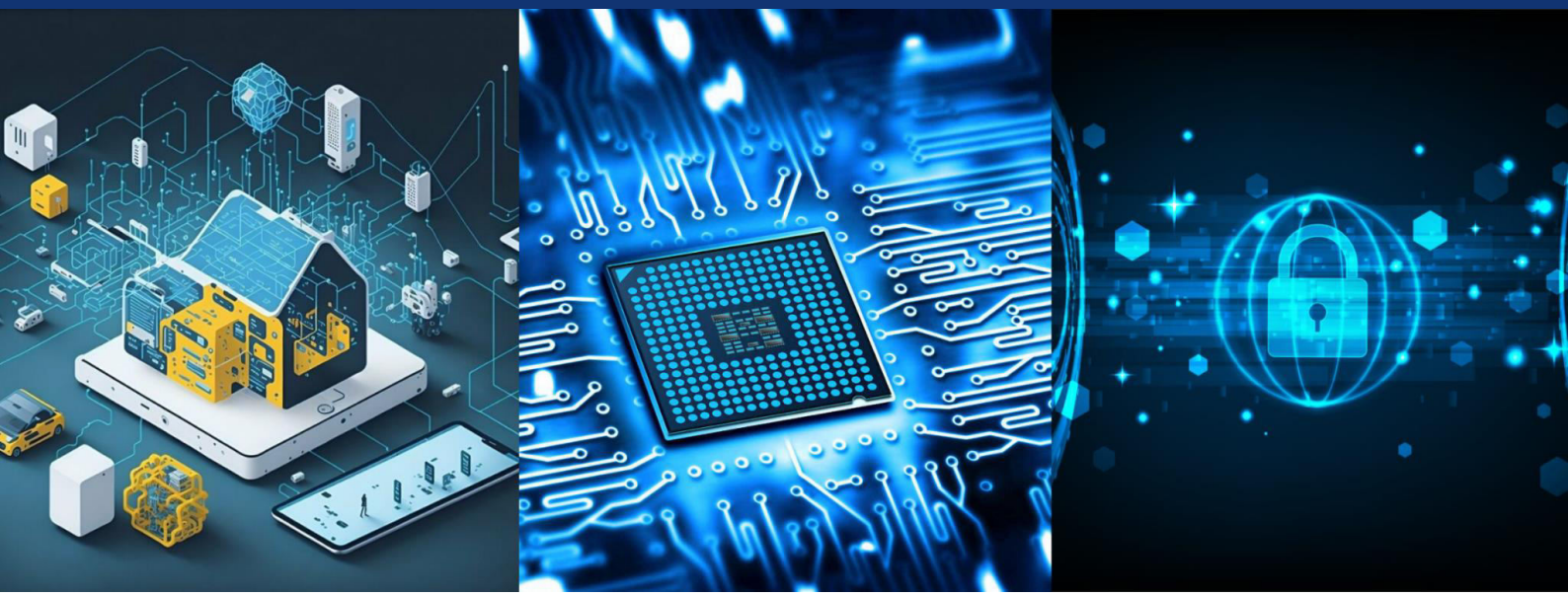
IJIRCCE

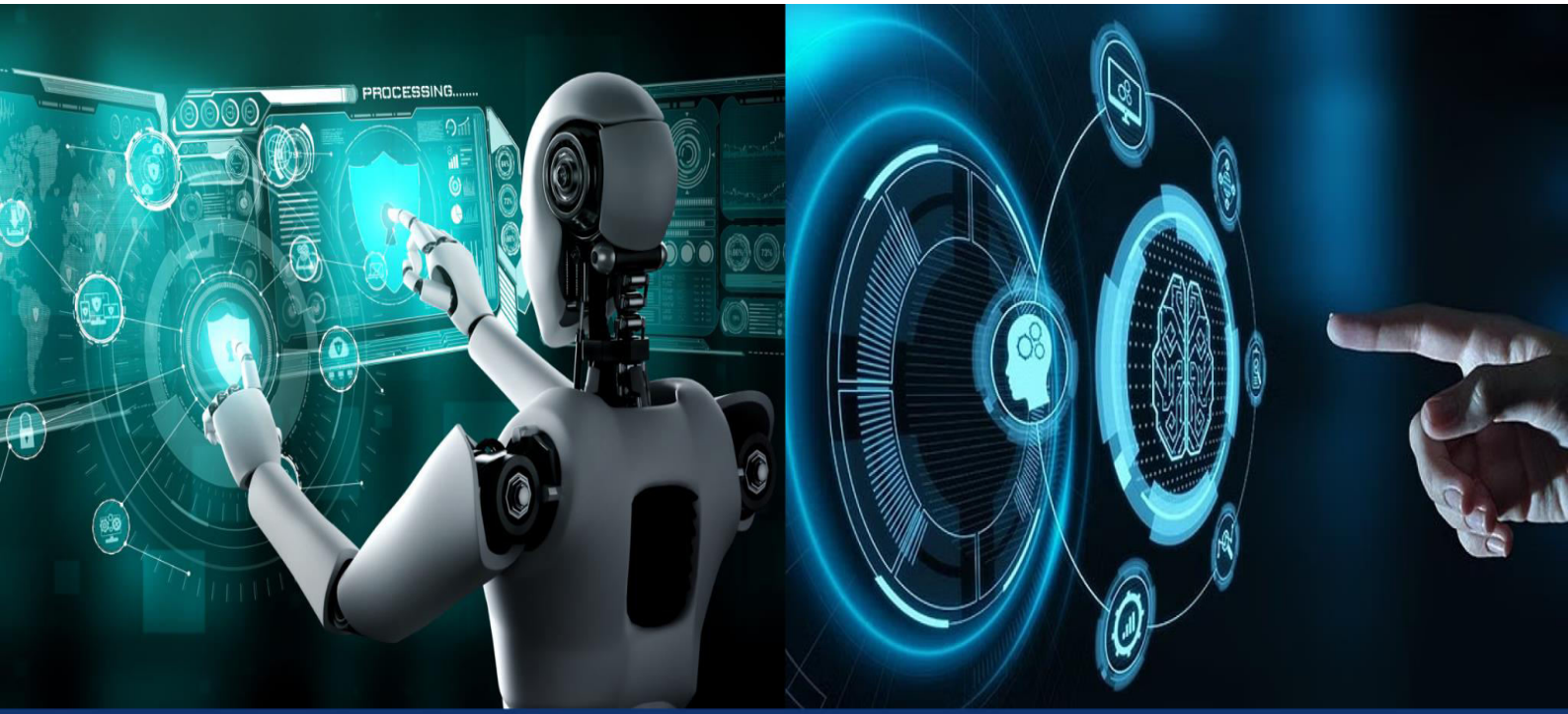
ISSN(O): 2320-9801

ISSN(P): 2320-9798



International Journal of Innovative Research in Computer and Communication Engineering

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)



Impact Factor: 9.927

Volume 14, Issue 5, May 2026

© 2026 IJIRCCE | Volume 14, Issue 5, May 2026 DOI: 10.15680/IJIRCCE.2026.1405001

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

From Intrusion Detection to Autonomous Breach Diagnosis Using Multi-Source Evidence Fusion

Serhani Aymane

PG Student, School of Computer Science, Nanjing University of Information Science & Technology, Nanjing, China

ABSTRACT: Network Intrusion Detection Systems are very good at detecting malicious network traffic; however, they do a poor job of providing analysts with the information to diagnose a breach or make decisions about what actions should be taken. This research describes an AI-based autonomous breach diagnostic system. The proposed autonomous breach diagnostic system uses machine learning algorithms, XGBoost and HistGradientBoosting, in combination with data from multiple sources, uncertainty-based classification, SHAP explanations, incident clustering/grouping, and corroborated evidence to determine which alerts may represent malicious activity. The proposed autonomous breach diagnostic system has been tested using three different datasets: CICIDS-2018, CICIDS-2017, and UNSW-NB15. On CICIDS-2018, the proposed model achieved F1 = 0.9892 and AUC-ROC = 0.9990. In addition, it reduced 405,032 raw detections down to 8,702 diagnosed breaches that included 499 high-priority alerts.

KEYWORDS: Network Intrusion Detection; Autonomous Breach Diagnosis; Uncertainty Quantification; SHAP Explainability; Evidence Corroboration.

I. INTRODUCTION

For decades researchers have been working on Network Intrusion Detection Systems (NIDS) and they generally have one goal in mind: to determine if incoming traffic on the network is benign or malicious. While determining if the traffic coming onto your network is malicious or benign is important, it isn't enough for today's Security Operations Centers (SOC). Even though you have a strong detection model, at scale the number of alerts produced will overwhelm analysts. Analysts then need to prioritize the alerts (which ones are more critical), explain what was used to generate them, determine if the alerts are related to each other (an incident), and verify if the available data supports escalating an issue.

There is a very real gap between detecting intrusions and diagnosing breaches. When a detection model gets a high F1-Score or AUC-ROC score, it means that the model works better than others when tested against a standard test set, it doesn't mean that the model provides any diagnostic context such as severity of the issue detected, certainty of the detection, evidence to support the detection, and/or urgency to respond to the issue. Without these types of additional diagnostic contexts, even the best detection models can contribute to alert fatigue and make responding to incidents take longer.

Therefore, this research fills that gap by developing an AI based autonomous breach diagnosis framework utilizing multi-source evidence. The proposed framework utilizes both XGBoost and HistGradientBoosting as part of a fusion model. Then, it enhances the output from the detection module with five states of uncertainty classification; SHAP-based explanations; groupings of incident activity; and weighted evidence to corroborate other pieces of evidence. Therefore, instead of providing a single malicious or benign classification to each piece of traffic, the proposed framework generates an incident-level diagnostic report that can assist analysts in making decisions about how to address an incident.

The proposed framework is evaluated using CICIDS-2018 as its primary benchmark, CICIDS-2017 as a secondary benchmark and corroboration source, and UNSW-NB15 as an external validation dataset. Also included in the evaluation is a temporal robustness assessment on a larger sanitized version of CICIDS-2018. Therefore, the objective of this research is to not only enhance the detection accuracy of NIDS but to demonstrate how NIDS outputs can be converted into actionable information that can aid analysts in their day-to-day operations.

© 2026 IJIRCCE | Volume 14, Issue 5, May 2026 DOI: 10.15680/IJIRCCE.2026.1405001

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

II. LITERATURE SURVEY

Machine Learning for NIDS

Network intrusion detection has been used with machine learning because flow records can be treated as structured feature vectors [1]. Classical models like logistic regression and random forest offer efficient baseline models [2], and ensemble models like XGBoost and gradient boosting are especially powerful for tabular features like CICFlowMeter [3], [4]. These models are used to represent the non-linear relationships between packet count, byte count, duration, inter-arrival time, and TCP flag statistics.

In addition to the above-mentioned models, deep learning models such as multi-layer perceptrons, convolutional networks, and CNN-LSTM hybrids have also been used in NIDS [5], [6]. The hosts can be represented as nodes, and the flows as edges; this is the direction used by graph neural networks [7]. But, deep and graph-based models can be expensive and may not outperform tree-based ensembles on tabular flow features.

Explainability, Generalization, and Alert Triage

Explainable AI techniques like SHAP and LIME aid analysts in comprehending why a model is alerting [8], [9]. If an attack has been successful, NIDS can determine if the destination port, packet length, TCP window size or inter-arrival time played a part in the malicious prediction. But this does not address the operational challenge of assigning severity, grouping of incidents or prioritizing responses.

Another longstanding limitation is the lack of cross-dataset generalization. The training process on one benchmark may be completely different from another, causing a model to be worse at generalizing to the other when tested on it. Normal traffic, attack implementation, feature distribution and labelling conventions can be different between the two benchmarks, making a model trained on one perform poorly on the other. So, a strong score on an in-distribution test needs to be validated externally and tested for temporal robustness.

The drive to prioritize is also fueled by SOC alert fatigue. Current work in the NIDS field generally deals with detection metrics and operational systems demand evidence-aware ranking. This paper is aimed at filling this gap by combining detection, uncertainty, explanation, corroboration, and priority assignment into a single reproducible framework.

Table 1: Positioning of the Proposed Framework

Direction	Detect.	Uncert.	XAI	Corr.	Priority
Traditional ML-NIDS	Yes	No	No	No	No
Deep-learning NIDS	Yes	No	Partial	No	No
XAI-based IDS	Yes	Partial	Yes	No	No
SOC prioritization	Partial	Partial	Partial	Partial	Yes
Proposed	Yes	Yes	Yes	Yes	Yes

III. METHODOLOGY / APPROACH

Framework Overview

The framework is organized as a 5-tiered autonomous breach diagnosis pipeline as proposed. Layer 1 is responsible for detection using a fusion model and baseline classifiers. Layer 2 transforms raw predictions into descriptions of the attack type, its severity, the confidence level, the uncertainty about the prediction, and the recommendation for response. The feature explanation, that is, the SHAP value is added in Layer 3. Layer 4 groups event-level detections together to create incidents and calculates corroboration scores. Layer 5 adds extra sources, like Zeek logs and live Scapy-based inference.

The difference between the framework and traditional NIDS is that the end result is not just a label, but an item of interest at the incident level. Probability, uncertainty state, explanation and corroboration score are associated with each object, along with response priority. This design enables the system to be considered as a classifier, but also as an operational diagnosis process, reducing the number of raw alerts and helping to triage incidents.

© 2026 IJIRCCE | Volume 14, Issue 5, May 2026 DOI: 10.15680/IJIRCCE.2026.1405001

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

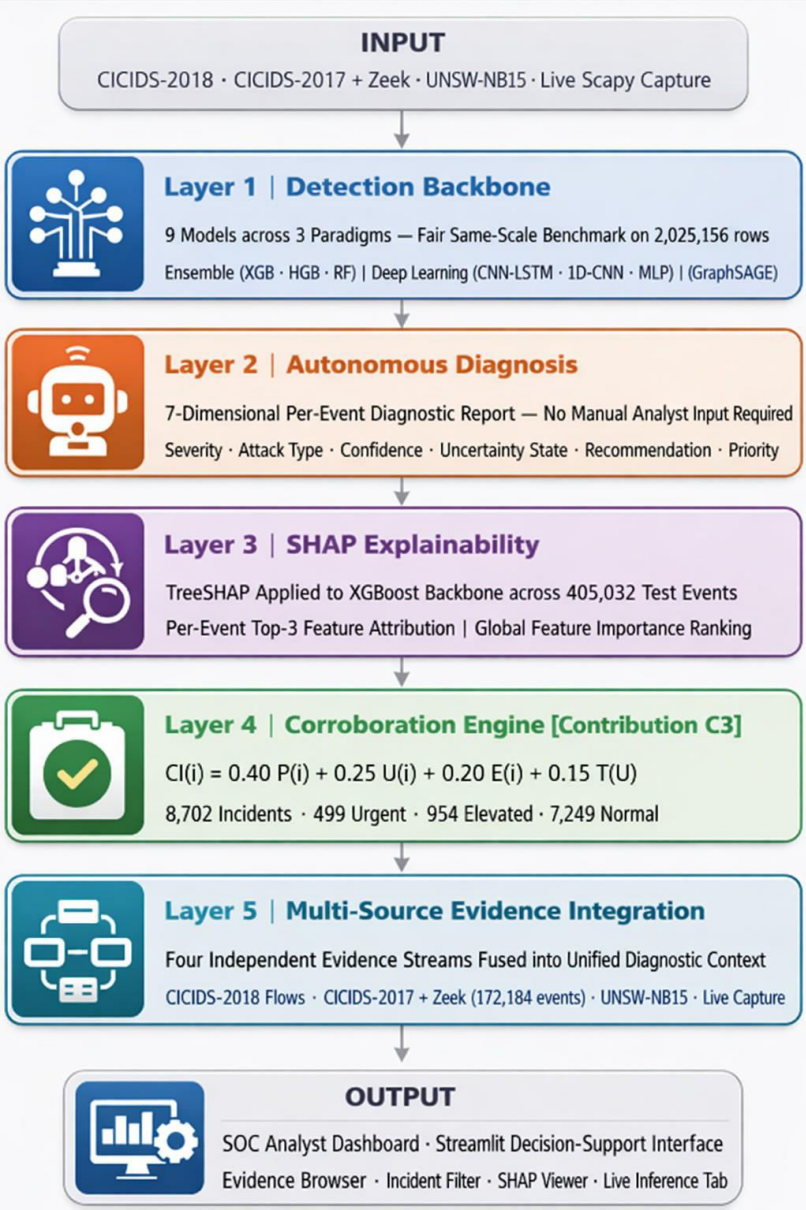


Figure 1: Five-layer autonomous breach diagnosis framework.

Data Sources and Preprocessing

The evaluation is done on four datasets, listed in Table 2. The main criterion is CICIDS-2018 [10]. The secondary benchmark and Zeek corroboration for CICIDS-2017 is given by the same benchmark [11]. UNSW-NB15 is used as an external benchmark [12]. Windowed temporal validation is done with a full sanitized CICIDS-2018 corpus.

Table 2: Datasets and Data Sources Used in the Evaluation

Dataset	Rows	Split	Role
CICIDS-2018	2,025,156	80/20	Primary benchmark
CICIDS-2017	2,522,362	80/20	Secondary / Zeek
UNSW-NB15	257,673	Official	External validation
Full CICIDS-2018	15,656,752	Windows	Temporal validation

© 2026 IJIRCCE | Volume 14, Issue 5, May 2026 DOI: 10.15680/IJIRCCE.2026.1405001

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

The preprocessing flow through a pipeline consists of five steps: duplicate removal, constant feature removal, infinite value cleaning, missing value replacement with medians (median imputation), and binary labeling. Label standardization is done for tree based methods and logistic regression/deep learning baselines. For all baselines chronological splitting was done in place of random splitting as it allows for preservation of time series data and avoids over-optimistic results from an overly favorable split.

Fusion Detection Backbone

The proposed detection model combines XGBoost and HistGradientBoosting using a logistic meta-learner [3], [4]. These two models are selected because both perform strongly on tabular network-flow features but produce different boundary behavior. Their probability outputs are combined as:

$$P_{ens}(x) = \sigma(w_1 P_{XGB}(x) + w_2 P_{HGB}(x) + b) \quad (1)$$

where P_XGB(x) and P_HGB(x) are malicious-class probabilities, w1 and w2 are learned weights, b is the bias term, and σ is the sigmoid function. A sample is classified as malicious when P_ens(x) > 0.5.

The individual backbone outputs are preserved because disagreement between them is useful for uncertainty classification. Therefore, the fusion backbone supports both detection performance and diagnostic interpretation.

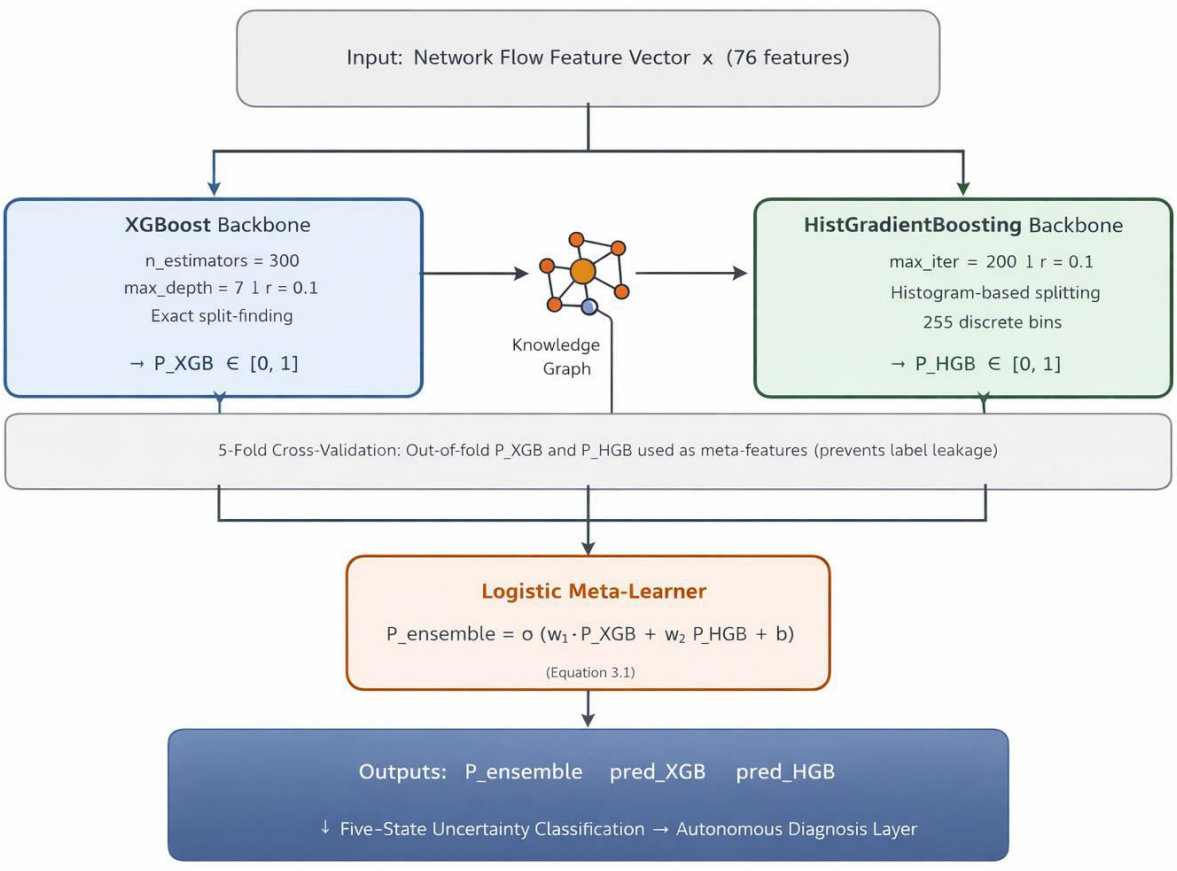


Figure 2: Detailed data-flow of the proposed fusion detection backbone.

Uncertainty, Explanation, and Corroboration

We define five states in the uncertainty mechanism: Stable Malicious, Moderate Confidence, Conflicted, Borderline, and Stable Benign, which show agreement between XGBoost and HistGradientBoosting models across five states. If there is no agreement, then that event should be treated with greater caution and should be examined in the light of supporting evidence, whereas if there is some agreement, then there is more confidence in that event.

© 2026 IJIRCCE | Volume 14, Issue 5, May 2026 DOI: 10.15680/IJIRCCE.2026.1405001

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

SHAP is used to explain the XGBoost backbone [8]. For feature f and event xi, the Shapley value is:

$$\phi_f(x_i) = \sum_{S \subseteq F \setminus \{f\}} \frac{|S|!\,(|F| - |S| - 1)!}{|F|!} [v(S \cup \{f\}) - v(S)] \quad (2)$$

Global feature importance is computed as the mean absolute SHAP value over all test events.
Incident-level corroboration is computed using:

$$C(i) = 0.40P(i) + 0.25U(i) + 0.20E(i) + 0.15T(i) \quad (3)$$

where P(i) is detection probability, U(i) is uncertainty-state score, E(i) is evidence-channel count, and T(i) is temporal coherence. Priority tiers are assigned as Urgent if $C(i) \geq 0.75$, Elevated if $0.50 \leq C(i) < 0.75$, and Normal if $C(i) < 0.50$.

Experimental Setup

All models were tested in exactly the same environment using the original CICIDS-2018 test set for the evaluation of all models. In addition to the model baselines (XGBoost, HistGradientBoosting, Random Forest, MLP, 1-Dimensional Convolutional Neural Network (CNN), CNN-LSTM, GraphSAGE, and Logistic Regression) we also used a baseline that included no machine learning at all. We have compared our system against each of these models across six different evaluation metrics including: accuracy, precision, recall, F1-score, Area Under Curve Receiver Operating Characteristic (AUC ROC) score, and run time.

The reason we selected F1-score as the first or primary detection metric is due to the nature of intrusion datasets which are typically highly imbalanced. As such, it is important to consider both false positive rates as well as false negative rates when evaluating detection performance.

To evaluate if our proposed approach improves upon prior approaches we measured four new metrics for the diagnosis layer. The new metrics evaluated how many raw events were reported by our system; how many groups of related incidents we detected; what proportion of the total number of incidents fell into each of three tiers of importance; whether or not there was corroborating evidence for each incident report; and finally, how many raw detection reports were reduced to the level of an urgent incident queued for analysis.

IV. RESULTS AND DISCUSSION

CICIDS-2018 Benchmark

Table 3 reports the primary benchmark results. The proposed model achieves the best F1-score and AUC-ROC among all evaluated models.

Table 3: Nine-Model Benchmark Results on CICIDS-2018

Model	F1	AUC	Acc.	Prec.	Rec.	s
Proposed Model	0.9892	0.9990	0.9848	0.9933	0.9851	87.5
HGB	0.9870	0.9988	0.9819	0.9971	0.9771	18.2
XGBoost	0.9854	0.9984	0.9797	0.9937	0.9774	51.3
Random Forest	0.9840	0.9975	0.9775	0.9869	0.9811	290.1
MLP	0.9740	0.9410	0.9643	0.9992	0.9500	39.6
1D-CNN	0.9737	0.9896	0.9639	0.9997	0.9490	61.1
GraphSAGE	0.9711	0.9937	0.9601	0.9923	0.9507	2934.0
CNN-LSTM	0.9704	0.9927	0.9593	0.9932	0.9486	649.7
Logistic Regression	0.9640	0.9147	0.9498	0.9735	0.9546	45.5

© 2026 IJIRCCE | Volume 14, Issue 5, May 2026 DOI: 10.15680/IJIRCCE.2026.1405001

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

The top four models are all tree-based ensembles, demonstrating that tabular features like CICFlowMeter are suitable for decision-tree based learning. Deep learning models are competitive but not surpass ensemble baselines, and GraphSAGE is significantly more time consuming.

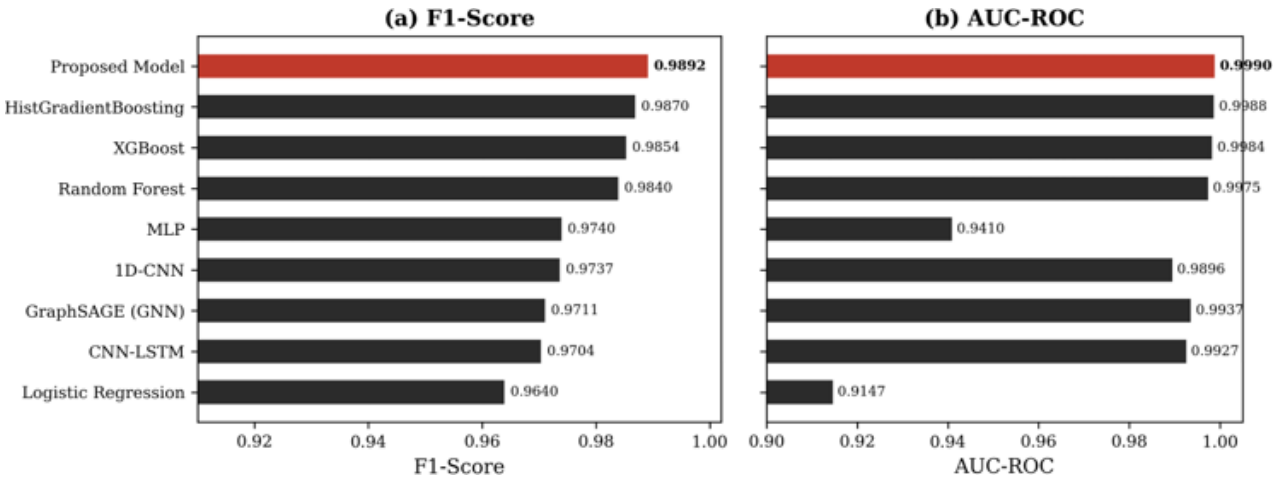


Figure 3: Model comparison on CICIDS-2018.

ROC-AUC and External Validation

The proposed model achieves AUC-ROC = 0.9990 on CICIDS-2018, confirming strong threshold-independent discrimination. Figure 4 shows the ROC comparison.

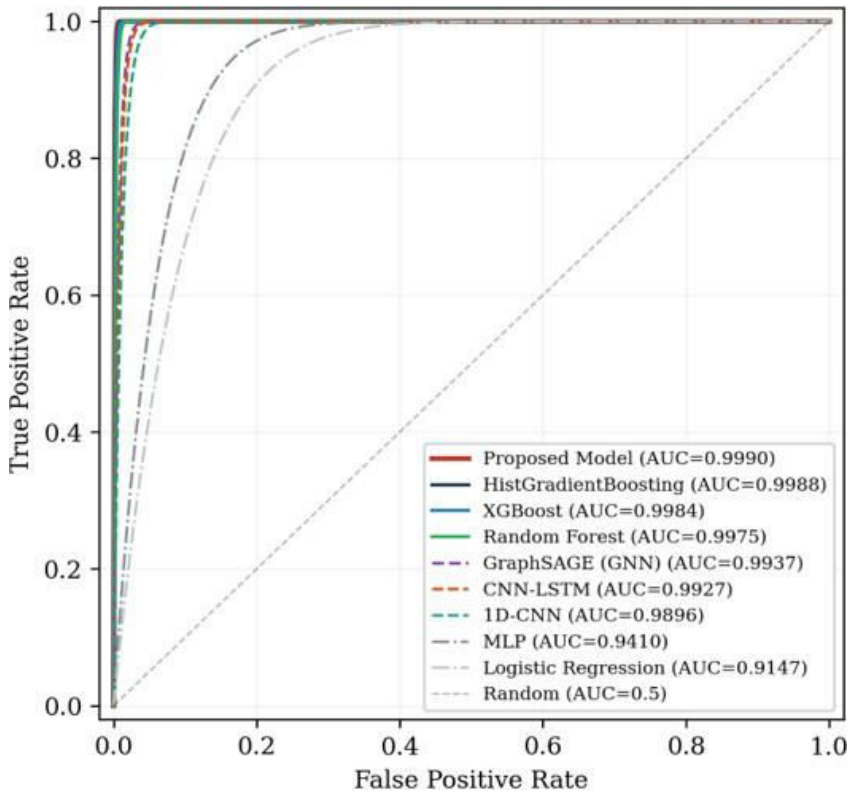


Figure 4: ROC curves for nine detection models on CICIDS-2018.

© 2026 IJIRCCE | Volume 14, Issue 5, May 2026 DOI: 10.15680/IJIRCCE.2026.1405001

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

The proposed model also achieves F1 = 0.9980 and AUC-ROC = 0.9990 on CICIDS-2017, and F1 = 0.9208 with AUC-ROC = 0.9844 on UNSW-NB15. These results support the model's usefulness as the detection base for the diagnosis framework while showing that performance varies across datasets.

Table 4: External Validation Summary

Dataset	F1	AUC	Accuracy
CICIDS-2017	0.9980	0.9990	0.9993
UNSW-NB15	0.9208	0.9844	--

Cross-Dataset and Temporal Robustness

Zero-shot transfer from CICIDS-2018 to CICIDS-2017 exposes a major robustness limitation. The proposed model drops from F1 = 0.9892 to F1 = 0.3214, while GraphSAGE retains the strongest transfer performance with F1 = 0.7149. This confirms that high in-distribution benchmark performance does not guarantee direct deployment robustness.

Table 5: Cross-Dataset Transfer from CICIDS-2018 to CICIDS-2017

Model	F1 2018	F1 2017	ΔF1
Proposed Model	0.9892	0.3214	-0.6678
GraphSAGE	0.9711	0.7149	-0.2562
XGBoost	0.9854	0.3526	-0.6328
HGB	0.9870	0.1433	-0.8437
Random Forest	0.9840	0.0214	-0.9626

Windowed temporal validation on the full sanitized CICIDS-2018 corpus also shows degradation: F1 decreases from 0.7980 to 0.6788 and then to 0.4543 across three sequential windows. This motivates the corroboration layer because raw detection probability alone becomes unreliable under temporal shift.

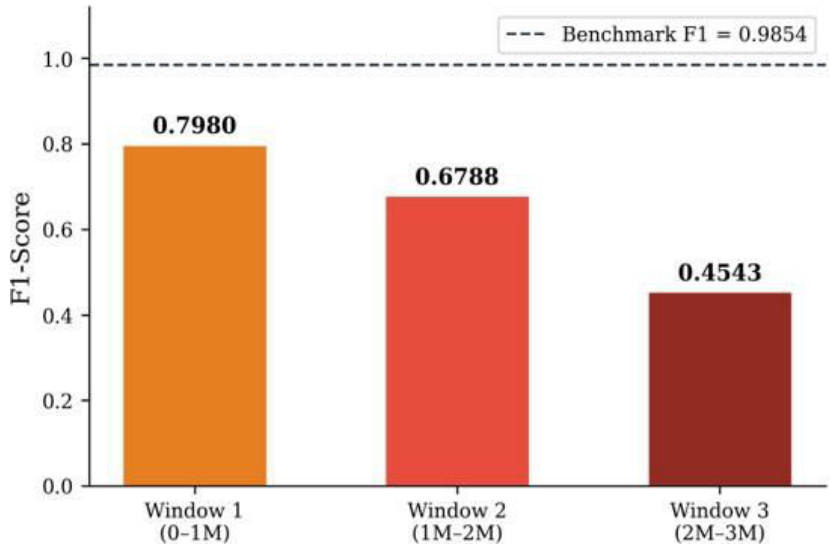


Figure 5: Windowed temporal validation on CICIDS-2018.

SHAP and Corroboration Results

SHAP analysis shows that the model is based on real TCP/IP features; for example, it looks at how big a first forward segment is, how many bytes are in the initial forward / backward windows, the length of packets sent out, what the destination port number is, how often the ECE (ECN Echo) flags have been set and the amount of time that has passed since each packet arrived. These results help to support this models' ability to be interpreted by helping the analyst understand why an event was labeled "malicious".

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

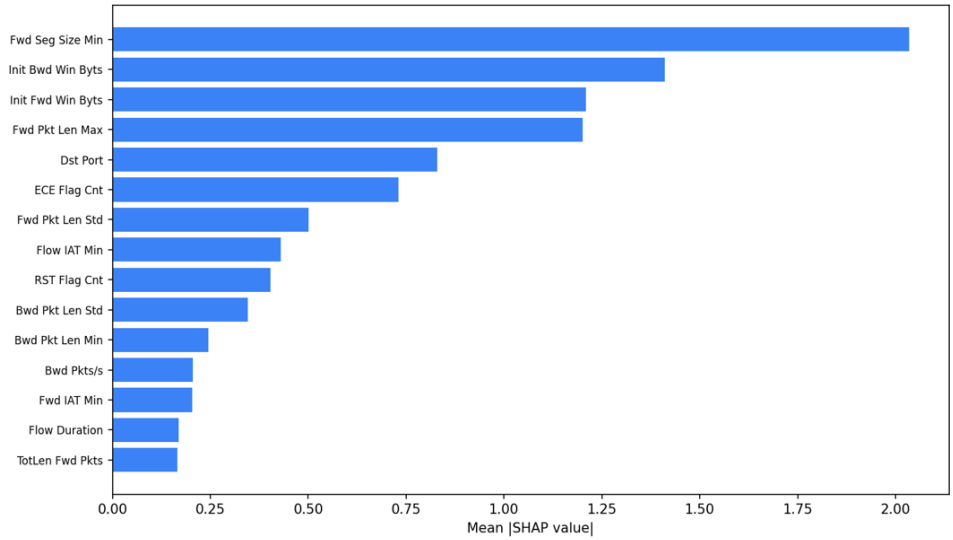


Figure 6: SHAP global feature importance for the XGBoost backbone.

Table 6: Autonomous Corroboration Engine Outputs

Priority Tier	Count	Percentage
Urgent	499	5.7%
Elevated	954	11.0%
Normal	7,249	83.3%
Strongly Corroborated	7,325	84.2%
Context-Corroborated	1,037	11.9%
Total Incidents	8,702	100%

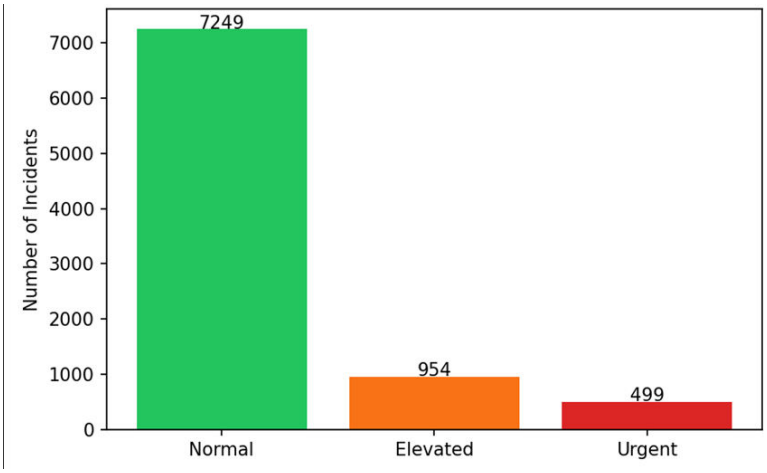


Figure 7: Priority-tier distribution of diagnosed CICIDS-2018 incidents.

Zeek integration identified 172,184 Zeek-corroborated flows within the CICIDS-2017 data set [13], [11] that correspond to 6.8% of the overall data set. The live Scapy-based Inference Module uses a subset of 15 IP addresses it has previously seen with a probability, uncertainty, severity, and type-of-attack output [14]. Because it was not feasible for the live Scapy capture to be able to replicate all of the same exact capabilities of CICFlowMeter, this live module is used as an example of how one would deploy the prototype.

© 2026 IJIRCCE | Volume 14, Issue 5, May 2026 DOI: 10.15680/IJIRCCE.2026.1405001

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

	Case_Source	id.resp_h	id.resp_p	proto	Evidence_Count	Unique_Sources	Top_Conn_State
0	Zeek DDoS Case	192.168.10.50	80	tcp	95983	1	RSTO
1	Zeek DDoS Case	192.168.10.3	53	udp	4051	1	SF
2	Zeek DDoS Case	192.168.10.50	22	tcp	982	9	SF
3	Zeek DDoS Case	192.168.10.50	137	udp	489	11	SF
4	Zeek DDoS Case	192.168.10.50	21	tcp	476	10	SF
5	Zeek DDoS Case	192.168.10.50	139	tcp	163	3	SF
6	Zeek DDoS Case	192.168.10.50	137	udp	167	3	SHR
7	Zeek DDoS Case	192.168.10.50	139	tcp	157	3	OTH
8	Zeek DDoS Case	192.168.10.50	443	tcp	241	1	REJ
9	Zeek DDoS Case	192.168.10.3	88	udp	198	1	SF

Figure 8: Multi-source corroboration and live inference summary.

Operational Value

The suggested model will generate a richer data record that contains more detailed information about an item to be analyzed, including probability, state of uncertainty, SHAP explanations, corroboration scores, tiers, and analyst recommendations, than what is produced in a typical NIDS, which is usually a very large number of potential malicious labels or probability values.

Limitations and Future Work

Cross-environment generalization to other environments is the first limitation. Although it outperforms baselines by large margins on its own training data (i.e., in-distribution), when the same environment is used for out-of-distribution (zero-shot) transfer, the model's accuracy drops significantly; therefore, in practice it would likely need to be retrained or recalibrated before being deployed into a different network. The second limitation of this method is that while it has strong temporal robustness based upon our results for windowed validation, if there are changes in traffic distributions, then we see an immediate drop-off in performance. The third limitation is the breadth of the type of data used as "evidence." In addition to using flow records, Zeek log files, the external testing data we have collected for use as a benchmark and live-packet capture as input to infer information about attacks in real time, the system currently does not support all types of evidence including, but not limited to, endpoint logs, firewall records, user/identity logs, cloud events and/or EDR alerts.

In future work we plan to utilize learned corroborative weights from analyst confirmed incident cases as well as implement techniques for adapting domains to new networks. Additionally, we will build a complete real-time version of CICFlowMeter compatible feature extraction and expand integration with production security telemetry source systems.

V. CONCLUSION

This paper has developed a novel AI-based autonomous breach diagnosis system which extends NIDS capabilities to go beyond simply classifying traffic as either "malicious" or "benign". The proposed fusion of two machine learning models, namely XGBoost and HistGradientBoosting, have produced excellent performance metrics in terms of both precision (F1) and accuracy (AUC-ROC) on all three datasets used. Specifically, the proposed framework produced a high level of precision with F1 scores of 0.9892 and 0.9980 on CICIDS-2018 and CICIDS-2017 respectively; while it demonstrated very good discrimination capability with an area under receiver operating characteristic curve score of 0.9990 on CICIDS-2018 and 0.9844 on UNSW-NB15. Importantly, the new framework includes four new features: uncertainty classification, SHAP explanation, incident grouping and weighted evidence corroboration.

www.ijircce.com | e-ISSN: 2320-9801, p-ISSN: 2320-9798| Impact Factor: 9.927| ESTD Year: 2013|

International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Ultimately, the practical application of the new framework was able to reduce the number of immediate analyst action required from 405,032 detections down to just 8,702 incidents and 499 high-priority cases thereby resulting in a reduction of the initial analyst action queue by 99.88%.

Therefore, this research is significant because it shows that operationally effective intrusion detection systems need to be capable of identifying potential attacks with some degree of certainty, explaining why they are indicating certain activity as suspicious, corroborating their findings for analysts, and providing a way to prioritize responses based upon urgency.

REFERENCES

[1] 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.
[2] Breiman, L. (2001). Random forests. Machine Learning, 45(1), 5-32.
[3] Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. Proceedings of ACM SIGKDD, 785-794.
[4] Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. Annals of Statistics, 29(5), 1189-1232.
[5] Chollet, F. (2017). Deep Learning with Python. Manning Publications.
[6] Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.
[7] Hamilton, W. L., Ying, R., & Leskovec, J. (2017). Inductive representation learning on large graphs. Advances in Neural Information Processing Systems, 1024-1034.
[8] Lundberg, S. M., & Lee, S.-I. (2017). A unified approach to interpreting model predictions. Advances in Neural Information Processing Systems, 4765-4774.
[9] Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). Why should I trust you? Explaining the predictions of any classifier. Proceedings of ACM SIGKDD, 1135-1144.
[10] Sharafaldin, I., Lashkari, A. H., & Ghorbani, A. A. (2018). Toward generating a new intrusion detection dataset and intrusion traffic characterization. Proceedings of ICISSP, 108-116.
[11] Zeek Project. (n.d.). Zeek Network Security Monitor. https://zeek.org
[12] Moustafa, N., & Slay, J. (2015). UNSW-NB15: A comprehensive data set for network intrusion detection systems. Proceedings of MilCIS, 1-6.
[13] Paxson, V. (1999). Bro: A system for detecting network intruders in real-time. Computer Networks, 31(23-24), 2435-2463.
[14] Scapy Project. (n.d.). Scapy: Packet manipulation tool. https://scapy.net

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi crossref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details