Jurnal Informatika Terpadu

https://journal.nurulfikri.ac.id/index.php/JIT
ISSN ONLINE : 2460-8998

PENGEMBANGAN ANTARMUKA WEB ANALITIK *LOG* DETEKSI INTRUSI JARINGAN BERBASIS SURICATA MENGGUNAKAN DASH

Nikita Putri Berniawan[1], Henry Saptono[2], Efrizal Zaida[3]

[1,2,3]Teknik Informatika, Sekolah Tinggi Teknologi Terpadu Nurul Fikri
Jakarta Selatan, DKI Jakarta, Indonesia 12640
niki21118ti@student.nurulfikri.ac.id, henry@nurulfikri.ac.id, efrizal@nurulfikri.com

Abstract

The rapid growth of internet usage correlates with an increasing risk of network security threats. Attacks on network traffic may result in confidential data breaches and system disruptions. Suricata, a powerful Intrusion Detection System (IDS) tool, is used to generate rich detection logs. However, raw log data in JSON format remains difficult to analyze and interpret directly due to its complexity and large volume. This study proposes the development of a web-based application using the Dash framework to visualize intrusion detection results from Suricata. Dash is capable of presenting data in an interactive and informative manner through various components such as histograms, line charts, tables, and filter features. The purpose of this research is to assess Dash's effectiveness in presenting intrusion data in a format that is accessible and easily interpreted by users. Evaluation results show that the Dash framework successfully visualized 24,526 alerts out of a total of 4,247,464 logs accurately. The application was also able to display all information components comprehensively and interactively. Thus, this application can contribute to improving both network security and operational efficiency.

Keywords: *Dash, Intrusion Detection System, Network Security, Suricata, Visualization*

Abstrak

Meningkatnya volume penggunaan internet, berbanding lurus dengan ancaman keamanan jaringan. Serangan pada lalu lintas jaringan dapat menyebabkan kebocoran informasi konfidensial dan gangguan terhadap sistem. Suricata sebagai alat *Intrusion Detection System* (IDS) yang andal, digunakan dalam menghasilkan *log* deteksi yang kaya akan informasi. Namun, data *log* mentah dalam format JSON masih sulit dianalisis dan dipahami secara langsung karena kompleksitas dan volumenya yang besar. Penelitian ini membahas tentang pengembangan aplikasi web berbasis Dash untuk menampilkan visualisasi hasil deteksi intrusi jaringan pada Suricata. *Framework* Dash dapat menampilkan data secara interaktif dan informatif melalui berbagai elemen, seperti grafik histogram, grafik garis, tabel, dan fitur filter. Tujuan dari penelitian ini untuk menguji efektivitas Dash dalam menyajikan data deteksi intrusi yang mudah dipahami oleh *user*. Hasil evaluasi yang didapat setelah pengerjaan serangkaian proses, menunjukkan bahwa *framework* Dash berhasil menampilkan 24.526 *alert* dari total 4.247.464 *log* secara akurat. Aplikasi juga dapat berhasil menampilkan keseluruhan komponen informasi secara lengkap dan interaktif. Dengan demikian, aplikasi ini dapat berkontribusi dalam mendukung peningkatan efisiensi dan keamanan jaringan.

Kata kunci: Dash, Deteksi Ancaman, *Intrusion Detection System*, Suricata, Visualisasi Web

1. PENDAHULUAN

Peran internet di zaman teknologi ini telah mengalami transformasi yang sangat signifikan, dari yang sebelumnya hanya sebagai sarana pendukung, kini telah menjadi bagian integral dari aktivitas warga Indonesia. Peningkatan pengguna internet semakin memperkuat fakta tersebut. Berdasarkan Asosiasi Penyelenggara Jasa Internet Indonesia (APJII), jumlah pengguna internet Indonesia di tahun 2024 mencapai 221.563.479 jiwa dari total populasi 278.696.200 jiwa penduduk Indonesia tahun 2023 atau setara dengan 79,50% dari total populasi. APJII juga memprediksi jumlah pengguna internet pada tahun ini bertambah sekitar 6 juta pengguna [1].

Berbanding lurus dengan pertumbuhan teknologinya, risiko yang dihadapi untuk menjaga keamanan informasi juga meningkat tajam. Semakin maraknya serangan siber seperti *phishing, ransomware*, dan *Distributed Denial of Service (*DDoS*)* dapat menjadi bukti [2]. Dimana, serangan tersebut dapat berdampak pada kebocoran data mengenai informasi konfidensial, detail bisnis, dan informasi pelanggan dapat dengan mudah diakses melalui berbagai situs [3].

Diterima 12 November 2025, Direvisi 13 November 2025, Diterima untuk publikasi 25 November 2025

Berdasarkan laporan *Interpol Cyber Assessment Report* 2021, terdapat sekitar 2,7 juta serangan *ransomware* yang terdeteksi di Asia Tenggara pada, dengan Indonesia berada di peringkat teratas dengan 1,3 juta kasus. Pada tahun 2022, terdapat 8.831 kasus kejahatan siber yang ditindak, meningkat hingga 14 kali lipat dibandingkan dengan tahun 2021. Selain itu, riset dari Fortinet mengungkapkan bahwa serangan siber *ransomware* di Indonesia meningkat dua kali lipat selama tahun 2023 [4]. Dan selama semester pertama tahun 2024, total seluruh serangan siber di Indonesia mencapai 2,5 miliar. Yang berarti Indonesia mengalami rata-rata 158 serangan siber per detik [5].

Demi mencegah terjadinya serangan siber, juga untuk melindungi data sensitif dan menjaga integritas infrastruktur digital, *Intrusion Detection System* (IDS) menjadi garis pertahanan yang sangat penting. Sistem ini dirancang untuk memonitor lalu lintas jaringan dan mendeteksi anomali dalam aktivitas yang berpotensi menandakan terjadinya serangan siber [6]. Dengan memantau dan menganalisis lalu lintas, IDS dapat memberikan peringatan dini kepada tim keamanan untuk mengambil tindakan yang diperlukan dan meminimalkan dampak serangan.

Salah satu jenis IDS *open-source* yang populer adalah Suricata. Suricata merupakan *software* berupa sistem deteksi dan pencegahan intrusi yang memiliki kemampuan untuk mendeteksi serta mencegah serangan pada lalu lintas jaringan [7]. Untuk mempermudah pemantauan dan analisis terhadap *log* serangan yang dihasilkan Suricata, dibutuhkan web analisis dengan *dashboard* yang interaktif dan *real-time*. Dash, sebagai salah satu *framework* Python, menawarkan kemudahan dalam membangun aplikasi web interaktif dengan elemen antarmuka pengguna web, layaknya visualisasi data, yang lengkap [8]. Sehingga, tim keamanan dapat dengan cepat mengidentifikasi pola serangan dan mengambil tindakan yang diperlukan.

Penelitian ini bertujuan untuk mengevaluasi efektivitas IDS berbasis Suricata dalam mendeteksi ancaman siber serta mengembangkan *dashboard* analisis berbasis Dash untuk mempermudah visualisasi *log* serangan. Dengan pendekatan yang menggabungkan analisis data sekunder dan simulasi sistem, diharapkan penelitian ini dapat memberikan solusi yang efektif dalam meningkatkan kemampuan deteksi ancaman siber secara cermat dan dapat membantu organisasi dalam melindungi aset digital mereka dari berbagai ancaman siber yang terus berkembang.

Web

Web merupakan sebuah sistem informasi berupa ekosistem digital yang luas, yang memanfaatkan berbagai format berkas untuk menampilkan teks, gambar, multimedia interaktif berupa animasi dan video, serta visualisasi data berupa grafik dan diagram yang statis maupun dinamis.

Lalu Lintas Jaringan

Merupakan jumlah data yang bergerak melintasi jaringan pada waktu tertentu. Perangkat komunikasi di setiap harinya bertugas untuk mengakses sumber informasi, menerima *request* untuk melaksanakan pekerjaan lain, dan merespons *request* yang telah diterima tadi. Dalam pertukaran informasi tersebut, terdapat data dalam bentuk sejumlah besar paket yang beredar di dalam jaringan.

Serangan Siber

Serangan siber merupakan setiap tindakan siber yang ilegal yang bertujuan untuk melanggar kebijakan keamanan suatu aset siber dan menyebabkan kerusakan, gangguan, atau penghentian layanan maupun penghambatan akses ke informasi yang terdapat dalam akses tersebut.

IDS

IDS merupakan sistem yang memonitor aktivitas dari lingkungan spesifik, seperti lalu lintas jaringan dan *syslog records*, untuk menentukan apakah aktivitas tersebut merupakan aktivitas yang sah atau merupakan gejala dari suatu serangan [9]. IDS berfokus pada mendeteksi kejadian yang berpotensi terjadi, menyimpan informasi mengenai kejadian tersebut, dan melaporkan informasi tersebut ke administrasi keamanan.

Suricata

Suricata merupakan sebuah aplikasi jaringan yang dapat memantau *log* dan memeriksa setiap *node* jaringan untuk memverifikasi apakah ada lalu lintas mencurigakan yang melewati jaringan. Suricata akan bertindak sebagai *third-party* mendukung fungsi *firewall* untuk mendeteksi aktivitas dalam lalu lintas jaringan setiap waktu [10]. Suricata memiliki beberapa fungsi, diantaranya menganalisis lalu lintas jaringan, mendeteksi dan mencegah terjadinya intrusi, menganalisis protokol, memonitor arus jaringan, dan pembuatan *log* [11].

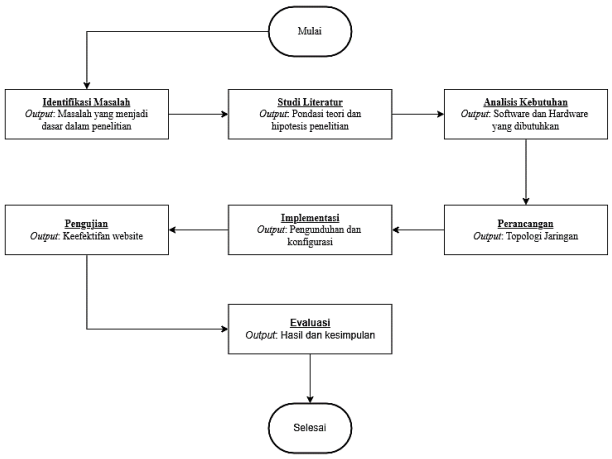
Dash

Dash merupakan *framework open source* yang memiliki fungsi untuk membuat *website* yang interaktif, responsif, dan dinamis menggunakan bahasa Python. *Framework* ini berperan sebagai penghubung antara kemampuan Python dalam analisis data dengan teknologi web modern. Komponen dalam Dash umumnya digunakan untuk membuat *tools* seperti, *drop down*, grafik, dan komponen web lainnya sehingga pengguna dapat berinteraksi dan mengeksplorasi data secara mendalam [12].

2. METODE PENELITIAN

2.1 Tahapan Penelitian

Bab ini akan menjelaskan secara sistematis mengenai tahapan penelitian dimulai dari identifikasi masalah hingga penarikan kesimpulan dari evaluasi yang didapat. Tahapan-tahapan penelitian ditunjukkan pada gambar 1 berikut:



Gambar 1. Tahapan Penelitian

a) Identifikasi Masalah

Proses penelitian ini dimulai dengan melakukan pengidentifikasian masalah utama yang akan diteliti secara mendalam. Pada tahapan ini dilakukan analisis secara terperinci mengenai permasalahan memvisualisasikan data pada sistem deteksi intrusi jaringan. Proses ini dilakukan agar penelitian lebih terfokus sehingga tujuan dapat disusun secara spesifik.

b) Studi Literatur

Proses ini berupa pengumpulan informasi dari berbagai sumber, seperti jurnal, buku, dan riset terdahulu, mengenai konsep Dash, sistem intrusi jaringan, dan Suricata. Melalui tahapan ini didapatkan pemahaman bahwa Dash menjadi salah satu *framework* dari Python yang banyak digunakan untuk membuat web berisi visualisasi dan analisis data. Sementara itu, sistem intrusi jaringan berperan penting dalam mendeteksi serangan yang terjadi pada lalu lintas jaringan. Dan Suricata sebagai aplikasi sistem intrusi jaringan, untuk divisualisasikan menggunakan Dash.

c) Analisis Kebutuhan

Tahapan analisis kebutuhan berfungsi untuk menentukan kebutuhan yang akan digunakan dalam penelitian ini. Kebutuhan tersebut mencakup *dataset*, sistem seperti *software* dan *hardware*, serta *library*, *framework*, dan *tools* untuk membangun program yang akan dijalankan. Dalam penelitian ini, *dataset* yang dibutuhkan berupa *log* serangan yang terjadi dalam lalu lintas jaringan dalam seminggu, *hardware* berupa laptop dan *software* yang mendukung kerjanya penelitian ini, serta *tools* dan *framework* berupa Suricata dan Dash.

d) Perancangan

Pada tahap ini, sistem pemantauan dirancang menggunakan arsitektur IDS sehingga dapat terintegrasi secara efektif dengan Dash. Desain yang dihasilkan mencakup perancangan topologi jaringan, pengumpulan data dari lalu lintas jaringan menggunakan Suricata, serta visualisasi data menggunakan Dash.

e) Implementasi

Tahap ini akan mengaplikasikan desain sistem sehingga dapat beroperasi. Tahap ini mencakup pengunduhan dan konfigurasi Suricata pada mesin virtual untuk memantau lalu lintas, membuat atau mengunduh *rule* yang sesuai untuk mendeteksi ancaman, serta mengintegrasi Suricata dengan Dash agar data yang dikumpulkan dan dianalisis oleh Suricata dapat divisualisasikan menggunakan Dash.

f) Pengujian

Pada tahap pengujian, sistem akan diuji berulang kali guna memastikan bahwa penerapan sistem dalam tahap implementasi dapat berfungsi dengan optimal. Aspek yang menjadi fokus perhatian meliputi keberhasilan Suricata dalam mendeteksi segala anomali yang ada dan keefektifan Dash dalam menampilkan visualisasi data yang didapat dari analisis Suricata.

g) Evaluasi

Tahapan evaluasi dilakukan untuk menilai dan menganalisis secara menyeluruh hasil pengujian yang telah dilakukan pada tahap sebelumnya. Proses ini bertujuan untuk menilai apakah Dash berhasil menampilkan visual dari analisis serangan, serta Suricata mampu memantau lalu lintas jaringan dengan baik.

2.2 Metode Pengumpulan Data

Sumber data utama diambil dengan menggunakan metode eksperimen, diantaranya dengan menguji web dengan memberikan berbagai serangan dan data berupa *log* aktivitas serangan lalu lintas menggunakan Suricata. Yang nantinya akan dilakukan serangkaian proses otomatisasi sistem untuk mendapati data akhir yang keberfungsian dan konsistensinya sesuai dengan tujuan dan manfaat penelitian yaitu perancangan sistem yang efektivitasnya terjamin.

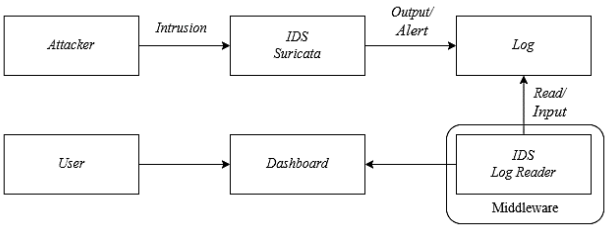
2.3 Metode Pengujian

Dalam penelitian ini, dilakukan dua metode dalam pengujian sebagai bagian dari kerangka penelitian, yaitu *Black Box testing* yang berfokus dalam mengevaluasi keakuratan *rule* yang digunakan dalam mendeteksi serangan, serta uji fungsionalitas, di mana Dash sebagai *framework* web analisis akan diuji keefektifannya dalam menampilkan data *log* serangan yang telah dihasilkan sebelumnya. Pengujian ini mencakup kemampuan *website* dalam menyajikan data secara akurat dan interaktif.

3. HASIL DAN PEMBAHASAN

3.1 Perancangan

a) Arsitektur Sistem

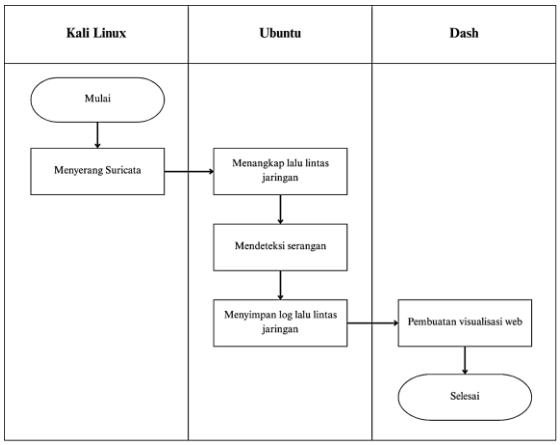


Gambar 2. Arsitektur Sistem

Gambar 2 menggambarkan rancangan arsitektur sistem yang dirancang untuk memastikan bahwa intrusi jaringan yang terdeteksi oleh Suricata dapat dikumpulkan dalam bentuk *log alert* dan disajikan dengan visualisasi yang informatif serta mudah dipahami dengan komponen sebagai berikut:

- *Attacker* sebagai penyerang yang digunakan untuk menguji efektivitas sistem deteksi intrusi Suricata.
- IDS Suricata yang bertugas untuk memonitor lalu lintas jaringan, dengan mendeteksi pola serangan dan mencatat aktivitas mencurigakan tersebut dalam bentuk *log alert*.
- *Log* hasil deteksi yang berisi sumber IP, tujuan IP, waktu insiden, identifikasi serangan, dan lainnya.
- IDS *Log Reader* yang berfungsi untuk membaca dan memproses *log alert* yang dihasilkan oleh Suricata. Pada penelitian ini, implementasi fungsi *log reader* terintegrasi ke dalam *framework* Dash, sehingga proses penyusunan data untuk visualisasi dilakukan secara langsung dalam modul *dashboard*.
- *Dashboard* yang merupakan antarmuka visual yang menampilkan hasil analisis data.
- *User* yang merupakan pengguna akhir sistem yang akan berinteraksi langsung dengan visual antarmuka untuk melakukan analisis terhadap hasil dari deteksi intrusi tersebut.

b) Alur Kerja Sistem



Gambar 3. Diagram Alur Kerja Sistem

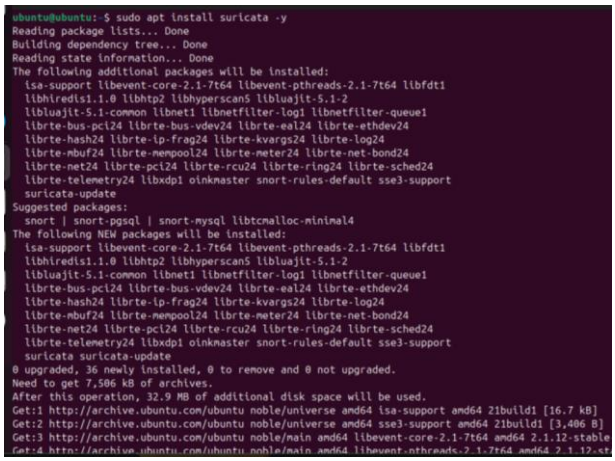
Berdasarkan gambar 3, proses sistem dimulai dengan melakukan simulasi serangan di Kali Linux terhadap Suricata untuk menguji efektivitas *rule* yang telah dikonfigurasikan pada Suricata. Suricata akan menganalisis lalu lintas yang keluar-masuk dan membandingkannya dengan *rule* yang tersedia. Jika ditemukan pola yang sesuai dengan *rule,* Suricata akan mengidentifikasikannya sebagai potensi serangan. Aktivitas yang mencurigakan itu nantinya akan disimpan ke dalam sebuah *file log* bernama eve.json. *File log* tersebut berisi informasi mengenai IP sumber dan tujuan serangan, waktu terjadinya serangan, tipe serangan, juga tingkat keparahannya.

Log tersebut akan diproses oleh aplikasi berbasis *framework* Dash. Dalam proses yang mencakup pembacaan, pengelompokan, dan visualisasi. Dash akan menampilkan informasi dalam bentuk antarmuka web interaktif.

3.2 Implementasi

a) Instalasi

Pengunduhan Suricata dilakukan pada sistem operasi Ubuntu menggunakan perintah `sudo apt install suricata` yang ditunjukkan pada gambar 4 di bawah.



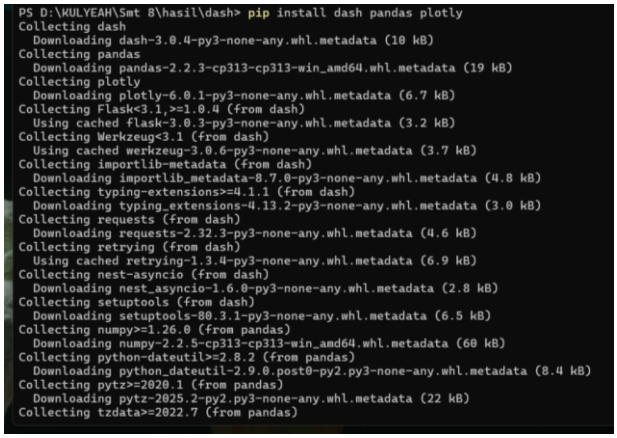
Gambar 4. Instalasi Suricata

Secara *default*, Suricata telah dilengkapi dengan *rule* bawaan. Namun, dalam konteks penelitian ini, untuk meningkatkan kemampuan sistem dan proses deteksi serangan yang lebih kompleks, diperlukan instalasi *rule* tambahan melalui perintah `sudo wget https://rules.emergingthreats.net/open/suricata-7.0.3/emerging.rules.tar.gz` seperti pada gambar 5 di bawah

Gambar 5. Instalasi *Rule* Suricata

Untuk memvisualisasikan *log* serangan yang dihasilkan oleh Suricata dalam bentuk antarmuka yang informatif dan

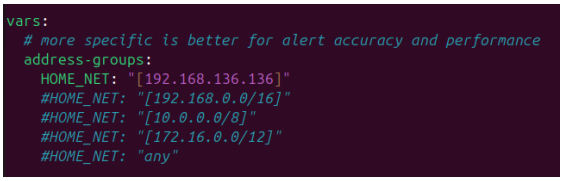
interaktif, digunakan *framework* Dash. Untuk dapat menggunakannya, diperlukan instalasi *library* Dash dan beberapa *library* tambahan seperti Pandas dan Plotly. Instalasi tersebut menggunakan perintah `pip install dash pandas plotly` seperti pada gambar 6 di bawah.



Gambar 6. Instalasi Dash

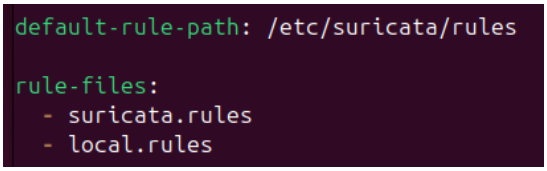
b) Konfigurasi

Konfigurasi yang dilakukan mencakup pengaturan alamat IP, pemilihan *rule* yang akan digunakan, dan lokasi *output log* serangan. Dimana, seluruh konfigurasi tersebut dilakukan di direktori /etc/suricata/suricata.yaml. Untuk dapat mengedit di dalam direktori tersebut, menggunakan perintah `sudo nano /etc/suricata/suricata.yaml` seperti pada gambar 7 di bawah.



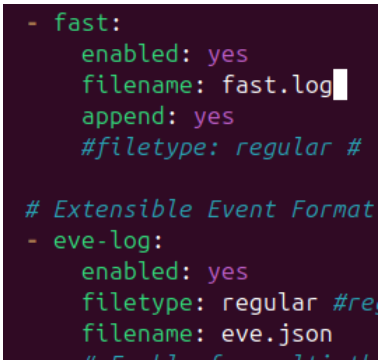
Gambar 7. Konfigurasi Alamat IP

Nilai pada parameter `HOME_NET`, diisi dengan target pemantauan lalu lintas yaitu alamat IP Ubuntu. Sehingga dapat melakukan pemantauan lalu lintas jaringan yang keluar dan masuk ke dalam alamat IP tersebut.



Gambar 8. Konfigurasi *Rule* Suricata

Gambar 8 menunjukkan *default-rule-path* yang merupakan direktori tempat *file rule* berada. Sedangkan, *rule-files* merupakan kumpulan dari nama-nama *file rule* yang akan digunakan oleh Suricata. Kedua parameter tersebut harus didefinisikan secara tepat dan lengkap, sehingga Suricata dapat mengakses dan memuat seluruh *rule* yang diperlukan untuk mendeteksi potensi ancaman dalam lalu lintas jaringan secara maksimal.



Gambar 9. Konfigurasi *Output Log* Serangan

Output yang dihasilkan berupa *log* yang tersimpan dalam *file* eve.json dan fast.log yang ditunjukkan pada gambar 9. Untuk menampilkan ringkasan *alert* secara cepat dan ringan, *file log* dapat dilihat di direktori /var/log/suricata/fast.log. Sementara itu, *log* dalam format JSON, dapat dilihat pada direktori /var/log/suricata/eve.json. Oleh karena itu, diperlukan konfigurasi untuk mengaktifkan *output log* eve.json, agar dapat diproses lebih lanjut dalam bentuk visual.

3.3 Pengujian

a) Simulasi Serangan

Tabel 1. Serangan

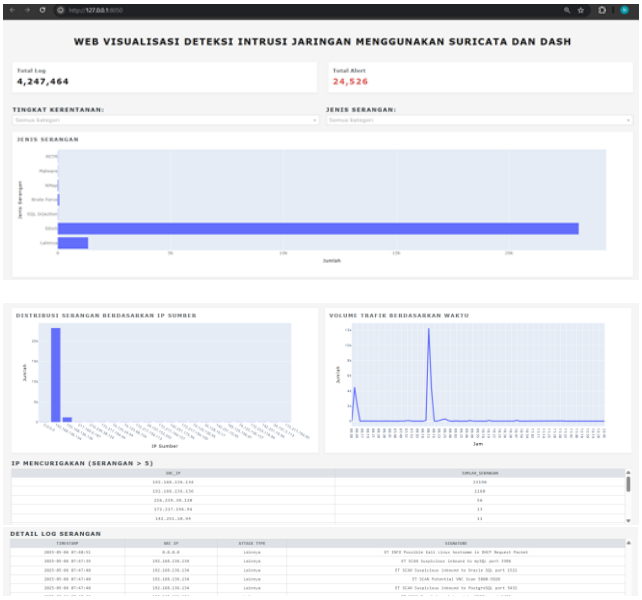
Serangan	***Script***

Malware	`UBUNTU: nc -lvp 444`
	`KALI: echo "bash -i >& /dev/tcp/192.168.136.136/4440&1" > shell.sh`
	
	`curl -X POST -d @shell.sh http://192.168.136.136/upload`

DDoS	`hping3 -S -p 80 --flood 192.168.136.136 --rand-source`
Brute Force	`hydra -l ftpuser -P /usr/share/wordlists/rockyou.txt ftp://192.168.136.136`
SQL *Injection*	`sqlmap -u http://192.168.136.136/dvwa/vulnerabilities/sqli/?id=1&Submit=Submit -cookie="PHPSESSID=…" --risk=3 --level=5 --batch`
MITM	`bettercap -iface eth0`
Nmap	`nmap -sS -T4 -A 192.168.136.136`

Tabel 1 menunjukkan jenis-jenis serangan yang disimulasikan beserta *script* atau perintah yang digunakan pada masing-masing skenario. Simulasi dilakukan untuk menguji respons sistem terhadap beberapa bentuk ancaman siber, yaitu serangan *malware*, DDoS, *brute force*, SQL *injection*, dan MITM (*Man-in-the-Middle*).

b) Pengujian Web Visualisasi



Gambar 10. Web Visualisasi Deteksi Intrusi Jaringan

Gambar 10 merupakan tampilan antarmuka dari aplikasi yang berhasil dibangun menggunakan *framework* Dash. Web tersebut menampilkan berbagai macam informasi diantaranya tingkat kerentanan, jenis serangan, total *log* dan *alert*, IP sumber, waktu kejadian serangan, dan detail *log* serangan.

3.4 Hasil

Tabel 2. Metode Pengujian *Black Box*

Skenario	Hasil yang Diharapkan	Hasil (Berhasil/Gagal)
Simulasi serangan *Malware*	*Alert* serangan *Malware* muncul di *log* eve.json secara *real-time*	Berhasil
Simulasi serangan DDoS	*Alert* serangan DDoS muncul di *log* eve.json secara *real-time*	Berhasil
Simulasi serangan *Brute Force*	*Alert* serangan Brute Force muncul di *log* eve.json secara *real-time*	Berhasil
Simulasi serangan SQL *Injection*	*Alert* serangan SQL *Injection* muncul di *log* eve.json secara *real-time*	Berhasil
Simulasi serangan MITM	*Alert* serangan MITM muncul di *log* eve.json secara *real-time*	Berhasil
Simulasi serangan Nmap	*Alert* serangan Nmap muncul di *log* eve.json secara *real-time*	Berhasil

Tabel 3. Metode Pengujian *Functional Testing*

Skenario	Langkah Uji	Hasil yang Diharapkan	Hasil (Berhasil/ Gagal)
Visualisasi data serangan	Menjalankan web Dash	Grafik menampilkan tren dan jenis serangan secara tepat	Berhasil
Visualisasi *log* data dan *alert*	Menjalankan web Dash	*Dashboard* menampilkan jumlah *log* persis seperti pada perhitungan manual	Berhasil
Filter *log*	Menggunakan filter berdasarkan severity dan jenis serangan	Tabel dan grafik menyesuaikan hasil dengan filter	Berhasil

Pengujian pada simulasi serangan dilakukan untuk memastikan bahwa sistem Suricata dapat mendeteksi segala serangan yang terdapat di lalu lintas jaringan. Dengan menggunakan *Black Box*, pengujian ini akan difokuskan pada *output log* IDS tanpa melihat struktur internal dan *rule* pada Suricata. Hasil pengujian pada tabel 2 menunjukkan bahwa seluruh serangan yang ditujukan ke Suricata berhasil dideteksi dan dicatat ke dalam log eve.json secara *real-time*.

Pengujian fungsionalitas pada web visualisasi dilakukan untuk mengevaluasi apakah seluruh komponen pada web, yang dikembangkan menggunakan *framework* Dash, dapat berfungsi dengan baik dan sesuai dengan tujuannya untuk menampilkan informasi mengenai *log* serangan Suricata secara informatif dan interaktif. Selain itu, juga akan dilakukan perbandingan antara hasil perhitungan keseluruhan *log* jaringan dan *alert* di terminal Ubuntu dengan data yang ditampilkan dalam *dashboard* visualisasi web. Pada pengujian ini, semua komponen yang dimiliki web visualisasi akan diuji berdasarkan *input user* dan *output* sistem yang diharapkan. Hasil pengujian pada tabel 3 menunjukkan bahwa komponen-komponen yang dimiliki dapat berjalan sesuai fungsinya, *dashboard* web menampilkan data yang informatif dan mudah dimengerti, hasil keseluruhan *log* jaringan dan *alert* di kedua tempat menunjukkan jumlah yang identik, dan penggunaan filter bekerja dengan baik sehingga menjadikan web lebih interaktif.

Berdasarkan hasil dari pengujian *Black Box* dan *Functional Testing*, sistem telah memenuhi semua aspek yang diuji dengan baik. Pada pengujian *Black Box*, didapat bahwa Suricata dapat mendeteksi berbagai jenis serangan yang dikirim secara *real-time* dan mencatatnya secara lengkap di eve.json. Hal ini membuktikan bahwa sistem deteksi intrusi bekerja sesuai fungsinya yaitu mendeteksi anomali pada lalu lintas jaringan. Sedangkan, *Functional Testing* web

visualisasi menunjukkan bahwa seluruh komponen antarmuka dapat berfungsi dengan baik. Informasi yang divisualisasikan juga identik dengan data *log* serangan. Hal ini menunjukkan bahwa, visualisasi yang dibangun memiliki keakuratan dalam merepresentasikan data serangan yang didapat dari Suricata. Selain itu, sistem dapat menjalankan seluruh komponen fungsional dengan baik dan aplikasi web mampu menampilkan elemen interaktif berupa filter *dropdown* dan grafik dinamis. Dengan demikian, dapat disimpulkan bahwa sistem yang dibangun tidak hanya mampu mendeteksi serangan pada lalu lintas jaringan secara akurat, tetapi juga menyajikan visualisasi *log* serangan hasil deteksi intrusi menggunakan Suricata secara informatif, interaktif dan mudah dipahami oleh *user*.

4. KESIMPULAN

Berdasarkan uji coba yang dilakukan, penelitian ini menjelaskan tahapan pengembangan web analisis untuk sistem deteksi intrusi jaringan berbasis Suricata menggunakan *framework* Dash. Proses pengembangan diawali dengan perancangan arsitektur sistem, kemudian konfigurasi *rule* deteksi dan simulasi serangan untuk menghasilkan *log* dari Suricata. Selanjutnya, *log* data akan melewati tahap pengolahan dan pembersihan, serta implementasi visualisasi data menggunakan komponen-komponen interaktif Dash. Desain *dashboard* web disusun sedemikian rupa agar dapat menampilkan informasi secara dinamis dan sesuai dengan kebutuhan atau pilihan *user*.

Efektivitas web analisis dapat dilihat dari keakuratan data, evaluasi fungsionalitas, dan evaluasi interaktivitas. Jika dibandingkan dengan melihat melalui terminal Ubuntu, jumlah *alert* yang terdeteksi dan ditampilkan tetap konsisten. Selain itu, seluruh komponen aplikasi dapat berjalan baik juga interaktif. Dengan demikian, dapat disimpulkan bahwa web analisis ini efektif dalam menyajikan data serangan jaringan secara akurat, informatif, interaktif dan mudah dipahami oleh *user*.

DAFTAR PUSTAKA

[1] H. R. Virdynata and B. H. Wiyono, "Implementasi *IBM QRadar* Sebagai Pelindung Serangan *BruteForce Attack* Pada Laboratorium SMK Taruna Bhakti," Skripsi, Sekolah Tinggi Teknologi Terpadu Nurul Fikri, Depok, Indonesia, 2025.

[2] A. Rahmawati, M. N. Ramadhani, N. Yevana, R. Maulina, A. D. Fattah and D. Pratama, "Optimalisasi Infrastruktur Keamanan Teknologi Infrastruktur Dalam Menghadapi Ancaman *Cybersecurity*," Jurnal Pendidikan Sosial dan Humaniora, vol. 4, no. 2, pp. 2587-2597, 2025.

[3] S. D. Anjuju, M. Ulfa and D. Irawan, "Analisis Keamanan Infrastruktur Teknologi Informasi Dalam menghadapi Ancaman *Cybersecurity*," *Journal of Data Analytics, Information, and Computer Science*, vol. 2, no.1, pp. 75-80, 2025.

[4] F. R. Najwa, "Analisis Hukum Terhadap Tantangan Keamanan Siber: Studi Kasus Penegakan Hukum Siber di Indonesia," Jurnal Ilmu Sosial, Politik, dan Hukum, vol. 2, no. 1, pp. 8-16, 2024.

[5] M. Fikrie, "Serangan Siber ke RI Naik 6 Kali Lipat pada H1 2024, Mayoritas dari Dalam Negeri," kumparanTECH, 28 Agustus 2024. [Online]. Available: https://kumparan.com/kumparantech/serangan-siber-ke-ri-naik-6-kali-lipat-pada-h1-2024-mayoritas-dari-dalam-negeri-23PnYQpafrf/full. [Accessed 20 Februari *2025]*.

[6] Y. S, "*Optimized Intrusion Detection Model For Identifying Known And Innovative Cyber Attacks Using Support Vector Machine (SVM) Algorithms*," *Journal of Science Technology and Research* (JSTAR), vol. 5, no. 1, pp. 402, 2024.

[7] M. Tahir, U. Wahyuningsih, M. I. P. Pratama and M. A. Effindi, "*Development of Network Security Using a Suricata-Based Intrusion Prevention," Innovation in Research of Informatics* (INNOVATICS), vol. 6, no. 2, pp. 41-48, 2024.

[8] A. Garcia-Robledo and M. Zangiabady, "*Dash Sylvereye: A Python Library for Dashboard-Driven Visualization of Large Street Networks*," *IEEE Access*, vol. 11, pp. 121142-121161, 2023.

[9] N. Oliveira, I. Praça, E. Maia and O. Sousa, "*Intelligent Cyber Attack Detection and Classification for Network-Based Intrusion Detection Systems*," *Applied Sciences*, vol. 11, no. 4, 2021.

[10] H. Setiawan and W. Sulistyo, "*SIEM (Security Information Event Management) Model for Malware Attack Detection Using Suricata and Evebox," Int. J. Eng., Technol. Nat. Sci.*, vol. 5, no. 2, pp. 138-147, 2023.

[11] M. Jarabaa, "*Assessing the Widely Used Cyber Security Tools*," Thesis, Mutah University, Mu'tah, Jordan, 2024.

[12] F. Clement, A. Kaur, M. Sedghi, D. Krishnaswamy and K. Punithakumar, "*Interactive Data Driven Visualization for COVID-19 with Trends, Analytics and Forecasting*," in 2020 24th *International Conference Information Visualisation* (IV), Melbourne, 2020.