

Research Digest on Engineering Management and Social Innovations

Available on: <http://rdems.in>

DOI: 10.46647/rdems0204025

Volume: 02 Issue: 04, April, 2026

p-ISSN: 3117-4418

e-ISSN: 3117-4426

Network Traffic Anomaly Detection Using Hybrid Machine Learning and Generative AI: A SOC-Integrated Approach

¹Mrs N. Samatha, ²CH. Snehith, ³D. Sharon, ⁴D. Meghana Reddy

^{1,2,3,4}Dept of Computer Science and Engineering (Data Science), Vidya Jyothi Institute of Technology, Hyderabad, India

Abstract—Modern enterprise networks face an ever-expanding threat landscape characterized by zero-day exploits, polymorphic malware, distributed denial-of-service campaigns, and sophisticated insider threats that consistently evade signature based detection systems. Traditional Intrusion Detection Systems (IDS), despite decades of refinement, suffer from high false-positive rates, inability to detect novel attacks, and an absence of contextual explanation that forces security analysts to manually interpret raw machine-generated data. This paper presents the design, implementation, and experimental evaluation of a Network Traffic Anomaly Detection Security Operations Center (SOC) Dashboard that addresses these limitations through the integration of hybrid unsupervised machine learning and generative artificial intelligence. The system deploys a weighted ensemble of Isolation Forest and Local Outlier Factor (LOF) to compute continuous anomaly scores across 60,000 synthetic network flow records, classifying detected anomalies into four severity tiers: LOW, MEDIUM, HIGH, and CRITICAL. A deterministic rule engine operates in parallel, applying domain specific security rules to escalate alert severity for high-confidence attack signatures including port scanning, distributed denial of-service patterns, and command-and-control port usage. The central innovation of this work is the integration of the LLaMA 3.1 8B Instant large language model via the Groq API to generate automated, human-readable, MITRE ATT&CK-mapped triage reports for each detected alert, eliminating the need for manual expert interpretation and substantially reducing mean time to respond. The Streamlit-based interactive dashboard presents results across three analytical modules: Incident Detail and Explainability, Network Graph Visualization, and AI-Powered Triage. Experimental results demonstrate successful detection of all injected attack scenarios, generation of 4,800 severity classified alerts from 60,000 traffic events, and LLM response latency averaging approximately two seconds per query. This work demonstrates that combining unsupervised behavioral detection with generative AI explanation bridges the semantic gap between machine output and analyst understanding, enabling faster, more accessible, and more effective cybersecurity operations.

Index Terms—Network anomaly detection, Isolation Forest, Local Outlier Factor, large language models, SOC dashboard, generative AI, MITRE ATT&CK, cybersecurity, Streamlit, hybrid ensemble, alert triage.

I. INTRODUCTION

The rapid-fire digitization of enterprise structure, the proliferation of Internet of effects(IoT) endpoints, and the global shift toward pall-native infrastructures have unnaturally converted the network security geography. ultramodern enterprise networks induce business at preliminarily unconceivable scale — frequently exceeding millions of packets per nanosecond across hundreds of geographically distributed bias, services, and third- party integrations. Within this immense data volume, relating vicious or anomalous geste in real time represents one of the most critical and technically demanding challenges in contemporary cybersecurity. The trouble terrain has grown similarly sophisticated. Adversaries now routinely employ zero- day exploits that target preliminarily undisclosed vulnerabilities, polymorphic and fileless malware that evades hand-



Research Digest on Engineering Management and Social Innovations

Available on: <http://rdems.in>

DOI: 10.46647/rdems0204025

Volume: 02 Issue: 04, April, 2026

p-ISSN: 3117-4418

e-ISSN: 3117-4426

grounded discovery, multi- stage side movement juggernauts that unfold gradationally over days or weeks, and bigwig pitfalls involving compromised credentials or vicious workers whose geste deviates only subtly from normal nascences. The fiscal and functional consequences of delayed or missed discovery are severe, and traditional approaches are decreasingly inadequate. Traditional Intrusion Discovery Systems(IDS), which underpin the security posture of the vast maturity of enterprise surroundings, operate through hand grounded pattern match- ing. Incoming network business is compared against databases of given attack autographs, and matches detector cautions. While these systems give dependable discovery of entered pitfalls, they suffer from four abecedarian limitations that significantly constrain their effectiveness in ultramodern surroundings. First, hand- grounded systems are innately reactive, re- quiring that an attack be preliminarily observed and proved before a corresponding hand can be written, rendering them ineffective against zeroday attacks. Second, exorbitantly broad rule delineations constantly flag licit business as vicious, generating inordinate false-positive alert volumes that overwhelm security judges and beget genuine pitfalls to be overlooked amidst the noise. Third, the nonstop emergence of new attack variants requires ongoing homemade sig- nature conservation, placing a substantial functional burden on security brigades. Fourth, and utmost critically for functional effectiveness, being systems produce raw cautions conforming of IP addresses, harborage figures, protocol flags, and anomaly scores without contextual explanation, taking deep sphere moxie to interpret rightly under time pressure. The Security Operations Center(SOC), as the organizational whim-whams center responsible for nonstop monitoring, discovery, and response to security incidents, is particularly exposed to these limitations. SOC judges are routinely needed to triage hundreds of diurnal cautions with minimum automated contextual support. Inferior judges may warrant the experience necessary to directly interpret nebulous machine- generated signals, leading to delayed responses, incorrect groups, or critical pitfalls being dismissed as false cons. Machine literacy offers a unnaturally different paradigm for anomaly discovery rather than matching against known autographs, ML algorithms learn statistical models of normal network geste and identify diversions that may indicate pitfalls, including new attacks not preliminarily encountered. Unsupervised approaches are particularly precious in the SOC environment because labelled attack datasets are infrequently available in functional surroundings. still, the deployment of machine literacy in real- world SOC surroundings introduces a critical semantic gap the labours of ML algorithms — anomaly scores, decision function values, and point significance vectors are not directly interpretable by mortal judges without significant sphere moxie and fresh disquisition trouble. This gap between machine discovery and mortal understanding has been a persis- roof handicap to the functional relinquishment of ML- grounded security tools. The emergence of Large Language Models(LLMs) as important, instruction- following natural language creators provides a compelling result to this challenge. LLMs can consume structured machine- generated alert data and produce plain- English security analyses, effectively rephrasing between the language of machine literacy and the language of security operations. This paper presents the Network Traffic Anomaly Detection SOC Dashboard, a comprehensive system that combines unsupervised ML grounded anomaly discovery with generative AI- powered alert triage. The top benefactions of this work are(1) a weighted mongrel ensemble combining insulation timber and Original Outlier Factor with adaptive percentile- grounded inflexibility bracket;(2) a deterministic rule machine furnishing high perfection hand overlay for given attack patterns;(3) integration of the LLaMA 3.1 8B Instant language model via the Groq API for automated, MITRE ATTCK- counterplotted alert explanation;(4) a CSV train upload module enabling analysis of real- world external datasets; and(5) an interactive Streamlit-



Research Digest on Engineering Management and Social Innovations

Available on: <http://rdems.in>

DOI: 10.46647/rdems0204025

Volume: 02 Issue: 04, April, 2026

p-ISSN: 3117-4418

e-ISSN: 3117-4426

grounded SOC dashboard with logical modules presenting discovery results, network topology visualization, and AI- generated triage reports.

II. LITERATURE REVIEW

A. Classical Anomaly Detection Algorithms

The theoretical foundations of this work rest on two major algorithms from the anomaly discovery literature. insulation timber detects anomalies by segregating rare and statistically distinct data points through recursive arbitrary partitioning. Since anomalous points bear smaller splits to be insulated, they accumulate shorter path lengths, which serve as anomaly scores. The algorithm is computationally effective and well-suited for large- scale network business analysis. Original Outlier Factor(LOF) is a viscosity- grounded anomaly discovery algorithm that evaluates the original viscosity divagation of a point relative to its neighbours. A point is considered anomalous if its original viscosity is significantly lower than that of girding points. LOF is particularly effective for subtle anomalies that may be missed by global styles. A broader check of anomaly discovery ways classifies styles into statistical, propinquity- grounded, bracket- grounded, and information theoretic orders. This theoretical foundation informed the choice of algorithms and evaluation strategy used in this work.

B. Machine Learning-Based Network Intrusion Detection

Previous work in machine literacy- grounded network intrusion discovery highlights both the pledge and the challenges of similar systems. Unsupervised ways are seductive because they exclude reliance on labelled datasets, which are delicate to gain in realistic SOC surroundings. More recent studies have shown that deep literacy and mongrel machine learning approaches can ameliorate discovery capability across dynamic network surroundings. At the same time, experimenters have emphasized functional enterprises, including false cons and the gap between model affair and critic interpretation. This limitation explosively motivated the addition of an AI triage subcaste in the current system.

C. Large Language Models in Cybersecurity

The integration of LLMs into cybersecurity operations is a fleetly expanding area of exploration. LLMs are now being explored for tasks similar as alert triage, log summarization, trouble intelligence enrichment, and incident response backing. Their crucial value lies in their capability to restate complex machine labours into accessible language. In this design, the LLM is not used as the sensor itself; rather, it functions as an explanation machine that sits on top of anomaly discovery and rule- grounded waking to help judges with contextual interpretation.

III. PROBLEM STATEMENT AND DESIGN OBJECTIVES

A. Problem Statement

This design addresses four interrelated problems in contemporary network security operations. First, the incapability of being rule- grounded IDS to descry new, evolving, or zeroday network pitfalls that produce business patterns not matching any predefined hand. Second, the absence of automated intelligent systems able of assaying large volumes of network business logs and relating anomalous patterns with high delicacy and nicely low false-positive rates without taking labelled training data. Third, the lack of a mortal- readable explanation subcaste that contextualizes machine- generated cautions for security judges of varying moxie situations. Fourth, the difficulty of presenting complex, multi-dimensional security data in a format that's contemporaneously accessible to specialized judges and scrutable to non-technical stakeholders.

B. Design Objectives

Research Digest on Engineering Management and Social Innovations

Available on: <http://rdems.in>

DOI: 10.46647/rdems0204025

Volume: 02 Issue: 04, April, 2026

p-ISSN: 3117-4418

e-ISSN: 3117-4426

The primary objects of this work are to develop an ML grounded anomaly discovery system able of relating both known and unknown pitfalls through behavioral birth modeling, apply and compare multiple unsupervised algorithms, design an interactive Streamlit dashboard presenting detected anomalies and inflexibility groups, and integrate a generative AI subcaste generating plain- English alert explanations with MITRE ATT CK mapping.

Secondary objects include comprehensive data preprocessing and point engineering on a 60,000-record dissembled dataset, perpetration of a four- league inflexibility bracket model, creation of useful visualizations including network graphs and trend summaries, and designing a modular system armature that can be extended in unborn work.

IV. SYSTEM ARCHITECTURE AND DESIGN

A. Architectural Overview

The proposed system is structured as a multi-stage data processing channel that transforms raw network business logs into inflexibility- classified, LLM- annotated security cautions presented through an interactive SOC dashboard. The armature follows a modular design principle wherein each subcaste performs a separate processing function and passes structured affair to the coming stage.

The three- league software armature separates enterprises across a Data Layer, an operation Subcaste, and a donation Subcaste. External service integration with the Groq API is insulated within the LLM Summarization module, icing that API attainability degrades only the AI Triage functionality without affecting discovery and visualization. The overall architecture of the proposed system is shown in Fig. 1.

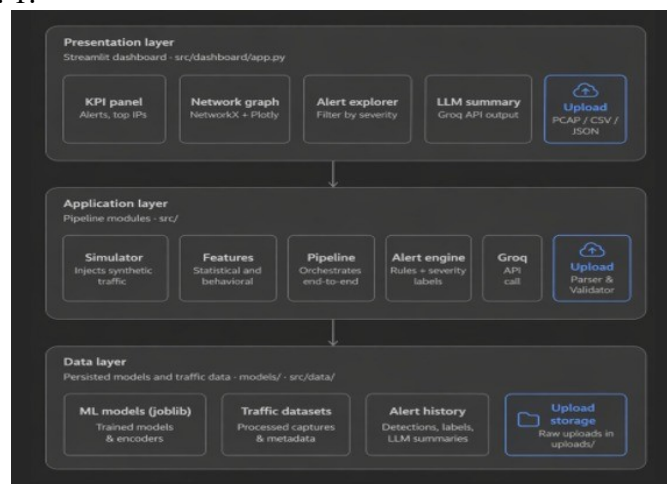


Fig. 1. System architecture of the proposed Network Traffic Anomaly Detection SOC Dashboard.

B. Feature Engineering Design

Feature engineering is the most critical preprocessing stage.

Raw network business fields similar as source IP, destination IP, harborage figures, protocol type, packet size, byte counts, and duration are inadequate collectively to distinguish attack patterns from licit variations. Behavioral features must be deduced through aggregation over time windows.

The system implements five-minute sliding window aggregations producing six primary behavioral features as defined in Table I.

Research Digest on Engineering Management and Social Innovations

Available on: <http://rdems.in>

DOI: 10.46647/rdems0204025

Volume: 02 Issue: 04, April, 2026

p-ISSN: 3117-4418

e-ISSN: 3117-4426

TABLE I
ENGINEERED FEATURES AND SECURITY SIGNIFICANCE

Engineered Feature	Security Significance
src_unique_ports_5m	Detects port scanning behaviour
src_unique_dsts_5m	Detects lateral movement
dst_unique_srcs_5m	Detects DDoS attack patterns
src_bytes_spike_score	Detects data exfiltration spikes
bytes_per_second	Detects high bandwidth anomalies
hour	Enables off-hours detection

The bytes spike score is computed using Z-score normalization relative to each source IP's historical transmission statistics:

$$\text{bytes sent} - \mu_i$$

$$\text{spike score} = \frac{\text{bytes sent} - \mu_i}{\sigma_i + \varepsilon} \quad (1)$$

$\sigma_i + \varepsilon$

where μ_i and σ_i represent the mean and standard deviation of bytes sent by source IP i , and $\varepsilon = 10^{-9}$ prevents division by zero.

C. Hybrid Anomaly Detection Engine

Two reciprocal unsupervised ML algorithms are trained on the regularized, finagled point matrix. insulation timber exploits the oddity and statistical distinctness of anomalous points to insulate them efficiently through arbitrary point partitioning. LOF computes the original reachability viscosity of each point relative to its neighborhood, producing a degree of- outlierness score.

The final hybrid score is computed as:

$$\text{score}_{final} = 0.6 \times \text{score}_{IF} + 0.4 \times \text{score}_{LOF} \quad (2)$$

The higher weight assigned to Isolation Forest reflects its stronger global anomaly detection performance. Severity thresholds are defined using adaptive percentile boundaries from the empirical score distribution.

D. Rule-Based Alert Engine

A deterministic rule machine operates in resemblant with the ML channel, applying sphere-specific security rules. Port checkup discovery fires when a source IP connections 25 or further unique destination anchorages within a five- nanosecond window. DDoS pattern discovery fires when 25 or further unique source IPs direct business toward a single destination. Suspicious harborage discovery flags business to anchorages 4444, 6667, and 9001. Rule successes escalate alert inflexibility.

E. Workflow and System Interaction

The internal flow of processes is illustrated through the Level 1 DFD in Fig. 2, while user interaction with the system is represented in Fig. 3. These diagrams summarize the path from data ingestion to alert generation, LLM summarization, and dashboard rendering.

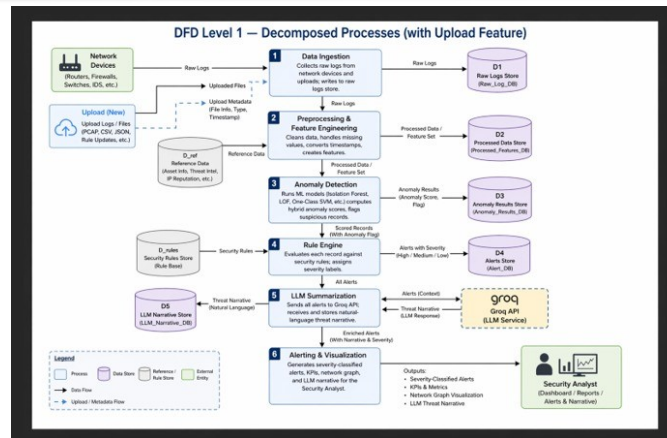


Fig. 2. Level 1 data flow diagram of the anomaly detection SOC system.

F. LLM Triage Integration

The AI Triage module integrates the Groq API with the LLaMA 3.1 8B Instant model. A structured prompt template containing alert metadata — source and destination IP, protocol, destination harboorage, anomaly score, inflexibility bracket, touched off rule, and crucial features — instructs the model to produce a plain- English explanation, likely attack type, MITRE ATT CK fashion mapping, and recommended critic conduct.

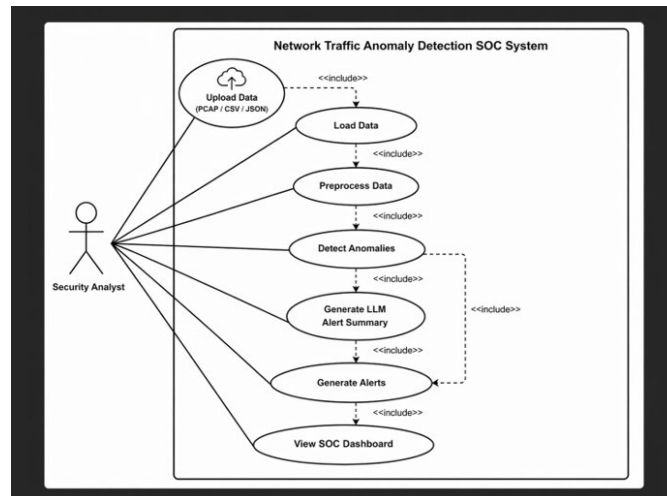


Fig. 3. Use case diagram showing interaction between the security analyst and the SOC system.

V. IMPLEMENTATION

A. Development Environment and Technology Stack

The system was developed on a Windows 11 workstation using Python 3.10 within an isolated virtual environment. Visual Studio Code served as the primary development environment. The implementation used Python, Streamlit, scikitlearn, Pandas, NumPy, NetworkX, Plotly, the Groq Python SDK, and Git.

TABLE II

DEVELOPMENT TOOLS AND TECHNOLOGIES

Tool Technology	Purpose
Python 3.10	Primary programming



Research Digest on Engineering Management and Social Innovations

Available on: <http://rdems.in>

DOI: 10.46647/rdems0204025

Volume: 02 Issue: 04, April, 2026

p-ISSN: 3117-4418

e-ISSN: 3117-4426

	language
Visual Studio Code	Integrated development environment
Streamlit	Web-based dashboard framework
Scikit-learn	Machine learning model implementation
Pandas / NumPy	Data manipulation and numerical computing
NetworkX / Plotly	Network graph construction and visualization
Groq API (LLaMA3)	LLM-powered alert triage
Git	Version control

B. Data Generation and Preprocessing Pipeline

A synthetic network business creator simulates realistic commercial enterprise business. The creator produces 60,000 inflow records containing source and destination IP addresses, anchorages, protocols, byte volumes, packet counts, and timestamps gauging a realistic business period.

Attack scripts are designedly fitted at a 5 rate to pretend real world pitfalls including harborage reviews, DDoS patterns, side movement, and out- hours data transfers. The preprocessing channel parses timestamps, handles missing values, removes duplicates, and caps outliers at the 99th percentile to help model deformation.

C. Model Training and Score Fusion

Following feature engineering, the processed dataset is used to fit both models. Isolation Forest is trained with 200 estimators and a contamination rate of 5%. LOF is trained with novelty detection enabled, allowing it to score new incoming traffic against the learned baseline.

The two model outputs are independently normalized and fused through weighted score averaging. Severity levels are assigned dynamically based on score percentiles.

D. Dashboard Architecture and Modules

The Streamlit dashboard integrates all system components into a unified interface. The main analytical tabs are:

- Incident Detail and Explainability
- Network Graph View
- AI Triage

The implemented dashboard supports both synthetic simulation and CSV upload for user-supplied traffic data. A representative dashboard output can be shown as in Fig. 4.

Research Digest on Engineering Management and Social Innovations

Available on: <http://rdems.in>

DOI: 10.46647/rdems0204025

Volume: 02 Issue: 04, April, 2026

p-ISSN: 3117-4418

e-ISSN: 3117-4426

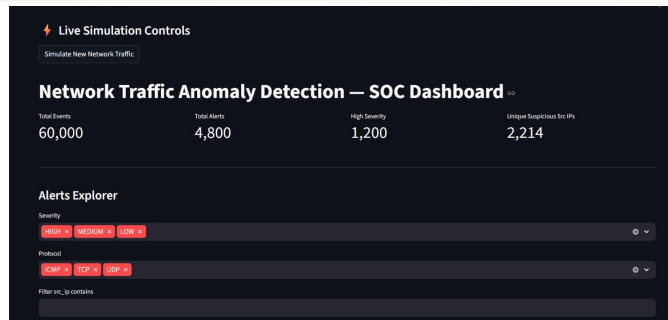


Fig. 4. SOC dashboard showing anomaly analysis, visualization, and AI powered triage.

E. File Upload Module

A file upload module was implemented using Streamlit's file_uploader, allowing users to upload custom datasets in CSV format. Uploaded files are validated for schema consistency and then passed through the same preprocessing, feature engineering, model scoring, rule evaluation, and alert generation pipeline used for synthetic data.

F. Testing and Validation

Unit testing validated each channel stage singly. The synthetic data creator was vindicated to produce datasets with anticipated field distributions and attack rates. The feature engineering module was validated using manually reckoned values for known scripts. Integration testing vindicated the end- to- end data inflow between modules.

VI. RESULTS

A. Pipeline Execution and Detection Metrics

Execution of the complete pipeline on the 60,000-record synthetic dataset produced the results summarized in Table IV. The complete pipeline executes in approximately 4.5 minutes on the development hardware. The hybrid ensemble identified 4,800 alerts, representing 8% of total traffic events.

TABLE III

INTEGRATION TESTING RESULTS SUMMARY

Test Case	Expected Result	Status
Data generation produces 60,000 records	60,000 rows in output CSV	PASS
Port scan feature correct for known case	src_unique_ports_5m = 30	PASS
Isolation Forest flags injected anomalies	Anomaly score above threshold	PASS
Groq API returns structured explanation	JSON-formatted triage response	PASS
Dashboard renders all three tabs	Interactive rendering	PASS



Research Digest on Engineering Management and Social Innovations

Available on: <http://rdems.in>

DOI: 10.46647/rdems0204025

Volume: 02 Issue: 04, April, 2026

p-ISSN: 3117-4418

e-ISSN: 3117-4426

TABLE IV
IMPLEMENTATION OUTPUT SUMMARY

Metric	Value
Total traffic events processed	60,000
Total alerts generated	4,800 (8.0%)
CRITICAL severity alerts	~ 240 (5% of alerts)
HIGH severity alerts	~ 384 (8% of alerts)
MEDIUM severity alerts	~ 720 (15% of alerts)
LOW severity alerts	~ 3,456 (72% of alerts)
Rule engine escalations	312 alerts escalated
Models trained	Isolation Forest + LOF
LLM model used	LLaMA3 Groq API
Average LLM response time	~ 2.1 seconds per query
Dashboard load time	<4 seconds
Complete pipeline runtime	~ 4.5 minutes

B. Severity Distribution Analysis

The severity distribution—72% LOW, 15% MEDIUM, 8% HIGH, and 5% CRITICAL—reflects the expected real-world skew toward lower-severity anomalies. High and CRITICAL alerts predominantly correspond to injected port scan and DDoS scenarios. MEDIUM alerts are often associated with suspicious ports and moderate-intensity lateral movement indicators. LOW alerts capture borderline behavioral deviations that may warrant monitoring.

C. Algorithm Comparative Analysis

Insulation timber demonstrated stronger global anomaly discovery performance, particularly for high- volume attack patterns including DDoS and large- scale harborage scanning. LOF handed reciprocal perceptivity to subtle localized anomalies. The weighted emulsion strategy prioritized the union of both models' strengths and bettered discovery content across different scripts.



Research Digest on Engineering Management and Social Innovations

Available on: <http://rdems.in>

DOI: 10.46647/rdems0204025

Volume: 02 Issue: 04, April, 2026

p-ISSN: 3117-4418

e-ISSN: 3117-4426

The rule machine further bettered performance by raising low- scoring but high- confidence suspicious actions, especially business involving well- known suspicious anchorages.

D. LLM Triage Performance

The LLaMA 3.1 8B model produced structured, contextually accurate alert analyses within roughly two seconds per query. Triage labors rightly linked likely attack types and counterplotted detected actions to MITRE ATT CK ways similar as T1046 for harborage reviews and T1498 for DDoS exertion. The explanations were presented in critic-friendly language and included suggested remediation way.

E. Dashboard Performance

The Streamlit dashboard loaded within four seconds following channel completion. The network graph view successfully rendered graphs with hundreds of bumps and thousands of edges without significant performance declination. The AI Triage tab maintained usable real- time response characteristics for critic workflows.

VII. DISCUSSION

A. Practical Implications

The hybrid ML-LLM architecture directly addresses three major limitations of traditional IDS: inability to detect unknown threats, difficulty in prioritizing alerts effectively, and lack of contextual explanation. Behavioral anomaly detection captures novel attack patterns, adaptive severity classification supports prioritization, and LLM-generated triage reduces the manual investigation burden.

The CSV upload feature increases the system's practical value by allowing analysts to apply the same detection and triage workflow to external datasets rather than relying only on simulation.

B. Comparison with Prior Work

Compared to traditional intrusion detection systems like Snort and Suricata based on the signatures method, our solution can detect the unknown threats with the help of learned baselines, automatic natural language parsing of alerts, and a dashboard that integrates all three functions: detection, visualization, and AI analysis. Compared to the pure machine learning algorithms, rule engines provide better accuracy in recognizing suspicious behavior patterns.

C. Limitations and Threats to Validity

Testing is performed using artificial data sets, and even though the attacks are created based on real-life scenarios, synthetic data does not fully represent actual settings. This could be remedied by evaluating the system in open-source test data sets, like the NSL-KDD or the CICIDS dataset. The LOF algorithm could also have difficulty scaling to massive data sets, and the need for an external LLM API creates problems in isolated networks.

VIII. CONCLUSION AND FUTURE WORK

A. Conclusion

The Network Traffic Anomaly Detection SOC Dashboard proposed in this study uses the concept of integrating unsupervised learning techniques with an alert triaging capability using a generative AI system. The use of Isolation Forest and Local Outlier Factor with weights, along with the rule engine and the explanation mechanism with LLaMA 3.1 8B, offers multidimensional anomaly detection and is also supported by human analysts. Experimental evaluation conducted on 60,000 simulated entries for the network traffic proved the capability of detecting the injection of malicious packets as well as generating alerts based on their severity classification.



Research Digest on Engineering Management and Social Innovations

Available on: <http://rdems.in>

DOI: 10.46647/rdems0204025

Volume: 02 Issue: 04, April, 2026

p-ISSN: 3117-4418

e-ISSN: 3117-4426

B. Future Work

The future development paths include real-time packet capture integration using libraries such as Scapy or PyShark, adding models like LSTM and Transformer for detecting multi-stage attacks, federated learning capabilities, containerization using Docker, integration into SOAR platforms for automated responses, deployment of local large language models for eliminating cloud dependency, and benchmarking against public data sources.

ACKNOWLEDGMENT

The authors sincerely thank N. Samatha, Assistant Professor, for her expert guidance, timely feedback, and continuous mentorship throughout this project. Gratitude is also extended to the department and institution for their support and infrastructure.

REFERENCES

- [1] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation Forest," in *Proc. 8th IEEE Int. Conf. Data Mining*, 2008, pp. 413–422.
- [2] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander, "LOF: Identifying Density-Based Local Outliers," in *Proc. ACM SIGMOD Int. Conf. Management of Data*, 2000, pp. 93–104.
- [3] R. Sommer and V. Paxson, "Outside the Closed World: On Using Machine Learning for Network Intrusion Detection," in *Proc. IEEE Symp. Security and Privacy*, 2010, pp. 305–316.
- [4] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly Detection: A Survey," *ACM Comput. Surv.*, vol. 41, no. 3, 2009.
- [5] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A Detailed Analysis of the KDD Cup 99 Data Set," in *Proc. IEEE Symp. Comput. Intell. Security Defense Appl.*, 2009.
- [6] S. Alsoufi *et al.*, "Leveraging Deep Learning and Feature Extraction for Robust Anomaly Detection in Network Traffic," in *Proc. IEEE Int. Conf. Emerging Technologies*, 2024.
- [7] Y. Zheng *et al.*, "Machine Learning-Based Network Anomaly Detection: Design, Implementation, and Evaluation," *AI*, vol. 5, no. 4, pp. 2967–2983, 2024.
- [8] N. Georgiou *et al.*, "AI-Augmented SOC: A Survey of LLMs and Agents for Security Automation," *Computers*, 2025.
- [9] B. Wei, L. Chen, and R. Panda, "CORTEX: Collaborative LLM Agents for High-Stakes Alert Triage," *arXiv preprint*, 2025.
- [10] F. Pedregosa *et al.*, "Scikit-learn: Machine Learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.
- [11] Streamlit Inc., "Streamlit Documentation: Build and Share Data Apps," 2024.
- [12] Groq Inc., "Groq API Documentation: LPU Inference Engine," 2024.
- [13] D. E. Denning, "An Intrusion-Detection Model," *IEEE Transactions on Software Engineering*, vol. SE-13, no. 2, pp. 222–232, 1987.
- [14] G. Creech and J. Hu, "A Semantic Approach to Host-Based Intrusion Detection Systems Using Contiguous and Discontiguous System Call Patterns," *IEEE Transactions on Computers*, vol. 63, no. 4, pp. 807–819, 2014.
- [15] W. Wang, M. Zhu, J. Wang, X. Zeng, and Z. Yang, "End-to-End Encrypted Traffic Classification With One-Dimensional Convolution Neural Networks," *IEEE Access*, vol. 5, pp. 17528–17539, 2017.
- [16] N. Moustafa and J. Slay, "UNSW-NB15: A Comprehensive Data Set for Network Intrusion Detection Systems," in *Proc. Military Communications and Information Systems Conference*, 2015.



Research Digest on Engineering Management and Social Innovations

Available on: <http://rdems.in>

DOI: 10.46647/rdems0204025

Volume: 02 Issue: 04, April, 2026

p-ISSN: 3117-4418

e-ISSN: 3117-4426

-
- [17] Y. Xin, L. Kong, Z. Liu, Y. Chen, Y. Li, H. Zhu, M. Gao, H. Hou, and C. Wang, "Machine Learning and Deep Learning Methods for Cybersecurity," *IEEE Access*, vol. 6, pp. 35365–35381, 2018.
 - [18] H. Hindy et al., "A Taxonomy of Network Threats and the Effect of Current Datasets on Intrusion Detection Systems," *IEEE Access*, vol. 8, pp. 104650–104675, 2020.
 - [19] OpenAI, "GPT-4 Technical Report," *arXiv preprint arXiv:2303.08774*, 2023.
 - [20] A. T. Nguyen and H. N. Nguyen, "Using Large Language Models for Cybersecurity Threat Analysis," *Computers & Security*, Elsevier, 2024.