi kemudian memungkinkan visualisasi data yang dianalisis.

2.2.3 *Large Language Model*

Large Language Models (LLM) adalah jenis model pembelajaran mesin berskala besar yang dirancang untuk memproses dan memahami bahasa alami. Model ini dilatih dengan jumlah data yang besar menggunakan metode *self-supervised learning* untuk membentuk representasi umum dari bahasa. Bommasani et al. (2022) menyebut model seperti ini sebagai *foundation models,* karena dapat diterapkan pada berbagai tugas tanpa harus dilatih ulang secara spesifik untuk setiap tugas. Selain itu, Zhao et al. (2023) menyatakan bahwa LLM digunakan untuk menangkap *general-purpose linguistic representations*, sehingga dapat dimanfaatkan dalam berbagai aplikasi NLP seperti, *question answering, text summarization*, hingga *code generation*. LLM adalah model probabilistik yang dibangun di atas jaringan saraf dengan miliaran atau bahkan triliunan parameter. Parameter ini berfungsi sebagai penyimpan pengetahuan linguistik dan faktual yang diekstraksi dari data pelatihan. Skala yang masif inilah yang menjadi kunci dari kemampuan LLM untuk melakukan penalaran, generalisasi, dan pemahaman bahasa yang kompleks. Berikut adalah cara kerja dari LLM;

Arsitektur Basis: Transformer

Menurut Vaswani et al. (2017), merupakan dasar dari arsitektur Large Language Models (LLM) kontemporer. Mekanisme perhatian diri, atau self-attention mechanism, adalah inovasi utama dalam arsitektur ini. Dengan bantuan mekanisme ini, model dapat menimbang dan mengevaluasi relevansi kontekstual dari setiap token dalam sekuens data secara bersamaan dengan

semua token lainnya. Kemampuan ini memungkinkan pemahaman konteks yang lebih akurat, mengatasi keterbatasan model sekuensial sebelumnya dalam menangani dependensi jangka panjang.

Pelatihan Dua Tahap Pengembangan LLM

Pre-training (Pelatihan Awal): Tahap pertama adalah pelatihan pra-latihan yang dilakukan sendiri. Pada tahap ini, model dilatih pada korpus teks yang sangat besar untuk tujuan pemodelan bahasa (bahasa modeling), yang mencakup hal-hal seperti memprediksi token yang akan datang. Metode ini memungkinkan model menginternalisasi representasi linguistik yang kaya, termasuk pengetahuan faktual, kemampuan penalaran dasar, dan tata bahasa.

Fine-tuning, atau penyelarasan: Tujuan dari tahap kedua penyelarasan adalah untuk menyelaraskan (align) perilaku model dengan instruksi dan preferensi manusia. Reinforcement Learning from Human Feedback (RLHF) adalah salah satu metodologi yang paling efektif untuk tujuan ini. Menurut Ouyang et al. (2022), proses RLHF menggunakan umpan balik manusia untuk melatih sebuah *reward model*, yang kemudian memandu LLM agar menghasilkan respons yang lebih bermanfaat, jujur, dan aman.

2.2.3.1 BERT

BERT, singkatan dari Bidirectional Encoder Representations from Transformers, adalah model bahasa inovatif yang diluncurkan oleh Google pada tahun 2018. Jika dibandingkan dengan model bahasa sebelumnya, model ini jauh lebih baik dalam memahami konteks bidireksional kata-kata dalam kalimat. Menurut Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019) BERT dibuat untuk mengajarkan representasi bahasa umum (replikasi bahasa umum) dari korpus teks besar, seperti BookCorpus dan Wikipedia, yang kemudian dapat disesuaikan dengan mudah untuk berbagai tugas pemrosesan bahasa alami (NLP) tanpa melakukan modifikasi arsitektur yang signifikan.

Arsitektur Encoder Transformer: Arsitektur encoder model Transformer adalah inti dari BERT. Menurut Vaswani et al. (2017), BERT menggunakan mekanisme self-attention untuk memproses seluruh urutan masukan secara paralel. Ini memungkinkan model untuk menangkap dependensi jarak jauh antar kata dengan sangat efisien dibandingkan dengan model recurrent (RNN) yang memproses sekuensial.

Bidireksionalitas Penuh: Berbeda dengan model bahasa sebelumnya, seperti GPT yang hanya unidireksional atau ELMo yang menggabungkan dua arah terpisah, BERT dilatih untuk memahami konteks kata berdasarkan kata di sebelah kiri dan kanannya sekaligus. Ini dicapai melalui dua tugas pra-pelatihan:

- Model bahasa yang disembunyikan (MLM): Sebagian kata disembunyikan (masked) dalam input, dan model BERT dilatih untuk memprediksi kata-kata yang disembunyikan tersebut berdasarkan konteks sekitarnya. Dengan tugas ini, model dapat memperoleh pemahaman bidireksional yang kaya tentang bahasa.
- Model Next Sentence Prediction (NSP) berfungsi untuk menentukan apakah dua bagian berurutan dari teks adalah pasangan kalimat logis (yaitu, apakah kalimat pertama benar-benar mengikuti kalimat kedua). BERT memperoleh pemahaman yang lebih baik tentang hubungan antar kalimat dari tugas ini, penting untuk tugas seperti menjawab pertanyaan dan inferensi bahasa.

Pra-pelatihan (Pre-training) dan Penyesuaian (Fine-tuning): Dua tahap utama dilakukan dalam model BERT:

- Tahap Pra-Pelatihan: Tugas MLM dan NSP digunakan untuk melatih model pada dataset teks yang sangat besar. Pada tahap ini, model belajar representasi bahasa yang umum dan mendalam.
- Tahap Penyesuaian: Hanya memerlukan penyesuaian (fine-tuning) dengan dataset yang jauh lebih kecil untuk tugas NLP khusus model BERT yang sudah dilatih sebelumnya. Ini secara signifikan mengurangi jumlah waktu dan sumber daya komputasi yang diperlukan untuk melatih model sejak awal.

Representasi Kontekstual: BERT dapat menghasilkan embedding (representasi numerik) kata yang kontekstual berkat bidireksionalitas dan arsitektur Transformer. Representasi kata-kata yang berbeda bergantung pada konteks kalimat, kata yang sama dengan makna yang sama akan memiliki representasi yang berbeda dalam beberapa kalimat.

Mekanisme kerja utama dari Bidirectional Encoder Representations from Transformers (BERT) adalah metode pelatihannya, yang dimaksudkan untuk membuat representasi kontekstual dua arah (bidirectional). BERT, yang dilakukan oleh Devlin et al. (2018), memproses seluruh sekuens kata secara bersamaan melalui arsitektur encoder Transformer. Ini membedakannya dari model sebelumnya yang memproses teks secara sekuensial (baik dari kiri ke kanan maupun dari kanan ke kiri). Metode ini dicapai melalui dua tugas *pre-training* yaitu, yaitu *Masked Language Model* (MLM) dan *Next Sentence Prediction* (NSP).

Tugas pre-training MLM adalah inovasi utama BERT, yang bertujuan untuk mengatasi keterbatasan model bahasa searah (unidirectional). Dalam praktiknya, sekitar lima belas persen dari token yang ada dalam sekuens masukan dipilih secara acak untuk "ditutupi" atau disembunyikan. Sebuah token khusus, [MASK], digunakan untuk menggantikan token-token yang telah dipilih sebelumnya. Menurut Devlin et al. (2018), tujuan model adalah untuk memprediksi token asli yang disembunyikan berdasarkan konteks kedua arah, yaitu token yang mendahului dan yang mengikutinya. Untuk ilustrasi, token "dilatih" dapat disembunyikan sebagai "Model ini [MASK] pada data besar" dalam kalimat "Model ini dilatih pada data besar". Oleh karena itu, untuk memprediksi token asli, model harus mempertimbangkan konteks "Model ini" dan "pada data besar" secara bersamaan.

Tujuan matematis dari MLM adalah untuk meminimalkan fungsi kerugian, atau fungsi kerugian, antara distribusi probabilitas prediksi model dan token asli yang disembunyikan. Fungsi kerugian ini biasanya disebut sebagai kerugian cross-entropy. Ini dapat digambarkan sebagai upaya untuk meningkatkan kemungkinan log kata yang benar dalam seluruh konteksnya:

$$\max_{\theta} \sum_{x \in \mathcal{D}} \log P(w_{masked} | x_{\backslash masked}; \theta)$$

Dimana D adalah korpus data, wmasked adalah himpunan token yang disembunyikan dalam sekuens x, dan x\masked adalah token yang tidak disembunyikan dalam sekuens yang sama.

BERT juga dilatih dengan tugas Next Sentence Prediction (NSP) untuk membantu model memahami hubungan antar kalimat, yang penting untuk tugas seperti Jawaban Pertanyaan (QA) dan Natural Language Inference (NLI). Sepasang kalimat (A dan B) diberikan kepada model dalam tugas ini, dan mereka kemudian diminta untuk melakukan klasifikasi biner untuk menentukan apakah kalimat B merupakan kelanjutan logis dari kalimat A dalam korpus asli (Devlin et al., 2018). Selama pelatihan, setengah dari pasangan kalimat yang diberikan adalah pasangan yang berurutan, dan setengah lagi adalah pasangan di mana kalimat B dipilih secara acak dari korpus. Untuk klasifikasi ini, representasi output dari token khusus [CLS] yang ditambahkan di awal setiap sekuens input digunakan sebagai representasi gabungan dari seluruh pasangan kalimat.

Setelah melalui prosedur diatas, BERT berhasil membuat representasi vektor yang kaya akan informasi kontekstual pada level kata dan kalimat dengan menggabungkan dua tugas pre-training tersebut. Dengan menambahkan hanya satu lapisan output tambahan yang khusus untuk tugas hilir, presentasi ini kemudian dapat disesuaikan dan disesuaikan secara efektif untuk berbagai tugas di bawahnya.