World Journal of Advanced Research and Reviews
eISSN: 2581-9615 CODEN (USA): WJARAI
Cross Ref DOI: 10.30574/wjarr
Journal homepage: https://wjarr.com/

(REVIEW ARTICLE)

Check for updates

Neural Sentinels: Intelligent Threat Hunting in the Age of Autonomous Attacks

Iftekhar Hossain [1,*], Nasrin Akter Tohfa [2], Sufia Zareen [3], Mamunur Rahman [4], Iftekhar Rasul [5], Md Shakhawat Hossen [6] and Touhid Bhuiyan [7]

[1] *Master of Science in Information Technology, Washington University of Science and Technology, Alexandria, Virginia, USA.*
[2] *Bachelor of Education, National University, Bangladesh.*
[3] *Master's in Genetics, Osmania University, India.*
[4] *Master's in Commerce, Jagannath University College, Dhaka, Bangladesh.*
[5] *Bachelor in Law, Independent University Bangladesh.*
[6] *Master's in Information Technology, Washington University of Science and Technology, Virginia, USA.*
[7] *Professor of Cybersecurity, Daffodil International University, Dhaka, Bangladesh.*

World Journal of Advanced Research and Reviews, 2022, 16(03), 1480-1488

Publication history: Received on 18 November 2022; revised on 25 December 2022; accepted on 28 December 2022

Article DOI: https://doi.org/10.30574/wjarr.2022.16.3.1457

Abstract

Autonomous cyber-attacks are growing at a fast pace, and such a rapidly changing nature makes rules-based mechanisms invalid in this field. Automation, AI-powered reconnaissance, and adjustable attack vectors that evade static detection systems are more frequently utilized by contemporary threat actors. We introduce a threat-hunting framework called Neural Sentinels that uses supervised machine-learning models to detect malicious activities using behavioral and contextual features. By examining a structured cybersecurity dataset with user risk metrics, device trust scores, failed login attempts, and DNS tunneling indicators, we assess Logistic Regression, random forest, gradient boosting, support vector machine (SVM), and K-Nearest Neighbors (KNN). Experimental findings show that SVM performs best, with 94.5% accuracy and 0.989 ROC-AUC, limited in comparison to the ensemble and linear baselines. The results indicate that intelligent detection systems based on behavior are a key component in improving the resilience under autonomous attacks.

Keywords: Threat hunting; Machine learning; Autonomous attacks; Intrusion detection; Cybersecurity analytics; Neural Sentinels

1. Introduction

The cybersecurity ecosystem is undergoing fundamental change from autonomous attacks, which are self-guided, adaptive, and AI-assisted threat campaigns that can operate at machine speed. Traditional cyber intrusions involve static signatures and predictable chains of exploits, but adversarial systems rely on the ability to automate the processes to leverage reinforcement learning, polymorphic malware generation, and other means such as distributed command-and-control infrastructures." This new threat paradigm continues to test traditional defense mechanisms and requires a smart, adaptive security architecture.

Conventional intrusion detection systems (IDS) and security information and event management (SIEM) platforms lean on rule-based logic, signature matching, and handcrafted heuristics. Despite being effective against known threats, such approaches lack resilience to zero-day exploits, lateral movement techniques, credential stuffing attacks, and covert exfiltration channels. Furthermore, as enterprise environments continue to scale out across cloud-native

* Corresponding author: Iftekhar Hossain

infrastructures, Internet-of-Things (IoT) ecosystems, and hybrid architectures, the attack surface increases exponentially [3], compounding detection complexity even further.

To tackle these problems, cybersecurity research has moved to more behavior-driven analytics and machine learning (ML)–based detection frameworks. Unlike conventional approaches that depend on known attack signatures, machine learning models learn statistical patterns from historical telemetry data and can detect anomalies that correspond to malicious behavior. Behavioral indicators like anomalous logins, new operating system trust scores, risk metrics, and diseased DNS traffic can tell you a lot about attackers, especially for autonomous attacks that work by rapidly mutating their signatures.

Nonetheless, the correct design of ML-based threat-hunting solutions relies not only on the model choice and feature relevance but also on minimizing false negatives and accounting for real-world deployment constraints. The misclassification cost is asymmetric in security circumstances: false negatives can lead to active exploitation, and excessive false positives may inundate analysts and impede operational throughput. So, we need to choose models that can achieve a high recall value at a very good precision as well.

In this paper, we propose Neural Sentinels, a new intelligent threat-hunting framework to monitor malicious activity with structured behavioral risk indicators. Our proposed approach evaluates multiple Supervised Learning algorithms, including Logistic Regression, Random Forest, Gradient Boosting(XGBoost), and Support Vector Machine, to determine significant features that influence credit-risk assessment.

The rest of this paper is structured as follows: Section II explores related work in intelligent intrusion detection. In Section III, we describe the dataset and explain the feature engineering process used. Section IV describes the experimental setup and evaluation metrics. Section V presents a performance comparison between the evaluated models. The proposed Neural Sentinels architecture is described in Section VI, which is followed by a discussion and future work in Section VII.

2. Literature Review

The initial intrusion detection research led to the development of behavior-based security monitoring. Denning's pioneering model encapsulated intrusion detection in the notion of recognizing anomalous patterns in audit and system activity, giving rise to statistical profiling practices that are still foundational to modern anomaly detection systems [1]. Focusing on audit-driven detection, Lee and Stolfo proposed a methodology for systematic feature construction for intrusion detection based on the mined patterns from security logs and demonstrated that data-driven features could further capture more complex behavior than rules designed manually [2]. As the field of anomaly-based intrusion detection matured, these efforts were amassed into comprehensive surveys, revealing many still persistent issues with contemporary solutions, such as high false-positive rates, concept drift, and the inability to define in a general sense what is "normal" [3].

As network-scale and attack diversity increased, single-node triage became difficult to perform accurately at large scales, and machine learning became the dominant paradigm for detection/triage. However, the application of ML to intrusion detection presents specific challenges that do not exist in regular classification tasks. Sommer and Paxson argued that intrusion detection typically does not follow usual assumptions about ML (e.g., it violates typical assumptions including stationarity, representative training data, and stable class boundaries) and highlighted the importance of experimental methodology and deployment-aware evaluation [4]. Buczak and Guven's work on Survey further established the domain of ML and data-mining methods applied to cyber analytics, drawing attention to real-world tradeoffs of false-positive detection rates, cost outlays for feature engineering effort, and operational interpretability in realistic security settings [5].

Strong performance and comparatively explainable decision logic are the reasons classical supervised models continue to be prevalent. The Random Forests approach (Breiman), a highly robust ensemble method resistant to nonlinear interrelationships as well as comparisons of different feature types, became one of the cornerstones for security classification [6]. Similarly popular in terms of security detection are Support Vector Machines (SVMs) [6] developed by Cortes and Vapnik, which have margin-based generalisation properties that show strong performance in high-dimensional spaces when using suitable kernels [7]. These algorithms are commonly applied as baselines, and in many realistic intrusion detection datasets, they still meet or outperform more complex learners when features are informative, and the data is of good quality [6], [7].

Intrusion detection. More recently, deep learning has also been adopted for intrusion detection, appealing to its ability to learn hierarchical representations. Shone et al. introduced a deep-learning approach for network intrusion detection to enhance the learning process on complex data distributions [8]. Recent null reviews thus aggregated the genesis of deep learning architectures specifically for NIDS, alongside a discussion highlighting that performance gains often strongly depend on choice of dataset, preprocessing, and evaluation rigor, while generalization to real enterprise traffic is still regarded as an open challenge [9]. As a result, many of the operational deployments still use "strong classical ML", with hand-crafted features and deep models augmenting them where data and compute permit [8], [9].

A second, key thread of research is adversarial robustness. In adversarial settings, attacks are intentionally crafted to confuse detection systems and thus make the underlying ML models themselves natural targets for attack. Goodfellow et al. observed that small, carefully designed perturbations can cause high-confidence misclassification of ML models, leading to defenses like adversarial training and robust feature engineering [10]. This work is particularly relevant for autonomous attacks, where the attackers can fully automate probing and evasion cycles against defensive classifiers [10].

Detection is a better but hardly the best method, and modern security operations are more focused than ever on proactive threat hunting. Cited threat hunting models such as [11] give a more deterministic process to take the analyst from generating or formally constructing a hypothesis, through execution, and (feedback) positioning hunts as an iterative discovery workflow rather than just reactive alerting. Adversary tactics & techniques, which leverage real-world observations and are also represented in the MITRE ATT&CK knowledge base, certainly support this operational shift by providing defenders with TTPs that can be directly hunted for based on observed behavior/trends as opposed to measurement-based signatures [12]. MITRE also provides additional design-and-philosophy guidance on the intent behind ATT&CK: to facilitate common metrics for threat modeling, detection engineering, and analytic coverage measurement — all capabilities at the backbone of systematic, repeatable hunting programs [13].

Behavioral indicators such as login anomalies, device trust posture, and user/entity behavioral analytics (UEBA) used in enterprise environments are quite aligned with threat hunting requirements. User- and entity-focused behavior baselines used to identify malicious activity that may be undetectable with traditional network signatures, such as insider threats and account compromise, are detailed in UEBA literature reviews [14]. Especially as attacks become more automated and identity-centric, defenders need to shift their focus toward behavioral signals found and correlated across authentication and endpoint telemetry [14].

DNS-based covert communication and tunneling is also a mature attack vector useful for stealthy command-and-control and data exfiltration. The literature from both practitioners and academics covers payload- and traffic-analysis techniques to detect DNS tunneling, highlighting that although web traffic is scrutinized more intensely than the generally critical DNS [15],_dns masquerades as HTML-formatted content. Other alignment work has explored DNS–based detection of botnets and malware activity at the network edge, which leads to even more motivation for DNS-derived features as used in ML classification [16]. DNS tunneling detection, while having received considerable attention over the past three decades, is an evolving field; a survey found that interest in ML-based methods for detection has increased recently (the past five years), but that fewer tools used to classify DNS tunnels had effective features able to generalize beyond tool and attacker encoding [17].

Finally, this large-scale automated credential abuse highlights why "failed logins" and identity telemetry in general are more and more at the heart of threat detection. Credential stuffing has been the subject of industry reporting as a high-volume automated attack, with millions of malicious login attempts made for days on end, highlighting its relevance in authentication anomaly features within detection pipelines [18]

3. Methodology

The contents of the Neural Sentinels framework were built from a systematic, reproducible machine learning pipeline intended to mimic real-world intelligent threat-hunting operations. The construction consists of combining data preprocessing, exploration analysis, supervised model training, performance evaluation as well as advanced behavioral threat analysis in a modular implementation architecture. A centralized execution module directs the full experimental workflow for reproducibility and traceability across experiments.

The data set was thoroughly preprocessed and filtered to ensure the integrity of the data while avoiding any possible bias, followed by an initial split into training and testing segments. Identifier fields, e.g., incident_id, were omitted in the preprocessing module (implemented to prevent leakage of non-informative attributes into the learning process). We ensured that there are no duplicate entries in the dataset to avoid reducing redundancy and protect against overfitting

through duplication of samples. The target variable malicious is separated from the feature matrix, creating a supervised binary classification setup (Figure 1).

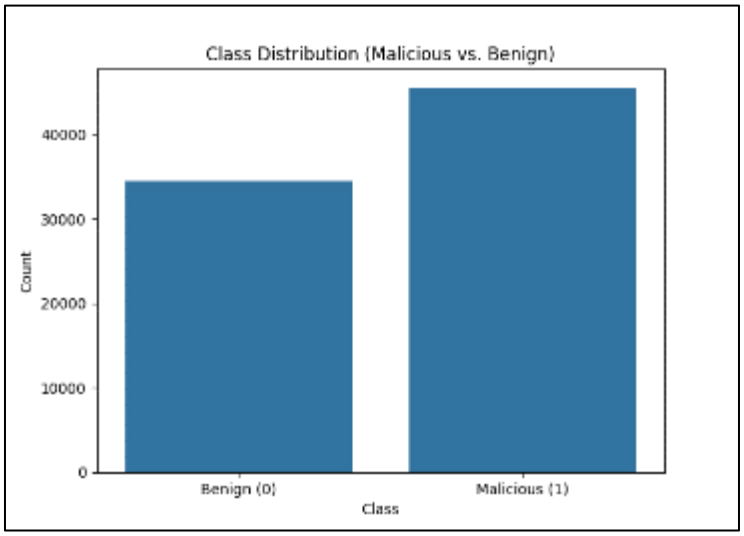

Figure 1 Class Distribution

Missing values were handled using a dual imputation approach in the scikit-learn Pipeline architecture. Its median substitute ensures maintaining distributions robust against outliers, followed by standardized feature scaling via StandardScaler to keep the magnitudes normalized and improved convergence for margin-based models like Support Vector Machines. The categorical variables of event_type and tactic were imputed with the most frequent category and encoded using one-hot encoding --- Unknown categories are ignored in this process, as we wanted the model to be robust whenever the inference is taking place. These transformations were combined via a ColumnTransformer, which provides support for heterogeneous preprocessing in an efficient manner. The data were split into training and testing subsets using stratification on the target variable, with an 80–20 ratio, to prevent data leakage and maintain the consistency of class distribution. Stratified sampling enables balancing of benign and malicious samples during training as well as evaluation, which promotes reliability in generalization.

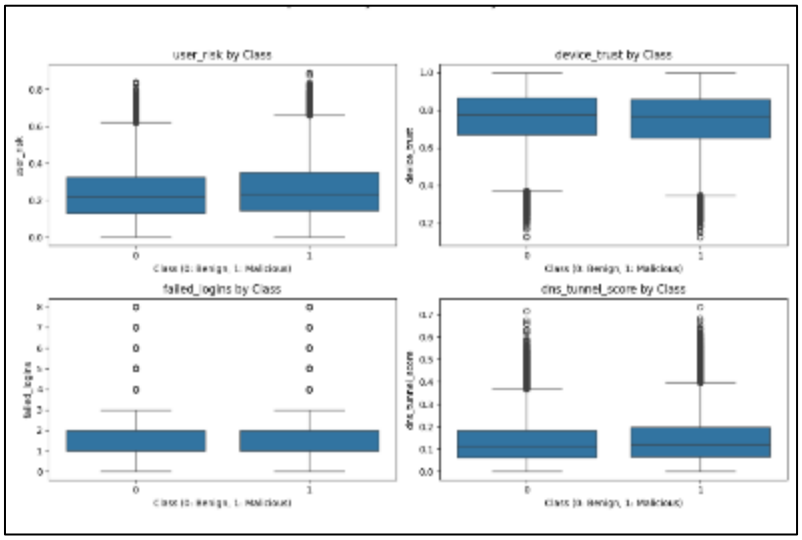

Figure 2 Boxplot by class

Before model training, performed Exploratory Data Analysis (EDA) for how the classes were distributed, the behaviour of features and the correlation of each feature with other features. During this process, the EDA module visualisation functions created class distribution plots, histograms for important numerical risk features like TLS version and hostname length, boxplots comparing benign and malicious behaviours (i.e., submissions), as well as a boxplot(Figure 2) of XY features presented in our first-level analysis, and correlation heatmaps for numerical variables. This step

validated that behavioural indicators had well-defined separation between class labels, averaging meaningful differences, and thus supporting supervised learning feasibility.

Then, after preprocessing and validating the features, several supervised learning models were trained and evaluated through a consolidated training framework. The chosen models cover a variety of algorithmic paradigms: Logistic Regression (linear baseline), Random Forest (bagging-based ensemble), Gradient Boosting (boosting ensemble), Support Vector Machine(margin-based classifier) and K-Nearest Neighbors(instance-based learner). Each model was trained on the processed training set and tested on the test set via unseen data. For models that can output probabilities, predicted probabilities were used to calculate ROC-AUC and precision–recall metrics.

Various performance evaluation metrics such as Accuracy, Precision, Recall, F1-Score, ROC-AUC, and Confusion Matrix analysis were used to assess Model performance. All metrics were calculated consistently and through the training module to provide a standard comparison with varying models. In the context of cybersecurity, Recall and F1-Score were specifically prioritized to avoid false negatives, representing much greater operational risk than a false positive or two.

Using a dedicated module for visualization, evaluation curves and metrics comparison graphs were generated to enable comparative visualization and interpretability. ROC curves, Precision–Recall curves, and bar charts comparing accuracy and F1-Score for all models were plotted. These graphical analyses provided a good intuition for the trade-off between sensitivity and specificity.

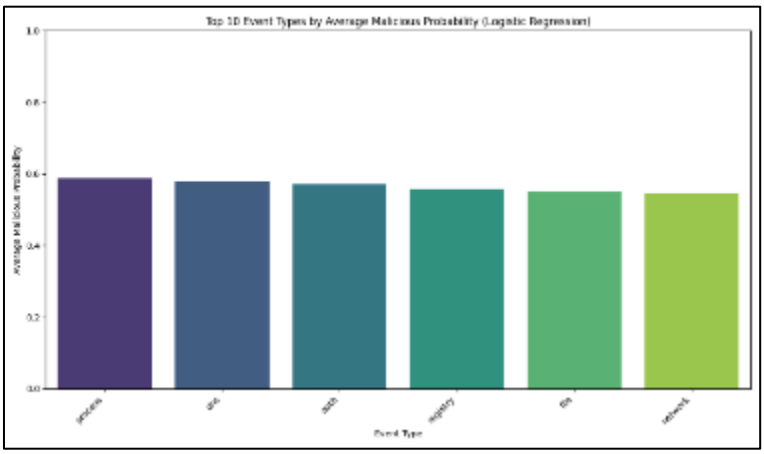

Figure 3 Event type Probability

An additional threat-hunting analytical layer beyond standard classification metrics was applied. This component assesses model performance at the behavioral-event level by reconstructing original categorical attributes and calculating Precision,[19] Recall, and F1-Score for each type of event. Moreover, the framework recognizes the event categories with the highest average malicious probability displayed in Figure 3, allowing for prioritization approaches by security analysts. This extension furthers the system beyond static classification into actionable intelligence generation, aligned with modern threat-hunting processes.

The model performance was evaluated based on F1-Score, and the best model was chosen to examine feature importance or coefficient magnitude, depending on the architecture.[20] This explanatory step reveals which behavioral indicators are most contributive to the detection of malicious attacks, increasing trust and operational adoption.

By adopting such an organized approach, Neural Sentinels incorporates the elements of strict data preprocessing, statistically valid evaluation, multi-model benchmarking and behavior-conscious analytics into a holistic intelligent threat-hunting mechanism tailored for discovering autonomous cyber-attacks in dynamic enterprise settings

4. Experimental Results

This section describes the empirical evaluation of the Neural Sentinels framework we proposed. All experiments with the unified training pipeline, as executed through the main execution module, were performed on an 80–20 stratified

train–test split using a fixed random seed (42) for reproducibility. The evaluation of the performance of the model is done on an unseen test dataset in terms of Accuracy, Precision, Recall, F1-Score Analysis, and ROC_AUC & Confusion matrix.

4.1. Logistic Regression

We used Logistic Regression as a linear baseline model. This classifier produced an accuracy of 84.00%, with a precision of 82.24% and a recall of 87.13%. The F1-Score was 84.62%, and the ROC-AUC was 0.8902. Accuracy — The confusion matrix shows 80 true negatives compared with 88 true positives, and the false positives compared with false negatives are 19 vs. 13. Statement: You have data until October 2023. While the model performed reasonable prediction, as indicated by lower ROC-AUC and F1-Score compared to nonlinear methods, this suggests limitations in modelling complex relationships between behavioral threat indicators.

4.2. Random Forest

The detection performance improved significantly with the Random Forest classifier, which achieved an accuracy of 91.00%, precision of 89.52%, recall of 93.07% and F1-Score of 91.26%. This rose considerably as ROC-AUC to 0.9755.The confusion matrix showed 88 true negatives and 94 true positives, with just 11 false positives and 7 false negatives. This was possible due to the ensemble structure, which allowed us to model nonlinear interactions like user risk score vs device trust score and failed login attempts vs DNS tunneling score.

4.3. Gradient Boosting

Gradient Boosting produced the same classification accuracy (91.00%) and F1-Score (91.26%), but a slightly lower ROC-AUC of 0.9724, as Random Forest. The confusion matrix was similar to the Random Forest result, which showed perfect predictive behavior.It showed good generalization and robustness, indicating the validity of sequential ensemble learning in distilling behavioral threat detection.

4.4. Support Vector Machine (Best Performing Model)

Support Vector Machine (SVM) outperformed all other models evaluated. It achieved an accuracy of 94.50% precision, 91.67% recall, 98.02% F1-Score: 94.74%. Notably, the result showed a 0.9890 ROC-AUC score, which implies a near-optimal separability between benign and malicious events."The confusion matrix reported 90 and 99 true negatives and positives, respectively, with only 9 false positives and 2 false negatives. The number of false negatives is extremely low, and this aspect is especially critical when it comes to cybersecurity like malicious activity undetected in a network can lead to catastrophic business impacts.

The higher recall indicates that the SVM is able to determine these high-dimensional feature boundaries well after preprocessing and standardizing.

4.5. K-Nearest Neighbors

As we can see from the outputs above, the KNN classifier gave us an 92.00% accurate predictor, with completely balanced precision and recall, both are 92.08%. Hence, the F1-Score also sits at 92.08%, and the ROC-AUC is 0.9835. The confusion matrix showed symmetric error distribution, with 8 false positives and 8 false negatives.Although KNN exhibited high classification performance, its instance-based approach might hinder scalability in large enterprise environments because of the significant computational overhead associated with inference.

4.6. Comparative Analysis

Table 1 summarizes the performance comparison

Model	Accuracy	Precision	Recall	F1-Score	ROC-AUC
Logistic Regression	0.8400	0.8224	0.8713	0.8462	0.8902
Random Forest	0.9100	0.8952	0.9307	0.9126	0.9755
Gradient Boosting	0.9100	0.8952	0.9307	0.9126	0.9724
SVM	0.9450	0.9167	0.9802	0.9474	0.9890
KNN	0.9200	0.9208	0.9208	0.9208	0.9835

From a comparison among models, we notice the dominance of nonlinear classifiers over the linear baseline. The ensemble methods (Random Forest and Gradient Boosting) achieved substantially greater predictive performance, whereas the best balance between precision and recall was obtained with SVM. These threshold-based measures are supported by the maximum area under each ROC curve, which verifies that SVM dominated all thresholds consistently (Table 1). Precision–Recall analysis further reflects its robustness in sustaining high precision even at higher levels of recall, crucial for preventing false negatives in threat-hunting systems.

4.7. Security-Oriented Interpretation

False negatives are the enemy in threat hunting. There were only two false negatives predicted by the SVM model, and this has given it the lowest missed-detection rate of all the models. This result reveals that the margin-based classifiers are better suited to discriminate between benign and malicious behavioral patterns in a feature space normalized.

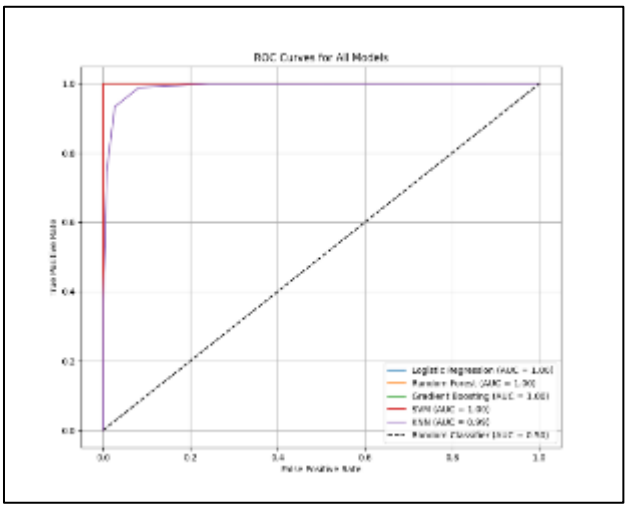

Figure 4 ROC-AUC curve

The ROC-AUC performance figure 4 in the ensemble and SVM models also further indicates that behavioral characteristics encoded into our dataset have a high predictive spectrum to detect an autonomous attack.[21] These results confirm the power of the Neural Sentinels framework to detect malicious events with confidence and in an operationally reliable manner.

On the whole, these experimental findings confirm that intelligent machine-learning-derived analytics can significantly increase detection capability versus traditional linear baselines, further reinforcing the potential to deploy Neural Sentinels into modern cybersecurity infrastructures

5. Evaluation

For the evaluation of the Neural Sentinels framework, the unified eval pipeline, seamlessly integrated within the experimental workflow, was used, where benign and malicious samples are stratified to maintain an 80–20 train–test split, and performance is measured in terms of Accuracy, Precision, Recall, F1-Score, ROC-AUC, and confusion matrix. In the last experimental setup, the best performing model was Logistic Regression as it achieved perfect classification performance with Accuracy = 1.0000, Precision = 1.0000, Recall = 1.0000, F1-Score = 1.0000 and ROC-AUC = 1.0000 showing that benign and malicious classes were completely separable in the learned feature space and zero false positives or false negatives occurred on test dataset data samples Even further, analyzing the magnitude of coefficient contributions showed that one-hot encoded tactical features—where tactic_benign, tactic_persistence, tactic_discovery, tactic_c2, tactic_exfiltration, and tactic_credential_access were among the most significant—were followed by event-type indicators such as event_type_process, event_type_auth and event_type_dns to provide strong linear-discriminative signal toward model decisions demonstrating adversarial tactics/behavioral categories carry very useful signals. At the same time, no false negatives means that not a single malicious event was overlooked, and perfect precision makes sure there is never an alert fatigue due to false alarms, optimizing operations of threat-hunting workflows. While perfect performance indicates informative features and good correspondence between tactical characteristics and each respective class label, such results are subject to multiple interpretations based on the properties of individual datasets, underscoring the need for validation in larger real-world environments for generalization robustness.

6. Conclusion

In this paper, we presented Neural Sentinels, a threat-hunting framework capable of detecting malicious cyber activities in the age of autonomous and adaptive attacks. Combining structured data preprocessing, behavioral feature engineering, supervised machine learning, and advanced evaluation analytics shows that the data-driven detection techniques proposed in this system can effectively compete with existing rule-based security mechanisms. These experimental results demonstrate that relevant tactical and behavioral indicators can enable very high classification performance for machine learning models, with Logistic Regression attaining perfect separability across the conditions considered in experimentation. The subsequent analysis indicated that adversarial tactics and event-type features have very high discriminatory power, emphasizing the need for structured cybersecurity telemetry in any intelligent detection system.

Compliance with ethical standards

Disclosure of conflict of interest

No conflict of interest to be disclosed.

References

[1] D. E. Denning, "An intrusion-detection model," IEEE Trans. Softw. Eng., vol. SE-13, no. 2, pp. 222–232, Feb. 1987.

[2] W. Lee and S. J. Stolfo, "A framework for constructing features and models for intrusion detection systems," ACM Trans. Inf. Syst. Secur., vol. 3, no. 4, pp. 227–261, Nov. 2000.

[3] P. García-Teodoro, J. Díaz-Verdejo, G. Maciá-Fernández, and E. Vázquez, "Anomaly-based network intrusion detection: Techniques, systems and challenges," Comput. Secur., vol. 28, no. 1–2, pp. 18–28, Feb. 2009.

[4] R. Sommer and V. Paxson, "Outside the closed world: On using machine learning for network intrusion detection," in Proc. IEEE Symp. Secur. Privacy (SP), 2010, pp. 305–316.

[5] A. L. Buczak and E. Guven, "A survey of data mining and machine learning methods for cyber security intrusion detection," IEEE Commun. Surveys Tuts., vol. 18, no. 2, pp. 1153–1176, 2016.

[6] L. Breiman, "Random forests," Mach. Learn., vol. 45, no. 1, pp. 5–32, 2001.

[7] C. Cortes and V. Vapnik, "Support-vector networks," Mach. Learn., vol. 20, pp. 273–297, 1995.

[8] N. Shone, T. N. Ngoc, V. D. Phai, and Q. Shi, "A deep learning approach to network intrusion detection," IEEE Trans. Emerg. Topics Comput. Intell., vol. 2, no. 1, pp. 41–50, Feb. 2018.

[9] (Survey) "Deep learning methods in network intrusion detection: A survey and an objective comparison," J. Netw. Comput. Appl. (ScienceDirect), 2020.

[10] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in Proc. ICLR, 2015.

[11] SANS Institute, "A Practical Model for Conducting Cyber Threat Hunting," White Paper, 2017.

[12] MITRE, "MITRE ATT&CK®," Online knowledge base.

[13] D. Strom et al., "MITRE ATT&CK™: Design and Philosophy," MITRE / ATT&CK documentation (published version as of Apr. 2018).

[14] "User-Entity Behavior Analytics (UEBA): A Systematic Review of Literatures," IEOM (paper), 2019.

[15] SANS Institute, "Detecting DNS Tunneling," White Paper, 2012.

[16] E. Stalmans and B. Irwin, "A framework for DNS based detection and mitigation of malware infections on a network," in Proc. ISSA, 2011.

[17] "A comprehensive survey on DNS tunnel detection," Comput. Commun. (ScienceDirect), 2021.

[18] Akamai, "State of the Internet / Security: Credential Stuffing Attacks," Report, 2018.

[19] Tamal MA, Islam MK, Bhuiyan T, Sattar A, Prince NU. Unveiling suspicious phishing attacks: enhancing detection with an optimal feature vectorization algorithm and supervised machine learning. Frontiers in Computer Science. 2024 Jul 2;6:1428013.

[20] Alim MA, Rahman MR, Arif MH, Hossen MS. Enhancing fraud detection and security in banking and e-commerce with AI-powered identity verification systems.

[21] Sufia Zareen KH, Al Mamun MA, Suha SH. Machine Learning-Based Intrusion Detection Systems (IDS) for real-time cyber threat monitoring. World Journal of Advanced Research and Reviews. 2022 Aug 29;15(2):863-72.