JITET (Jurnal Informatika dan Teknik Elektro Terapan)
Vol. 14 No. 1, pISSN: 2303-0577 eISSN: 2830-7062 http://dx.doi.org/10.23960/jitet.v14i1.8219

RANCANG BANGUN SISTEM ANALISIS LOG JARINGAN BERBASIS WEB UNTUK DETEKSI REAL-TIME ANCAMAN CANGGIH DAN GERAKAN LATERAL

Aldo Bonifasius Simbolon[1*], Dedy Kiswanto[2], Dean Siregar[3], Anwar Shaleh Lbn Gaol[4]

[1,2,3,4]Ilmu Komputer, Universitas Negeri Medan; Jl. Williem Iskandar Pasar V, Medan Estate, Medan, Sumatera Utara

Keywords:
Analisis Log;
Deteksi Ancaman;
Grafana;
Observability;
Pergerakan Lateral.

Corespondent Email:
aldo.4233250031@mhs.unimed.ac.id

Abstrak. Lanskap keamanan siber saat ini ditandai oleh evolusi ancaman yang terus-menerus dan dinamis, yang bergerak jauh melampaui cakupan malware konvensional dan serangan-serangan sederhana. Pelaku ancaman modern menggunakan teknik tersembunyi seperti pergerakan lateral (*lateral movement*) yang menghindari sistem keamanan tradisional berbasis tanda tangan (*signature-based*). Hal ini menciptakan masalah *signal-to-noise* yang signifikan di dalam data log yang bervolume besar. Penelitian ini menjawab tantangan tersebut dengan merancang dan mengimplementasikan sistem analisis log jaringan berbasis web. Sistem ini dibangun di atas tumpukan *observability* modern dan *open-source* yang mencakup Grafana, Loki, dan Prometheus. Solusi yang diusulkan mengintegrasikan alur data hibrida (*hybrid data pipeline*) untuk log dan metrik dengan logika deteksi berbasis aturan (*rule-based*) serta dasbor interaktif untuk visualisasi yang berpusat pada analis. Temuan utama dari skenario serangan simulasi menunjukkan bahwa sistem ini secara efektif mendeteksi upaya *brute-force* SSH dan aktivitas pergerakan lateral berikutnya secara *real-time*. *Intrusion Prevention System* (IPS) yang terintegrasi di dalam sistem berhasil memberikan respons otomatis dengan memblokir alamat IP penyerang di tingkat *firewall*. Studi ini menyimpulkan bahwa platform *open-source* yang terintegrasi secara holistik dapat berhasil menjembatani kesenjangan antara data log mentah dan intelijen keamanan yang dapat ditindaklanjuti. Hal ini memungkinkan pergeseran dari pemantauan pasif ke strategi pertahanan aktif.

1. PENDAHULUAN

Lanskap keamanan siber saat ini ditandai oleh evolusi ancaman yang terus-menerus dan dinamis. Ancaman ini telah bergerak jauh melampaui cakupan malware konvensional dan serangan-serangan sederhana. Pelaku ancaman modern, terutama yang tergolong *Advanced Persistent Threats* (APTs), menggunakan strategi multi-tahap yang kompleks untuk menyusup, bertahan di dalam, dan akhirnya membobol jaringan organisasi. Salah satu fase kritis dalam siklus hidup serangan canggih ini, seperti yang dirumuskan dalam kerangka kerja MITRE ATT&CK, adalah pergerakan lateral (*lateral movement*). Teknik ini memungkinkan pelaku ancaman memanfaatkan satu titik awal yang berhasil dibobol untuk bergerak semakin dalam ke jaringan, meningkatkan hak akses, dan bernavigasi menuju aset bernilai tinggi seperti basis data sensitif, *domain controller*, atau kendali infrastruktur kritis. Pentingnya fase strategis ini telah terbukti dalam insiden dunia nyata, seperti serangan canggih terhadap jaringan KA-SAT milik Viasat. Serangan itu menunjukkan bagaimana penyerang dapat berpindah-pindah di dalam jaringan untuk mencapai gangguan yang meluas. Realitas operasional ini menandakan pergeseran penting

dalam paradigma pertahanan. Medan pertempuran utama tidak lagi hanya di perimeter (batas luar) jaringan, tetapi telah bergeser secara tegas ke bagian dalam jaringan. Hal ini menuntut ketersediaan kemampuan yang kuat untuk mendeteksi aktivitas jahat setelah pembobolan terjadi.

Deteksi pergerakan lateral menjadi tantangan besar bagi operasi keamanan modern. Pelaku ancaman sengaja menggunakan taktik yang dirancang untuk menghindari langkah-langkah keamanan tradisional. Caranya adalah dengan berbaur di antara lalu lintas data yang sah dan bervolume besar dalam jaringan perusahaan. Hal ini sering dicapai melalui teknik "*living-off-the-land*". Dalam teknik ini, penyerang memanfaatkan alat administrasi sistem dan protokol bawaan yang memang umum digunakan di lingkungan target, seperti PowerShell, *Windows Management Instrumentation* (WMI), dan *Remote Desktop Protocol* (RDP). Karena alat-alat ini sah (legitim), penggunaannya tidak memicu alarm pada *Intrusion Detection Systems* (IDS) konvensional yang berbasis tanda tangan (*signature-based*). Sistem IDS semacam itu pada dasarnya dirancang untuk mengidentifikasi pola jahat yang sudah dikenal dan sebagian besar tidak efektif melawan ancaman baru atau yang selalu berubah bentuk. Oleh karena itu, kesulitan utamanya bukanlah kekurangan data, karena bukti tindakan jahat hampir selalu tercatat dalam log sistem dan jaringan. Sebaliknya, ini adalah masalah rasio sinyal terhadap kebisingan (*signal-to-noise ratio*). Indikator serangan yang samar terkubur di dalam terabyte data operasional normal. Hal ini membuat analisis manual menjadi tidak praktis, dan deteksi otomatis yang hanya berdasarkan aturan sederhana menjadi sangat rentan terhadap kegagalan.

Salah satu pendekatan yang efektif dalam meningkatkan keamanan website adalah melalui analisis log aktivitas jaringan [1]. Untuk melawan ancaman yang tersembunyi dan canggih ini, komunitas keamanan siber telah mengalihkan fokusnya ke paradigma baru yang berpusat pada analisis log secara proaktif dan *real-time*. Namun, tinjauan lebih dekat pada lanskap penelitian mengungkapkan tantangan yang signifikan. Banyak penelitian cenderung berkonsentrasi pada pengembangan algoritma deteksi yang kuat (misalnya yang berbasis *machine learning* atau teori graf), namun seringkali dilakukan secara terpisah [2]. Fokus pada komponen individual ini, meskipun berharga, sering mengabaikan kerumitan praktis dalam mengintegrasikannya menjadi satu sistem yang padu dan menyeluruh. Masalah ini diperparah oleh masalah lain yang diakui secara luas dan menghambat pengembangan algoritma itu sendiri: kurangnya ketersediaan *dataset* yang realistis, berskala besar, dan berlabel akurat yang mencerminkan seluk-beluk jaringan perusahaan modern [3]. Hal ini menciptakan kesenjangan yang jelas antara kemajuan algoritma teoretis dan implementasi praktisnya dalam alat keamanan yang fungsional dan berpusat pada analis. Di sinilah peran dasbor visualisasi berbasis web menjadi sangat penting, yang berfungsi sebagai antarmuka esensial untuk kemitraan antara manusia dan mesin. Sistem semacam itu memberdayakan analis untuk menjelajahi peristiwa keamanan yang kompleks, mengidentifikasi pola, dan membuat keputusan penting yang mendesak tanpa memerlukan keahlian pemrograman atau ilmu data yang mendalam. Model ini memungkinkan mesin untuk unggul dalam pemrosesan data skala besar dan deteksi anomali, sementara analis manusia menyediakan pengetahuan domain dan pemahaman kontekstual yang penting untuk menyelidiki dan memvalidasi potensi ancaman.

Makalah ini menyajikan desain dan pengembangan sistem analisis log jaringan berbasis web yang dirancang untuk deteksi *real-time* terhadap ancaman canggih dan pergerakan lateral. Sistem yang diusulkan mengintegrasikan alur pemrosesan log (*log processing pipeline*) yang skalabel dengan teknik deteksi canggih berbasis perilaku. Sistem ini juga menyediakan dasbor visualisasi interaktif untuk memberdayakan analis keamanan dalam aktivitas perburuan ancaman dan respons insiden mereka. Kebaruan (*novelty*) dari penelitian ini terletak pada integrasi holistik dari komponen-komponen canggih (*state-of-the-art*) ini ke dalam satu sistem yang padu dan fungsional. Sistem ini menjembatani kesenjangan antara data log mentah dan intelijen keamanan yang dapat ditindaklanjuti.

2. TINJAUAN PUSTAKA

2.1. Tumpukan Observability Modern untuk Analisis Keamanan

Dalam dunia keamanan siber, data adalah landasan pertahanan. Namun, memiliki data saja tidak cukup. Data harus dikumpulkan, diproses, dan disajikan dengan cara yang tepat waktu dan mudah dipahami. Sistem manajemen log tradisional, meskipun kuat, seringkali membawa beban operasional (*overhead*) yang signifikan dalam hal biaya dan kerumitan. Hal ini telah membuka jalan bagi generasi baru tumpukan *observability open-source* yang lebih fleksibel, skalabel, dan hemat biaya. Contoh utama dari pendekatan modern ini adalah kombinasi Grafana, Loki, dan Prometheus [4]. Grafana berfungsi sebagai platform terpadu untuk *observability*. Grafana menawarkan antarmuka berbasis web yang kuat dan fleksibel untuk memvisualisasikan data dari berbagai sumber. Berbeda dengan sistem lama yang mungkin memerlukan alat terpisah untuk jenis data yang berbeda, Grafana menciptakan *single pane of glass* (satu panel tunggal) di mana analis keamanan dapat mengkorelasikan berbagai aliran informasi.

Ini membawa kita ke sumber data itu sendiri. Arsitektur tumpukan ini sangat sederhana namun kuat. Prometheus unggul dalam mengumpulkan dan menyimpan data *time-series* (data deret waktu), seperti penggunaan CPU, volume lalu lintas jaringan, dan penggunaan memori. Metrik-metrik ini sangat penting untuk memahami kesehatan dan kinerja sistem. Metrik ini juga bisa menjadi indikator pertama dari jenis serangan tertentu, seperti *Denial-of-Service* (DoS). Melengkapi Prometheus adalah Grafana Loki, sebuah sistem agregasi log yang terinspirasi oleh Prometheus. Loki mengambil pendekatan unik dan sangat efisien dengan tidak mengindeks konten penuh dari log. Sebaliknya, Loki hanya mengindeks sekumpulan label yang lebih kecil untuk setiap aliran log. Hal ini secara drastis mengurangi biaya penyimpanan dan meningkatkan kecepatan *ingestion* (pemasukan data). Log dikumpulkan dari sistem target oleh agen bernama Promtail. Promtail dirancang untuk menemukan dan meneruskan file log, seperti /var/log/auth.log atau log *firewall*, ke instansi pusat Loki. Kombinasi *backend* metrik (Prometheus) dan *backend* log (Loki), yang semuanya divisualisasikan dalam Grafana, membentuk alur data hibrida (*hybrid data pipeline*) yang menyediakan pandangan *real-time* dan komprehensif mengenai postur keamanan jaringan.

Pelengkap dari pendekatan tumpukan (*stack*) terintegrasi ini adalah praktik mendasar untuk menggunakan alat analisis khusus. Walidin misalnya, menyoroti penggunaan Kali Linux sebagai platform untuk analisis keamanan jaringan [5]. Secara spesifik, penggunaan Wireshark untuk menangkap dan menginspeksi lalu lintas secara detail berfungsi sebagai bentuk pencatatan log (*logging*) jaringan yang sangat granular (terperinci). Pendekatan *hands-on* (praktik langsung) ini memberikan pemahaman mendasar tentang aktivitas jaringan yang melengkapi pemantauan otomatis tingkat tinggi yang ditawarkan oleh tumpukan *observability* modern.

Evolusi tumpukan observability tidak berhenti pada visualisasi pasif, tetapi meluas menjadi fondasi untuk sistem otonom yang lebih cerdas. Penelitian terkini menunjukkan bahwa komponen seperti Prometheus bukan lagi hanya alat pemantauan, melainkan pemicu inti dalam arsitektur *AIOps* (*AI for IT Operations*) yang dapat beradaptasi dan memperbaiki diri (*self-healing*). Sebagai contoh, Elsayed mendemonstrasikan arsitektur infrastruktur virtualisasi yang ditingkatkan dengan AI, di mana metrik *real-time* dari Prometheus digunakan oleh agen AI untuk melakukan penskalaan sumber daya prediktif dan respons insiden cerdas secara otomatis [6]. Dalam model ini, platform otomasi *open-source* seperti n8n berfungsi sebagai orkestrator yang menerjemahkan data dari tumpukan *observability* menjadi tindakan konkret. Pergeseran ini secara fundamental mengubah peran analis keamanan, dari sekadar responden insiden menjadi arsitek sistem pertahanan otonom. Tugas mereka bergeser dari pemantauan dasbor secara manual ke perancangan, pelatihan, dan validasi logika yang mengatur agen AI, memastikan tindakan otonom selaras dengan kebijakan operasional yang ketat melalui teknik seperti *Retrieval-Augmented Generation* (RAG) untuk mencegah keputusan AI yang tidak terduga.

Saat membandingkan pendekatan dalam membangun sistem analisis keamanan *open-source*, terdapat spektrum antara arsitektur modular yang dapat disesuaikan, seperti

tumpukan Grafana-Loki-Prometheus, dan platform *Security Information and Event Management* (SIEM) terintegrasi seperti Wazuh. Wazuh menawarkan rangkaian kapabilitas yang luas "di luar kotak", termasuk analisis log, deteksi kerentanan dengan mengintegrasikan basis data eksternal seperti National Vulnerability Database (NVD), pemantauan integritas file, penilaian konfigurasi otomatis, dan modul respons aktif dengan skrip bawaan untuk mitigasi ancaman. Pendekatan terintegrasi ini mempercepat implementasi, namun pendekatan modular menawarkan fleksibilitas yang lebih besar untuk kustomisasi dan integrasi. Keberhasilan kedua pendekatan tersebut sangat bergantung pada kemampuan mereka untuk beroperasi dalam ekosistem keamanan yang lebih luas. Proyek penelitian seperti TestCat berfokus pada penciptaan lingkungan pengujian yang fleksibel untuk mengevaluasi sistem deteksi intrusi secara objektif [7], sementara inisiatif Eropa seperti NEWSROOM bertujuan untuk membangun platform *Cyber Situational Awareness* (CSA) yang kolaboratif. Hal ini menggarisbawahi bahwa nilai sejati dari sistem pemantauan modern tidak hanya terletak pada analisis data internal, tetapi juga pada kemampuannya untuk menyesuaikan data tersebut dengan intelijen ancaman eksternal yang dibagikan oleh komunitas [8].

2.2. Deteksi Ancaman Cerdas pada Data Log dan Metrik

Dengan aliran data log dan metrik yang terus mengalir ke sistem terpusat, tantangan berikutnya adalah memahami semua data tersebut. Di sinilah kekuatan kecerdasan buatan (AI) dan *machine learning* (ML) berperan. Volume data yang sangat besar yang dihasilkan oleh jaringan modern membuat analisis manual untuk deteksi ancaman menjadi tugas yang mustahil [9]. Akibatnya, bidang ini telah mengalami pergeseran signifikan menuju teknik analisis cerdas yang otomatis. Teknik ini dapat mengidentifikasi pola-pola aktivitas jahat yang samar, yang tersembunyi di antara data operasional normal (*noise*). Metode-metode ini bergerak melampaui keterbatasan sistem berbasis tanda tangan (*signature-based*) tradisional, yang hanya efektif melawan ancaman yang sudah dikenal. Metode ini menerapkan pendekatan yang lebih dinamis dan berorientasi pada perilaku [10].

Salah satu paradigma paling efektif di bidang ini adalah *User and Entity Behavior Analytics* (UEBA). Alih-alih mencari tanda tangan (*signature*) malware tertentu, sistem UEBA menggunakan machine learning untuk menetapkan *baseline* (garis dasar) perilaku normal untuk setiap pengguna dan perangkat di jaringan. *Baseline* ini dibangun dari berbagai sumber data, termasuk log sistem, lalu lintas jaringan, dan metrik penggunaan aplikasi. Setelah *baseline* ditetapkan, sistem akan terus memantau adanya penyimpangan. Tindakan yang secara statistik tidak mungkin terjadi (probabilitasnya rendah), misalnya pengguna masuk dari lokasi geografis baru pada waktu yang tidak biasa, atau server yang tiba-tiba memulai koneksi keluar dalam jumlah besar, akan ditandai sebagai potensi ancaman. Pendekatan ini sangat cocok untuk mendeteksi ancaman orang dalam (*insider threats*), akun yang disusupi, dan teknik tersembunyi yang digunakan dalam pergerakan lateral [2].

Teknik yang lebih canggih memanfaatkan model *deep learning*, seperti jaringan *Long Short-Term Memory* (LSTM) dan *Transformers*, untuk memahami sifat sekuensial (berurutan) dari data log. Model-model ini memperlakukan rangkaian peristiwa sistem seperti kalimat dalam bahasa alami. Hal ini memungkinkan mereka mempelajari "tata bahasa" dari operasi normal dan mendeteksi serangan sebagai rangkaian tindakan yang anomali [9]. Bidang lain yang menjanjikan adalah penggunaan pembelajaran berbasis graf (*graph-based learning*). Dalam pendekatan ini, aktivitas jaringan dimodelkan sebagai graf, di mana pengguna dan *host* adalah *node* (simpul) dan autentikasi adalah *edge* (penghubung). Model *machine learning* kemudian dapat mengidentifikasi pergerakan lateral dengan mendeteksi terciptanya edge yang berprobabilitas rendah, misalnya koneksi dari *workstation* di departemen pemasaran ke domain *controller* yang kritis [11]. Dengan mengkorelasikan teknik analitik canggih ini dengan data log (contoh: autentikasi berhasil dan gagal) dan data metrik (contoh: lonjakan penggunaan CPU), sistem dapat membangun gambaran fidelitas tinggi (sangat akurat) dari serangan yang sedang berlangsung, mulai dari

upaya brute-force hingga pergerakan lateral sesudahnya di dalam jaringan.

Untuk mengatasi tantangan skalabilitas dan privasi data yang melekat dalam analisis log terpusat, paradigma *Federated Learning* (FL) telah muncul sebagai pendekatan mutakhir. FL memungkinkan pelatihan model *machine learning* secara kolaboratif di berbagai perangkat atau silo data terdistribusi tanpa perlu memindahkan data mentah ke server pusat. Sebaliknya, setiap node melatih modelnya secara lokal, dan hanya pembaruan parameter model yang diagregasi secara terpusat. Haddad dan Anbar mengusulkan NAIIDS4IoT, sebuah arsitektur deteksi intrusi untuk lingkungan IoT yang menggunakan FL untuk melatih model *Deep Autoencoder* di setiap perangkat *edge* [12]. Arsitektur ini tidak hanya menjaga privasi data tetapi juga mengurangi beban komunikasi jaringan dan menggunakan teknologi blockchain untuk memverifikasi integritas pembaruan model. Demikian pula, Karunamurthy menunjukkan bahwa FL dapat secara efektif melatih *classifier deep learning* untuk IDS di lingkungan IoT, mengatasi ketergantungan pada dataset terpusat yang besar dan beragam [13]. Paradigma ini membuka kemungkinan strategis baru untuk keamanan siber kolaboratif antar-organisasi. Sebagai contoh, sebuah konsorsium lembaga keuangan dapat bersama-sama melatih model deteksi penipuan yang unggul dengan memanfaatkan data dari semua anggota tanpa pernah berbagi data transaksi rahasia mereka, menciptakan postur pertahanan kolektif yang lebih kuat terhadap ancaman baru [12] [13].

2.3. Visualisasi Interaktif dan Peringatan Proaktif

Kualitas sebuah sistem keamanan bergantung pada tindakan yang dapat dihasilkannya. Model deteksi AI paling canggih sekalipun tidak akan banyak berguna jika temuannya tidak dikomunikasikan secara efektif kepada analis manusia. Di sinilah peran dasbor visualisasi berbasis web menjadi sangat penting. Dasbor interaktif, seperti yang dibangun di Grafana, berfungsi sebagai antarmuka utama antara analis keamanan dan lautan data yang dikumpulkan oleh sistem. Dasbor ini lebih dari sekadar kumpulan grafik; ini adalah *workbench* (meja kerja) interaktif untuk *threat hunting* (perburuan ancaman) dan investigasi insiden [14].

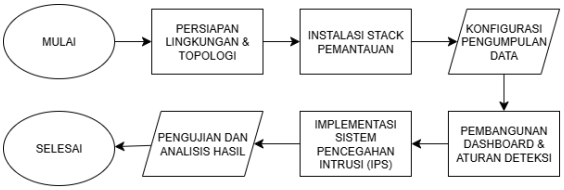
Visualisasi yang efektif menerjemahkan kumpulan data yang kompleks dan keluaran (output) algoritma menjadi wawasan yang jelas dan dapat ditindaklanjuti. Wawasan ini mendukung penalaran kognitif dan pengambilan keputusan manusia. Melalui dasbor interaktif, analis dapat menjelajahi hubungan antara titik data yang berbeda, mengkorelasikan peristiwa dari waktu ke waktu, dan melakukan *drill down* (penelusuran mendalam) dari peringatan tingkat tinggi ke entri log spesifik yang memicunya. Metodologi visual ini tidak hanya mempercepat deteksi ancaman, tetapi juga memperdalam pemahaman analis tentang lanskap ancaman secara keseluruhan. Dengan menyediakan antarmuka yang intuitif dan berpusat pada pengguna, sistem ini memberdayakan analis untuk menyelidiki dan memvalidasi potensi ancaman tanpa mereka harus menjadi seorang ilmuwan data [15].

Lebih jauh lagi, operasi keamanan modern sedang beralih dari postur yang murni pasif dan reaktif ke postur proaktif. Pendukung utama dari pergeseran ini adalah sistem peringatan otomatis dan cerdas. Daripada mengharuskan analis untuk terus-menerus mengawasi dasbor, sistem dapat dikonfigurasi untuk secara otomatis mengirimkan notifikasi ketika kondisi ancaman tertentu terpenuhi. Sebagai contoh, peringatan dapat dipicu jika jumlah upaya *login* yang gagal dari satu alamat IP melebihi ambang batas tertentu dalam waktu singkat (mengindikasikan serangan *brute-force*). Peringatan juga dapat dipicu jika penggunaan CPU server tetap tinggi secara kritis (tanda potensial serangan DoS atau aktivitas malware). Peringatan proaktif ini mengurangi *mean time to detect and respond* (waktu rata-rata untuk mendeteksi dan merespons). Hal ini membebaskan analis dari tugas pemantauan biasa dan memungkinkan mereka memfokuskan keahlian mereka untuk menyelidiki dan memitigasi ancaman yang telah terkonfirmasi [16]. Kombinasi antara visualisasi interaktif untuk investigasi mendalam dan peringatan otomatis untuk notifikasi segera ini menciptakan pertahanan berlapis yang kuat. Pertahanan ini sangat penting untuk memerangi ancaman siber canggih saat ini.

Pentingnya dasbor interaktif dalam sistem keamanan modern dapat diformalkan dalam kerangka akademis *Visual Analytics* dan *Software Security Visualization*. Bidang interdisipliner ini berfokus pada penggabungan algoritma penambangan data dengan representasi visual interaktif untuk mengubah data keamanan yang bervolume besar dan kompleks menjadi format yang dapat dicerna secara kognitif oleh manusia [17]. Dengan demikian, dasbor Grafana yang dikembangkan dalam penelitian ini bukan sekadar alat pelaporan, melainkan implementasi praktis dari prinsip-prinsip *Visual Analytics*. Dasbor ini berfungsi sebagai jembatan kognitif yang sangat penting, menghubungkan efisiensi pemrosesan data skala besar oleh mesin dengan keahlian domain, pemahaman kontekstual, dan kemampuan pengambilan keputusan yang unik dari analis manusia. Dalam konteks sistem keamanan yang semakin didorong oleh AI, peran visualisasi ini meluas lebih jauh. Ia menjadi mekanisme validasi dan *explainability* (XAI) yang krusial untuk model *machine learning* yang sering kali beroperasi sebagai "kotak hitam". Ketika sebuah model AI, seperti *Deep Autoencoder* atau *Random Forest*, menandai suatu anomali, analis tidak dapat hanya mempercayai hasilnya secara buta. Dasbor interaktif memungkinkan analis untuk melakukan *drill-down* dari peringatan tingkat tinggi ke entri log mentah, lonjakan metrik, atau pola lalu lintas jaringan yang mendasarinya. Proses ini memungkinkan analis untuk menginterogasi, memahami, dan pada akhirnya memvalidasi atau menolak keputusan yang dibuat oleh AI, yang merupakan komponen vital untuk adopsi AI yang bertanggung jawab dan dapat dipercaya dalam operasi keamanan.

3. METODE PENELITIAN

Sistem ini dikembangkan dan divalidasi melalui metodologi yang sistematis, sebagaimana ditunjukkan pada alur kerja penelitian di Gambar 1. Proses ini melibatkan serangkaian tahapan yang berurutan, dimulai dengan penyiapan lingkungan dan berpuncak pada pengujian dan analisis.



Gambar 1. Flowchart penelitian

Gambar 1 menunjukkan metodologi pengembangan sistem yang sistematis melalui beberapa tahapan penting. Fase awal, Persiapan Lingkungan & Topologi, mencakup penyiapan lab virtual untuk eksperimen. Tahap ini termasuk instalasi dua mesin virtual (VM) utama di dalam VirtualBox: sebuah Ubuntu Server untuk bertindak sebagai host target dan analisis, dan sebuah VM Kali Linux untuk berfungsi sebagai mesin penyerang. Topologi jaringan diatur untuk mendukung akses internet (guna instalasi perangkat lunak) melalui jaringan NAT, sekaligus lingkungan lab internal yang terisolasi (untuk simulasi serangan) melalui jaringan Host-Only.

Setelah penyiapan lingkungan selesai, fase Instalasi Tumpukan Pemantauan (Monitoring Stack) dilaksanakan. Tahap ini meliputi instalasi dan konfigurasi komponen perangkat lunak inti di Ubuntu Server: Grafana untuk visualisasi, Loki untuk agregasi log, Promtail sebagai agen pengumpul log, Prometheus untuk pengumpulan metrik, dan Node Exporter untuk menyajikan metrik sistem. Setiap komponen diatur agar berjalan sebagai system service (layanan sistem) untuk memastikan startup otomatis dan operasi yang berkelanjutan.

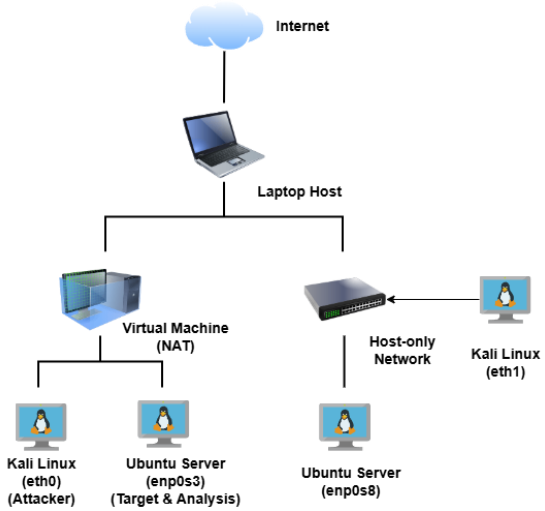
Pada tahap Konfigurasi Pengumpulan Data, para agen diatur untuk mengumpulkan data keamanan yang diperlukan. File promtail.yaml diubah untuk membaca dan meneruskan file log penting, khususnya auth.log (untuk peristiwa autentikasi) dan ufw.log (untuk aktivitas firewall). Pada saat yang sama, file prometheus.yml diatur untuk mengambil (scrape) data metrik time-series dari endpoint Node Exporter.

Kemampuan analisis inti dikembangkan dalam fase Pengembangan Dasbor & Aturan Deteksi. Di dalam Grafana, serangkaian panel visualisasi dibuat. Setiap panel ditenagai oleh query (kueri) analisis khusus yang ditulis dalam LogQL (untuk Loki) dan PromQL (untuk Prometheus). Query ini dirancang untuk mengidentifikasi pola ancaman yang telah

ditentukan sebelumnya, seperti upaya brute-force dan indikator pergerakan lateral.

Untuk menambahkan kemampuan respons aktif, dikembangkan pula Implementasi Intrusion Prevention System (IPS). Tahap ini meliputi pembuatan aplikasi web Python/Flask sederhana yang bertindak sebagai penerima webhook. Fitur peringatan (alerting) Grafana kemudian diatur untuk memicu webhook ke aplikasi ini ketika aturan deteksi ancaman terpenuhi. Pemicu ini selanjutnya akan menjalankan skrip untuk memblokir alamat IP penyerang secara otomatis di tingkat firewall.

Terakhir, fase Pengujian dan Analisis Hasil dilakukan untuk memvalidasi efektivitas sistem. Serangkaian serangan simulasi, termasuk SSH Brute Force dan aktivitas pergerakan lateral setelahnya, diluncurkan dari VM Kali Linux ke Ubuntu Server. Kemampuan sistem untuk mendeteksi serangan-serangan ini, menghasilkan notifikasi, dan menjalankan tindakan pemblokiran otomatis, kemudian diamati dan dianalisis melalui dasbor Grafana.



Gambar 2. Topologi jaringan dan lingkungan percobaan

Gambar 2 menyajikan representasi visual dari topologi jaringan virtual yang digunakan untuk pengujian dan analisis sistem. Seluruh lingkungan ini di-*host* di dalam VirtualBox pada satu laptop *host*. VM Kali Linux (192.168.56.103) ditetapkan sebagai penyerang, sementara VM Ubuntu Server (192.168.56.104) berfungsi sebagai target serangan sekaligus server pemantauan. Pengaturan ini memungkinkan simulasi serangan yang realistis dalam segmen jaringan yang terkendali dan terisolasi (192.168.56.0/24). Pengaturan ini sekaligus mengizinkan mesin *host* untuk mengakses antarmuka web Grafana guna melakukan pengamatan.

4. HASIL DAN PEMBAHASAN

4.1. Observabilitas Sistem Holistik melalui Visualisasi Data Hibrida

Dasbor utama dari sistem yang dikembangkan, ditunjukkan pada Gambar 3, berfungsi sebagai pusat komando terpusat bagi analis keamanan. Dasbor ini menyediakan gambaran umum (*overview*) yang holistik dan *real-time* mengenai status server. Dasbor ini menjadi contoh nyata dari pendekatan data hibrida yang digunakan sistem dengan mengintegrasikan dua jenis data yang berbeda namun saling melengkapi: metrik kinerja dari Prometheus dan log peristiwa keamanan dari Loki.

Gambar 3. Dasbor utama untuk pemantauan metrik kesehatan server

Panel untuk Penggunaan Memori (RAM), Penggunaan Disk, Penggunaan CPU, dan Lalu Lintas Jaringan menyediakan aliran berkelanjutan dari metrik kesehatan sistem. Ini adalah lapisan pertahanan pertama, karena lonjakan anomali pada metrik-metrik ini seringkali bisa menjadi indikator paling awal dari serangan siber. Sebagai contoh, peningkatan penggunaan CPU dan lalu lintas jaringan yang tiba-tiba dan berkelanjutan dapat menandakan serangan *Denial-of-Service* (DoS) atau upaya *brute-force* yang memakan banyak sumber daya. Dengan memantau tanda-tanda vital ini, analis dapat memperoleh kesadaran situasional langsung mengenai status operasional server.

Secara krusial, dasbor ini lebih dari sekadar pemantauan kinerja sederhana dengan mengkorelasikan metrik-metrik ini dengan

wawasan keamanan yang diambil dari log. Panel "Top Pengguna yang Mencoba Login" adalah contoh utamanya. Panel ini menganalisis log autentikasi (auth.log) untuk memvisualisasikan akun pengguna mana yang paling sering menjadi sasaran dalam upaya *login*. Hal ini mengubah data log mentah menjadi wawasan yang mudah dipahami, yang segera menarik perhatian analis ke potensi serangan *credential stuffing* atau *brute-force* yang menargetkan akun tertentu. Penggabungan data metrik dan log dalam satu panel tunggal (*single pane of glass*) ini sangat mendasar bagi desain sistem. Hal ini memungkinkan analis untuk beralih dengan cepat dari mengamati anomali kinerja (misalnya, CPU tinggi) ke mengidentifikasi potensi penyebab terkait keamanan (misalnya, volume *login* gagal yang tinggi).

4.2. Analisis Log Mendetail untuk Investigasi Forensik dan Perburuan Ancaman

Meskipun dasbor tingkat tinggi penting untuk kesadaran situasional, analisis keamanan yang efektif menuntut kemampuan untuk menggali data mentah lebih dalam (*drill down*). Gambar 4 menunjukkan kemampuan sistem untuk analisis log mendalam. Sistem ini menyediakan alat yang dibutuhkan analis, baik untuk investigasi forensik maupun perburuan ancaman (*threat hunting*) proaktif.

Gambar 4. Panel untuk analisis log mentah dan aktivitas sudo

Panel "LOG" di bagian atas menampilkan entri log mentah (raw) dan belum difilter dari /var/log/auth.log, persis saat data itu diterima (*ingest*) oleh Loki. Hal ini memberikan visibilitas *ground-truth* (data apa adanya) yang lengkap terhadap setiap peristiwa autentikasi yang terjadi di sistem. Bagi seorang analis keamanan, tampilan tanpa filter ini sangat berharga selama investigasi pasca-insiden. Tampilan ini memungkinkan mereka merekonstruksi linimasa aktivitas penyerang tanpa abstraksi (peringkasan) apa pun.

Namun, memilah ribuan entri log mentah bisa jadi tidak efisien untuk perburuan ancaman secara *real-time*. Untuk mengatasi hal ini, sistem menyediakan tampilan khusus yang sudah difilter (*pre-filtered*) untuk aktivitas berisiko tinggi. Panel "SUDO" adalah contoh nyata dari prinsip ini. Panel ini menggunakan *query* LogQL untuk memfilter aliran auth.log dan hanya menampilkan peristiwa saat sebuah perintah dieksekusi dengan hak akses (*privileges*) sudo. Tindakan ini segera mengisolasi (memisahkan) tindakan administratif. Tindakan ini sangat menarik perhatian karena mewakili potensi *privilege escalation* (peningkatan hak akses) atau perintah jahat yang dijalankan oleh akun yang telah dibobol. Kemampuan ganda ini—menyediakan tampilan *wide-angle* (sudut lebar) tanpa filter sekaligus tampilan *telescopic* (mendalam) yang terfilter—memberdayakan analis untuk bekerja secara efisien. Mereka dapat dengan cepat mengidentifikasi pola mencurigakan di panel yang telah dikurasi (dipilih) sebelum menyelam lebih dalam ke log mentah untuk mendapatkan bukti konklusif.

4.3. Deteksi Real-Time Serangan Brute-Force dan Pergerakan Lateral

Inti dari kemampuan deteksi ancaman sistem ditampilkan pada Gambar 5. Gambar ini berisi panel-panel yang dirancang khusus untuk mengidentifikasi dua tahap kritis serangan: akses awal melalui *brute force* dan pergerakan lateral (*lateral movement*) setelahnya.

Gambar 5. Panel untuk deteksi ancaman spesifik

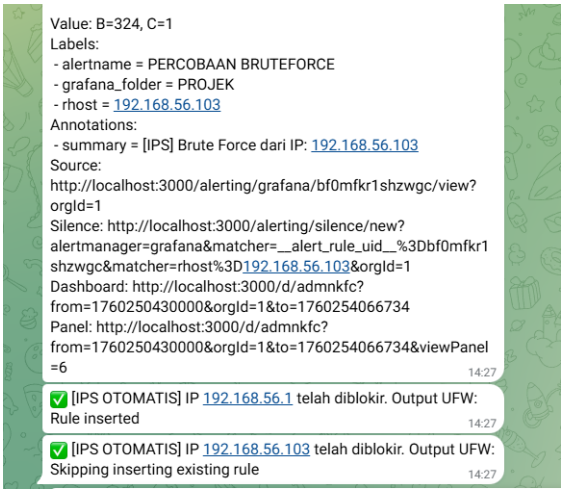
Panel "PERCOBAAN BRUTEFORCE" (*Brute Force Attempts*) berfungsi sebagai penghitung yang sederhana namun efektif. Panel ini ditenagai oleh *query* LogQL yang menghitung jumlah entri log berisi "Failed password" dalam rentang waktu tertentu. Hasil yang ditampilkan pada gambar—374 upaya gagal—adalah indikator yang jelas dan tegas

mengenai serangan *brute-force* yang sedang berlangsung terhadap layanan SSH. Metrik kuantitatif ini memberikan pemahaman sekilas (*at-a-glance*) dan langsung mengenai skala serangan.

Setelah penyerang mendapatkan akses awal, tujuan mereka berikutnya seringkali adalah bergerak di dalam sistem atau ke sistem lain di jaringan. Panel "Detektor Gerakan Lateral" (*Lateral Movement Detector*) dirancang untuk mendeteksi perilaku ini. Panel ini menghitung jumlah *login* sukses yang berasal dari sumber mencurigakan, seperti *localhost* atau alamat IP internal lainnya. Hal ini dapat mengindikasikan bahwa penyerang yang telah berhasil mendapatkan pijakan (*foothold*) sedang mencoba untuk berpindah (*pivot*) atau meningkatkan hak akses (*escalate privileges*). Panel "GERAKAN LATERAL" di bawahnya menyediakan entri log mentah yang sesuai untuk *login* yang sukses ini. Ini berfungsi sebagai bukti langsung, yang menunjukkan kepada analis secara pasti akun mana yang dibobol, IP sumber *login* tersebut, dan *timestamp* (penanda waktu) kejadiannya. Secara bersama-sama, panel-panel ini menciptakan narasi serangan. Panel ini memungkinkan analis melihat rentetan (*barrage*) *brute-force* awal, yang diikuti oleh pembobolan yang berhasil dan pergerakan internal setelahnya.

4.4. Respons Otomatis melalui Intrusion Prevention System (IPS)

Deteksi tanpa respons adalah postur keamanan yang tidak lengkap. Komponen final dan paling penting dari sistem ini adalah kemampuannya untuk beralih dari deteksi pasif ke pencegahan aktif. Gambar 6 menyajikan bukti definitif mengenai fungsionalitas *end-to-end* (menyeluruh) dari *Intrusion Prevention System* (IPS) yang terintegrasi.



Gambar 6. Bukti notifikasi dan tindakan pemblokiran otomatis oleh IPS

Gambar ini menunjukkan siklus respons otomatis yang lengkap. Ketika jumlah upaya *brute-force* yang terdeteksi di panel pada Gambar 5 melebihi ambang batas (*threshold*) yang telah ditentukan, *alerting engine* (mesin peringatan) Grafana akan terpicu. Pemicu ini mengirimkan *webhook* yang berisi alamat IP penyerang (*rhost*) ke aplikasi Python/Flask kustom (yang dibuat khusus). Tangkapan layar (*screenshot*) dari aplikasi Telegram menunjukkan bagian pertama dari respons ini: sebuah notifikasi langsung dikirimkan ke tim keamanan. Notifikasi ini memberikan kesadaran (*awareness*) *real-time* mengenai serangan dan IP sumber (192.168.56.103).

Hampir pada saat yang bersamaan, skrip Flask memproses informasi ini dan menjalankan (*executes*) perintah sistem untuk memperbarui aturan UFW (*Uncomplicated Firewall*) di server. Pesan kedua di tangkapan layar, "IP penyerang telah berhasil diblokir" (*The attacker's IP has been successfully blocked*), yang diikuti oleh keluaran (*output*) UFW "Rule inserted," mengonfirmasi bahwa alamat IP penyerang telah secara otomatis ditambahkan ke *deny list* (daftar tolak) *firewall*. Tindakan ini secara efektif memutus akses penyerang dari server, memitigasi ancaman secara *real-time* tanpa memerlukan intervensi manual apa pun dari seorang analis. Pengujian yang berhasil ini memvalidasi kemampuan sistem, tidak only untuk mendeteksi dan memvisualisasikan ancaman, tetapi juga untuk merespons ancaman tersebut secara aktif dan otonom. Hal ini secara signifikan mengurangi

window of opportunity (jendela peluang) bagi penyerang.

5. KESIMPULAN

Penelitian ini berhasil menjawab tantangan yang diuraikan di pendahuluan. Tantangan itu adalah kebutuhan akan sistem terintegrasi yang praktis untuk deteksi ancaman *real-time*, yang melampaui fokus teoretis pada algoritma yang terisolasi. Kontribusi utama dari penelitian ini adalah desain, implementasi, dan validasi platform analisis keamanan *end-to-end* (menyeluruh) yang lengkap. Platform ini dibangun seluruhnya di atas tumpukan *observability open-source* yang modern. Dengan mengintegrasikan Grafana, Loki, dan Prometheus secara holistik, studi ini menunjukkan model yang layak dan hemat biaya. Model ini menjembatani kesenjangan penting antara data log mentah bervolume tinggi dan intelijen keamanan yang dapat ditindaklanjuti. Sistem ini berhasil mengubah entri log dan metrik sistem yang terpisah-pisah menjadi dasbor yang padu dan berpusat pada analis. Dasbor ini mampu memberikan kesadaran situasional (*situational awareness*) secara langsung.

Hasil dari skenario serangan simulasi mengonfirmasi bahwa sistem ini memenuhi tujuan utamanya. Sistem ini secara efektif menyediakan *observability* hibrida dengan mengkorelasikan metrik kinerja dengan log keamanan. Hal ini memungkinkan visualisasi dampak serangan terhadap sumber daya sistem. Mesin deteksi berbasis aturan (*rule-based*) terbukti mampu mengidentifikasi upaya akses *brute-force* awal dan pergerakan lateral setelahnya secara *real-time*. Yang paling penting, implementasi sukses dari komponen *Intrusion Prevention System* (IPS) otomatis memvalidasi kemampuan sistem untuk menyelesaikan siklus keamanan penuh. Siklus ini mulai dari *ingestion* (pemasukan) data dan analisis, hingga deteksi, peringatan (*alerting*), dan mitigasi ancaman otonom. Hal ini menunjukkan pergeseran nyata dari postur pemantauan pasif ke strategi pertahanan aktif. Ini mengurangi *mean time to respond* (MTTR) secara drastis.

Meskipun hasil-hasil ini sukses, penelitian ini memiliki beberapa keterbatasan yang membuka jalan untuk penelitian di masa depan. Sistem ini divalidasi di lingkungan virtual berskala kecil dan terkendali. Kinerja dan skalabilitasnya di bawah beban jaringan perusahaan skala besar belum dievaluasi. Selain itu, logika deteksi bergantung pada *query* LogQL dan PromQL yang telah ditentukan sebelumnya. Meskipun efektif untuk pola yang dikenal, *query* ini kurang mampu mengidentifikasi ancaman baru atau *zero-day*. Penelitian selanjutnya harus berfokus pada peningkatan kecerdasan sistem. Caranya dengan mengintegrasikan model *machine learning* dan *deep learning* canggih untuk deteksi anomali yang sesungguhnya. Penerapan *User and Entity Behavior Analytics* (UEBA) akan memungkinkan sistem mempelajari *baseline* (garis dasar) perilaku dan mendeteksi penyimpangan yang signifikan secara statistik. Sementara itu, analisis berbasis graf (*graph-based*) dapat menawarkan cara yang lebih intuitif untuk memvisualisasikan dan melacak jalur pergerakan lateral yang kompleks. Sebagai tambahan, kemampuan respons otomatis dapat diperluas melampaui pemblokiran IP. Respons ini dapat mencakup tindakan yang lebih canggih, seperti isolasi *endpoint* atau penangguhan (*suspension*) akun pengguna. Dengan membangun di atas fondasi terintegrasi ini, penelitian di masa depan dapat menciptakan sistem pemantauan keamanan yang lebih cerdas, adaptif, dan tangguh.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada semua pihak yang telah memberikan dukungan dalam penyelesaian penelitian ini. Terima kasih disampaikan kepada dosen pembimbing dan rekan-rekan di lingkungan akademik yang telah memberikan masukan, saran, serta motivasi selama proses merancang dan membangun Sistem Analisis Log Jaringan Berbasis Web untuk Deteksi Real-Time Ancaman Canggih dan Gerakan Lateral. Ucapan terima kasih juga ditujukan kepada institusi yang telah menyediakan fasilitas penelitian serta lingkungan yang kondusif untuk pengembangan sistem ini. Tanpa dukungan moral, teknis, dan akademik dari berbagai pihak, penelitian ini tidak akan terselesaikan dengan baik.

DAFTAR PUSTAKA

[1] Delvita aulia artika, D., Daniel Rumahorbo, Muhammad haikal Al-Majid, & Dedy Kiswanto. (2025). IMPLEMENTASI SISTEM KEAMANAN WEBSITE DENGAN ANALISIS LOG DAN DETEKSI AKTIVITAS ANOMALI MENGGUNAKAN ISOLATION FOREST. *Jurnal Informatika Dan Teknik Elektro Terapan*, *13*(3S1). https://doi.org/10.23960/jitet.v13i3S1.8133

[2] Smiliotopoulos, C., Kambourakis, G., & Kolias, C. (2024). Detecting lateral movement: A systematic survey. *Heliyon, 10*(4), e26317. https://doi.org/10.1016/j.heliyon.2024.e26317

[3] Mohamed, A., Al-Shaer, E., & El-Sayed, H. (2024). LMDG: Advancing Lateral Movement Detection Through High-Fidelity Dataset Generation. *arXiv preprint arXiv:2405.12345*.

[4] Haridas, S. (2024). *Centralized Log Monitoring with Loki, Promtail, & Grafana*. Medium.

[5] Walidin, A. P., Putri, F. P., & Kiswanto, D. (2025). Kali Linux sebagai alat analisis keamanan jaringan melalui penggunaan Nmap, Wireshark, dan Metasploit. *JATI (Jurnal Mahasiswa Teknik Informatika)*, *9*(1), 1188-1196.

[6] Elsayed, M. (2024). *Enhancing hosting infrastructure management with AI-powered automation*. Theseus. https://www.theseus.fi/handle/10024/882571

[7] Skopik, F. (2024). *TestCat - Automated Testbeds for the Evaluation of Intrusion Detection Capabilities*. AIT Austrian Institute of Technology. https://www.skopik.at/

[8] Initmax. (2024). *Wazuh SIEM for security monitoring*. Initmax. https://www.initmax.com/wazuh-siem-for-security-monitoring/

[9] Makinde, M., & Aishat, T. (2025). *Automated Threat Hunting Using Natural Language Processing (NLP) on Security Logs*. ResearchGate.

[10] Chowdhury, A. R., Rahman, M. M., & Islam, M. S. (2024). The Integration of Machine Learning and Deep Learning for Advanced Intrusion Detection Systems: A Comprehensive Survey. *International Journal of Computer Applications, 186*(58), 1-10.

[11] Bian, S., Hu, Y., Liu, A. X., & Boutaba, R. (2021). Uncovering Lateral Movement Using Authentication Logs. *IEEE Transactions on Network and Service Management, 18*(4), 4593-4606. https://doi.org/10.1109/TNSM.2021.3108605

[12] Haddad, M. J., & Anbar, M. (2025). NAIIDS4IoT: A Novel Artificial Intelligence-Based Intrusion Detection Architecture for the Internet of Things. *Inteligencia Artificial*, *28*(76), 253-271.

[13] Karunamurthy, A., Vijayan, K., Kshirsagar, P. R., & Tan, K. T. (2024). An optimal federated learning-based intrusion detection for IoT environment. *Scientific Reports*, *14*(1), Article 29381. https://doi.org/10.1038/s41598-024-80387-9

[14] Forensic Visualization Toolkit. (2025). *Enhancing Cyber Threat Hunting — A Visual Approach with the Forensic Visualization Toolkit*. ResearchGate.

[15] Zhu, Y., Wang, F., & Zhang, J. (2024). VisualSphere: A Web-Based Interactive Visualization System For Clinical Research Data. *Journal of Biomedical Informatics, 150*, 104521. https://doi.org/10.1016/j.jbi.2024.104521

[16] Illumio Inc. (2025). *Lateral Movement in Cyberattacks Continues to Evade Detection, Exposing Critical Visibility Gaps, Illumio Research Finds*. Illumio News.

[17] Wagner, M., & Rind, A. (2024). Software security visualization: A systematic literature review. *IEEE Transactions on Visualization and Computer Graphics*, *30*(1), 123-134.