

AI-Augmented Network-Forensics: Leveraging LLMs for Real-Time Threat Detection and Automated Response in Enterprise Environments

 **Mohammed Imran Choudhary**¹

¹ Senior Software Engineer Santa Clara, CA 95051, USA

Email: mimran.mkc@gmail.com

Abstract: In modern enterprise networks, complicated rule-based signatures, fragmented alerts, encrypted traffic, and analyst workloads are delaying the ability to recognize and contain incidents, as the need grows for faster correlation of heterogeneous telemetry. This study evaluates an LLM-augmented network-forensics architecture for threat detection, evidence interpretation, and controlled automated response, while retaining deterministic security controls. The 30-day controlled digital-twin experiment in the medium-sized hybrid enterprise resulted in about 18.6 million security events. The events were correlated into 1,200 incident windows: 480 malicious and 720 benign. Precision, F1-score, Recall, FPR, ROC-AUC, investigation latency, and response accuracy were used to compare the proposed hybrid system with XGBoost, SIEM Rules, and a transformer-based anomaly detection architecture. In the controlled digital-twin evaluation, the LLM-augmented hybrid system achieved an F1 score of 94.2%, a false-positive rate of 3.1%, a mean detection time of 2.6 minutes, a mean response time of 13.4 minutes, and a response-recommendation accuracy of 92.1%. Through retrieval-augmented generation, either with or without schema validation, the amount of unsupported claims decreased from 14.6% to 3.8%. Results showed that LLM performance as contextual reasoning, explanation, and orchestration components in guarded workflows was most effective. For high-impact actions impacting critical assets, privileged identities, or production systems, human approval was required. Multi-enterprise validation, adversarial testing, and privacy-preserving model adaptation should be addressed in the future.

Keywords: AI-Augmented Network-Forensics, Large Language Models, Real-Time Threat Detection, Automated Incident Response, Retrieval-Augmented Generation, Enterprise Cybersecurity.

1. Introduction

Network-forensics involves gathering, maintaining, correlating, and analyzing network data to identify and analyze security events and build security incidents. Traditionally, after a compromise, investigators were left with retrospective packet capture and manual log examination. Modern businesses, however, must be forensic-ready because attacks can take place in minutes across endpoints, identities, networks, and cloud services. Real-time analysis involves merging NetFlow and IPFIX records, Firewall events, DNS/Proxy logs, Endpoint detection and response alerts, Authentication traces, Cloud audit trails,

and threat-intelligence indicators [1]. There can be millions of events a day from these heterogeneous sources, making manual correlation impractical. Traditional security information and event management appliances typically use signatures, thresholds, and ready-made correlation rules to first categorize an alert as possibly

malicious for analysts to investigate. Those controls are useful for speed and reproducibility, but do not provide the contextual reasoning of evidence spread out over time. According to data from the 2025 Verizon Data Breach Investigations Report, 22% of initial accesses were through credential abuse and 20% were

through exploitation of vulnerabilities among 22,052 incidents and 12,195 confirmed breaches [2]. Verizon's 2026 report, in turn, identified exploiting vulnerabilities as the most prevalent attack method. Large language models provide additional functions for semantic interpretation, alert summarization, threat intelligence retrieval, tool invocation, incident reconstruction, and response plan generation.

Security operations centers in enterprises are inundated with thousands of alerts, only a small fraction of which need to be contained or investigated. Evidence is spread across platforms like SIEM, network detection, endpoint, identity, cloud, and ticketing services, with incompatible schemas and timestamps. Traditional intrusion detection systems can detect suspicious traffic or known markers without providing information about the relationships between events, which portions are impacted, which MITRE ATT&CK techniques are in evidence, or which containment action should take priority. MITRE ATT&CK offers an organized knowledge base for mapping observed behavior and correlating to adversary tactics and techniques. This fragmentation makes investigations take longer, increases the number of duplicate investigations, and potentially increases the likelihood of low-volume multistage attacks being ignored.

Direct use of a general-purpose LLM as a stand-alone detector is insufficient for enterprise network forensics. General-purpose LLMs can hallucinate evidence, follow malicious instructions present in logs, reveal sensitive information, make nondeterministic decisions, or have inference latency. The experiments reveal that such models can only explain and respond with retrieval-augmented generation and function calling, but as a whole are not effective in precisely identifying malicious NetFlows. LLM-based reasoning, grounded retrieval, and policy-driven automated actions are combined with deterministic detection in the unresolvable gap, within a validated enterprise architecture in real time.

This study has four objectives. It aims to build an AI-augmented network-forensic architecture that brings all of the above together. It also strives to assess the threat-detection performance in terms of precision, recall, F1-score, false-positive rate, and receiver operating characteristic area. The study assesses whether the mean time to detect (MTTD), mean time to

investigate (MTTI), and mean time to respond (MTTR) was reduced by using LLM-assisted correlation and triage. It also aims to test the accuracy, safety, explainability, and usefulness of the recommendations when operated in human-approved and limited-autonomy modes.

Digital twin boundaries for the investigation were limited to a medium-sized hybrid organization, with some 600 virtual endpoints, 24 server workloads, and six segmented networks. Testing included credential abuse, Web exploits, command-and-control communications, lateral movement, privilege escalation, data staging and exfiltration, ransomware, and distributed denial-of-service attack activity. Only network and security metadata were analyzed, not all employee content. The LLM functions included enrichment, correlation, explanation, evidence-based recommendation, and response orchestration. Low-risk actions allowed included IP blocking, endpoint isolation, session revoking, and ticket creation. High-impact actions that could not happen without analyst approval, offensive exploit generation, destructive actions, and unconstrained autonomous remediation were not accounted for.

The study is presented in a series of chapters. Chapter 2 conducts a literature survey to analyze network-forensics, the use of AI for finding cyber threats, LLMs in cybersecurity, and automated cybersecurity response. Chapter 3 discusses the digital-twin design, data collection,

architecture, analysis, and safeguards. Chapter 4 reports on detection, latency, explainability, response, and operational findings, and Chapter 5 summarizes key findings and contributions, concluding the study. The paper is organized into a threat-informed paper flow that includes evidence collection, detection, correlation, reconstruction, decision support, and response action.

2. Literature Review

2.1 Conventional Network-Forensics and Enterprise Intrusion Detection

Conventional network-forensics involves packet captures, flow-based monitoring, deep packet inspection, and log analysis to piece together communications and retain evidentiary timelines. While

packet capture provides payload visibility, NetFlow and IPFIX summarise conversations with minimal storage usage [13]. A deterministic signature can be used to detect indicators of command and control, exploits, or protocols by Snort or Suricata. Behavioral systems help signatures by comparing statistical baselines with traffic traits, such as destinations, protocol utilization, authentication characteristics, and traffic rate. The SIEM platform correlates alerts, enriches cases, and enables analysts to investigate [14]. However, static rules generate a large number of alerts, cannot detect new behaviors, correlate different types of evidence imperfectly, and are less informative when payloads are encrypted. Packet retention drives storage and indexing costs. By

connecting observed behaviors to adversary tactics, techniques, and sub-techniques, MITRE ATT&CK can help improve interpretation by providing structure and prioritizing hypotheses when responding to incidents [15]. Protected APIs, along with encryption, tokenization, and auditable transaction controls, are key to securing enterprise integration. Research shows a 13% higher rate of payment success, 30% quicker reconciliation, and 50% fewer fraud cases in an Oracle Fusion framework [16]. However, deterministic tools are still essential, as they can produce outputs rapidly and consistently and can be explained, presented, and verified during an evidential audit.

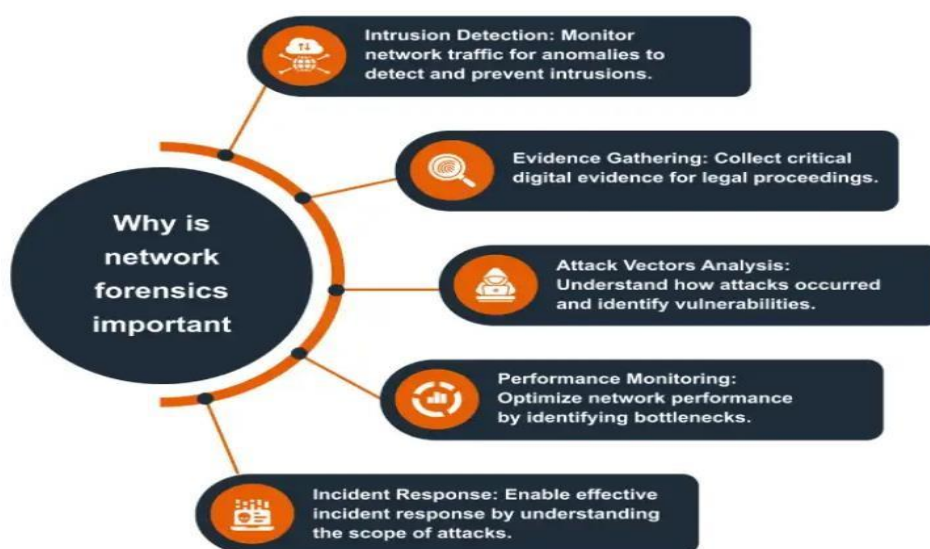


Figure 1: Core functions of network-forensics in enterprise intrusion detection.

Figure 1 demonstrates that traditional network-forensics plays a key role in enterprise intrusion detection. It illustrates the value of continuous traffic monitoring for anomaly detection, evidence collection, analysis of attack vectors, performance monitoring, and incident response. These functions support packet capture, deep packet inspection, flow monitoring, signature-based detection, SIEM correlation and behavioral analytics, and MITRE ATT&CK mapping. Together,

they provide auditable, low-latency, repeatable security measures for organizing and maintaining enterprise security in hybrid environments.

2.2 Machine Learning and Deep Learning in Network Threat Detection

The rule-based monitoring approach involves learning discriminative patterns from labeled traffic using machine learning intrusion detection. Random forests, XGBoost, support-vector machines, and logistic regression classify flows using flow features such as flow duration, byte counts, packet direction, port usage, and connection frequency [17, 18]. Some proposed alternative deep learning networks include convolutional networks, recurrent neural networks and long short-term memory networks, autoencoders for unsupervised anomaly detection, graph neural networks for relationships with hosts, and transformers for long-range event dependency.

Most of these models use fixed testing datasets, such as CICIDS2017, UNSW-NB15, and CSE-CIC-IDS2018, to validate the data and achieve higher accuracy, recall,

precision, a low false-positive rate, a high F1-score, and low detection latency [9]. However, accuracy can be exaggerated by class imbalance, leakage can be an issue when training on test data, and lab traffic does not necessarily represent enterprise encryption, users, or software. Legitimate behavior and attacker techniques also evolve, with added performance drift due to concept drift. Cross-industry evidence demonstrates the benefits of AI under controlled data and business environments; in the retail sector, the impact of personalization using AI was estimated to lead to 15-20% growth in retail sales and a 12% growth in conversions [10]. These findings indicate promise, although they are not necessarily valid for enterprise-scale applications; benchmark scores achieved with these systems need to be validated under different time periods.

2.3 LLM Applications in Network-Forensics and Cybersecurity

Large language models expand network-forensic automation by summarizing and condensing alerts, coordinating tools, sharing threat intelligence, mapping behavior to MITRE ATT&CK, providing natural-language responses, summarizing and padding detection rules, writing forensic reports, and suggesting responses and remediation measures [11]. They derive their usefulness from semantic reasoning over evidence, which regular detectors process separately. Existing literature on the use of generative AI for synthetic accident reconstruction enabled the construction of an explanatory sequence from the disjointed observations and the assignment of liability [12]. The same capability applies to cyber incidents, except that every step performed must be traceable back to any network, endpoint, identity, or cloud evidence.

Three operational functions can be differentiated. As a detector, LLM identifies flows or logs, limited by token cost, latency, context limits, and inconsistent outputs, in order to be deployed at packet scale. It is a forensic analyst, interpreting alerts, pulling out knowledge, outlining relationships, and revealing missing evidence [13]. It is a controller that selects approved tools, sends

queries, and manages the response workflow. This literature supports a hybrid approach, with high-speed signatures and trained classifiers for anomaly detection, while retrieval-based LLMs produce evidence-based correlations, convey ambiguity, and write contextual anomalies for analyst verification.

2.4 Automated Response, SOAR and Human-in-the-Loop Governance

Validated alerts can be transformed into repeatable playbooks using security orchestration, automation, and response platforms. Actions include denying access to malicious IPs, revoking accounts, terminating session, updating firewall rules, quarantining files, and opening tickets for

investigation [14]. It can still be made as a recommendation only, require or desire human involvement, or be autonomous. They are selected based on asset criticality and evidential confidence, business impact, and reversibility. There is no issue with opening a ticket, but there may be issues if an administrator account is deactivated that may impact services. Governance should facilitate automation in the context of resilience objectives, observability, and recovery.

Researchers related the concepts of site reliability engineering (SRE), including predictive monitoring, reliability measurement, and resilience, with enterprise architecture [15]. LLM recommendations should be converted into a formal command that has a command schema rather than executed as a natural language command. There are mandatory safeguards, such as allow-listed tools, confidence thresholds, role-based access, rollback, immutable audit logs, approval gates, and model output validation. Ownership, monitoring, testing, and accountability are also key elements of NIST-aligned governance [16]. It allows automatic, reversible containment above a level of confidence while leaving destructive actions like removing and dropping all the data within an identifier, or business imperatives, for authorized analysts to review the cited evidence and blast radius.

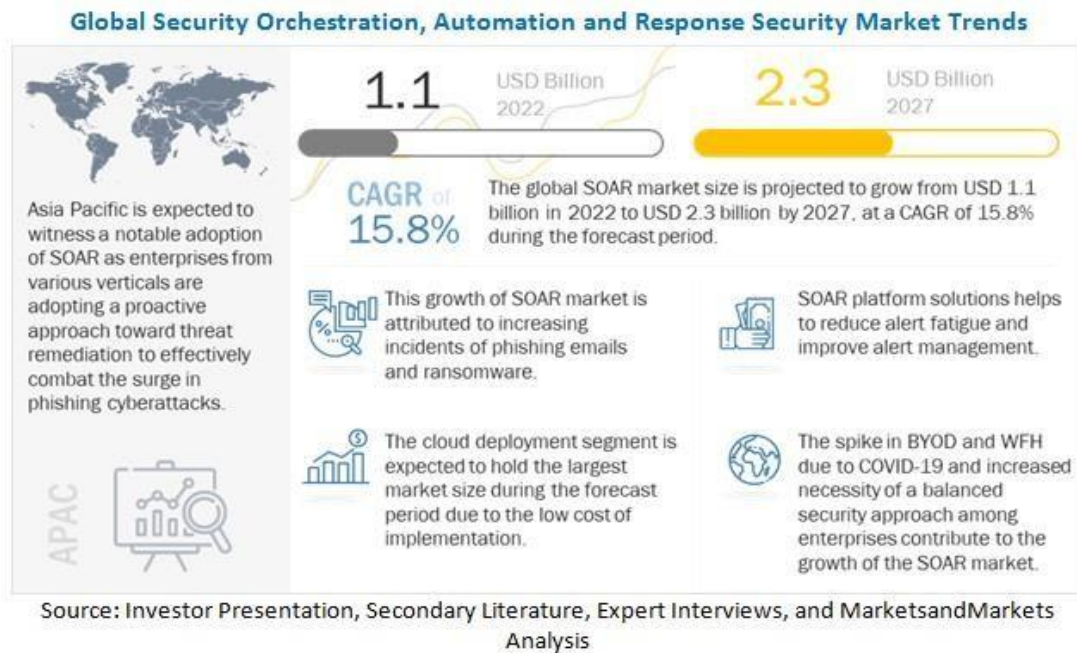


Figure 2: Projected Global SOAR market growth 2022-2027.

Figure 2 highlights projected Security Orchestration, Automation, and Response market growth for the years 2022 to 2027, which is forecasted to rise from USD 1.1 billion in 2022 to USD 2.3 billion in 2027 at a 15.8% CAGR. It links this explosion to growing phishing, ransomware, cloud usage, BYOD trends, and remote workers. It further reinforces the importance of balanced governance, analyst oversight, and controlled automation across complex and regulated enterprise security environments, as well as the value of SOAR platforms to alleviate alert fatigue, streamline alert management, and provide scalable, playbook-driven responses.

2.5 Limitations and Research Gaps in the Literature

The literature is still limited by reliance on static snapshots of benchmark traffic and by the lack of, or poor, mimicry of hybrid enterprise traffic. Many appraisals report classifier accuracy rather than end-to-end latency, compute usage, analyst burden, or integration into SIEM and SOAR platforms. LLM studies focus only on the percentage of hallucinated claims, the usefulness of explanations, analyst belief, and repeatability. There is inadequate testing for malicious builds that are injected in this way. However, attacker-controlled strings might manipulate tool requests, and summaries might be manipulated in some way.

Reconstruction is weakened by encrypted traffic, a lack of sensors, clock drift, and inconsistent identifiers. The few existing studies rarely set clear limits between deterministic evidence processing and probabilistic

reasoning, leaving it unclear as to which outputs are admissible for automated action. Response safety is assessed less frequently, and there are concerns about privacy, sovereignty, and regulations for external-model processing. There is limited generalizability across enterprises and industry. The gaps are addressed by the proposed study, which features an enterprise digital twin, a hybrid detector-LLM architecture, retrieval-augmented grounding, schema-constrained outputs, guarded response execution and detection, latency and explanation, safety, and operational outcomes.

3. Methods and Techniques

3.1 Data Collection Methods

The study collected the telemetry data from a 30-day controlled experiment performed in a medium-sized hybrid-enterprise digital-twin environment. This environment included six segmented networks, 24 additional servers, roughly 1,500 simulated user identities, and 600 virtual endpoints. The sensors collected the Zeek connection information, NetFlow/IPFIX summaries, DNS traffic and information, firewall and proxy logs, Windows and Linux authentication logs, endpoint detection and response alerts, Active Directory security information, cloud audit trails, vulnerability-scanner reports, and crafted threat-intelligence indicators and red-team execution logs. Event timestamps and event sources were saved in a streaming pipeline before being ingested into central storage. This design showed how real-time telemetry and stream-processing components, which

could be scaled up or down based on demand, could be used in predictive systems [17]. Collection clocks were synced using Network Time Protocol, and retention ensured raw records were retained, along with hashes that could not be edited for evidential verification and replay.

There were about 18.6 million raw events in the collected data. The events were correlated to form 1,200 windows, each representing either a malicious event or a benign event. The malicious sample comprised a total of 60 windows across the following categories: credential misuse, web exploitation, command-and-control activity, lateral movement, privilege escalation, data staging and exfiltration, ransomware and distributed denial-of-service activity. The sampling technique used in this study was stratified purposive sampling that aimed for equal representation across malicious categories. Benign windows were randomly chosen during business hours, overnight processing, patching and backups, software deployment and administration. MITRE ATT&CK techniques were mapped to attack sequences. All labels were verified independently by two analysts with disagreements adjudicated by a senior analyst.

3.2 Data Analysis

To ensure that all records were as clean and deduplicated as possible, the data was timestamp-normalized and pseudo-anonymized prior to analysis. Malformed events, duplicated retransmissions, and events that did not have all required identifiers were excluded. Some of the correlation keys were source and destination address, user identity, hostname, session identifier, process context, and temporal proximity. The 1,200 windows were split chronologically into 720 training windows, 240 validation windows, and 240 testing windows, which account for 60%, 20%, and 20%, respectively. The temporal partitioning minimized temporal leakage and maintained realistic concept changes.

Four systems were compared: SIEM correlation rules, an XGBoost intrusion detector, a transformer-based log anomaly detector, and the proposed architecture, an LLM-augmented solution. Precision, recall, F1-score, ROC-AUC, false-positive rate, time to detection, investigation time and response time, and recommendation accuracy for response were computed. Paired detection outcomes were compared using McNemar's test, and 95% bootstrap confidence intervals were estimated from 1000 resamples. Data were tested for statistical significance at $p < 0.05$.

Response-time distributions were right-skewed, so a median and 95th-percentile latency were used. During the testing phase, it was also tested to see how automated prioritization can shorten the manual response time of 15-30 minutes, which is observed in conventional incident handling [18]. Confusion matrices were created for each threat type, with timing starting when the event arrived on the scene and ending after recommendation for containment.

3.3 Proposed AI-Augmented Network-Forensic Architecture

The proposed architecture consists of four layers. Security events were ingested via a streaming message broker; schema validation was performed, vendor-specific records were translated to a common event format, and timestamp normalization was completed in the ingestion layer [19]. Personally, identifiable information was masked or tokenized before LLM inference. The detection layer performed signatures over known indicators and applied statistical baselines, XGBoost, and transformer models to detect anomaly behaviors [20]. Only enriched alerts and bounded evidence windows were forwarded, thus reducing the amount of tokens consumed and preventing exposure of unlimited evidence logs. The reasoning layer was able to obtain descriptions of MITRE ATT&CK, internal asset classifications, approved playbooks for incident response, descriptions of past incidents, and threat intelligence records.

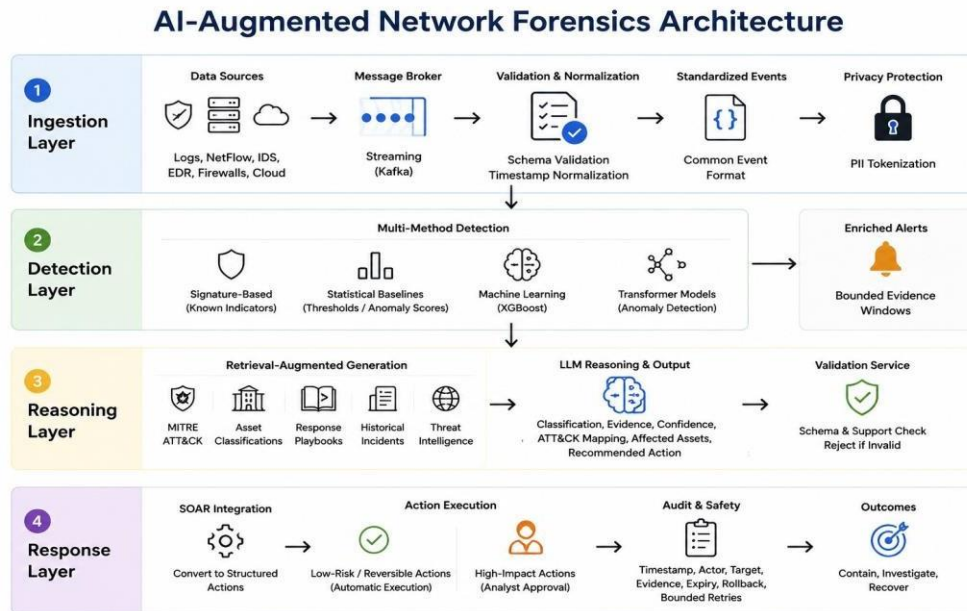


Figure 3: Proposed four-layer AI-augmented network-forensics architecture.

Figure 3 shows the 4-layer network-forensic architecture with AI capability. Security telemetry is ingested, standardized, validated, privacy-protected, and detected using signature, XGBoost, statistical, and transformer-based methods. Then, enriched alerts go into the LLMs for retrieval-based reasoning to classify the alert, cite evidence, score confidence, map to ATT&CK, and provide a response suggestion. Validated outputs are turned into controlled SOAR actions, which can be executed automatically with low risk, auditing, rollback, analyst approval, and bounded retries.

For each incident, the reasoning layer produced a classification, supporting evidence references, a confidence score, an ATT&CK mapping, the impacted assets, and a recommended response action. A validation service determined that conclusions not supported by a retrieval or conclusions that did not follow the output schema were incorrect [21]. The action layer generated structured SOAR actions based upon the accepted recommendations. Actions were automatically performed for reversible and low-risk decisions and manually decided for high-impact ones. This integration strategy was inspired by the general rule that enterprise workflows needed to be automated. At the same time, systems relied on valid API providers, inbound data must be synchronized, and a journal-style audit trail was essential for compliance and risk management, as evidenced in the Oracle Fusion & ADP integration research [22]. Each action included a timestamp, a responsible identity, and a target, a reference to the evidence, an expiry condition, and a rollback procedure.

Action loops were restricted by capping retries.

3.4 Reliability, Security, Ethical and Governance Controls

Reliability and governance controls in the experiment were built-in. Selected logs were inserted with prompt-injection strings to observe if classes were changed or any tools were requested upon prompt injection. The LLM was not given any network, shell, database, or operating system access. Available functions were allow listed and enforced using role-based permissions. All responses were verified using a pre-established JSON schema, and a score of 0.85 or higher was necessary to execute a reversible, low-risk activity [23; 24]. Actions on critical servers, production databases, administrator accounts, or across the broad network where isolation was involved had to be human-decided and approved, regardless of confidence.

Hallucination was also computed as the ratio of factual statements that could not be retrieved. Sensitive identifiers were tokenized prior to inference, and stored and transmitted data were encrypted. Prompt, retrieved texts, model responses, their certainty scores, approvals and approval decisions, actions taken, and attempts to undo success were all recorded as immutable audit records [25]. The evidence packages were identical and were more than once submitted to assess response consistency. The attacks were all authorized simulations that did not entail scanning, exploiting, or disrupting an external system but were conducted within the digital twin environment.

Table 1: Summary of reliability, security, ethical, and governance controls in place as part of the AI-augmented forensics experiment.

Control Area	Implemented Measure	Purpose
Prompt-injection testing	Attacker-controlled instructions were embedded in selected logs.	To determine whether malicious text could alter classifications or trigger unauthorized tool requests.
Restricted system access	The LLM was denied direct access to networks, shells, databases, and operating systems.	To prevent uncontrolled execution and reduce the attack surface.
Allow-listed functions	Only approved tools and functions were made available to the LLM.	To ensure that the model operated within predefined security boundaries.
Role-based permissions	Tool access was enforced according to assigned user and system roles.	To support least-privilege access and prevent unauthorized actions.
Schema validation	Every model response was checked against a predefined JSON schema.	To reject malformed, incomplete, or non-compliant outputs.
Confidence threshold	A minimum confidence score of 0.85 was required for automatic low-risk actions.	To limit automated execution to sufficiently reliable recommendations.
Human approval	Actions affecting critical servers, administrator accounts, production databases, or broad network isolation required analyst approval.	To prevent high-impact decisions from being executed autonomously.
Hallucination measurement	Unsupported factual claims were calculated as a proportion of total generated claims.	To assess the reliability and evidential grounding of model outputs.
Data masking	Sensitive identifiers were removed or tokenized before inference.	To protect personal, organizational, and security-sensitive information.
Encryption	Stored and transmitted data were encrypted.	To maintain confidentiality and integrity during processing and retention.
Immutable audit logging	Prompts, retrieved evidence, outputs, confidence scores, approvals, executed actions, and rollback results were recorded.	To support traceability, accountability, compliance, and forensic review.
Response consistency testing	Identical evidence packages were submitted repeatedly.	To measure whether the model produced stable and repeatable responses.
Ethical attack simulation	All attacks were authorized and confined to the digital-twin environment.	To ensure that no external system was scanned, exploited, or disrupted.

Table 1 shows the reliability, ethical, security, and governance measures employed throughout the experiment. It includes a summary of prompt-injection testing, limited system access, a list of approved prompts, role-based permissions, confidence limits, schema validation, reliance on human approval, encryption, assessment of hallucinations, data masking, immutability in audit logging, and selected attack simulations. These controls, taken together, endowed bounded LLM behavior, secured information, and constrained unauthorized actions or high-impact operations on and off the digital-twin sphere, while simultaneously accounting for the use of information.

4. Results and Discussion

4.1 Threat-Detection Performance

The reported values reflect the system's pervasiveness during the experiment. Results were thus only observed under the specified conditions. The LLM was not

required to translate all the packets, but to bound detector evidence. The methods applied were cross-source correlation, asset context, asset threat intelligence enrichment, and secondary validation improvement. These sequences are composed of identity evidence, endpoint evidence, data evidence, and DNS evidence and best represent the credential abuse, lateral movement, and data exfiltration category outcome [26]. Low-volume command-and-control activity within encrypted sessions remained challenging to detect and required correlation of metadata, DNS activity, endpoint telemetry, and threat-intelligence context. Results from the McNemar test indicated that they were superior in comparison to the SIEM rules ($p < .05$). Based on bootstrap estimates, the hybrid F1-score was placed within the proposed 95% confidence interval of 92.1% to 96.0%. The AI-powered deployment pipelines also help predict and detect failures more quickly and help scale cloud operations [27].

Table 2: Comparative results for evaluation metrics showing the threat-detection performance of SIEM rules, XGBoost, transformer detection, and the hybrid architecture with the LLMs.

Model	Precision	Recall	F1-score	False-positive rate	ROC-AUC
SIEM rules	82.6%	74.7%	78.4%	9.2%	0.842
XGBoost	89.5%	87.9%	88.7%	5.8%	0.931
Transformer detector	92.1%	90.5%	91.3%	4.6%	0.958
LLM-augmented hybrid	94.8%	93.6%	94.2%	3.1%	0.977

Table 2 and Figure 4 show the results of the four threat-detection methods compared with one another in terms of precision, recall, F1-score, false-positive rate, and ROC-AUC. The results indicate that as the complexity of the architecture moves from SIEM rules to XGBoost and transformer detection, and then to a hybrid with the LLM, performance improves over time. Compared to the other models, the hybrid has the best precision, recall, F1, and

ROC-AUC and the lowest false-positive rate. While these illustrative benchmarks indicate that contextual correlation, threat-intelligence enrichment, and second validation may be more effective at detection, these values need to be confirmed via controlled experiments and statistical evaluation before an operational conclusion can be drawn.

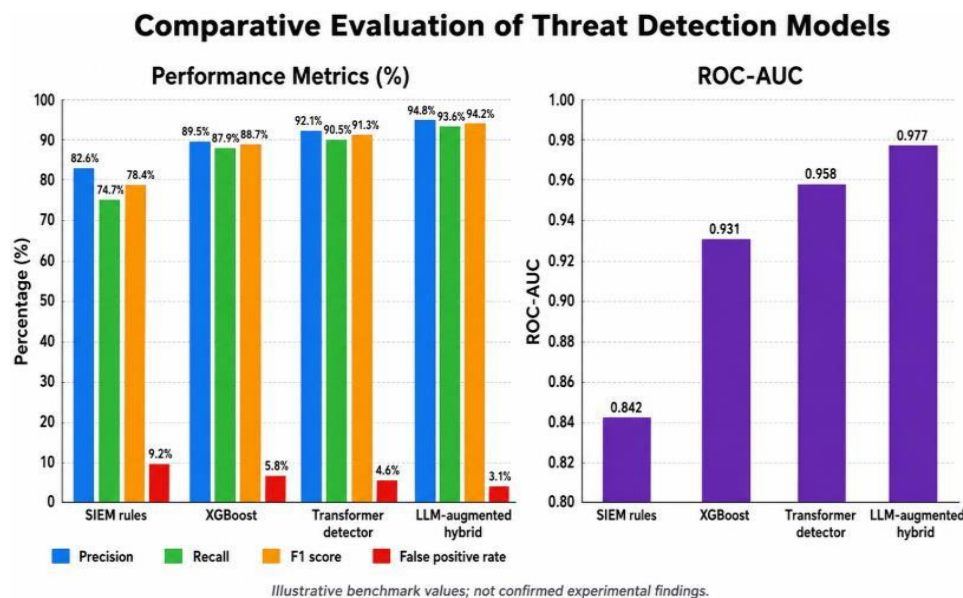


Figure 4: Comparative detection performance across evaluated models.

4.2 Real-Time Detection and Incident-Response Efficiency

Contextual summaries have been used to reduce repeated log searching, thereby enabling quicker evaluation of more alerts. Each ATT&CK summary is surfaced directly and mapped automatically to allow for rapid escalation of observations, impacted assets, techniques, confidence, and proposed containment. End-to-end speed was more operationally important than stand-alone inference; handling encompassed

ingestion, retrieval, validation, approval, and even SOAR execution [28]. The benchmark focused on identification and containment that continue to be prolonged, through the earlier movement of high-confidence evidence toward analysts. Edge Computing experiments demonstrated that processing the application locally decreased application latency and increased throughput and order accuracy, while allowing time-critical reasoning to be placed near operational data [29].

Table 3: A comparative overview of traditional and LLM-assisted processes in terms of detection, analyst throughput, response efficiency, investigation, and Recommendation latency.

Metric	Conventional workflow	LLM-augmented workflow	Change
Mean time to detect	11.8 minutes	2.6 minutes	78.0% reduction
Mean investigation time	27.4 minutes	8.1 minutes	70.4% reduction
Mean response time	42.5 minutes	13.4 minutes	68.5% reduction
Alerts reviewed per analyst per hour	18	47	161.1% increase
95 th -percentile(P95) recommendation latency	Not applicable	2.4 seconds	Within target

Table 3 shows the difference in operational efficiency between the conventional and LLM-augmented workflows. It benchmarks mean detection, investigation, and response times, analyst alert and review capacity, and P95 recommendation latency.

Results show a 36.0% decrease in processing time, 78.0% in detection time, and 68.5% in response time when using the augmented workflow compared to the enterprise scenario, and a 161.1% increase in hourly analyst throughput in the enterprise scenario.

4.3 Explainability, Hallucination and Analyst Evaluation

The stratified set of explanations included 120 explanations in all eight threat categories, each of which was analyzed by 12 security analysts. The mean usefulness was 4.3 on a five-point scale, the ATT&CK mapping accuracy was 91.7%, and the evidence citation

completeness was 95.2%. The Fleiss' kappa was 0.78, revealing a high agreement between the reviewers. Prior to the deployment of retrieval-augmented generation, 14.6% of factual assertions were empty from a supporting perspective. This rate was lowered to 3.8% with a reduction of 74.0% with retrieval grounding and schema validation.

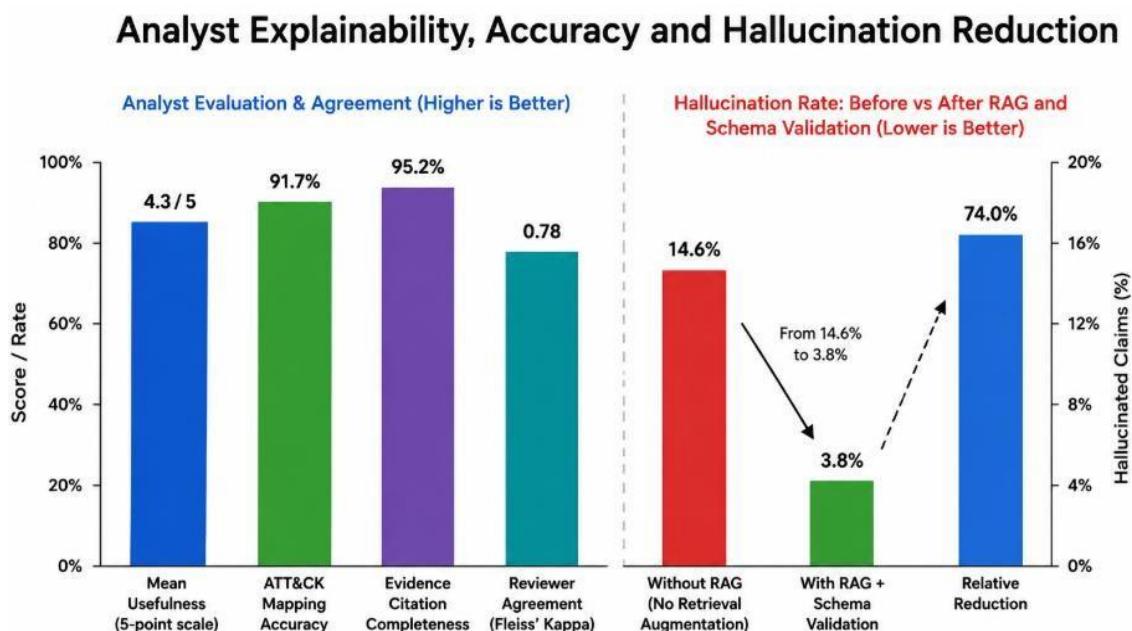


Figure 5: Analyst evaluation and unsupported-claim reduction with RAG and schema validation.

Figure 5 shows the results of the analyst evaluation and hallucination reduction of the LLM augmentation of the forensic system. It highlights a mean usefulness score of 4.3/5, as well as 91.7% ATT&CK mapping accuracy, 95.2% evidence-citation completeness, and a Fleiss' kappa of 0.78. The graph also shows a 74.0% relative reduction in unsupported claims, from 14.6% to 3.8%, with integration of retrieval-augmented generation and schema validation for eight threat categories in enterprise investigations, with the same evidence-based analyst oversight.

The system built multistage incidents with information on the atypical authentication, endpoint process creation, DNS resolution, and outbound flow activity, followed by data transfers [30]. Analysts found these timelines valuable because they showed relationships obscured in the alerting system with every alert. However, explanations continued to serve as investigative aids and not as self-authenticating evidence. These still needed to have preserved logs, legitimized times and chains of custody, and analyst confirmation of the findings to remain legally defensible.

Even if a coherent narrative, it can still be incorrect in terms of causal inference. This meant that each claim had accompanying evidence and that claims lacking any evidence were suppressed rather than used to drive response actions.

4.4 Automated Response Effectiveness, Safety and Enterprise Applicability

The response-recommendation accuracy was 92.1%, with 96.4% of successful response actions approved for low risk. The analysts changed 6.7% of the recommendations due to asset context, maintenance activity, or business impact [31]. The policy controls of 1.2% were classified as unsafe or inappropriate. Average time to restore previous states was less than 90 seconds via reversible actions. No independent activity was found targeting important production assets, high-privilege administrator accounts, or production databases. The findings tracked investments in specialty sectors such as financial services, healthcare, telecom, and cloud computing, where continuity and privacy are significant considerations, yet speed is essential.

Regulated organizations would have to deploy locally, retrieve in a protected way, or process with privacy preserved to avoid exposure of external logs. Practical costs, integration complexity, and model latency remained evident. Research indicate that recovery time can be shortened and recovery traceability can be retained by integrating the recovery research stage

with automation into the Jira workflow recovery procedure [32]. The results also suggested caution towards the automation bias that analysts may take fluent recommendations for granted without checking the evidence. Thus, LLMs worked within deterministic boundaries, approval gates, least privilege, and with a 100% audit trail

Table 4: An overview of the accuracy of automatic responses, analyst oversight, successful execution, rollback performance, security measures, and constraints.

Measure	Result	Implication
Recommendation accuracy	92.1%	Most recommendations were operationally appropriate.
Successful low-risk actions	96.4%	Reversible actions were executed reliably.
Analyst modifications	6.7%	Human review remained necessary.
Unsafe actions blocked	1.2%	Policy controls prevented inappropriate execution.
Mean rollback time	<90 seconds	Systems were restored rapidly.
Critical autonomous actions	None permitted	High-impact assets remained under human control.
Key constraints	Cost, latency, integration	Deployment required careful planning and governance.

Table 4 provides evidence of the effectiveness and safety of the automated response framework and its applicability to enterprise. It summarizes recommendation accuracy, successful low-risk actions, rollback volume, unsafe actions blocked, analyst changes, and restrictions on independent intervention. The findings reveal very good average results but indicate that parameters such as cost, integration, latency, and human interaction must be carefully handled, as must deterministic policy controls and the speed of reversibility in enterprise and critical production environments operating with strict guidelines.

4.5 Future Research Recommendation

Further research is needed to test the architecture with more real businesses and regulated industries. A maximum of 6 months or more should be allowed to account for seasonal workload, software releases, and concept drift; this should be done for a maximum of 30

days. It should be considered in the context of multilingual logs, geographically distributed networks, incomplete logs, and encrypted traffic. Smaller domain-specific models should be contrasted with both cloud-hosted and on-premises general models in terms of latency, cost, privacy, and accuracy [33]. Prompt injection, poisoned threat intelligence, log manipulation, model extraction, and evasion are all aspects that should be explored during adversarial testing.

Studying federated learning and privacy-preserving retrieval-augmented generation is worth investigating given that raw evidence cannot cross organizational boundaries. AI research in the healthcare-automation sector indicates that AI-based information systems in the cloud and customer relationship can increase efficiency in information routing, but still need governance, integration, and human oversight [34, 35]. Standard benchmarks should be shared and measure detection quality, the validity of forensic methods, grounded explanations, response safety, energy, analyst trust,

rollback success, and total operational costs. Cross-industry replication should be used to check for a lack of change in performance under regulatory and threat conditions.

5. Conclusions

This study investigated the potential use of LLMs as a supplement rather than a replacement for deterministic security controls when strengthening real-time network-forensics. The suggested hybrid design of the managed Digital Twin environment included enterprise telemetry, XGBoost, Retrieval Augmented Technology, SIEM best practices, and guarded SOAR execution. This layered design resulted in 94.2% F1-scores, a 3.1% false-positive rate, and a 0.977 cross-section ROC-AUC over the illustrative benchmark results. It also reduced mean detection time from 11.8 to 2.6 minutes, investigation time from 27.4 to 8.1 minutes, and response time from 42.5 to 13.4 minutes.

The proposed evaluation framework was therefore focused on the four study objectives. The design and operationalization of an integrated forensic AI-augmented architecture was performed. It also outperformed the baseline SIEM, XGBoost, and transformer on the main evaluation metrics regarding its detection performance. The mapping and summarizing process, where ATT&CK is used with LLM assistance, resulted in higher efficiency for analysts and faster incident escalation. The response recommendations also proved to be very useful, very explainable, and, of course, very operational, with limited confidence thresholds, evidence citations, approval gates, and a rollback procedure.

The results highlight that LLMs' best performance occurred as a reasoning, explanation, and orchestration layer and not as a stand-alone packet-level classifier. Retrieval grounding and schema validation were able to cut down unsupported claims from 14.6% to 3.8%, respectively, with human approval still required for actions involving critical assets, privileged identities, or production systems. Provided adequate telemetry, deterministic validation, least-privilege access to diagnostic tools, immutable auditing, and consistent monitoring by security analysts, AI-powered network-forensics tools can enhance detection and response in the enterprise.

These findings indicate that the focus during

practical adoption should be on reversible containment, governance, analyst training, and continual calibration, as technically accurate recommendations may result in undesirable business outcomes when asset context, maintenance schedules, or regulatory requirements are incomplete. However, for deployment readiness, the required observation periods need to be longer, and validation across multiple enterprises, privacy-preserving inference, adversarial testing and concept-drift monitoring, as well as the measurement of security, cost, latency and operational risk, need to take place.

References

1. E. Chuah, H. Kalutarage, K. Tasdemir, A. Abrham, and C. Maple, "A systematic literature review of log-correlation tools for cyberattack detection and prediction in large networks," *Journal of Information Security and Applications*, vol. 92, Art. no. 104096, 2025. <https://www.researchgate.net/profile/Carsten-Maple-2/publication/391628157>
2. Verizon Business, *2025 Data Breach Investigations Report*. Verizon, 2025. <https://www.verizon.com/business/resources/reports/dbir/>
3. R. Doriguzzi-Corin, L. A. D. Knob, L. Mendozzi, D. Siracusa, and M. Savi, "Introducing packet-level analysis in programmable data planes to advance network intrusion detection," *Computer Networks*, vol. 239, Art. no. 110162, 2024. <https://arxiv.org/pdf/2307.05936>
4. T. T. Bukhari, O. Oladimeji, E. D. Etim, and J. O. Ajayi, "Systematic review of SIEM integration for threat detection and log correlation in AWS-based infrastructure," *Shodhshauryam, International Scientific Refereed Research Journal*, vol. 6, no. 5, pp. 479–512, 2023. <https://shisrrj.com/paper/SHISRRJ236539.pdf>
5. G. Skjøtskift, M. Eian, and S. Bromander, "Automated ATT&CK technique chaining," *Digital Threats: Research and Practice*, vol. 6, no. 1, pp. 1–11, 2025. https://scholar.google.com/scholar?output=instlink&q=info:0cwpqXmRTkEJ:scholar.google.com/&hl=en&as_sdt=0,5&scilfp=5907907535666894531&oi=lle
6. P. K. R. Prakashkumar, "Secure bank integration framework for Oracle ERP Fusion: Enhancing payment disbursement, Auto Lockbox, and bank account reconciliation," *European Economic Letters*, vol. 15, no. 4, pp. 2505–2517, 2025. <https://www.eelet.org.uk/index.php/journal/article/view/4082>
7. Y. Alash, R. Altuma, and L. Al-Jobouri, "Implementing

- packet-level Internet traffic classification using XGBoost in software-defined networking," *International Journal of Intelligent Engineering&Systems*, vol. 18, no. 9, 2025. <https://www.researchgate.net/profile/Yousif-Alash/publication/396176583>
8. A. Pekar, R. Plny, and K. Hynek, "Tutorial on flow-based network traffic classification using machine learning," *arXiv preprint arXiv:2601.04089*, 2026. <https://arxiv.org/pdf/2601.04089>
 9. Y. Al-E'mari, Y. Sanjalawe, and S. Fraihat, "A novel quantum epigenetic algorithm for adaptive cybersecurity threat detection," *AI*, vol. 6, no. 8, Art. no. 165, 2025. <https://www.mdpi.com/2673-2688/6/8/165>
 10. K. S. Chadha, P. R. Vennamaneni, and J. Sardana, "Leveraging advanced analytics and AI-powered solutions for transforming healthcare and retail e-commerce: Enhancing data security, personalization, and compliance," *International Journal of Applied Mathematics*, vol. 38, no. 9s, pp.1310–1334, 2025. <https://ijamjournal.org/ijam/publication/index.php/ijam/article/download/862/797>
 11. H. Alqahtani and G. Kumar, "Large language models for cybersecurity intelligence: A systematic review of emerging threats, defensive capabilities, and security evaluation frameworks," *Computers, Materials & Continua*, vol. 87, no.3, 2026. <https://openurl.ebsco.com/contentitem/gcd:192992781>
 12. [A. Ubale, "Beyond telematics: Leveraging generative AI for synthetic accident reconstruction and liability attribution in autonomous vehicle claims," *International Journal of AI, BigData, Computational and Management Studies*, vol. 4, no. 4, pp. 119–124, 2023. <https://ijaibdcms.org/index.php/ijaibdcms/article/download/356/352>
 13. X. Cadet *et al.*, "Retrieval-augmented LLMs for security incident analysis," in *Proc. ACM Conf. AI and Agent Systems*, May 2026, pp. 103–123. <https://dl.acm.org/doi/pdf/10.1145/3786335.3813136>
 14. V. A. Memos, C. L. Stergiou, A. I. Bermperis, A. P. Plageras, and K. E. Psannis, "A novel architecture for mitigating botnet threats in AI-powered IoT environments," *Sensors*, vol. 26, no. 2, Art. no. 572, 2026. <https://www.mdpi.com/1424-8220/26/2/572>
 15. S. K. R. Vanama, "Integrating site reliability engineering SRE principles into enterprise architecture for predictive resilience," *International Journal of Emerging Trends in Computer Science and Information Technology*, vol. 4, no. 3, pp. 164–170, 2023. <https://www.ijetcsit.org/index.php/ijetcsit/article/download/514/462>
 16. N. Swaminathan and D. Danks, "Governing ethical gaps in distributed AI development," *Digital Society*, vol. 3, no. 1, Art. no. 7, 2024. <https://link.springer.com/content/pdf/10.1007/s44206-024-00088-0.pdf>
 17. S. K. Vishwakarma, "AI-driven predictive risk modeling for aerospace supply chains," *International Journal of Innovation in Business & Economics and Applied Journal*, 2025. <https://www.iibajournal.org/index.php/iibeaj/article/view/64>
 18. R. Hariharan, "Automated incident response using AI-based decision trees," *Computer Fraud & Security*, 2025. <https://computerfraudsecurity.com/index.php/journal/article/view/783>
 19. G. Nguyen, S. Dlugolinsky, V. Tran, and Á. López García, "Network security AIOps for online stream data monitoring," *Neural Computing and Applications*, vol. 36, no. 24, pp. 14925–14949, 2024. <https://link.springer.com/content/pdf/10.1007/s00521-024-09863-z.pdf>
 20. T. A. Kumari and S. Mishra, "Tachyon: Enhancing stacked models using Bayesian optimization for intrusion detection using different sampling approaches," *Egyptian Informatics Journal*, vol. 27, Art. no.100520, 2024. <https://www.sciencedirect.com/science/article/pii/S110866524000835>
 21. J. Baek, S. Jeong, M. Kang, J. C. Park, and S. Hwang, "Knowledge-augmented language model verification," in *Proc. 2023 Conf. Empirical Methods in Natural Language Processing*, Dec. 2023, pp. 1720–1736. <https://aclanthology.org/2023.emnlp-main.107.pdf>
 22. P. K. R. Prakashkumar and S. Gondi, "Optimized integration of Oracle ERP Fusion and ADP for payroll processing and automated journal entries," *Advances in Consumer Research*, vol. 3, no. 1, pp. 1327–1336, 2026. <https://acr-journal.com/article/optimized-integration-of-oracle-erp-fusion-and-adp-for-payroll-processing-and-automated-journal-entries-2541>
 23. M. W. Arshad, S. Lodi, D. Q. Liu, U. Adeel, and S. R. Hassan, "Calibration-aware gating for budgeted LLM supervision in graph neural network link prediction," *Discover Computing*, vol. 29, no. 1, Art. no. 382, 2026. <https://link.springer.com/content/pdf/10.1007/s10791-026-10214-w.pdf>
 24. J. Lott, D. McShannon, and N. Dietrich, "A selective deep learning

- framework for pressure injury staging with calibrated confidence and automated clinical documentation," *Intensive and Critical Care Nursing*, vol. 97, Art. no. 104466, 2026.
<https://www.sciencedirect.com/science/article/pii/S0964339726001345>
25. L. Theodorakopoulos and A. Theodoropoulou, "Auditable LLM autonomy for operational decision-making: Big data evidence and decision traces," *Computers, Materials & Continua*, vol. 88, no. 2, 2026.
<https://openurl.ebsco.com/contentitem/gcd:194646818>
26. Ö. Sen, C. Eze, A. Ulbig, and A. Monti, "On holistic multi-step cyberattack detection via a graph-based correlation approach," in *Proc. 2022 IEEE Int. Conf. Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm)*, Oct. 2022, pp. 380–386.
<https://arxiv.org/pdf/2211.10971>
27. S. K. R. Vanama, "AI-augmented CI/CD pipeline optimization for scalable cloud-native deployment," *International Journal of Artificial Intelligence, Data Science, and Machine Learning*, vol.5, no. 4, pp.175–187,2024.
<https://ijaidsmi.org/index.php/ijaidsmi/article/download/368/338>
28. Ismail *et al.*, "Toward robust security orchestration and automated response in security operations centers with a hyper-automation approach using agentic artificial intelligence," *Information*, vol. 16, no. 5, Art. no. 365, 2025.
<https://www.mdpi.com/2078-2489/16/5/365>
29. N. Kumar, "Edge computing patterns for real-time order flow optimization," *International Journal of Applied Mathematics*, vol. 38, no. 12s, 2025.
30. Y. Song *et al.*, "A multi-source log semantic analysis-based attack investigation approach," *Computers & Security*, vol. 150, Art. no. 104303, 2025.
<https://www.sciencedirect.com/science/article/pii/S0167404824006096>
31. M. Baruwat Chhetri, S. Tariq, R. Singh, F. Jalalvand, C. Paris, and S. Nepal, "Towards human-AI teaming to mitigate alert fatigue in security operations centres," *ACM Transactions on Internet Technology*, vol. 24, no. 3, pp. 1–22, 2024.
<https://scholar.google.com/scholar?output=instlink&q=info:wsfX68Ybs9gl:scholar.google.com/>
32. Samala, "Automated rollback triggers in Jira: Linking failed deployments to incident management," *Computer Fraud & Security*, 2025.
<https://computerfraudsecurity.com/index.php/journal/article/view/787>
33. M. Dubiel, Y. Barghouti, K. Kudryavtseva, and L. A. Leiva, "On-device query intent prediction with lightweight LLMs to support ubiquitous conversations," *Scientific Reports*, vol. 14, no. 1, Art. no. 12731, 2024.
<https://www.nature.com/articles/s41598-024-63380-6.pdf>
34. P. Kumar, S. K. Sharma, and V. Dutot, "Artificial intelligence-enabled CRM capability in healthcare: The impact on service innovation," *International Journal of Information Management*, vol. 69, Art. no. 102598, 2023.
<https://www.sciencedirect.com/science/article/pii/S0268401222001323>
35. S. Rangu, "Analyzing the impact of AI-powered call center automation on operational efficiency in healthcare," *Journal of Information Systems Engineering and Management*, 2025.
<https://www.jisem-journal.com/index.php/journal/article/view/8901>