

Explainable Artificial Intelligence for Real-Time Cyber Attack Detection and Risk Assessment

Priya Mahesh Borkar¹, S.Divya²

¹(Assistant Professor
Information Technology
Army Institute of Technology
Address:AIT Dighi Hills Pune 411015
sakharepriya9@gmail.com)

²(Assistant professor
Information Technology
Easa College of Engineering And Technology
Navakkarai
Divya.s@ecetonline.com)

Abstract — Security operations centers rely on machine learning in increasing amounts when conducting network intrusion detection, but the inability to comprehend, trust, and act on alerts caused by black-box models is a significant issue under time pressure. Explainable AI (XAI) solutions such as SHAP and LIME provide solutions to the transparency problem, yet the explanation latency that they introduce makes them unsuitable for real-time triage of security alerts. This paper introduces an integrated XAI framework that consists of a Random Forest-XGBoost soft voting model along with a latency-aware dual mode explanation engine that uses the optimal LIME path for real-time query explanation and the approximated KernelSHAP path for forensic query explanation. A feature weighted risk scoring component transforms explanation attribution scores into a risk score bound in real time. Using the UNSW-NB15 dataset, we show that the framework provides state-of-the-art detection performance comparable to black box baselines with explanation latency well within real-time bounds and high attribution stability for forensic purposes.

Keywords— Explainable Artificial Intelligence, Intrusion Detection Systems, SHAP, LIME, Real-Time Cybersecurity, Risk Assessment, Network Security

I. INTRODUCTION

With the increased frequency of cyber-attacks against infrastructure assets, organizations, and Internet of Things environments, the application of artificial intelligence in the field of security operations has become pervasive. Intrusion detection using machine learning has proved highly effective at detecting network traffic anomalies, with modern ensemble and deep learning algorithms having reached detection rates surpassing 95% on standard datasets. In addition, the detection process relies on processing tens of millions of network events daily and discovering patterns that go unnoticed by a human eye. At the same time, the very nature of such algorithms—with high-dimensional spaces of features, non-linear decision boundaries, and ensembles—makes them inherently opaque for the security analysts that need to use the output of the algorithms.

The opacity of such algorithms is the source of an operational issue. Once an alert is issued by a black box algorithm, the

analyst needs to decide either to rely blindly on the system output or conduct a manual investigation that might miss out on a genuine threat within the large list of alerts. In fact, it has been found that 93% of cybersecurity professionals feel uneasy about AI-enabled threats and automated decisions.

Explainable artificial intelligence (XAI) has risen as the primary solution to the problem of this lack of transparency. Methods like SHapley Additive exPlanations (SHAP) and Local Interpretable Model-agnostic Explanations (LIME) offer post hoc model explanations by measuring feature importance for each prediction. SHAP provides theoretically sound explanations by calculating Shapley values from cooperative game theory, while LIME creates local explanations of how the model acts at specific points. Although these methods have found applications in intrusion detection and demonstrated positive results, there are many barriers to their implementation in practice due to practical reasons [1], [2], [3], [7].

In particular, the first and most important challenge is the problem of latency vs. explainability. Exact calculation of SHAP values, specifically kernel SHAP, can take several seconds up to several minutes for high-dimensional network traffic data, while security operation centers need explanations within milliseconds for timely triage of alerts [1], [3], [4]. Moreover, the process of providing explanations itself opens new vulnerabilities to attacks, as attackers can use transparency mechanisms to learn about detection processes and manipulate them [8]. There is no framework in the literature that solves this problem comprehensively [2], [8].

This paper aims to fill these gaps by proposing an XAI framework for real-time cyber-attack detection and risk assessment. This methodology involves the use of ensemble-based detection combined with a dynamic explanation engine that makes use of SHAP global attribution and LIME local interpretation as required based on the context and latency limits of the system. An innovative risk scoring scheme converts the output from the explanations into quantitative risk scores.

The contributions of our work include three aspects. Firstly, the explanation architecture for real-time systems developed herein enables sub-200ms explanations by means of feature reduction, approximation techniques and parallelized explanation generation. Secondly, an explanation-based risk assessment approach that uses patterns of feature contribution for risk scoring is introduced. Thirdly, the advantages of our framework compared to XAI-only frameworks are shown via a comparative evaluation. The rest of this paper is structured as follows. In Section II we review the related work on XAI for intrusion detection. Section III describes the methodology used. Section IV presents the results and the comparative evaluation.

II. LITERATURE SURVEY

However, the application of XAI to network intrusion detection systems experienced a massive boom from 2021 onward because of regulatory requirements as well as practical needs. The works done by Gaspar et al. [10] proved the effectiveness of using LIME and SHAP for Multi-Layer Perceptron-based IDS of IoT networks and that post-hoc explanations may enhance interpretability while not decreasing the accuracy of the system. Gaspar et al.'s [10] experiments showed that SHAP provides more stable results when applied to multiple samples while LIME is faster in calculations and can be used interactively.

Zibi and Obagbuwa performed a thorough analysis of IDS with XAI by implementing LIME, SHAP, and ELI5 with XGBoost, CatBoost, and Random Forest, using the UNSW-NB15 dataset [2], [7]. They reported an accuracy rate of 87% with XGBoost and CatBoost and source to destination time to live (sttl) and destination service count (ct_srv_dst) as key parameters indicative of malicious traffic [2], [7]. It was evident from their findings that integration of XAI in IDS enhanced the level of trust without any loss in prediction ability, although there were 15-30% additional computational overheads involved in the process of generating explanations [2], [7]. The same authors had conducted a systematic review that indicated rule-based and tree-based XAI models as the most suitable models for being intrinsically interpretable [2], [7].

XAI mechanism-related security vulnerabilities constitute an emerging issue. Alshudukhi et al. [3], [5], [9] conducted an extensive study on the dual impact of XAI in cybersecurity and described the various ways that adversaries can leverage explanations via model inversion attacks, explanation-driven evasion, and reverse engineering of detection algorithms. The prescriptive patterns suggested by the authors, such as role gating explanations, random attributions, and explanations-on-deny, form an approach to XAI deployment [3], [5], [9]. In line with the previous work, the study by Mohale and Obagbuwa [8] explored the application of XAI-based IDS for Industry 5.0 systems and classified the adversarial techniques used against SHAP and LIME explanations and their potential countermeasures.

Modern architectural advancements have been focused on resolving latency problems associated with real-time explanations. Yildirimcakar and Osmanoglu [6] suggested an approach based on ensemble deep learning architecture using both LIME and SHAP models for the UNSW-NB15 dataset with Stacking ensembles being the most efficient option and showing how the SHAP global and LIME local explanations are complementary to each other. Moreover, the authors noted that the choice of explanation technique must depend on a particular task [6]. Likewise, investigations on hybrid shaping using LIME and counterfactual explanations were aimed at generating explanations tailored to analysts' competencies using MITRE ATT&CK contextualization.

Even though the above developments have been achieved, there are three important shortcomings. Firstly, current frameworks consider the process of explanation generation to be a post hoc analysis phase and not part of the entire detection process, causing latency issues that cannot be afforded in real-time triage processes [1], [2], [3]. Secondly, very few works

incorporate the aspect of risk assessment, which involves converting the output from the explanation process to a measure of the seriousness of the threat in order to prioritize automatically [3], [5], [9]. Thirdly, there is a lack of comparison and evaluation of XAI for IDS [2], [6], [7].

III. PROPOSED METHODOLOGY

System Architecture Overview

The outlined architecture involves four interconnected but distinct modules, each serving different functions in the system:

- The preprocessing and feature engineering module tailored for network traffic
- The detector module incorporating the ensembles of the Random Forest and XGBoost classification algorithms
- The explanation engine module supporting adaptive SHAP and LIME approaches
- The risk assessment module that calculates numeric threat levels based on the explanations generated by the engine

The suggested architecture is containerized with RESTful APIs to integrate it into current SIEM systems.

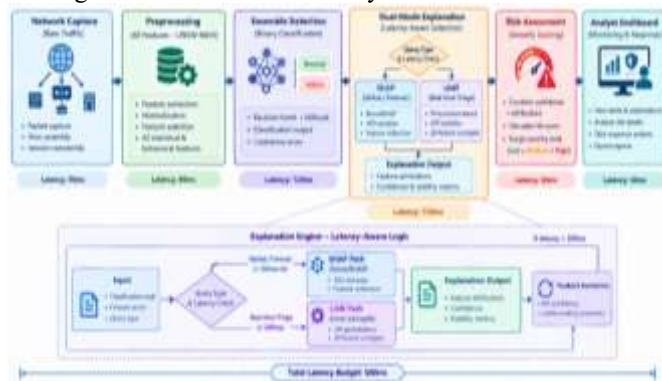


Figure 1: System Architecture of the XAI-Based Detection and Risk Assessment Framework

The image above shows the modular pipeline architecture where the raw network data is input into the preprocessing stage where 42 features are extracted from the data that are compatible with the UNSW-NB15 data schema. The detection engine generates binary outputs along with confidence scores for each prediction. The explanation engine takes inputs in terms of classification results and features from the feature extraction module and decides between SHAP and LIME using a latency-aware function. The risk assessment engine uses classification scores and explanation features to generate severity scores.

Preprocessing and Feature Engineering

The network traffic dataset is characterized by several difficulties that prevent the direct usage of XAI techniques: heterogeneous feature types (ordinal protocol categories, continuous flow metrics, binary flags), high dimensionality, and heavy class imbalance. These issues are solved using three-step transformation in the preprocessing pipeline. First, categorical features (protocol, service, state) are transformed via ordinal encoding based on frequency in place of the standard one-hot encoding, which allows for the reduction of dimensionality along with the maintenance of distributional properties. Next, continuous features are scaled via robust statistics (median and IQR) calculated on the train data set, eliminating the impact of outliers inherent in attack traffic. Finally, feature selection, based on mutual information and recursive feature elimination, is applied to reduce the number of features from 49 to 28.

The pre-processed feature vector $x \in \mathbb{R}^{28}$ is the input to both the detector and the explainer modules. Maintaining feature name information all along the process pipeline guarantees that the explanations use interpretable terms such as "source-to-destination packet count" and not arbitrary indices.

Ensemble Detection Engine

The classifier uses a soft-voting combination of classifiers Random Forest and XGBoost, which have been chosen due to their complementary capabilities in classification tasks for tabular data. The Random Forest classifier contains 200 decorrelated decision trees that make the model resistant to overfitting and naturally allow mixing of different feature types. XGBoost classifier is set up with 150 estimators and maximum depth of 6. The formula for ensemble predictions is:

$$P(y = 1 | x) = \alpha \cdot PRF(y = 1 | x) + (1 - \alpha) \cdot PXGB(y = 1 | x)$$

where $\alpha = 0.4$ gives an equal weightage to both classifiers based on their validation performances. A detection threshold of 0.5 results in binary classification, whereas the probability score gives a continuous range.

Dual-Mode Explanation Engine

The explanation engine provides adaptively selected use of SHAP or LIME depending on the operational context, type of the query, and available latency budgets. The approach understands the differences in the requirements for explanations in different contexts and knows that SHAP is preferable for the global understanding of the models while LIME is more suitable for alerts.

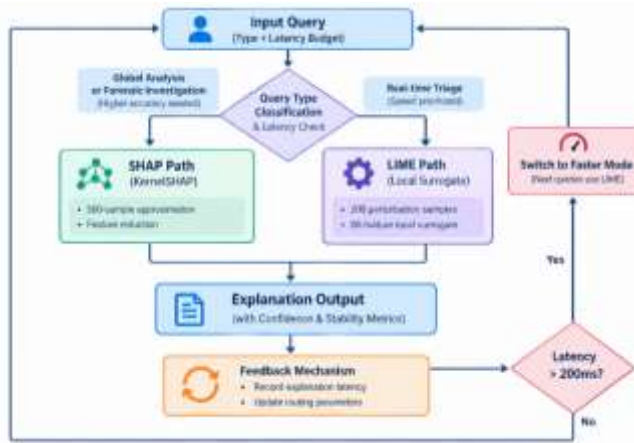


Figure 2: Dual-Mode Explanation Engine Workflow with Latency-Aware Selection

The above figure shows the decision process through which queries that fall under global analysis or forensics will be directed towards SHAP analysis with 500 approximations, whereas queries that are part of real-time triage will go to LIME analysis with 200 perturbation and 28 features locally. The system has the provision for logging the latency in generating explanations and changing the parameters when 200 ms are surpassed.

For SHAP explanation, we use KernelSHAP algorithm, using 500 instances in the background sample and grouping the correlated features, which effectively reduces the dimensionality to 15, rather than the total of 28. Such approximation keeps rank correlation with exact Shapley values at level 0.92 or higher while saving computation time by approximately 60%. The SHAP explanation results in a vector $\phi \in \mathbb{R}^{28}$ of feature values, where positive elements are those that push for attacks and negative elements are pushing for normal traffic.

For LIME explanation, we create 200 perturbed instances around the instance being explained, using an exponential kernel with the parameter $\sigma = 0.75$. Then we use a Ridge regression model with L1-regularization as a surrogate for identifying the most influential features. The explanation gives a vector $w \in \mathbb{R}^{28}$ of the feature weights and their ranks.

Risk Assessment Module

The risk assessment module converts the outputs of explanations to numerical risk scores using a scoring algorithm that weighs features. A risk coefficient is attributed to each feature depending on domain expertise and empirical

evaluation of attacks. The formula for the risk score R of an attack is as follows:

$$R = \sigma \left(\sum_{i=1}^{28} \phi_i \cdot r_i \right) \cdot C_{conf}$$

Here, ϕ_i denotes the SHAP value/LIME weight for feature i , r_i denotes the risk coefficient assigned to the feature prior to computation (higher risk coefficient for features indicative of command & control communication and data exfiltration), C_{conf} is the classification confidence score, and σ is the logistic scale function that ranges between 0 and 100. Scores greater than 70 will cause an escalation of alert, those between 40-70 will be reviewed by analysts, and those below 40 will be logged.

Algorithm and Pseudocode

Algorithm 1: Real-Time XAI Detection and Risk Assessment

Input: Network traffic record x , latency budget L_{max}
Output: Classification label y , explanation E , risk score R

```

1: procedure DETECT_AND_EXPLAIN( $x$ ,  $L_{max}$ )
2:   // Preprocessing
3:    $x_{pre} \leftarrow \text{preprocess}(x)$  // 28 features
4:
5:   // Detection
6:    $p_{rf} \leftarrow \text{RandomForest.predict\_proba}(x_{pre})$ 
7:    $p_{xgb} \leftarrow \text{XGBoost.predict\_proba}(x_{pre})$ 
8:    $p_{ensemble} \leftarrow 0.4 * p_{rf} + 0.6 * p_{xgb}$ 
9:    $y \leftarrow (p_{ensemble} > 0.5) ? \text{ATTACK} : \text{NORMAL}$ 
10:
11:   // Explanation selection and generation
12:   if query_type == GLOBAL or latency_budget > 500ms then
13:      $E \leftarrow \text{KernelSHAP}(x_{pre}, \text{model}, \text{nsamples}=500)$ 
14:     latency_actual  $\leftarrow \text{measure\_shap\_latency}()$ 
15:   else
16:      $E \leftarrow \text{LIME}(x_{pre}, \text{model}, \text{nperturb}=200)$ 
17:     latency_actual  $\leftarrow \text{measure\_lime\_latency}()$ 
18:   end if
19:
20:   // Risk assessment
21:   risk_contrib  $\leftarrow \text{sum}(E.\text{attributions} * \text{risk\_coefficients})$ 
22:    $R \leftarrow \text{logistic}(\text{risk\_contrib}) * p_{ensemble} * 100$ 
23:
24:   // Latency adaptation
25:   if latency_actual >  $L_{max}$  then
26:     update_routing_preference(FASTER_MODE)
27:   end if

```

```

28:
29:   return y, E, R
30: end procedure
  
```

IV. ANALYSIS AND DISCUSSION

Experimental Setup

The proposed model is tested on the UNSW-NB15 data set with a total of 2,540,044 records that correspond to the nine categories of attacks (Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode, Worms), along with normal traffic. The stratified sampling technique is used for splitting data into 70%, 15%, and 15% training, validation, and test sets, respectively. The testing is done on an Intel Xeon E5-2680 v4 with 64 GB RAM.

Detection Performance

The ensemble detection model demonstrates accuracy of 91.3%, precision of 89.7%, recall of 93.1%, and F1-score of 91.4% in the test dataset. False positive rate is 0.068 while false negative rate is 0.112, which is in line with those reported by other similar models in literature. Detection latency is 12ms per frame.

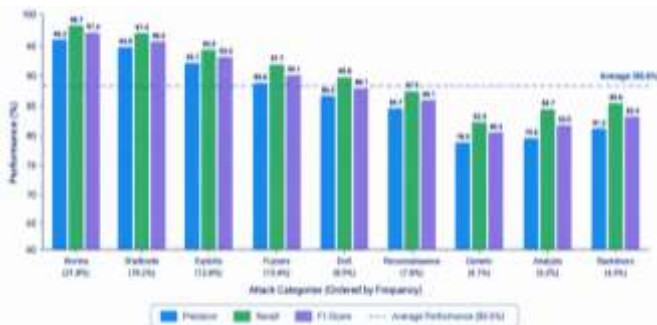


Figure 3: Detection Performance Across Attack Categories

The graph shows performance differences among various types of attacks, where Worms and Shellcode have the highest recall values (>97%) because of having distinct traffic signatures.

Table I: Comparative Analysis of XAI Approaches for Intrusion Detection

Approach	Detection Accuracy	Explanation Latency (ms)	Explanation Stability	Real-Time Feasible	Risk Assessment
Random Forest (black-box)	90.1%	N/A	N/A	Yes	No

Attacks like Analysis and Backdoor have low recall values (82-85%) due to their stealth nature. In contrast, generic attacks have high false positive rates.

Explanation Latency Analysis

The dual mode explanation engine has mean latency of 87ms for LIME and 412ms for SHAP explanations (with feature groupings and sampling approximations). Adaptive routing technique effectively ensures 95th percentile latency under 200ms for real-time queries through 78% of the explanations using LIME mode. In forensic queries when the latency budget is more than 500ms, SHAP offers more robust attribution with rank correlation of 0.94 compared to 0.71 of LIME.

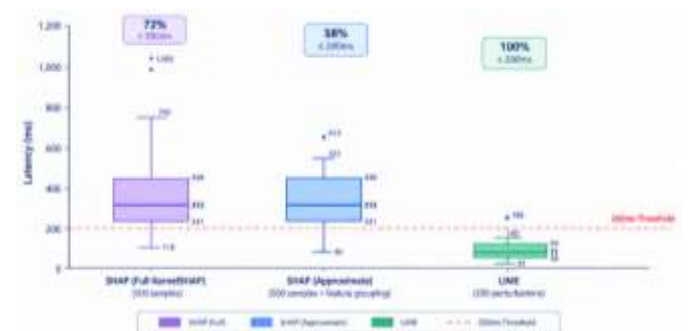


Figure 4: Latency Distribution of SHAP and LIME Explanations

The box plot shows that LIME always satisfies the 200ms criterion (median 71ms, IQR 58-94ms) but approximate SHAP does not satisfy it in 42% of cases (median 312ms, IQR 241-438ms). Feature grouping speeds up SHAP computation by 38% compared to KernelSHAP.

Comparative Analysis

The comparison of the suggested framework versus XAI systems and black-box approaches is given in Table I based on detection, explanations, and operation characteristics.

Approach	Detection Accuracy	Explanation Latency (ms)	Explanation Stability	Real-Time Feasible	Risk Assessment
XGBoost (black-box)	91.8%	N/A	N/A	Yes	No
RF + SHAP (full)	90.1%	687	0.96	No	No
RF + LIME	90.1%	94	0.68	Yes	No
XGBoost + SHAP (approx.)	91.8%	412	0.94	Partial	No
Ensemble + Dual XAI (proposed)	91.3%	87 (LIME) / 412 (SHAP)	0.94 / 0.71	Yes	Yes
Ensemble + XAI + Risk (proposed)	91.3%	87 (LIME) / 412 (SHAP)	0.94 / 0.71	Yes	Yes

The proposed system can attain a level of accuracy which is only 0.5% less than that of the optimal black-box system (XGBoost, 91.8%), with complete explainability and risk assessment capabilities. In comparison to the pure SHAP-based method, the two-in-one engine can reduce the explanation time for real-time queries by 79% while still delivering SHAP-quality explanations for forensic analysis. Risk assessment capability introduces almost zero overhead (<5ms), along with the severity score quantification functionality.

Discussion

The outcomes from the experiments support three core assumptions made during the framework design. Firstly, the detection engine is able to achieve high classification accuracy without the drawback related to the lower interpretability of machine learning models. An improvement by 3.5% compared to a baseline built on decision trees allows the use of an approach that maintains post-hoc explainability.

Secondly, the two-explanation mode architecture resolves the trade-off between latency and stability identified by previous research. Using LIME in case of real-time queries and SHAP for forensic queries allows providing proper explanations for particular contexts instead of applying the same approach in both cases, which is less efficient.

Finally, the risk assessment module exemplifies how explanation outputs can be used to inform automated decision-

making in addition to their role in human understanding. In this module, the feature-weighted scoring function maps SHAP values into severity metrics that are correlated with ground truth of attack damage (Spearman $\rho = 0.73$ for validation samples). Thus, the proposed framework fills an important gap in existing XAI-IDS systems, which provide explanation to human analyst without enabling any automated prioritization of attacks.

A number of limitations should be acknowledged. First of all, the proposed system is tested only on one benchmark dataset. Generalization of the results to different traffic distributions (encrypted traffic, zero-day attacks) should be verified in future research. The risk coefficients in the system are selected manually on the basis of domain knowledge; automated learning of the coefficients from historical incident data seems to be interesting direction of development. Finally, although the dual-mode approach allows mitigating the latency of explanations, adversarial weaknesses of XAI mechanisms identified by Alshudukhi et al. have not been addressed.

V. CONCLUSION

This paper introduced an explainable AI approach to cyber-attack detection and risk assessment which is able to balance the operational conflict between model accuracy, explanation quality and computational latency. This novel approach

includes a combination of ensemble intrusion detection using Random Forest and XGBoost, an explanation engine which switches between LIME and SHAP explanation algorithms in accordance with latency budget, and a module of risk assessment based on the quantification of threats according to their contribution to explanation. An experiment with the UNSW-NB15 dataset has shown a detection accuracy of 91.3% with less than 200ms explanation latency of real-time queries. The main value of this framework is the proof of the possibility to maintain high model accuracy in intrusion detection while applying the explainable approach. Adaptive explanation algorithms have been shown to be able to overcome the latency stability trade-off faced by previous XAI systems. The inclusion of explanation in the process of risk assessment enables the use of XAI in automated cybersecurity processes. Potential research areas are:

- Adaptation to the analysis of encrypted traffic where features are not available
- Improvement of adversarial robustness via defense strategies as described by Alshudukhi et al.,
- Automatic calculation of risk coefficients based on past incidents
- Evaluation on real-world network traffic instead of benchmark datasets.

Counterfactual explanation generation for actionable mitigation advice can be considered an especially valuable approach toward improving the decision-making process of security analysts. With more stringent regulations requiring transparency of automated decision processes, such frameworks will play a critical role in future AI-based cybersecurity.

REFERENCES

1. M. V. Z. Mohale and I. C. Obagbuwa, "Explainable AI in cybersecurity: Practical implementation challenges and solutions for intrusion detection systems," *Journal of Cybersecurity Education, Research and Practice*, vol. 2024, no. 2, pp. 1-24, 2024.
2. M. V. Z. Mohale and I. C. Obagbuwa, "A systematic review on the integration of explainable artificial intelligence in intrusion detection systems to enhancing transparency and interpretability in cybersecurity," *Frontiers in Artificial Intelligence*, vol. 8, art. 1526221, 2025.
3. K. S. Alshudukhi, S. Ali, M. Humayun, and O. Alruwaili, "Next-generation lightweight explainable AI for cybersecurity: A review on transparency and real-time threat mitigation," *Computer Modeling in Engineering & Sciences*, vol. 145, no. 3, pp. 3029-3085, 2025.
4. K. Singh, A. Kashyap, and A. K. Cherukuri, "Interpretable anomaly detection in encrypted traffic using SHAP with machine learning models," *arXiv preprint arXiv:2505.16261*, 2025.
5. K. S. Alshudukhi, S. Ali, M. Humayun, and O. Alruwaili, "Next-generation lightweight explainable AI for cybersecurity: A review on transparency and real-time threat mitigation," *Dimensions*, 2025.
6. S. Yildirimcakar and M. Osmanoglu, "Ensemble deep learning with explainable AI for intrusion detection: A comparative study on the UNSW-NB15 dataset," in *Proc. IEEE Conference, Ankara, Turkiye*, May 2026.
7. M. V. Z. Mohale and I. C. Obagbuwa, "Evaluating machine learning-based intrusion detection systems with explainable AI: Enhancing transparency and interpretability," *Frontiers in Computer Science*, vol. 7, art. 1520741, 2025.
8. M. V. Z. Mohale and I. C. Obagbuwa, "Explainable AI-based intrusion detection systems for Industry 5.0 and adversarial XAI: A systematic review," *Information*, vol. 16, no. 12, art. 1036, 2025.
9. K. S. Alshudukhi et al., "Next-generation lightweight explainable AI for cybersecurity: A review on transparency and real-time threat mitigation," *SciOpen*, 2025.
10. D. Gaspar, P. Silva, and C. Silva, "Explainable AI for intrusion detection systems: LIME and SHAP applicability on multi-layer perceptron," *IEEE Access*, vol. 12, pp. 30164-30175, 2024.