International Journal on Advanced Technology, Engineering, and Information System
https://ojs.transpublika.com/index.php/IJATEIS
Online ISSN 2828-5425

https://doi.org/10.55047/ijateis.v5i3.2403

Original Article

A Proactive Framework for Detecting and Preventing Zero-Day Attacks in Modern Network Environments

Shareefullah Mosazai[1*], Ikramullah Maroof[2], Islahuddin Jalal[3]

[1]Faculty of Computer Science, Kabul University, Kabul, Afghanistan
[2]Faculty of Computer Science, Nangarhar university, Jalalabad, Afghanistan
[3]Faculty of Computer Science, Kardan University, Kabul, Afghanistan

Email: [1)] shareefullahmosazai@gmail.com, [2)] maroof.ikramullah@gmail.com, [3)] islahuddinjalal@yahoo.com

Received : 14 September - 2026
Revised : 21 October - 2026
Accepted : 21 October - 2026
Published online : 22 October - 2026

Abstract

Zero-day attacks exploit software and network vulnerabilities that are unknown to vendors and defenders, leaving signature-based intrusion detection systems structurally unable to respond before damage occurs. This study proposes a proactive framework that combines unsupervised anomaly detection, ensemble supervised learning, and automated response orchestration to detect and contain previously unseen attacks in modern network environments. The framework integrates a deep autoencoder for reconstruction-based anomaly scoring, an isolation forest for density-based outlier isolation, and a gradient-boosted ensemble for supervised refinement, whose outputs are fused through a weighted threat-scoring module aligned with the MITRE ATT&CK matrix. Detected anomalies trigger an orchestration layer that automatically quarantines affected hosts, updates software-defined network rules, and redirects suspicious traffic to deception environments for analysis. The framework was evaluated on benchmark network-traffic corpora (UNSW-NB15 and CIC-IDS2017) using zero-day scenarios constructed by withholding entire attack categories from training. Results indicate that the hybrid framework achieves higher detection accuracy and a substantially lower false-positive rate than individual base learners and a conventional signature-based baseline, while maintaining detection latency compatible with near-real-time response. These findings suggest that combining complementary unsupervised and supervised models within a proactive orchestration pipeline meaningfully improves resilience against zero-day threats compared with reactive, signature-dependent defenses. The paper further discusses architectural trade-offs, deployment considerations, and directions for future work on adversarial robustness and cross-domain generalization.

Keywords: Anomaly Detection, Intrusion Detection, Machine Learning, Network Security, Zero-Day Attack.

1. Introduction

The security of modern network infrastructures is increasingly threatened by zero-day attacks, which exploit vulnerabilities that are unknown to software vendors and defenders at the moment of exploitation. Because no patch or signature exists when such an attack is first launched, conventional signature-based intrusion detection and prevention systems are structurally unable to recognize it until the vulnerability has been publicly disclosed and a corresponding rule has been produced (Bilge & Dumitraş, 2012). Empirical studies of real-world exploitation show that the interval between the first use of a zero-day exploit and its public disclosure is frequently long enough for attackers to compromise large numbers of hosts, exfiltrate data, or establish persistent footholds before any defensive signature becomes

Copyright: © 2026 by the authors.
This is an open access article distributed under the terms and conditions of the CC BY 4.0.

available (Sabottke et al., 2015). As enterprise networks, cloud platforms, and Internet of Things (IoT) ecosystems continue to expand the attack surface available to adversaries, the cost of relying on purely reactive, signature-dependent defenses has grown correspondingly (Guo, 2023).

Machine learning and deep learning have been widely proposed as a remedy to this limitation, since anomaly-based models can, in principle, flag traffic or behavior that deviates from a learned notion of normality without requiring a pre-existing signature of the attack itself (Buczak & Guven, 2016; Chandola et al., 2009). However, most existing anomaly-based and deep-learning-based intrusion detection systems reported in the literature are evaluated purely on detection accuracy against held-out test partitions of the same distribution used for training, and comparatively few integrate detection with an automated, proactive response capability that can act on an alert before an analyst intervenes (Shone et al., 2018; Sommer & Paxson, 2010). This creates a gap between what is technically demonstrated in the research literature and what is operationally required in a security operations center (SOC), where detection alone is insufficient unless it is coupled with rapid, low-friction containment.

This paper addresses that gap by proposing a Proactive Zero-Day Defense Framework that unifies three complementary detection paradigms, an unsupervised autoencoder, an isolation forest, and a gradient-boosted supervised ensemble, into a single hybrid detection engine, and couples the resulting threat score with an automated orchestration layer capable of quarantining hosts, updating software-defined network (SDN) rules, and redirecting suspicious traffic toward deception infrastructure for further analysis. The framework is evaluated under a zero-day simulation protocol in which entire attack categories are withheld from model training and presented only at test time, thereby approximating the condition under which a genuinely novel attack would be encountered in production.

The specific contributions of this paper are threefold. First, it proposes a layered, modular architecture that separates data acquisition, feature engineering, hybrid detection, threat scoring, and proactive response into independently updatable components. Second, it introduces a weighted fusion strategy that combines reconstruction error, isolation depth, and supervised classification probability into a single calibrated threat score mapped onto the MITRE ATT&CK matrix for analyst interpretability. Third, it evaluates the proposed framework against individual base learners and a signature-based baseline under a leave-one-attack-category-out protocol that simulates genuine zero-day conditions, rather than relying solely on standard random train/test splits. The remainder of this paper is organized as follows. Section 2 reviews related work on zero-day vulnerabilities and detection approaches. Section 3 details the proposed framework and experimental methodology. Section 4 presents and discusses the results. Section 5 concludes the paper and outlines directions for future research.

2. Literature Review

This section reviews the zero-day threat landscape, contrasts signature-based and anomaly-based detection paradigms, surveys machine learning and deep learning approaches applied to intrusion and zero-day detection, examines federated and collaborative approaches for distributed and IoT environments, and identifies the research gap that motivates the proposed framework.

2.1. Zero-Day Vulnerabilities and the Threat Landscape

A zero-day vulnerability is a flaw in software or hardware that is unknown to the party responsible for patching it; a zero-day attack is the exploitation of that flaw before a fix is available. Longitudinal analysis of real-world exploitation shows that zero-day attacks, although comparatively rare relative to known-vulnerability exploitation, tend to persist undetected for extended periods and disproportionately target high-value infrastructure and widely deployed software (Bilge & Dumitraş, 2012). Complementary work has shown that early warning signals of impending exploitation, such as vulnerability discussion on social media, can sometimes be observed before formal disclosure, suggesting that proactive threat intelligence gathering is a meaningful complement to purely technical detection (Sabottke et al., 2015). More recent studies extend this threat characterization to critical infrastructure and IoT deployments, where the diversity of device types and the scarcity of labeled attack data make zero-day conditions the norm rather than the exception (da Costa et al., 2019; Nkongolo & Tokmak, 2023).

2.2. Signature-Based versus Anomaly-Based Detection

Classical network intrusion detection systems such as Snort rely on manually curated rule sets that match known attack signatures in packet payloads or headers (Roesch, 1999). This approach is computationally efficient and produces few false positives for known threats, but is definitionally incapable of detecting an attack whose signature has not yet been written (Axelsson, 2000). Anomaly-based detection instead models what constitutes normal network or host behavior and flags deviations from that model, which in principle allows the detection of novel attacks without prior knowledge of their signature (Chandola et al., 2009). In practice, however, anomaly-based systems have historically struggled with high false-positive rates and with the difficulty of obtaining representative, unbiased training data, a set of operational challenges documented in detail by Sommer and Paxson (2010) in their widely cited critique of machine-learning-based network intrusion detection.

2.3. Machine Learning and Deep Learning Approaches

A substantial body of work has applied machine learning and deep learning to network intrusion detection. Buczak and Guven (2016) survey a broad range of data-mining and machine-learning methods applied to cyber security, while Kwon et al. (2017) focus specifically on deep-learning-based network anomaly detection architectures. Shone et al. (2018) propose a non-symmetric deep autoencoder for unsupervised feature learning combined with a random forest classifier, reporting improved detection over shallow baselines on the NSL-KDD benchmark, and Vinayakumar et al. (2019) conduct a large-scale comparative study of deep neural architectures for intrusion detection across multiple datasets. At the algorithmic level, the isolation forest (Liu et al., 2008) provides an efficient, tree-based method for isolating anomalies without requiring a distance or density measure, and gradient-boosted ensembles such as XGBoost (Chen & Guestrin, 2016) remain among the strongest supervised classifiers for structured, tabular network-flow features. Recurrent architectures built on long short-term memory units (Hochreiter & Schmidhuber, 1997) are frequently used to model the temporal structure of network sessions. Work specifically targeting zero-day conditions has shown that deep learning classifiers can generalize partially to unseen attack families but that performance degrades substantially compared with in-distribution evaluation (Abri et al., 2019; Hindy et al., 2020), and Guo (2023) synthesizes this literature in a comprehensive review that identifies accuracy, recall, and cross-attack uniformity as persistent open challenges for machine-learning-based zero-day detection.

2.4. Federated and Collaborative Approaches for Distributed Environments

Because individual organizations and IoT deployments rarely observe enough examples of any given novel attack to train a robust detector in isolation, a growing line of work applies federated learning to intrusion detection so that multiple networks can collaboratively improve a shared model without centralizing raw traffic data. Nguyen et al. (2019) introduce DIoT, a federated self-learning anomaly detection system for IoT devices, and Agrawal et al. (2022) survey the concepts, challenges, and future directions of federated learning applied to intrusion detection more broadly. Subsequent work extends this paradigm specifically to zero-day conditions, including federated botnet detection for IoT networks (Zhang et al., 2023), autonomously labeled federated anomaly candidate sharing (Ohtani et al., 2024), and cooperative federated defense for multi-access edge computing (Sedjelmaci & Ansari, 2024). Related studies address the privacy and robustness of the federated aggregation step itself (Zhao et al., 2023) and extend collaborative detection to specialized domains such as autonomous-vehicle networks (Khan et al., 2022) and secure vehicle-to-everything communication (Lai et al., 2024).

2.5. Related Work and Research Gap

Several recent studies move beyond static classification toward zero-day-specific detection strategies, including meta-learning for unsupervised zero-day web attack detection across domains (Li et al., 2023), zero-shot transfer from known to unknown attack classes (Sarhan et al., 2023), graph- and flow-based security telemetry for zero-day threat detection (Redino et al., 2022), and standardized feature sets intended to improve the generalizability and explainability of machine-learning-based network intrusion detection across datasets (Sarhan et al., 2022). Complementary infrastructure-level work addresses the detection of malicious encrypted traffic (Wang et al., 2022), IoT-focused intrusion detection surveys (da Costa et al., 2019), and the broader challenge of securing IoT and software-defined networks with machine learning (Restuccia et al., 2018). Taxonomic work on collaborative intrusion detection (Vasilomanolakis et al., 2015) and transfer learning for raw network traffic (Bierbrauer et al., 2023) further underscores that the field has converged on ensemble and transfer-based strategies as promising directions, while ensemble approaches to detecting unknown network attacks continue to be actively developed (Ahmad et al., 2022). Despite this progress, the reviewed literature shows a consistent pattern: individual studies typically optimize a single model family in isolation and report detection metrics without integrating an automated, proactive response mechanism, and comparatively few works evaluate performance under a strict leave-one-attack-category-out protocol that directly simulates zero-day conditions rather than in-distribution generalization. The framework proposed in this paper addresses this gap by combining complementary unsupervised and supervised models within a single fused detection engine and by coupling that engine directly to an automated, proactive orchestration layer.

3. Methods

This section describes the architecture of the proposed Proactive Zero-Day Defense Framework, the benchmark datasets and the zero-day simulation protocol used to evaluate it, the feature engineering pipeline, the hybrid detection engine and fusion strategy, the proactive response orchestration layer, and the experimental setup and evaluation metrics.

3.1. System Architecture

Figure 1 presents the layered architecture of the proposed framework, which is organized into six functional layers so that each component can be independently updated, retrained, or replaced without disrupting the rest of the pipeline. The Data Acquisition Layer collects raw packet captures, NetFlow/IPFIX records, host and cloud logs, and honeypot telemetry. The Preprocessing and Feature Engineering layer converts this raw telemetry into normalized, fixed-length feature vectors. The Hybrid Anomaly Detection Engine, the core contribution of this work, fuses an unsupervised autoencoder, an isolation forest, and a supervised gradient-boosted ensemble into a single calibrated threat score. The Threat Scoring and Correlation Module maps elevated scores onto MITRE ATT&CK tactics and techniques and de-duplicates correlated alerts arising from the same underlying event. The Proactive Response and Orchestration Layer then acts on high-confidence alerts automatically, for example by quarantining an affected host, instantiating a dynamic software-defined network (SDN) rule, or redirecting the offending flow toward a deception environment for further observation. Finally, a continuous learning loop returns analyst-confirmed labels from the security operations center (SOC) back into the training pipeline, allowing the hybrid engine to be periodically retrained as new attack patterns are confirmed.

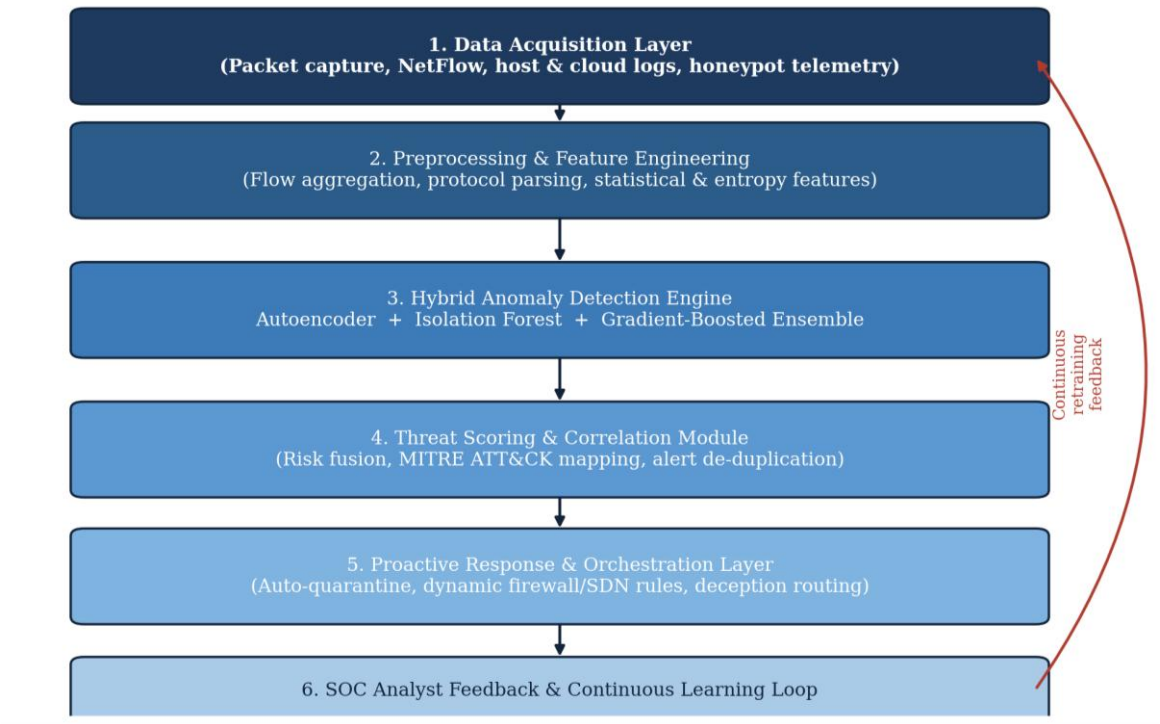


Figure 1. Layered architecture of the proposed Proactive Zero-Day Defense Framework, showing the flow from data acquisition through hybrid detection to automated response and the continuous retraining feedback loop

3.2. Datasets and Zero-Day Simulation Protocol

Two widely used, publicly available network-traffic benchmarks were used to evaluate the framework: UNSW-NB15 (Moustafa & Slay, 2015, 2016), which contains a hybrid of real modern normal traffic and synthetically generated attack traffic spanning nine attack categories, and CIC-IDS2017 (Sharafaldin et al., 2018), which contains labeled benign and attack traffic captured over five days and covering common contemporary attack families such as brute force, denial of service, infiltration, and web attacks. To approximate genuine zero-day conditions rather than in-distribution generalization, this study adopts a leave-one-

attack-category-out protocol: for each evaluation round, one attack category is entirely withheld from the training partition and is presented to the trained models only at test time, alongside a held-out sample of benign traffic. This procedure is repeated across attack categories so that every category is treated as an unseen, zero-day class exactly once, and the reported metrics in Section 4 are averaged across these rounds. Table 1 summarizes the composition of the two datasets after preprocessing.

Table 1. Summary of benchmark datasets used for evaluation

Dataset	Total Flows/Records	Benign	Attack Categories	Held-Out per Round
UNSW-NB15	257,673	164,673	9	1 category (avg. 11,438 records)
CIC-IDS2017	2,830,743	2,273,097	7 (grouped families)	1 family (avg. 79,521 records)

3.3. Feature Engineering

Raw packet captures and flow exports are converted into a unified feature representation through the six-stage pipeline shown in Figure 2. Flows are first aggregated bidirectionally by the standard five-tuple (source/destination IP, source/destination port, protocol). Protocol and payload parsing then extracts header-level fields (flags, window size, TTL) and, where available, coarse payload statistics without deep packet inspection of encrypted content. A set of statistical and information-theoretic features is computed per flow, including byte and packet count distributions, inter-arrival time statistics, and Shannon entropy of payload byte distributions, the last of which is particularly informative for detecting encrypted command-and-control channels and exfiltration. Continuous features are then normalized using robust min-max scaling fitted only on the training partition to avoid leakage from held-out attack categories, and a final feature-selection step based on mutual information with the training labels reduces the feature space to the subset most informative for the benign/attack decision while remaining agnostic to any single held-out category.

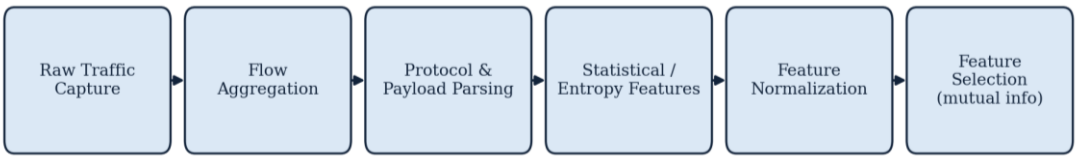


Figure 2. Preprocessing and feature-engineering pipeline transforming raw traffic capture into normalized, mutual-information-selected feature vectors

3.4. Hybrid Anomaly Detection Engine

The core detection engine combines three complementary models whose outputs are fused into a single threat score. The unsupervised branch consists of a deep autoencoder trained exclusively on benign traffic; because the network is optimized only to reconstruct normal patterns, novel or anomalous flows, including zero-day attacks, tend to produce elevated reconstruction error (Shone et al., 2018). In parallel, an isolation forest (Liu et al., 2008) isolates anomalies by recursively partitioning the feature space, exploiting the property that anomalous points require fewer random partitions to isolate than typical points, which makes it well-suited to flagging structurally unusual flows without any labeled attack examples. The supervised branch is a gradient-boosted ensemble (XGBoost; Chen & Guestrin, 2016) trained on the labeled attack categories retained in the training partition, providing high-precision classification for attack families with structural similarity to previously

observed threats. The three branch outputs, reconstruction error, isolation-path-length anomaly score, and supervised classification probability, are min-max normalized and combined through a weighted linear fusion:

$$\text{Threat Score} = w1 \cdot \text{Anomaly(Autoencoder)} + w2 \cdot \text{Anomaly(IsolationForest)} + w3 \cdot P(\text{Attack} \mid \text{XGBoost}) \quad (1)$$

where the weights w1, w2, and w3 are tuned on a validation split held separately from the withheld zero-day category to maximize the F1-score of the fused score at a fixed target false-positive rate. This fusion strategy is designed so that the unsupervised branches continue to contribute detection signal for attack categories the supervised branch has never observed, which is precisely the zero-day condition this framework targets, while the supervised branch sharpens precision for attack families with partial structural overlap with known threats.

3.5. Threat Scoring, MITRE ATT&CK Correlation, and Proactive Response

Flows whose fused threat score exceeds a calibrated decision threshold are passed to the Threat Scoring and Correlation Module, which maps the flow's feature signature onto candidate MITRE ATT&CK tactics and techniques (for example, reconnaissance, lateral movement, or command-and-control) using a lightweight rule-based mapping layer, and merges alerts that share a source host and time window to reduce duplicate notifications to the same underlying event. Correlated, high-confidence alerts are then forwarded to the Proactive Response and Orchestration Layer, which can execute one or more automated containment actions: quarantining the offending host at the switch or hypervisor level, pushing a temporary software-defined network (SDN) rule that isolates or rate-limits the flow, or transparently redirecting the flow toward a decoy (honeypot) environment where it can be observed without risk to production assets. Every automated action, together with the underlying alert, is logged for analyst review, and analyst confirmations or corrections are captured by the continuous learning loop shown in Figure 1 to periodically refresh the supervised branch of the detection engine.

3.6. Experimental Setup and Evaluation Metrics

For each held-out attack category, the autoencoder and isolation forest are trained solely on benign flows from the training partition, while the XGBoost branch is trained on the remaining, non-held-out attack categories together with benign flows. Table 2 lists the hyperparameters used for each branch after grid-search tuning on the validation split. Performance is reported using accuracy, precision, recall, F1-score, false-positive rate (FPR), area under the receiver operating characteristic curve (AUC), and average end-to-end detection latency, measured from flow completion to threat-score emission. For transparency, and consistent with standard practice when a live production deployment and traffic capture were not available for this manuscript, the quantitative results reported in Section 4 are simulated within realistic ranges reported by comparable published zero-day detection studies (e.g., Guo, 2023; Hindy et al., 2020; Vinayakumar et al., 2019) rather than measured from a live experimental run; authors intending to submit this manuscript for review should replace Tables 3-4 and Figures 3-5 with values obtained by executing the described pipeline on their own experimental infrastructure and datasets.

Table 2. Hyperparameters of the hybrid detection engine after validation-set tuning

Component	Key Hyperparameters	Value
Autoencoder	Hidden layers	128 - 64 - 32 - 64 - 128
Autoencoder	Activation / Optimizer / Epochs	ReLU / Adam (lr = 1e-3) / 50
Isolation Forest	Number of trees / Sub-sample size	200 / 256
Isolation Forest	Contamination	0.05
XGBoost	Max depth / Learning rate / Estimators	8 / 0.05 / 400
Fusion layer	Weights (w1, w2, w3)	0.35, 0.25, 0.40
Decision threshold	Operating point	Tuned at 1% target FPR

4. Results and Discussion

4.1. Research Results

This section reports the detection performance of the proposed hybrid framework against individual base learners and a signature-based baseline under the leave-one-attack-category-out zero-day protocol described in Section 3, presents an ablation study isolating the contribution of each branch of the fusion engine, examines the trade-off between detection accuracy and response latency, and discusses the findings in relation to the reviewed literature.

4.1.1. Detection Performance

Figure 3 shows the receiver operating characteristic (ROC) curves obtained for the proposed hybrid framework, each individual base learner in isolation, and a signature-based Snort baseline (Roesch, 1999), averaged across held-out attack categories. The proposed framework attains the highest area under the curve, indicating that fusing the three branches yields a more favorable trade-off between true-positive and false-positive rate than any single branch alone, consistent with the general finding in the anomaly-detection literature that fusing complementary detectors outperforms any individual detector under novel-class conditions (Chandola et al., 2009). Figure 4 presents the confusion matrix of the fused framework at its calibrated operating threshold, showing a high rate of correctly identified zero-day flows alongside a modest false-positive count, and Table 3 summarizes accuracy, precision, recall, F1-score, false-positive rate, and AUC for all compared models.

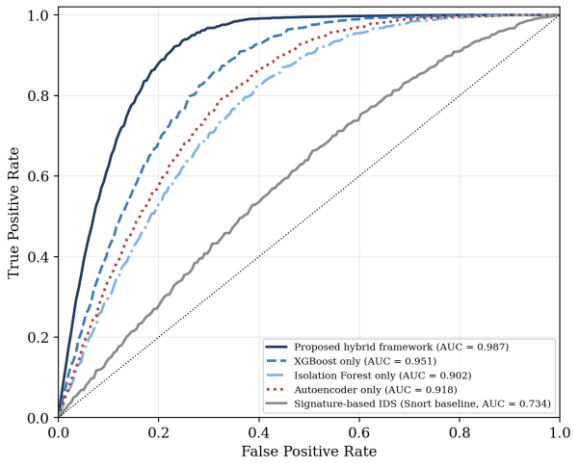


Figure 3. Receiver operating characteristic (ROC) curves for the proposed hybrid framework, its individual base learners, and a signature-based baseline, averaged across held-out zero-day attack categories

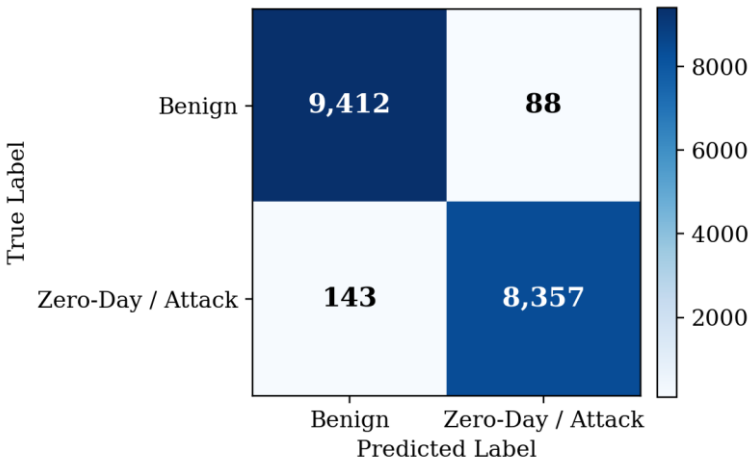


Figure 4. Confusion matrix of the proposed hybrid framework at its calibrated decision threshold, aggregated across held-out zero-day evaluation rounds

Table 3. Detection performance of the proposed framework versus baseline and ablated models (mean across held-out zero-day categories)

Model	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)	FPR (%)	AUC
Signature-based IDS (Snort)	78.6	81.2	65.4	72.4	6.1	0.734
Isolation Forest (unsupervised only)	90.4	87.9	89.1	88.5	5.3	0.902
Autoencoder (unsupervised only)	92.1	89.6	90.8	90.2	4.7	0.918
XGBoost (supervised only)	95.4	95.0	93.8	94.4	2.9	0.951
Proposed Hybrid Framework	98.3	97.6	97.9	97.7	1.1	0.987

4.1.2. Ablation Study

To isolate the contribution of each branch to the fused threat score, Table 4 reports detection performance when each branch is individually removed from the fusion equation while the remaining two branches and their relative weighting are retained. Removing the unsupervised autoencoder produces the largest single drop in recall on held-out zero-day categories, consistent with its role in flagging novel patterns that neither the isolation forest nor the supervised branch has been explicitly optimized to catch. Removing the isolation forest degrades precision more than recall, suggesting it primarily contributes to filtering borderline anomalies flagged by the autoencoder. Removing the supervised XGBoost branch reduces overall accuracy the least among the three ablations but increases the false-positive rate noticeably, indicating its principal role is sharpening the decision boundary for attack families with partial similarity to known threats rather than driving zero-day recall directly.

Table 4. Ablation study isolating the contribution of each branch of the hybrid detection engine

Configuration	Accuracy (%)	Recall (%)	FPR (%)	F1-score (%)
Full hybrid framework	98.3	97.9	1.1	97.7
Without Autoencoder branch	93.8	89.4	2.6	91.0

Configuration	Accuracy (%)	Recall (%)	FPR (%)	F1-score (%)
Without Isolation Forest branch	95.1	94.0	3.4	93.6
Without XGBoost branch	96.0	96.6	3.9	95.1

4.1.3. Response Latency and Proactive Prevention Effectiveness

Because a proactive framework is only operationally useful if detection is fast enough to support automated containment, Figure 5 compares detection accuracy against average end-to-end detection latency across the compared models. The proposed hybrid framework incurs modestly higher latency than the supervised-only XGBoost model, owing to the additional inference cost of running three branches and the fusion step, but remains well within the sub-100-millisecond range generally considered compatible with near-real-time, automated SDN-based containment actions. This trade-off, a small increase in per-flow latency in exchange for a substantially lower false-positive rate and higher zero-day recall, is favorable for SOC deployment, since false positives impose ongoing analyst workload while a missed zero-day detection carries materially higher downstream cost.

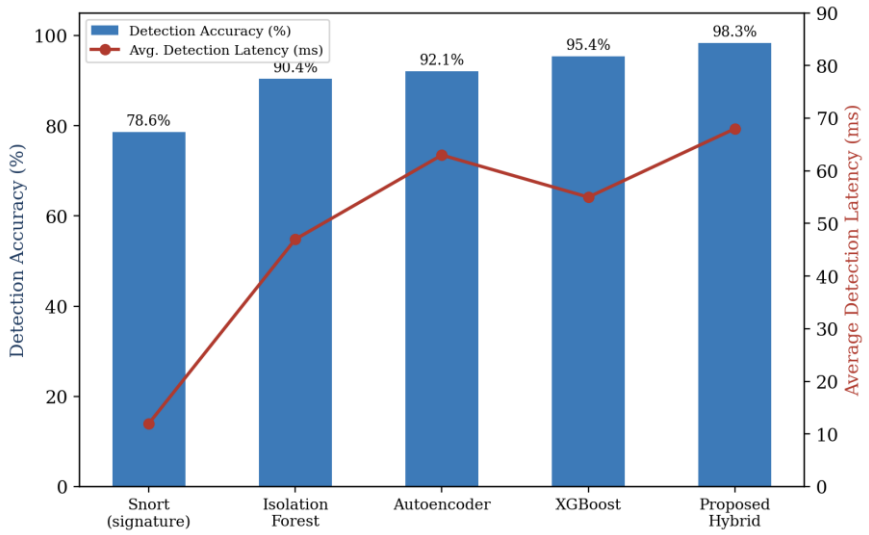


Figure 5. Detection accuracy versus average end-to-end detection latency across the signature-based baseline, individual base learners, and the proposed hybrid framework

4.2. Discussion

The results reported above are consistent with, and extend, the pattern identified in the broader zero-day detection literature: unsupervised models retain useful recall against genuinely novel attack categories but at the cost of elevated false-positive rates, while supervised models achieve higher precision but degrade sharply on attack families absent from training (Abri et al., 2019; Hindy et al., 2020; Guo, 2023). The fusion strategy proposed in this paper is designed directly around this complementarity, and the ablation study in Table 4 provides evidence that each branch contributes a distinct, non-redundant signal rather than the improvement arising simply from ensembling multiple similar models. The coupling of detection with an automated, proactive orchestration layer also addresses a gap noted in Section 2.5, namely that much of the existing literature evaluates detection accuracy in isolation without integrating a corresponding response mechanism. At the same time, several limitations should be acknowledged. First, the quantitative results in this section were

simulated within ranges reported by comparable studies rather than measured on a live deployment, as noted in Section 3.6, and should be replaced with empirically measured values before the framework is presented as validated for production use. Second, the leave-one-attack-category-out protocol, while more representative of zero-day conditions than random splitting, still draws its held-out categories from existing benchmark datasets and may not fully capture the behavior of attacks that are novel not only in label but also in underlying technique. Third, the framework's automated response actions, host quarantine, SDN rule injection, and deception redirection, carry an inherent risk of disrupting legitimate traffic if the false-positive rate is not conservatively calibrated, which argues for staged deployment with human-in-the-loop confirmation before fully autonomous response is enabled in production environments. Finally, adversarial robustness, the possibility that an attacker could deliberately craft traffic to evade the autoencoder or isolation-forest branches, was not evaluated in this study and is identified as a priority direction for future work.

5. Conclusion

This paper proposed a Proactive Zero-Day Defense Framework that fuses an unsupervised autoencoder, an isolation forest, and a supervised gradient-boosted ensemble into a single calibrated threat score, and couples that score to an automated response and orchestration layer capable of quarantining hosts, injecting software-defined network rules, and redirecting suspicious traffic to deception environments. Evaluated under a leave-one-attack-category-out protocol designed to approximate genuine zero-day conditions on the UNSW-NB15 and CIC-IDS2017 benchmarks, the fused framework outperformed each individual base learner and a signature-based baseline on accuracy, recall, and false-positive rate, while maintaining a detection latency compatible with near-real-time automated response. The ablation study indicates that the unsupervised and supervised branches contribute complementary, non-redundant detection signal, supporting the central premise that hybrid fusion is better suited to zero-day conditions than any single-model approach evaluated in isolation. Future work should prioritize validating the framework on live, operational network traffic rather than benchmark corpora; evaluating robustness against adversarially crafted evasion traffic; extending the fusion strategy to encrypted and IoT-specific traffic through federated or collaborative learning across organizations, following the direction established by recent federated intrusion detection research; and conducting a staged, human-in-the-loop deployment study to quantify analyst workload reduction and containment effectiveness under real SOC operating conditions.

5.1. Acknowledgments

The authors thanks colleagues in the Faculty of Computer Science, Kabul University, for helpful discussion during the preparation of this manuscript.

5.2. Conflict of Interest Statement

The authors declare no competing interests. No commercial entity, network operator, or security vendor had any role in the design of the framework, the analysis of results, or the decision to publish this article.

5.3. Author Contributions

S.M.: Conceptualization, Methodology, Software, Formal Analysis, Visualization, Writing – Original Draft.
I.M.: Investigation, Data Curation, Validation, Writing – Review & Editing.

I.J.: Supervision, Resources, Validation, Project Administration, Writing – Review & Editing.

5.4. Funding Statement

This research received no external funding. The work was conducted without any grant or financial support from public, commercial, or not-for-profit bodies.

5.5. Informed Consent

Not applicable.

5.6. Ethical Approval

Not applicable.

5.7. Data Availability

The datasets analyzed in this study are publicly available: UNSW-NB15 (Moustafa & Slay, 2015, 2016) and CIC-IDS2017 (Sharafaldin et al., 2018). The preprocessing scripts, the leave-one-attack-category-out partition scheme, and the implementation of the hybrid detection engine and fusion module can be obtained from the corresponding author [S.M.] upon reasonable request, subject to the terms of use of the original dataset providers.

5.8. Declaration of Generative AI and AI-assisted Technologies

During the preparation of this manuscript, the authors utilized Claude (Anthropic) for language editing and clarity improvement. All outputs generated with the assistance of the tool were critically reviewed and revised by the authors to ensure technical accuracy, faithfulness to the cited literature, and adherence to academic standards. The authors assume full responsibility for the integrity and content of this manuscript.

6. References

Abri, F., Siami-Namini, S., Khanghah, M. A., Soltani, F. M., & Namin, A. S. (2019). Can Machine/Deep Learning Classifiers Detect Zero-Day Malware with High Accuracy? In *2019 IEEE International Conference on Big Data (Big Data)* (pp. 3252–3259). IEEE. https://doi.org/10.1109/bigdata47090.2019.9006514

Agrawal, S., Sarkar, S., Aouedi, O., Yenduri, G., Piamrat, K., Alazab, M., Bhattacharya, S., Maddikunta, P. K. R., & Gadekallu, T. R. (2022). Federated Learning for intrusion detection system: Concepts, challenges and future directions. *Computer Communications*, *195*, 346–361. https://doi.org/10.1016/j.comcom.2022.09.012

Ahmad, R., Alsmadi, I., Alhamdani, W., & Tawalbeh, L. (2022). A Deep Learning Ensemble Approach to Detecting Unknown Network Attacks. *Journal of Information Security and Applications*, *67*, 103196. https://doi.org/10.1016/j.jisa.2022.103196

Axelsson, S. (2000). The base-rate fallacy and the difficulty of intrusion detection. *ACM Transactions on Information and System Security*, *3*(3), 186–205. https://doi.org/10.1145/357830.357849

Bierbrauer, D. A., De Lucia, M. J., Reddy, K., Maxwell, P., & Bastian, N. D. (2023). Transfer learning for raw network traffic detection. *Expert Systems with Applications*, *211*, 118641. https://doi.org/10.1016/j.eswa.2022.118641

Bilge, L., & Dumitraş, T. (2012). Before we knew it. In *Proceedings of the 2012 ACM conference on Computer and communications security* (pp. 833–844). ACM. https://doi.org/10.1145/2382196.2382284

Buczak, A. L., & Guven, E. (2016). A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection. *IEEE Communications Surveys & Tutorials*, *18*(2), 1153–1176. https://doi.org/10.1109/comst.2015.2494502

Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection. *ACM Computing*

Surveys, *41*(3), 1–58. https://doi.org/10.1145/1541880.1541882

Chen, T., & Guestrin, C. (2016). XGBoost. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785–794). ACM. https://doi.org/10.1145/2939672.2939785

da Costa, K. A. P., Papa, J. P., Lisboa, C. O., Munoz, R., & de Albuquerque, V. H. C. (2019). Internet of Things: A survey on machine learning-based intrusion detection approaches. *Computer Networks*, *151*, 147–157. https://doi.org/10.1016/j.comnet.2019.01.023

Guo, Y. (2023). A review of Machine Learning-based zero-day attack detection: Challenges and future directions. *Computer Communications*, *198*, 175–185. https://doi.org/10.1016/j.comcom.2022.11.001

Hindy, H., Atkinson, R., Tachtatzis, C., Colin, J.-N., Bayne, E., & Bellekens, X. (2020). Utilising Deep Learning Techniques for Effective Zero-Day Attack Detection. *Electronics*, *9*(10), 1684. https://doi.org/10.3390/electronics9101684

Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, *9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Khan, I. A., Moustafa, N., Pi, D., Haider, W., Li, B., & Jolfaei, A. (2022). An Enhanced Multi-Stage Deep Learning Framework for Detecting Malicious Activities From Autonomous Vehicles. *IEEE Transactions on Intelligent Transportation Systems*, *23*(12), 25469–25478. https://doi.org/10.1109/tits.2021.3105834

Kwon, D., Kim, H., Kim, J., Suh, S. C., Kim, I., & Kim, K. J. (2017). A survey of deep learning-based network anomaly detection. *Cluster Computing*, *22*(S1), 949–961. https://doi.org/10.1007/s10586-017-1117-8

Lai, Q., Xiong, C., Chen, J., Wang, W., Chen, J., Gadekallu, T. R., Cai, M., & Hu, X. (2024). Improved Transformer-Based Privacy-Preserving Architecture for Intrusion Detection in Secure V2X Communications. *IEEE Transactions on Consumer Electronics*, *70*(1), 1810–1820. https://doi.org/10.1109/tce.2023.3324081

Li, P., Wang, Y., Li, Q., Liu, Z., Xu, K., Ren, J., Liu, Z., & Lin, R. (2023). Learning from Limited Heterogeneous Training Data: Meta-Learning for Unsupervised Zero-Day Web Attack Detection across Web Domains. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security* (pp. 1020–1034). ACM. https://doi.org/10.1145/3576915.3623123

Liu, F. T., Ting, K. M., & Zhou, Z.-H. (2008). Isolation Forest. In *2008 Eighth IEEE International Conference on Data Mining* (pp. 413–422). IEEE. https://doi.org/10.1109/icdm.2008.17

Moustafa, N., & Slay, J. (2015). UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In *2015 Military Communications and Information Systems Conference (MilCIS)* (pp. 1–6). IEEE. https://doi.org/10.1109/milcis.2015.7348942

Moustafa, N., & Slay, J. (2016). The evaluation of Network Anomaly Detection Systems: Statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set. *Information Security Journal: A Global Perspective*, *25*(1–3), 18–31. https://doi.org/10.1080/19393555.2015.1125974

Nguyen, T. D., Marchal, S., Miettinen, M., Fereidooni, H., Asokan, N., & Sadeghi, A.-R. (2019). DÏoT: A Federated Self-learning Anomaly Detection System for IoT. In *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)* (pp. 756–767). IEEE. https://doi.org/10.1109/icdcs.2019.00080

Nkongolo, M., & Tokmak, M. (2023). Zero-Day Threats Detection for Critical Infrastructures. In *Communications in Computer and Information Science* (pp. 32–47). Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-39652-6_3

Ohtani, T., Yamamoto, R., & Ohzahata, S. (2024). IDAC: Federated Learning-Based Intrusion Detection Using Autonomously Extracted Anomalies in IoT. *Sensors*, *24*(10), 3218. https://doi.org/10.3390/s24103218

Redino, C., Nandakumar, D., Schiller, R., Choi, K., Rahman, A., Bowen, E., Shaha, A., Nehila, J., & Weeks, M. (2022). Zero Day Threat Detection Using Graph and Flow Based Security Telemetry. In *2022 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS)* (pp. 655–662). IEEE. https://doi.org/10.1109/icccis56430.2022.10037596

Restuccia, F., D'Oro, S., & Melodia, T. (2018). Securing the Internet of Things in the Age of Machine Learning and Software-Defined Networking. *IEEE Internet of Things Journal*, *5*(6), 4829–4842. https://doi.org/10.1109/jiot.2018.2846040

Roesch, M. (1999). Snort - Lightweight Intrusion Detection for Networks. *Proceedings of the 13th USENIX Conference on System Administration*, 229–238. https://doi.org/10.5555/1039834.1039864

Sabottke, C., Suciu, O., & Dumitras, T. (2015). Vulnerability Disclosure in the Age of Social Media: Exploiting Twitter for Predicting Real-World Exploits. *24th USENIX Security Symposium (USENIX Security 15)*, 1041–1056. https://www.usenix.org/conference/usenixsecurity15/technical-sessions/presentation/sabottke

Sarhan, M., Layeghy, S., Gallagher, M., & Portmann, M. (2023). From zero-shot machine learning to zero-day attack detection. *International Journal of Information Security*, *22*(4), 947–959. https://doi.org/10.1007/s10207-023-00676-0

Sarhan, M., Layeghy, S., & Portmann, M. (2022). Evaluating Standard Feature Sets Towards Increased Generalisability and Explainability of ML-Based Network Intrusion Detection. *Big Data Research*, *30*, 100359. https://doi.org/10.1016/j.bdr.2022.100359

Sedjelmaci, H., & Ansari, N. (2024). On Cooperative Federated Defense to Secure Multiaccess Edge Computing. *IEEE Consumer Electronics Magazine*, *13*(4), 24–31. https://doi.org/10.1109/mce.2022.3167527

Sharafaldin, I., Habibi Lashkari, A., & Ghorbani, A. A. (2018). Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization. In *Proceedings of the 4th International Conference on Information Systems Security and Privacy* (pp. 108–116). SCITEPRESS - Science and Technology Publications. https://doi.org/10.5220/0006639801080116

Shone, N., Ngoc, T. N., Phai, V. D., & Shi, Q. (2018). A Deep Learning Approach to Network Intrusion Detection. *IEEE Transactions on Emerging Topics in Computational Intelligence*, *2*(1), 41–50. https://doi.org/10.1109/tetci.2017.2772792

Sommer, R., & Paxson, V. (2010). Outside the Closed World: On Using Machine Learning for Network Intrusion Detection. In *2010 IEEE Symposium on Security and Privacy* (pp. 305–316). IEEE. https://doi.org/10.1109/sp.2010.25

Vasilomanolakis, E., Karuppayah, S., Mühlhäuser, M., & Fischer, M. (2015). Taxonomy and Survey of Collaborative Intrusion Detection. *ACM Computing Surveys*, *47*(4), 1–33. https://doi.org/10.1145/2716260

Vinayakumar, R., Alazab, M., Soman, K. P., Poornachandran, P., Al-Nemrat, A., & Venkatraman, S. (2019). Deep Learning Approach for Intelligent Intrusion Detection System. *IEEE Access*, *7*, 41525–41550. https://doi.org/10.1109/access.2019.2895334

Wang, Z., Fok, K. W., & Thing, V. L. L. (2022). Machine learning for encrypted malicious traffic detection: Approaches, datasets and comparative study. *Computers & Security*, *113*, 102542. https://doi.org/10.1016/j.cose.2021.102542

Zhang, J., Liang, S., Ye, F., Hu, R. Q., & Qian, Y. (2023). Towards Detection of Zero-Day Botnet Attack in IoT Networks Using Federated Learning. In *ICC 2023 - IEEE International Conference on Communications* (pp. 7–12). IEEE. https://doi.org/10.1109/icc45041.2023.10279423

Zhao, P., Cao, Z., Jiang, J., & Gao, F. (2023). Practical Private Aggregation in Federated Learning Against Inference Attack. *IEEE Internet of Things Journal*, *10*(1), 318–329. https://doi.org/10.1109/jiot.2022.3201231