Jurnal Teknik Informatika (JUTIF)
P-ISSN: 2723-3863
E-ISSN: 2723-3871
Vol. 6, No. 6, December 2025, Page. 5371-5385
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.6.4694

Improving Detection Accuracy of Network Intrusions Using a Hybrid Network Intrusion Detection System Based on Isolation Forest and Random Forest Algorithms

Ryan Christensen Wang*[1], Refgiufi Patria Avrianto[2]

[1,2]Informatics, Pradita University, Indonesia

Email: [1]ryan.christensen@student.pradita.ac.id

Received : May 26, 2025; Revised : May 29, 2025; Accepted : Jun 16, 2025; Published : Dec 22, 2025

Abstract

The growing sophistication of cyberattacks has increased the urgency of securing organizational networks, especially those handling sensitive and large-scale data. Traditional intrusion detection systems (IDS) such as Suricata rely on signature-based methods and often fail to detect zero-day or evolving threats. To address this gap, this research proposes a hybrid intrusion detection model that integrates Suricata with machine learning algorithms—Isolation Forest and Random Forest. Suricata performs real-time packet inspection and anomaly filtering, while the machine learning component enhances detection of novel threats and reduces false positives. The methodology involves capturing real-time network traffic, pre-processing data, training models on both CICIDS2017 and simulated attack data, and evaluating performance using accuracy, precision, recall, and F1-score. Experimental results show that the hybrid model achieves high detection accuracy—99.86% on simulated data and 96.33% on the CICIDS2017 dataset. Compared to standalone Suricata, the hybrid model detects more unknown threats and reduces alert fatigue by minimizing false positives. This study contributes a scalable and adaptive IDS framework that combines anomaly- and signature-based detection techniques. The proposed system enhances threat detection capabilities in enterprise-level networks and offers practical implications for intelligent cybersecurity defences. The findings advance research in computer science, particularly in the domains of machine learning applications and network security systems.

Keywords : *Hybrid Machine Learning, Isolation Forest, Network Intrusion Detection System (NIDS), Random Forest, Suricata.*

1. PENDAHULUAN

Keamanan jaringan menjadi aspek yang sangat krusial dalam era digital saat ini, terutama bagi perusahaan besar yang bergantung pada infrastruktur teknologi informasi dan menyimpan data privasi dalam skala besar. Meningkatnya adopsi teknologi digital secara masif telah memicu eskalasi ancaman terhadap jaringan internal, termasuk serangan Distributed Denial of Service (DDoS), malware, dan eksploitasi zero-day yang mampu menimbulkan kerugian finansial maupun reputasi [1]. Serangan-serangan tersebut tidak hanya mengganggu operasional, namun juga dapat menurunkan kepercayaan publik terhadap organisasi yang terdampak [2]. Sebagai contoh, serangan ransomware oleh kelompok Daixin Team terhadap maskapai AirAsia pada tahun 2022 mengakibatkan kebocoran data lebih dari 5 juta pelanggan dan karyawan, termasuk data sensitif seperti nama, nomor paspor, dan alamat email [3]. Insiden ini menunjukkan betapa rentannya sistem keamanan jaringan dan pentingnya deteksi dini terhadap ancaman siber.

Dalam menghadapi tantangan tersebut, perusahaan perlu mengadopsi solusi yang tidak hanya reaktif tetapi juga proaktif dalam mendeteksi potensi serangan siber. Salah satu pendekatan yang menjanjikan adalah penggunaan **Network Intrusion Detection System (NIDS)** berbasis **Suricata**, yang mampu memonitor dan menganalisis lalu lintas jaringan secara real-time dengan efisiensi tinggi.

P-ISSN: 2723-3863 https://jutif.if.unsoed.ac.id
E-ISSN: 2723-3871 DOI: https://doi.org/10.52436/1.jutif.2025.6.6.4694

Suricata menggunakan pendekatan kombinasi antara deteksi berbasis tanda tangan dan deteksi anomali untuk mengenali pola-pola serangan yang sudah dikenal maupun yang belum terdokumentasi sebelumnya. Kinerja Suricata dapat ditingkatkan lebih lanjut melalui integrasi dengan teknologi machine learning, khususnya model **Hybrid Machine Learning**, yang terbukti secara empiris dapat meningkatkan akurasi deteksi hingga 99% dibandingkan model tunggal [4]. Pendekatan hybrid menggabungkan kelebihan dari berbagai algoritma untuk menangani kompleksitas ancaman siber modern yang semakin variatif dan canggih. Model ini secara khusus relevan untuk skenario enterprise yang memiliki volume lalu lintas tinggi dan arsitektur sistem yang heterogen [5].

Penelitian ini bertujuan untuk menganalisis performa pengembangan keamanan jaringan internal perusahaan dengan menerapkan NIDS berbasis Suricata yang diintegrasikan dengan model Hybrid Machine Learning. Fokus analisis mencakup efektivitas sistem dalam mendeteksi serangan siber, dampaknya terhadap performa jaringan, serta evaluasi komparatif antara hasil deteksi Suricata dan algoritma hybrid yang diajukan. Berdasarkan hal tersebut, penelitian ini merumuskan lima pertanyaan utama: (1) Mengapa dibutuhkan Hybrid Machine Learning untuk mendeteksi ancaman siber dalam jaringan? (2) Bagaimana cara membangun model hybrid tersebut? (3) Seberapa efektif model hybrid dalam mendeteksi ancaman pada jaringan internal? (4) Bagaimana cara penghitungan metrik performa seperti akurasi, presisi, recall, dan F1-score? serta (5) Bagaimana perbandingan kinerja antara Suricata dan model Hybrid ML?

Secara teoritis, penelitian ini memberikan kontribusi terhadap pengembangan ilmu di bidang keamanan siber, khususnya mengenai efektivitas integrasi Suricata dengan algoritma machine learning dalam konteks deteksi intrusi jaringan. Penelitian ini juga memperkaya literatur terkait penerapan hybrid model sebagai pendekatan yang lebih adaptif dalam mengidentifikasi ancaman. Sementara itu, secara praktis, hasil penelitian diharapkan dapat membantu perusahaan dalam membangun sistem keamanan jaringan yang lebih responsif dan cerdas, dengan kemampuan deteksi dini yang lebih baik terhadap potensi serangan siber sebelum berdampak signifikan terhadap sistem maupun pengguna [6]. Pendekatan ini juga sejalan dengan arah transformasi digital nasional dan tuntutan global terhadap sistem keamanan yang otonom dan berbasis kecerdasan buatan [7].

Sebagai landasan teori, IDS dibedakan menjadi NIDS dan HIDS [8], serta menggunakan metode deteksi berbasis tanda tangan dan anomali [9]. Suricata sebagai IDS berbasis tanda tangan memiliki keunggulan dalam kecepatan dan efisiensi pemrosesan [10]. Sementara itu, pendekatan hybrid machine learning mengombinasikan model supervised dan unsupervised untuk meningkatkan deteksi serangan kompleks [11]. Isolation Forest bekerja dengan mengisolasi anomali secara efisien tanpa memerlukan label data [12], sedangkan Random Forest memberikan klasifikasi yang akurat dengan daya tahan terhadap overfitting [13]. Integrasi keduanya memungkinkan sistem IDS untuk mendeteksi ancaman yang dikenal maupun yang belum dikenali secara adaptif dan responsif.

Penelitian oleh Veerasingam et al. [14] mengkaji penerapan Suricata pada skala usaha kecil menengah (UKM) menggunakan perangkat hemat daya seperti Raspberry Pi 2B, dan membuktikan efektivitas deteksi real-time terhadap pemindaian jaringan dan malware. Dalam konteks ini, Suricata juga terbukti fleksibel dalam implementasi dengan biaya rendah namun tetap efisien. Lebih lanjut, Sangadji et al. [15] membandingkan Suricata dan Snort dalam jaringan laboratorium komputer, dan menyimpulkan bahwa meskipun Suricata memerlukan pemrosesan CPU yang lebih tinggi, kemampuannya dalam mendeteksi serangan berskala besar lebih unggul dibandingkan Snort. Keunggulan ini juga didukung oleh penelitian Bhosale dan Mane [16], serta Gupta dan Sharma [17], yang menegaskan bahwa arsitektur multi-threaded Suricata memungkinkan pemrosesan data dalam jaringan berkecepatan tinggi secara lebih efisien, menjadikannya solusi unggul untuk skenario jaringan modern yang menuntut skalabilitas tinggi. Dalam ranah industri spesifik seperti SCADA, Wong et al. [1] berhasil mengembangkan modul Suricata untuk mendeteksi ancaman berbasis protokol EtherNet/IP

Vol. 6, No. 6, December 2025, Page. 5371-5385
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.6.4694

dengan implementasi pada perangkat keras berdaya rendah, menunjukkan kapabilitas adaptif Suricata untuk berbagai skenario keamanan kritikal.

Selain pendekatan berbasis tanda tangan, integrasi Suricata dengan teknologi machine learning juga telah dieksplorasi dalam berbagai studi. Chiba et al. [13] mengusulkan kerangka kerja hybrid yang menggabungkan Suricata dengan Isolation Forest Algorithm (IFA) untuk meningkatkan akurasi deteksi dan mengurangi false positive. Kombinasi ini memperluas kapabilitas sistem dalam mengenali anomali baru di luar serangan yang telah dikenal sebelumnya. Di sisi lain, pendekatan machine learning secara umum telah menghasilkan banyak solusi inovatif dalam sistem deteksi intrusi. Misalnya, Momand et al. [18], Santhosh Kumar et al. [19], dan Sahani et al. [6] menunjukkan bahwa arsitektur hybrid dan deep learning semakin mendominasi pengembangan IDS modern untuk lingkungan seperti IoT, smart grid, dan jaringan kendaraan. Penelitian-penelitian ini memperkuat keyakinan bahwa integrasi metode deteksi berbasis aturan seperti Suricata dengan pembelajaran mesin modern mampu menciptakan sistem yang lebih adaptif dan cerdas dalam menghadapi spektrum ancaman siber.

Berbagai studi sebelumnya menunjukkan efektivitas Suricata dalam berbagai konteks penerapan sistem deteksi intrusi, dibawah ini merupakan tabel komparatif dari beberapa penelitian terdahulu sebagai visualisasi gap literatur:

Tabel 1. Ringkasan Literatur Terkait IDS dan Hybrid ML

Peneliti	Metode	Dataset	Algoritma	Hasil Utama
Tahir et al. [20]	Suricata + Waterfall	DDoS Simulasi	-	Akurasi 96,5%, FP 2,5%
Hussain et al. [21]	Hybrid ML	Simulasi	Random Forest + OC-SVM	Akurasi 95,95%
Dini et al. [22]	Supervised ML	3 Datasets	SVM + Random Forest	Akurasi 100% (biner)

Berdasarkan tinjauan pustaka tersebut, dapat disimpulkan bahwa arah pengembangan IDS masa kini bergerak ke arah solusi yang bersifat hybrid—menggabungkan kekuatan deteksi berbasis tanda tangan dengan fleksibilitas pembelajaran mesin [23]. Maka dari itu, penelitian ini tidak hanya mengkaji efektivitas Suricata sebagai sistem NIDS dalam mendeteksi ancaman, tetapi juga mengeksplorasi potensinya jika dikombinasikan dengan model Hybrid Machine Learning, khususnya dalam konteks jaringan internal perusahaan berskala besar. Penelitian ini diharapkan dapat mengisi celah dalam literatur yang masih terbatas dalam pembahasan langsung antara performa Suricata dengan model hybrid berbasis ML, serta memberikan solusi komprehensif yang relevan dan aplikatif bagi dunia industri dalam meningkatkan ketahanan siber.

2. METODE

2.1. Pengumpulan Data

Pada Dalam penelitian ini, data jaringan dikumpulkan dari jaringan internal yang terhubung melalui switch. Proses pengambilan data dilakukan menggunakan sistem operasi Kali Linux. Kali Linux adalah sistem operasi berbasis Debian yang dikhususkan untuk menguji penetrasi dan keamanan jaringan, serta memiliki berbagai alat untuk melakukan sniffing, logging, hingga simulasi serangan jaringan nirkabel dan kabel [24].

Proses sniffing data dilakukan dengan merekam lalu lintas jaringan menggunakan tool seperti tcpdump atau Wireshark, lalu disimpan dalam bentuk file .pcap atau CSV untuk digunakan dalam eksperimen lebih lanjut.

2.2. Pengolahan Awal Data

Setelah data jaringan sudah dikumpulkan, data tersebut akan digunakan dan diproses / filter terlebih dahulu menggunakan Suricata NIDS untuk mengecek apakah serangan tersebut sudah terdeteksi dalam rules yang sudah terdapat pada database Suricata, jika serangan terdaftar maka akan otomatis terdeteksi. Kemudian data jaringan tersebut akan digunakan juga dalam pelatihan dan pengujian Hybrid Machine Learning menggunakan Google Colab. Google Colab merupakan platform komputasi berbasis cloud dan memungkinkan user untuk menulis serta menjalankan kode Python langsung dari browser mereka. Platform ini mendukung lingkungan notebook interaktif dan telah dilengkapi dengan berbagai pustaka untuk pemrosesan data dan pembelajaran mesin [25].

2.3. Metode Yang Diusulkan

Metode yang akan digunakan oleh penulis untuk penelitian ini adalah Hybrid Machine Learning yang merupakan penggabungan dari model Isolation Forest (IF) dan Random Forest (RF) dimana IF digunakan sebagai detektor anomali untuk memetakan skor anomali pada setiap instance sementara, RF digunakan sebagai klasifikasi supervised berdasarkan fitur asli dan skor anomali hasil dari Isolation Forest. Berikut adalah konfigurasi model yang akan digunakan di dalam penelitian ini:

Isolation Forest:

- N_estimators = 100
- Contamination = 0.1
- Max_samples = 'auto'
- Random_state = 42

Random Forest:

- N_estimators = 100
- Max_depth = None
- Min_samples_split = 2
- Max_features = 'sqrt'
- Criterion = 'gini'
- Random_state = 42

Kedua model dilatih secara terpisah, namun digabung dalam proses inferensi: skor anomali dari Isolation Forest ditambahkan ke fitur hasil pre-processing, lalu menjadi input ke dalam Random Forest.

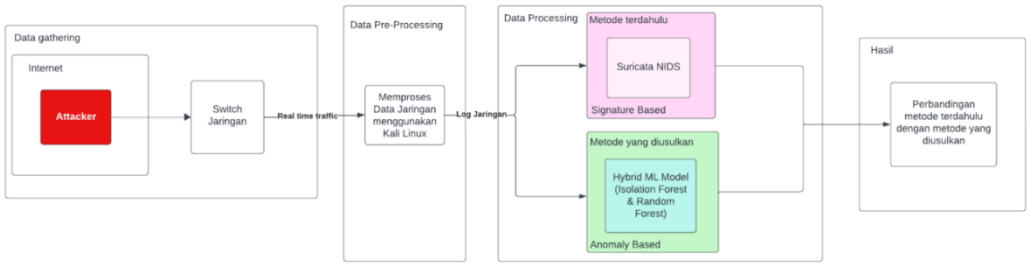
2.4. Eksperimen dan Pengujian Metode

Berikut merupakan diagram alur eksperimen beserta dengan penjelasannya yang akan dilakukan di dalam penelitian ini:

- Monitoring & Data Gathering: Melakukan pengumpulan data menggunakan switch yang terhubung kepada jaringan dan akan digunakan kali linux untuk mengumpulkan data jaringan lokal yang aktif.
- Data Pre-processing: Data jaringan yang sudah dikumpulkan akan dibersihkan (handling missing/infinite values), setelah itu akan dilakukan balancing dataset dengan SMOTE (Synthetic Minority Over-sampling Technique) agar tidak terjadi oversampling dan menjaga keseimbangan kelas pada dataset, kemudian dilakukan seleksi fitur menggunakan SelectKBest dengan metode ANOVA F-Statistic dan terakhir normalisasi fitur menggunakan StandardScaler.
- Training Machine Learning Model: Menggunakan dataset yang sudah dikumpulkan dari internet dan juga log jaringan untuk pelatihan model Hybrid Machine Learning; Isolation Forest akan dilatih terlebih dahulu untuk menghasilkan skor anomali, skor tersebut akan diproses dan ditambahkan pada fitur oleh Random Forest.

Jurnal Teknik Informatika (JUTIF)
P-ISSN: 2723-3863
E-ISSN: 2723-3871
Vol. 6, No. 6, December 2025, Page. 5371-5385
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.6.4694

- Comparative Study: Akan dilakukan perbandingan antara NIDS Suricata dan juga model Hybrid Machine Learning yang sudah diajukan untuk membandingkan efektivitas menggunakan skor *Accuracy, Precision, Recall* dan *F1-Score* dari kedua metode tersebut.



Gambar 1. Diagram Alur Penelitian

2.5. Evaluasi dan Validasi Hasil

Dalam Penelitian ini akan dilakukan perbandingan performa dari NIDS Suricata dan Hybrid ML yang diajukan dalam penelitian ini menggunakan skor akurasi, skor false positives dan false negatives. Evaluasi performa model dilakukan dengan menghitung metrik sebagai berikut :

1. Accuracy: $\boldsymbol{Accuracy} = \frac{\boldsymbol{TP+TN}}{\boldsymbol{TP+TN+FP+FN}}$
2. Precision: $\boldsymbol{Precision} = \frac{\boldsymbol{TP}}{\boldsymbol{TP+FP}}$
3. Recall: $\boldsymbol{Recall} = \frac{\boldsymbol{TP}}{\boldsymbol{TP+FN}}$
4. F1-Score: $\boldsymbol{F1-score} = 2 \times \frac{\boldsymbol{Precision \times Recall}}{\boldsymbol{Precision + Recall}}$

Keterangan:

- TP (True Positive): Serangan yang benar terdeteksi.
- TN (True Negative): Normal traffic yang benar diabaikan.
- FP (False Positive): Normal traffic yang salah diklasifikasikan sebagai serangan.
- FN (False Negative): Serangan yang gagal dideteksi.

3. HASIL

3.1. Implementasi Sistem

3.1.1. Konfigurasi dan Pengaturan Suricata

Suricata dikonfigurasi sebagai NIDS yang ditempatkan dalam jaringan. Suricata bertugas untuk menginspeksi lalu lintas jaringan dan mendeteksi ancaman berdasarkan aturan yang telah ditentukan. Berikut adalah langkah-langkah utama implementasi Suricata:

- Instalasi Suricata pada server keamanan jaringan.
- Konfigurasi aturan deteksi yang mencakup serangan berbasis tanda tangan (signature-based) dan aturan yang dapat dikustomisasi.
- Logging dan Analisis menggunakan Suricata untuk mencatat setiap anomali yang terdeteksi.

3.1.2. Pengembangan Hybrid Machine Learning

Sistem Hybrid Machine Learning dikembangkan untuk meningkatkan deteksi anomali dengan mengkombinasikan:

Vol. 6, No. 6, December 2025, Page. 5371-5385
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.6.4694

- Isolation Forest (IF) untuk mendeteksi anomali berdasarkan distribusi data jaringan yang tidak wajar.
- Random Forest (RF) sebagai model klasifikasi untuk meningkatkan akurasi dan mengurangi false positives.

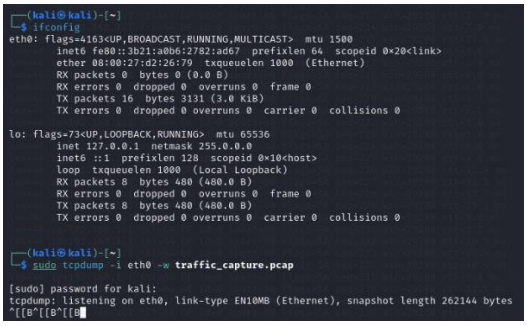
Proses pengembangan model Hybrid Machine Learning meliputi:

1. Pengumpulan dataset dari log jaringan dan hasil simulasi serangan menggunakan traffic generator.
2. Preprocessing data, termasuk normalisasi dan ekstraksi fitur dari paket jaringan.
3. Pelatihan model menggunakan Isolation Forest untuk mendeteksi anomali dan Random Forest untuk mengklasifikasikan ancaman.
4. Validasi model dengan membandingkan hasil deteksi terhadap ground truth.

3.2. Objek Penelitian

3.2.1. Deskripsi Dataset (Simulasi)

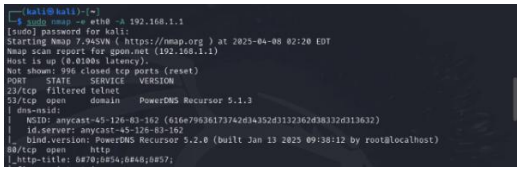
Dataset simulasi dibuat menggunakan kali linux seperti pada gambar dibawah:



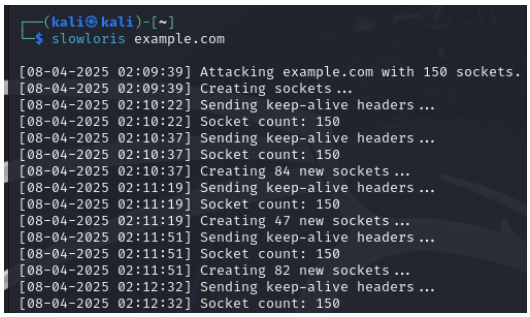
Gambar 2. Mengumpulkan Data Jaringan Menggunakan Tcpdump

Pada Gambar 2, Penulis menggunakan command tcpdump untuk mengumpulkan data jaringan kemudian data tersebut akan disimpan pada traffic_capture.pcap

Selanjutnya pada gambar 3, 4 dan 5 penulis melakukan beberapa simulasi serangan dan juga browsing pada internet seperti normal untuk mensimulasikan traffic nyata. Beberapa jenis simulasi yang dilakukan melainkan, nmap, slowloris, synflood:



Gambar 3. Simulasi nmap



Gambar 4. Simulasi Serangan Slowloris

Jurnal Teknik Informatika (JUTIF)
P-ISSN: 2723-3863
E-ISSN: 2723-3871
Vol. 6, No. 6, December 2025, Page. 5371-5385
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.6.4694

Gambar 5. Simulasi Serangan Synflood

Dari hasil pcap simulasi yang telah dilakukan penulis melakukan konversi menjadi csv bernama attack_data.csv agar dapat diproses lebih mudah menggunakan Hybrid Machine Learning. Dataset attack_data.csv berisi 9407 sampel dengan 7 label tanpa adanya label attack spesifik.

Encoding Label:

Label dikonversi ke bentuk numerik dengan encoding sebagai berikut:

Anomaly → 1

Normal → 0

Pembagian Data:

Dataset dibagi menjadi 6,584 sampel untuk training dan 2,823 sampel untuk testing.

3.2.2. Deskripsi Dataset (CICIDS2017)

Dataset yang digunakan bernama Wednesday-workingHours.pcap_ISCX.csv dari CICIDS2017, yang berisi 692,703 sampel dengan 78 fitur dan 1 label. Distribusi label dalam dataset adalah sebagai berikut:

- BENIGN: 440,031 sampel
- DoS Hulk: 231,073 sampel
- DoS GoldenEye: 10,293 sampel
- DoS Slowloris: 5,796 sampel
- DoS Slowhttptest: 5,499 sampel
- Heartbleed: 11 sampel

Penanganan Data Hilang dan Nilai Tak Terhingga:

Dataset memiliki 1,008 missing values dan 1,586 infinity values pada fitur Flow Bytes/s dan Flow Packets/s. Penanganan dilakukan dengan mengganti nilai tersebut dengan median fitur terkait.

Encoding Label:

Label dikonversi ke bentuk numerik dengan encoding sebagai berikut:

- BENIGN → 0
- DoS GoldenEye → 1
- DoS Hulk → 2
- DoS Slowhttptest → 3
- DoS Slowloris → 4
- Heartbleed → 5

Pembagian Data:

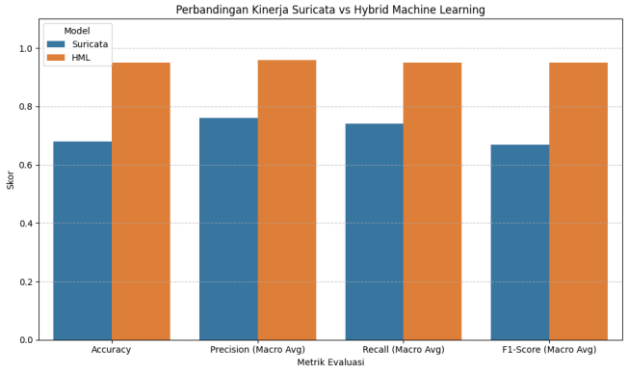
Dataset dibagi menjadi 1,848,130 sampel untuk training dan 792,056 sampel untuk testing

3.3 Visualisasi Perbandingan Akurasi Suricata dan Hybrid Machine Learning:

Gambar 6 menunjukkan grafik batang perbandingan kinerja antara Suricata dan model Hybrid Machine Learning (HML) berdasarkan empat metrik evaluasi utama: Accuracy, Precision (Macro Avg), Recall (Macro Avg), dan F1-Score (Macro Avg). Dari grafik terlihat jelas bahwa HML secara konsisten unggul di semua metrik dengan skor mendekati 1.0, menandakan performa deteksi yang sangat baik dan seimbang antar kelas. Sebaliknya, Suricata menunjukkan performa yang jauh lebih rendah, khususnya pada metrik F1-Score dan Accuracy, yang menunjukkan bahwa meskipun Suricata dapat mendeteksi

Vol. 6, No. 6, December 2025, Page. 5371-5385
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.6.4694

serangan, tingkat kesalahannya dalam mengklasifikasikan trafik normal masih tinggi. Visualisasi ini menegaskan bahwa pendekatan berbasis machine learning lebih efektif dalam mendeteksi lalu lintas jaringan secara akurat dibandingkan sistem rule-based seperti Suricata.



Gambar 6. Grafik Perbandingan Akurasi Suricata dan Hybrid Machine Learning

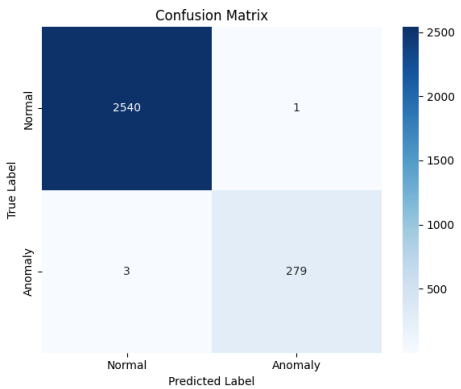
4. DISKUSI

4.1. Evaluasi Performa Model dengan dataset simulasi:

Confusion matrix menunjukkan hasil klasifikasi model terhadap dataset (simulasi):

Analisis Confusion Matrix Dari Gambar 7:

- True Positive (TP): 279
  Model berhasil mendeteksi 279 serangan/anomaly dengan benar. Ini menunjukkan bahwa model mampu mengenali anomaly secara efektif.
- True Negative (TN): 2540
  Sebanyak 2540 koneksi normal dikenali dengan benar sebagai normal. Ini berarti model memiliki tingkat akurasi tinggi dalam mengenali lalu lintas normal.
- False Positive (FP): 1
  Hanya ada 1 kasus normal yang salah diklasifikasikan sebagai anomaly. Ini artinya tingkat false alarm sangat rendah, sangat penting dalam implementasi nyata agar tidak mengganggu operasional jaringan.
- False Negative (FN): 3
  Ada 3 serangan yang gagal dideteksi, diklasifikasikan sebagai normal. Ini berarti masih ada potensi risiko keamanan, tapi angkanya kecil.



Gambar 7. Confusion Matrix (Dataset Simulasi)

Jurnal Teknik Informatika (JUTIF)
P-ISSN: 2723-3863
E-ISSN: 2723-3871
Vol. 6, No. 6, December 2025, Page. 5371-5385
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.6.4694

Perhitungan *Accuracy, Precision, Recall, dan F1-Score:*

- $\boldsymbol{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} = \frac{(279+2540)}{(279+279+2540+1+3)} = \frac{2819}{2823} \approx 99.86\%$
- $\boldsymbol{Precision} = \frac{TP}{TP+FP} = \frac{279}{(279+1)} \approx 99.64\%$
- $\boldsymbol{Recall} = \frac{TP}{TP+FN} = \frac{279}{(279+3)} \approx 98.94\%$
- $\boldsymbol{F1-score} = 2 \times \frac{Precision \times Recall}{Precision + Recall} \approx 99.29\%$

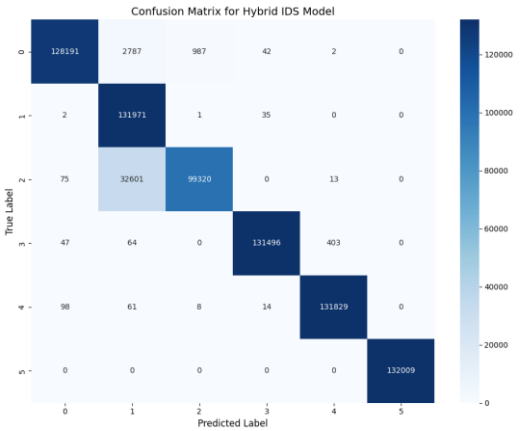
Laporan Klasifikasi:

Tabel 2. Laporan Klasifikasi (Data Simulasi)

Kelas	Accuracy	Recall	F1-Score
0 (Normal)	1,00	1,00	1,00
1 (Anomaly)	1,00	0,99	0,99

4.2. Evaluasi Performa Model dengan dataset CICIDS2017:

Confusion matrix menunjukkan hasil klasifikasi model terhadap dataset (CICIDS2017):



Gambar 8. Confusion Matrix (Data CICIDS2017)

Analisis Confusion Matrix Dari Gambar 8:

Berdasarkan confusion matrix yang ditampilkan, model Hybrid Intrusion Detection System (IDS) menunjukkan performa klasifikasi yang sangat baik. Hal ini terlihat dari dominasi angka yang tinggi pada diagonal utama, yang mengindikasikan bahwa sebagian besar lalu lintas jaringan berhasil diklasifikasikan sesuai dengan label sebenarnya. Sebagai contoh, trafik dengan label 0 dan 5 memiliki jumlah prediksi benar yang sangat tinggi, masing-masing sebanyak 128.191 dan 132.009 instance. Ini mencerminkan kemampuan model dalam mendeteksi trafik normal maupun jenis serangan tertentu dengan tingkat akurasi yang tinggi.

Meskipun demikian, masih terdapat sejumlah kesalahan klasifikasi yang perlu diperhatikan. Salah satu yang paling menonjol terjadi pada label 2, di mana sekitar 32.601 instance diklasifikasikan sebagai label 1. Hal ini kemungkinan disebabkan oleh kesamaan karakteristik trafik antara DoS Hulk dan DoS GoldenEye, terutama pada pola burst packet dan penggunaan protokol HTTP yang serupa. Fenomena

Vol. 6, No. 6, December 2025, Page. 5371-5385
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.6.4694

ini juga dicatat oleh Bekerman et al. [26],yang menyebutkan bahwa beberapa jenis serangan DoS memiliki atribut temporal dan struktural yang hampir identik sehingga menyulitkan model dalam melakukan klasifikasi yang tepat. Meski begitu, distribusi kesalahan secara umum relatif kecil jika dibandingkan dengan jumlah prediksi yang benar, sehingga secara keseluruhan model ini dapat dikatakan cukup andal dalam mendeteksi anomali pada jaringan. Temuan ini sejalan dengan penelitian oleh Zhang et al. [27], yang menunjukkan bahwa model berbasis pembelajaran mesin mampu mencapai akurasi tinggi dalam mendeteksi trafik abnormal pada dataset yang kompleks seperti CICIDS2017.

Perhitungan *Accuracy, Precision, Recall*, dan *F1-Score* menggunakan contoh pada kelas BENIGN(0):

- $\boldsymbol{Accuracy = \frac{TP+TN}{TP+TN+FP+FN} = \frac{(128{,}191+892{,}460)}{(1{,}024{,}691)} \approx 99.61\%}$
- $\boldsymbol{Precision = \frac{TP}{TP+FP} = \frac{128{,}191}{(128{,}191+222)} \approx 99.83\%}$
- $\boldsymbol{Recall = \frac{TP}{TP+FN} = \frac{128{,}191}{(128{,}191+3818)} \approx 97.09\%}$
- $\boldsymbol{F1-score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \approx 98.45\%}$

Laporan Klasifikasi:

Tabel 3. Laporan Klasifikasi BENIGN (Data CICIDS2017)

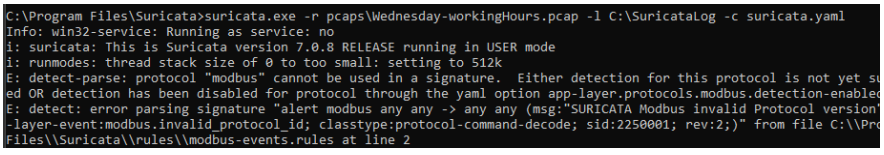
Kelas	Akurasi	Recall	F1-Score
0 (BENIGN)	1,00	0,97	0,98
1 (DoS GoldenEye)	0,79	1,00	0,88
2 (DoS Hulk)	0,99	0,75	0,86
3 (DoS Slowhttptest)	1,00	1,00	1,00
4 (DoS slowloris)	1,00	1,00	1,00
5 (Heartbleed)	1,00	1,00	1,00

Akurasi model pada data uji mencapai 96.33%, menunjukkan performa yang sangat baik dalam mendeteksi serangan siber.

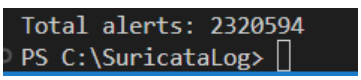
4.3. Perbandingan Performa Suricata dengan Hybrid Machine Learning

Pada perbandingan antara Suricata dan juga Hybrid Machine Learning, akan digunakan satu dataset saja agar hasil yang didapatkan adalah seakurat mungkin, dataset yang digunakan merupakan dataset CICIDS 2017 khususnya adalah Wednesday-workingHours.pcap.

Pertama penulis akan mengolah dataset menggunakan Suricata dan hasil tersebut akan di simpan dalam bentuk json seperti yang dapat dilihat pada gambar 8 dan 9:



Gambar 8. Konfigurasi Suricata Menggunakan Data CICIDS2017



Gambar 9. Hasil Proses Data CICIDS2017 Menggunakan Suricata

Jurnal Teknik Informatika (JUTIF)
P-ISSN: 2723-3863
E-ISSN: 2723-3871
Vol. 6, No. 6, December 2025, Page. 5371-5385
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.6.4694

Dari hasil Suricata pada gambar 9 dapat dilihat bahwa terdeteksi sebanyak 2,320,594 alerts yang dianggap oleh Suricata sebagai potensi serangan dan Suricata otomatis menyimpan hasilnya pada eve.json, selanjutunya penulis akan memparsing eve.json dan hasilnya diubah menjadi csv agar bisa dilakukan perbandingan menggunakan kode seperti pada gambar 10 dibawah ini:

```
import json
import pandas as pd


count = 0
alerts = []
with open("eve.json", "r", encoding="utf-8") as f:
    for line in f:
        count += 1
        if count % 10000 == 0:
            print(f"Processed {count} lines...")

        record = json.loads(line)
        if record.get("event_type") == "alert":
            alerts.append({
                "timestamp": record.get("timestamp"),
                "src_ip": record.get("src_ip"),
                "dest_ip": record.get("dest_ip"),
                "category": record["alert"]["category"],
                "signature": record["alert"]["signature"]
            })

print(f"Total alerts: {len(alerts)}")
suricata_df = pd.DataFrame(alerts)
suricata_df.to_csv("C:/SuricataLog/suricata_alerts.csv", index=False)
```

Gambar 10. Kode Parsing Untuk Mengubah Json Menjadi CSV

Hasil Signature suricata_alerts.csv menunjukan hanya terdapat 1 kategori yaitu, “Generic Protocol Command Decode” yang tidak dapat dipastikan adanya serangan dan beberapa signature seperti dibawah ini:

Tabel 4. Hasil dan Jumlah Signature Suricata Dalam Bentuk CSV

Signature	Jumlah
SURICATA TCPv4 invalid checksum	2,283,035
SURICATA HTTP unable to match response to request	23,570
SURICATA STREAM Packet with invalid timestamp	5,880
SURICATA STREAM FIN out of window	1,597
SURICATA STREAM Packet with invalid ack	1,553

Dari hasil tabel 4 dapat dilihat ternyata Suricata tidak dapat menkonfirmasi adanya serangan nyata dan alerts yang diberikan oleh Suricata merupakan sebuah peringatan dari anomali yang terdeteksi. Oleh karena itu penulis mencoba langsung melakukan evaluasi deteksi Suricata dan membandingkan langsung dengan ground truth yaitu file csv dataset CICIDS2017. Pada gambar 11 dibawah ini penulis melakukan parsing langsung pada eve.json dengan menggunakan column yang sama yaitu destination port dan label, kemudian hanya mengambil event yang berlabel alert pada eve.json tersebut :

Gambar 12 menunjukkan hasil evaluasi dari Suricata dalam memproses dataset CICIDS2017, dapat dilihat bahwa Suricata memiliki precision yang sangat tinggi untuk kelas BENIGN (1.00), namun recall-nya hanya 0.49, yang berarti banyak lalu lintas normal yang salah diklasifikasikan sebagai serangan. Sebaliknya, untuk kelas ATTACK, recall sangat tinggi (1.00), tetapi precision rendah (0.53), yang menunjukkan banyak false positive. Nilai akurasi keseluruhan Suricata hanya 68%, dengan F1-score rata-rata 0.67, menunjukkan bahwa performanya masih kurang seimbang dalam membedakan antara trafik normal dan anomali. Hal ini kemungkinan besar disebabkan karena Suricata merupakan

Jurnal Teknik Informatika (JUTIF)
P-ISSN: 2723-3863
E-ISSN: 2723-3871
Vol. 6, No. 6, December 2025, Page. 5371-5385
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.6.4694

Signature-based NIDS sehingga jika ada *rules* yang belum diperbaharui maka Suricata akan mempunyai kesulitan untuk mendeteksi serangan baru.

```python
import pandas as pd
import json
from sklearn.metrics import classification_report

print("Loading dataset CIC-IDS2017...")
df = pd.read_csv("Wednesday-workingHours.pcap_ISCX.csv", low_memory=False)
df.columns = df.columns.str.strip()

df = df[["Destination Port", "Label"]]
df["Destination Port"] = pd.to_numeric(df["Destination Port"], errors='coerce')
df = df.dropna(subset=["Destination Port"])
df["Destination Port"] = df["Destination Port"].astype(int)

df["GroundTruth"] = df["Label"].apply(lambda x: 0 if "BENIGN" in x.upper() else 1)

print("Parsing Suricata eve.json...")
alert_ports = set()

with open("eve.json", "r", encoding="utf-8") as f:
    for line in f:
        try:
            event = json.loads(line)
            if event.get("event_type") == "alert":
                dest_port = event.get("dest_port")
                if dest_port is not None:
                    alert_ports.add(int(dest_port))
        except json.JSONDecodeError:
            continue

print(f"Jumlah port yang terdeteksi oleh Suricata: {len(alert_ports)}")

df["Suricata_Predict"] = df["Destination Port"].apply(lambda port: 1 if port in alert_ports else 0)

print("\n=== Evaluasi Deteksi Suricata Berdasarkan Destination Port ===")
print(classification_report(df["GroundTruth"], df["Suricata_Predict"], target_names=["BENIGN", "ATTACK"]))
```

Gambar 11. Kode Evaluasi Deteksi Suricata

```
=== Evaluasi Deteksi Suricata Berdasarkan Destination Port ===
              precision    recall  f1-score   support

      BENIGN       1.00      0.49      0.66    440031
      ATTACK       0.53      1.00      0.69    252672

    accuracy                           0.68    692703
   macro avg       0.76      0.74      0.67    692703
weighted avg       0.83      0.68      0.67    692703
```

Gambar 12. Hasil Evaluasi Deteksi Suricata

Temuan ini memiliki implikasi penting secara praktis dan akademis. Secara praktis, penggunaan model Hybrid Machine Learning dapat memberikan keunggulan dalam deteksi anomali real-time dengan false positive rendah, yang dapat meningkatkan efisiensi operasional sistem keamanan jaringan. Sementara itu, secara akademis, hasil temuan ini membuka peluang pengembangan IDS berbasis pembelajaran mesin yang lebih adaptif dan otomatis terhadap perubahan pola serangan baru. Pada tabel 5 dibawah ini merupakan perbandingan yang dapat disimpulkan dari penelitian terhadap Suricata dan Hybrid Machine Learning:

Tabel 5. Perbandingan Suricata dan Hybrid Machine Learning

Aspek	Suricata	Hybrid Machine Learning
Metode:	Rule/Signature based.	Anomaly based.
Kustomisasi:	Memerlukan adanya update pada rule secara manual.	Mempelajari pattern secara otomatis dari data jaringan.
False Positives/Negatives:	Tinggi.	Sangat Rendah.
Maintenance:	Memerlukan update setiap kali ada serangan baru yang ditemukan.	Melakukan training ulang menggunakan data baru.

Vol. 6, No. 6, December 2025, Page. 5371-5385
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.6.4694

5. KESIMPULAN

Dari hasil pengujian dan perbandingan yang telah dilakukan, NIDS berbasis Suricata terbukti efektif dalam mendeteksi serangan yang sudah dikenal melalui pendekatan signature-based. Namun, sistem NIDS berbasis Hybrid Machine Learning menunjukkan keunggulan signifikan dalam hal akurasi dan kemampuan generalisasi, terutama dalam mendeteksi serangan baru atau zero-day. Hal ini ditunjukkan oleh hasil evaluasi menggunakan dataset simulasi, di mana model Hybrid ML mencapai tingkat akurasi hingga 99% dengan tingkat kesalahan yang sangat rendah karena penulis menerapkan Teknik SMOTE (Synthetic Minority Over-sampling Technique) untuk menangani ketidakseimbangan kelas dalam dataset serta kombinasi antara Isolation Forest dan juga Random Forest yang memungkinkan model untuk memfilter data anomali terlebih dahulu, kemudian mengklasifikasikannya dengan akurasi tinggi.

Kontribusi utama penelitian ini adalah menunjukkan bahwa pendekatan Hybrid Machine Learning tidak hanya meningkatkan performa deteksi serangan, tetapi juga memberikan arah baru dalam pengembangan sistem keamanan jaringan yang adaptif dan data-driven. Selain itu, hasil ini memperkuat pentingnya integrasi metode pembelajaran mesin dalam desain NIDS masa depan, khususnya pada konteks dinamika serangan siber yang terus berkembang.

Untuk penelitian selanjutnya, disarankan untuk mengembangkan model sehingga dapat berjalan dalam lingkungan jaringan nyata (*real-world deployment*) untuk mengevaluasi performa dan stabilitas sistem secara lebih komprehensif. Penambahan metode explainable AI (XAI) juga dapat menjadi pendekatan yang menarik untuk meningkatkan transparansi dan interpretabilitas sistem deteksi berbasis machine learning.

REFERENSI

[1] K. Wong, C. Dillabaugh, N. Seddigh, and B. Nandy, “Enhancing Suricata intrusion detection system for cyber security in SCADA networks,” in *2017 IEEE 30th Canadian Conference on Electrical and Computer Engineering (CCECE)*, IEEE, Apr. 2017, pp. 1–5. doi: 10.1109/CCECE.2017.7946818.

[2] I. Friedberg, F. Skopik, G. Settanni, and R. Fiedler, “Combating advanced persistent threats: From network event correlation to incident detection,” *Comput Secur*, vol. 48, pp. 35–57, 2015, doi: 10.1016/j.cose.2014.09.006.

[3] C. Fam, “AirAsia allegedly hit with ransomware attack, data of five million passengers and employees reportedly compromised ,” TheStar. Accessed: Dec. 17, 2024. [Online]. Available: https://www.thestar.com.my/tech/tech-news/2022/11/23/airasia-allegedly-hit-with-ransomware-attack-data-of-five-million-passengers-and-employees-reportedly-compromised

[4] B. S. Bari, K. Yelamarthi, and S. Ghafoor, “Intrusion Detection in Vehicle Controller Area Network (CAN) Bus Using Machine Learning: A Comparative Performance Study,” *Sensors*, vol. 23, no. 7, Apr. 2023, doi: 10.3390/s23073610.

[5] H. Chen, G.-R. You, and Y.-R. Shiue, “Hybrid Intrusion Detection System Based on Data Resampling and Deep Learning,” *International Journal of Advanced Computer Science and Applications*, vol. 15, no. 2, 2024, doi: 10.14569/IJACSA.2024.0150214.

[6] N. Sahani, R. Zhu, J. H. Cho, and C. C. Liu, “Machine Learning-based Intrusion Detection for Smart Grid Computing: A Survey,” *ACM Transactions on Cyber-Physical Systems*, vol. 7, no. 2, Apr. 2023, doi: 10.1145/3578366.

[7] S. Praptodiyono, T. Firmansyah, M. H. Anwar, C. A. Wicaksana, A. S. Pramudyo, and A. Al-Allawee, “Development of hybrid intrusion detection system based on Suricata with pfSense method for high reduction of DDoS attacks on IPv6 networks,” *Eastern-European Journal of Enterprise Technologies*, vol. 5, no. 9 (125), pp. 75–84, Oct. 2023, doi: 10.15587/1729-4061.2023.285275.

Vol. 6, No. 6, December 2025, Page. 5371-5385
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.6.4694

[8] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: techniques, datasets and challenges," *Cybersecurity*, vol. 2, no. 1, Dec. 2019, doi: 10.1186/s42400-019-0038-7.

[9] M. A. Al Hilmi and E. Khujaemah, "NETWORK SECURITY MONITORING WITH INTRUSION DETECTION SYSTEM," *Jurnal Teknik Informatika (Jutif)*, vol. 3, no. 2, pp. 249–253, Apr. 2022, doi: https://doi.org/10.20884/1.jutif.2022.3.2.117.

[10] D. H. K. Raharjo and Muhammad Salman, "ANALYZING SURICATA ALERT DETECTION PERFORMANCE ISSUES BASED ON ACTIVE INDICATOR OF COMPROMISE RULES," *Jurnal Teknik Informatika (Jutif)*, vol. 4, no. 3, pp. 601–610, Jun. 2023, doi: 10.52436/1.jutif.2023.4.3.1013.

[11] M. Sajid *et al.*, "Enhancing intrusion detection: a hybrid machine and deep learning approach," *Journal of Cloud Computing*, vol. 13, no. 1, Dec. 2024, doi: 10.1186/s13677-024-00685-x.

[12] R. Primartha and B. A. Tama, "Anomaly detection using random forest: A performance revisited," in *2017 International Conference on Data and Software Engineering (ICoDSE)*, IEEE, Nov. 2017, pp. 1–6. doi: 10.1109/ICODSE.2017.8285847.

[13] Z. Chiba, N. Abghour, K. Moussaid, A. El Omri, and M. Rida, "Newest collaborative and hybrid network intrusion detection framework based on suricata and isolation forest algorithm," in *ACM International Conference Proceeding Series*, Association for Computing Machinery, Oct. 2019. doi: 10.1145/3368756.3369061.

[14] P. Veerasingam, S. Abd Razak, A. F. A. Abidin, M. A. Mohamed, and S. D. Mohd Satar, "INTRUSION DETECTION AND PREVENTION SYSTEM IN SME'S LOCAL NETWORK BY USING SURICATA," *Malaysian Journal of Computing and Applied Mathematics*, vol. 6, no. 1, pp. 21–30, Mar. 2023, doi: 10.37231/myjcam.2023.6.1.88.

[15] V. I. Sangadji, A. H. Muhammad, and E. Gunawan, "Penerapan Metode Signature Base Berbasis IDS Snort dan IDS Suricata Pada Keamaan Jaringan Laboratorium Komputer.," *Jurnal Teknik Informatika (J-Tifa)*, vol. 6, no. 1, pp. 18–22, Mar. 2023, doi: 10.52046/j-tifa.v6i2.1678.

[16] D. A. Bhosale and V. M. Mane, "Comparative study and analysis of network intrusion detection tools," in *2015 International Conference on Applied and Theoretical Computing and Communication Technology (iCATccT)*, IEEE, Oct. 2015, pp. 312–315. doi: 10.1109/ICATCCT.2015.7456901.

[17] A. Gupta and L. Sen Sharma, "Performance Evaluation of Snort and Suricata Intrusion Detection Systems on Ubuntu Server," 2020, pp. 811–821. doi: 10.1007/978-3-030-29407-6_58.

[18] A. Momand, S. U. Jan, and N. Ramzan, "A Systematic and Comprehensive Survey of Recent Advances in Intrusion Detection Systems Using Machine Learning: Deep Learning, Datasets, and Attack Taxonomy," 2023, *Hindawi Limited*. doi: 10.1155/2023/6048087.

[19] S. V. N. Santhosh Kumar, M. Selvi, and A. Kannan, "A Comprehensive Survey on Machine Learning-Based Intrusion Detection Systems for Secure Communication in Internet of Things," *Comput Intell Neurosci*, vol. 2023, no. 1, Jan. 2023, doi: 10.1155/2023/8981988.

[20] M. Tahir, U. Wahyuningsih, M. I. Putra Pratama, and M. A. Effindi, "Development of Network Security Using A Suricata-Based Intrusion Prevention System Againts Distributed Denial of Service," *Innovation in Research of Informatics (Innovatics)*, vol. 6, no. 2, pp. 41–48, Sep. 2024, doi: 10.37058/innovatics.v6i2.11187.

[21] A. Hussain, F. Aguilo-Gost, E. Simo-Mezquita, E. Marin-Tordera, and X. Massip, "An NIDS for Known and Zero-Day Anomalies," in *2023 19th International Conference on the Design of Reliable Communication Networks, DRCN 2023*, Institute of Electrical and Electronics Engineers Inc., 2023. doi: 10.1109/DRCN57075.2023.10108319.

[22] P. Dini, A. Elhanashi, A. Begni, S. Saponara, Q. Zheng, and K. Gasmi, "Overview on Intrusion Detection Systems Design Exploiting Machine Learning for Networking Cybersecurity," Jul. 01, 2023, *Multidisciplinary Digital Publishing Institute (MDPI)*. doi: 10.3390/app13137507.

[23] A. L. Buczak and E. Guven, "A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection," *IEEE Communications Surveys and Tutorials*, vol. 18, no. 2, pp. 1153–1176, Apr. 2016, doi: 10.1109/COMST.2015.2494502.

Vol. 6, No. 6, December 2025, Page. 5371-5385
https://jutif.if.unsoed.ac.id
DOI: https://doi.org/10.52436/1.jutif.2025.6.6.4694

[24] R. R. Asaad, "Penetration Testing: Wireless Network Attacks Method on Kali Linux OS," *Academic Journal of Nawroz University*, vol. 10, no. 1, pp. 7–12, Feb. 2021, doi: 10.25007/ajnu.v10n1a998.

[25] R. Gelar Guntara, "Pemanfaatan Google Colab Untuk Aplikasi Pendeteksian Masker Wajah Menggunakan Algoritma Deep Learning YOLOv7," *Jurnal Teknologi Dan Sistem Informasi Bisnis*, vol. 5, no. 1, pp. 55–60, Feb. 2023, doi: 10.47233/jteksis.v5i1.750.

[26] D. Bekerman, B. Shapira, L. Rokach, and A. Bar, "Unknown malware detection using network traffic classification," in *2015 IEEE Conference on Communications and Network Security (CNS)*, IEEE, Sep. 2015, pp. 134–142. doi: 10.1109/CNS.2015.7346821.

[27] F. Zhang, H. A. D. E. Kodituwakku, J. W. Hines, and J. Coble, "Multilayer Data-Driven Cyber-Attack Detection System for Industrial Control Systems Based on Network, System, and Process Data," *IEEE Trans Industr Inform*, vol. 15, no. 7, pp. 4362–4369, Jul. 2019, doi: 10.1109/TII.2019.2891261.