ISSN: 2581-3404 (Online) **IF: 5.86 (SJIF)**

IJIRTM, Volume-10, Issue-3, June-2026

Smart Network Security System: A Review

[1]Anmol Bhalla, [2]Manish Kumar, [3]Neha Tuteja

[1,2]Student, [3]Assitant Professor

[1,2,3] Department of Computer Science & Engineering (Cyber Security)

[1,2,3] Panipat Institute of Engineering and Technology, Samalkha, Panipat, Haryana, India.

[3]neha.cse@piet.co.in

Abstract. *Digital operations require cybersecurity as their fundamental base because cyber-attacks have developed into more sophisticated forms while network systems become more complex and new advanced malware threats keep appearing. The current security threats create problems for Traditional Intrusion Detection Systems (IDS) because signature-based solutions fail to identify the new threat types that keep emerging. Machine Learning (ML) presents a promising alternative which provides adaptable solutions and strong precision rates and detection capabilities for unknown attack types. The research paper examines network security through ML-based approaches while creating a Smart Network Security System which performs real-time intrusion detection. The system operates by using supervised ML models, which include Random Forest, Support Vector Machines, and Gradient Boosting that learn from the UNSW-NB15 dataset to identify network traffic patterns. The paper analyses past research work by identifying datasets for IDS research and evaluates machine learning algorithms through performance metrics and real-time implementation assessment. The research study identifies various areas that require further investigation in its findings. This will help improve network security in real-time systems.*

Keywords: Intrusion Detection System (IDS), Machine Learning, Network Security, Real-Time Detection, UNSW-NB15, Ensemble Learning.

Introduction

The fast growth of internet-connected systems and cloud infrastructures and distributed applications have made modern networks more complicated to manage. Digital operations face sophisticated cyber threats which need immediate development of modern security systems to protect essential digital assets. Traditional security systems including signature-based Intrusion Detection Systems need existing attack signatures to identify dangerous network operations. The systems protect against documented threats but they fail to recognize new or modified attack methods which makes them vulnerable to zero-day exploits and polymorphic attacks and new malware threats. Network-Based cyberattacks have experienced two major developments during the last ten years because their occurrence rate has surged while their attack methods have grown more sophisticated.

The research field focuses on data-driven and adaptive detection models which use machine learning to solve these challenges. Machine learning techniques can identify complex network traffic patterns to improve the detection of malicious behaviour through better classification results.

The UNSW-NB15 and CICIDS 2017 and NSL-KDD datasets have shown improved detection rates through Random Forest and Gradient Boosting algorithms and Convolutional Neural Networks and Long Short-Term Memory networks deep learning models.

Related Work

The Intrusion Detection Systems (IDS) has undergone significant improvement since the first design as signature-based systems that now has advanced machine learning and deep learning features. The first IDS systems worked by referring to signature matching, to compare incoming packets with a database containing known harmful patterns. Snort and Suricata are signature based detection tools that detect threats by matching their signature to the available threat databases [1]. The systems should also be updated on the signature on a regular basis to operate adequately, and they are not capable of identifying new or concealed type of attack that necessitates them to have been updated with the signature [2]. To address these issues, the developers developed the anomaly-based IDS system that is used to identify network activities that do not follow the normal baseline behaviour. The systems identify abnormal behaviour by use of statistical thresholds and behavioural profiling systems that determine abnormal activities [3]. The key issue of using anomaly-based models to detect unknown attacks is that they have low sensitivity to false alarm since they do not comprehend the normal traffic behavior adequately [4]. This weakness of the conventional intrusion detection systems (IDS) compelled researchers to use machine learning (ML) algorithms in their investigations.

The adoption of machine learning systems is a promising prospect of network intrusion detection as established through scientific literature. Majority of the well-known algorithms Support Vector Machines (SVM) and Logistic Regression and k-Nearest Neighbors (KNN) and Decision Trees (DT) and Naive Bayes are tested on the benchmark datasets KDD 99 and NSL-KDD and UNSW-NB15. The classification techniques are producing acceptable outcomes although they are poor with the complex nature of the modern network transmission incorporating various features and intricate nonlinear patterns. SVM algorithm demonstrates great effectiveness in binary classification but its effectiveness declines when handling large volumes of data [6]. Logistic Regression is also a good option but it does not work well when representing complex attack patterns.

Scholars are now investigating ensemble learning algorithms, which include Random Forest (RF) and Gradient Boosting (GB) due to the models, which depict better detection rates and system stability. The high results of the Random Forest algorithm in processing the existing data sets are due to the generation of multiple decision trees that avoid data overfitting and address the data anomaly [8]. The results of research indicate that RF is more effective in dealing with unbalanced sets of attacks when compared to standard classifiers [9]. Gradient Boosting Gradient Boosting algorithms LightGBM and XGBoost obtain superior classification outcomes along with additional hyperparameter optimization expenses and higher computational costs [10]. The trend in the machine learning has inspired the use of deep learning (DL). As stated in [11] and [12], the identification of more sophisticated patterns of attacks through the aid of Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) and Long short-term

memory (LSTM) networks has become possible. The hybrid types of DL with classical ML have shown better results when delivered using example of CNN-RF and auto-encoders-SVM system [13]. The problems faced by organizations in applying the use of DL models in real time detection include the fact that the models are computationally power consuming and that they are at the state of the art levels of accuracy. The main problem with the existing research is that it is based on the KDD 99 and NSL-KDD data sets that do not reflect the existing network traffic and current cyber threats [14]. The following are tackled in the UNSW-NB15 and the CICID 2017 datasets which provide real-life attacks and different types of network traffic patterns [6],[15]. Model testing on offline data remains the primary issue of most of the research projects who fail to show their actual behavior under the real-life situations.

The research does not address the real contents of the system that is related to the utilization of the packet-capture tools and live traffic preprocessing modules and security operator graphical interfaces. The models are theoretical because they fail to apply in the context of the practice of operational networks in the real world [16]. The Intrusion Detection Systems (IDS) have been greatly developed since their initial development as signature based systems that now have advanced machine learning and deep learning capabilities.

First IDS systems have been based on signature matching so that they check incoming packets against a database of learned harmful patterns. Snort and Suricata are signature based detection tools which detect threats according to commonality with available threat database of their signature [1]. The systems have to be updated on the signatures frequently in order to operate and fail to detect new or concealed forms of attack that make them inappropriate in changing threat landscapes [2]. The developers came up with the system of anomaly based IDS to solve these problems by detecting the network behaviours which do not fall in the normal baseline behaviour. The systems detect the abnormal behaviour with the utilisation of statistical thresholds as well as the behavioural profiling systems that detect suspicious behaviours [3]. The main problem of the anomaly-based detection models of unknown attacks would occur as a result of the tendency to create high volumes of false alarms, as the models fail to learn the normal traffic patterns satisfactorily [4]. The limitation of the traditional intrusion detection system (IDS) forced researchers to continue investigating machine learning (ML) algorithms in their research. The introduction of machine learning systems is one of the bright futures of network intrusion detection since it was supported by scientific studies. Traditional algorithms Support Vector Machines (SVM) and Logistic Regression and k-Nearest Neighbors (KNN) and Decision Trees (DT) and Naive Bayes are tested on the benchmark datasets that include KDD49 and NSL-KDD and UNSW-NB15.

The classification methods are effective even though they are not able to address the complications of the modern network transmission which encompasses a number of characteristics and nonlinear shapes. SVM algorithm works best on binary classification tasks and its performance is deteriorated when dealing with huge volumes of data [6]. The use of Logistic Regression is an alternative that fails to comply with the modeling of intricate patterns of assaults.

The fact that the current research direction examines ensemble learning methods that incorporate the Random Forest (RF) and Gradient Boosting (GB) models is dictated by the fact that the models have been established to work better in the detection and stability within the system. The current data sets are favourable to the performance of the Random Forest algorithm because it generates numerous decision trees to prevent overfitting of the model and incorporates irregularities of the data [8]. Those studies have indicated that RF is superior to conventional classifiers in terms of dealing with skewed classes of attacks [9]. LightGBM and XGBoost algorithms are gradient Boosting algorithms, which are more optimally suited to classification, as well as optimization of hyperparameters and more costly to compute [10]. The improved machine learning has made deep learning (DL) more popular. According to [11] and [12], the identification of complex attack patterns via automatic extraction of raw traffic feature with Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) and Long Short-Term Memory (LSTM) networks has been made possible. Hybrid models that combine DL with classical ML have demonstrated improved results and some examples such as CNN-RF and Autoencoder-SVM have been used [13]. The difficulties associated with the real-time deployment of the DL models in organizations are that they need extensive computational power and the systems operate at the state-of-the-art accuracy levels.

The current literature has a significant issue since it is reliant on KDD 99 and NSL- KDD data sets that do not reflect current network behavior and current cyber threat [14]. These issues are addressed by the UNSW-NB15 and CICIDS 2017 datasets, which offer real-life scenarios of attacks and various categories of network traffic patterns [6],[15]. The testing of the models with offline datasets has been the most important of all the research works that do not demonstrate the performance of the models in the real life scenarios. The study fails to provide real system elements that are the packet-capture tools and live traffic preprocessing modules and security operator graphical interfaces. The models remain theoretical since they cannot be implemented in real world in operational networks [16].

System Design and Workflow

The whole process of constructing and evaluating the Smart Network Security System is explained in the following section. The methodology provides a single building block of architecture that integrates the datasets retrieval and preprocessing as well as feature engineering and model training and real-time packet acquisition and system implementation. The system design adheres to the traditional IDS development practices when it integrates machine learning intelligence to improve the accuracy and flexibility of threat detection [6], [11]. The proposed system works based on three key elements which are the main components of the system. Offline model development requires cleaning and transformation of the dataset prior to the commencement of the machine learning model training.
Real-Time Detection Engine: On-the-fly feature extraction, live packet sniffing, and model inference. The Graphical Monitoring Interface has a lightweight GUI, which displays intrusion alerts and traffic logs.

The entire process flow will be depicted as illustrated below.
A. Importing Dataset
UNSW-NB15 dataset was selected. There are 9 attack families in the dataset that encompass Reconnaissance and Fuzzers and Shellcode and Exploits and Worms and Generic attacks to offer all-

inclusive description of the contemporary cyber danger [6]. The UNSW-NB15 data set gives the real network data of the IXIA PerfectStorm tool. It is more appropriate to train machine learning classifiers to be used in reality deployment settings compared to the older KDD99 or NSL-KDD datasets. B.Data Preprocessing: The dataset analysis has shown that significant gaps needed to be fixed and the dataset was to be corrected significantly before it was possible to use it as the training data to the model. These issues were visible in the dataset as it included missing data, as well as duplicated records and multiple issues of data formats mismatch and unstructured random values. The machine learning preliminary phase involves cleaning the data and a series of preprocessing procedures to get ready to utilize the data.

1) Data Cleaning

The invalid values and duplications of flows and unused features were filtered out in order to remove the noise and inconsistencies in the data. The missing fields were handled using the median or mode imputation process that relied on the nature of attribute.

2) Categorical Encoding

The network attributes like protocol type, service and state are categorical and were converted using:

- One-Hot Encoding of low cardinality categories.
- Label Encoding of categories of high cardinality.

3) Feature Scaling

Since there is a difference in the magnitude of network attributes, the feature space was normalized using Min-Max scaling and Standardization.

4) Feature Reduction

Redundancy of features was removed using correlation analysis and Recursive Feature Elimination (RFE). Final selection was also guided by random forest importance of features, which led to a parsimonious informative subset of features.

B. Development of machine learning model:

The 3 classification algorithms tested were: Random Forest, Gradient Boosting and Support Vector machine (SVM). The models have been chosen because they have been used successfully in IDS applications [8], [9], [13].

1) Random Forest Classifier

The random forest produces several decision trees that operate independently and then combine their results to come up with one result. This method is highly resistant to noise and has high-speed processing enormous amounts of data and therefore makes it a suitable selection to real-time IDS [8].

2) Gradient Boosting Classifier. Gradient

Boosting processes also build decision trees sequentially, to correct the mistakes caused by the former trees. The system is very precise in its prediction but it takes long time to train and thus the system is not able to give results on-demand.

3) Support Vector Machine

SVM performs well under specific binary classification tasks, but is weak when used with the large and many-class UNSW-NB15. The cost involved in the training to utilize the approach makes it infeasible to identify intrusion at the system wide level.

4) Model Evaluation Strategy

The model training phase was divided into 8020 data split and 5 fold cross validation used in the model validation. The criteria of performance evaluation were based on five broad measures such as accuracy and precision and recall and F1-score and false-positive rate (FPR).

C. Live Intrusion Detection Engine:
The system incorporates a real-time detection engine to convert the offline model into an operational IDS with the use of Scapy. The engine performs:

- Packet Sniffing -captures real time packets.
- Feature Extraction - transforms raw packets to feature vectors.
- Model Inference - uses the trained ML model to classify traffic.
- Generation of alerts - signifies an alert to the GUI on malicious flows.

This provides the ability to identify a threat immediately, satisfying the needs of the operation as it is not found in most academic IDS research [12].

Comparative Analysis

The Smart Network Security System was tested in terms of performance in two different ways that involved offline testing on the UNSW-NB15 dataset and real-time monitoring of traffic tests. The section gives the accuracy of classification results of machine learning models and displays both comparative results and confusion elements analysis and system functionality during operation in an active network. The discussion indicates the advantages of the proposed system besides comparing its performance with other known IDS solutions that have been reported in [8], [13], and [15].

A. Training models assessment:
To ensure that the results were reliable, the dataset was broken down into 80 percent train and 20 percent test, and 5-fold cross-validation. Random Forest, Gradient Boosting, and Support Vector machine learning classifiers were tested against their accuracy, precision, recall, F1-score, and false-positive rate.

Model	Accuracy	Precision	Recall	F1-Score	FPR
Random Forest	95%	94%	93%	94%	Low
Gradient Boosting	94%	93%	92%	93%	Moderate
SVM	92%	91%	90%	91%	Higher

The results indicate that Random Forest produced the highest results that are in line with past studies that indicate that ensemble learning algorithms are more effective in intrusion detection systems [9], [12]. The

SVM model performed decently on small but crashed to process large size and complicated data patterns of UNSW-NB15.

B. Confusion Matrix Analysis:
The Random Forest classifier proves to have good predictive performance based on its confusion matrix that indicates a high predictive accuracy and low false negative rates.
The system was able to identify all the key categories of attacks that include Exploits and Reconnaissance and Generic attacks with the highest confidence levels. The misclassification choices were largely due to categories that are quite similar in terms of statistical features.

C. Real-Time Detection Performance:
The system analysis was carried out by real time network testing that utilized Scapy software in capturing actual network packets. The system was used to extract flow-level and packet-level features in real-time operation to use the trained models to classify them.
Measures of observed performance:
- Lateness of detection: 25-40 ms packet-1 however, update rate of the GUI: close to real-time.
- Live alert generation: immediate on all malicious flows.
- Usage of the CPU: moderate, has been maintained at that level with constant monitoring.

The system was continuously highly responsive when dealing with the multiple bursts of incoming traffic. The literature
[10] and [16] reveals that the ensemble ML approaches have viable solutions to the operational IDS deployments that are deployed in real-time settings. The results of the proposed system are comparable to the prior research and have higher performance in some areas. Key observations include:

The Random Forest model performs higher than all the base ML models that demonstrates its suitability to be used in IDS application.
The results of the evaluation of Gradient Boosting are that the model has had a good result but at a faster rate the speed of inference has reduced.

SVM does not cope well with large-scale data sets that contain numerous features that make it unsuitable to the present day IDS systems [7]. The use of real-time integration is the distinguishing feature between this work and most works that have been done before, where models are only tested offline [12], [15].
The Smart Network Security System is capable of precise and reliable intrusion detection owing to its tests in offline and live operation that makes it appropriate in being implemented by small and medium scale.
The graphical user interface (GUI) remains unchanged; however, I have noticed it might require enhancement due to its not being user-friendly.<|human|>D. Graphical User Interface (GUI): The graphical user interface (GUI) has not been modified, yet I have observed that this may need to be improved because it is not user-friendly.
The simple graphical interface was created with Tkinter/PyQt and to show:
- Live network traffic.

- Findings (normal or malicious).
- Time-stamped alerts.
- Model confidence levels.

This renders the system useful and practical to the administrators as compared to CLI-based research prototypes [15].

D. System Deployment Environment:

The system was executed in a medium-range computer:

- Intel Core i5/i7 Processor.
- 8–16 GB RAM.
- Python 3.x environment.
- Scikit-learn, Pandas, NumPy, Scapy.
- Windows/ Linux operating system.

This proves that the system does not require specialized hardware to be used successfully.

Conclusion and Research Directions

The conventional security systems are limited in the fact that the cyber-attack has evolved to become more complex thus necessitating the use of adaptive intelligent Intrusion Detection Systems. By using machine learning, the research team has come up with a Smart Network Security System that addresses the shortcuts of signature-based and traditional anomaly-based IDS systems.

UNSW-NB15 dataset assisted the system to be acquainted with existing contemporary attack patterns that led to enhanced behaviour analysis and intrusion detecting findings. It is demonstrated in the experimental results that Random Forest is found to be more accurate and has lower false-positive percentages and greater generalization systems compared to Support Vector Machine and Gradient Boosting classifiers. The observed research results can be connected with those already of the past that have shown that ensemble learning techniques provide superior intrusion detection results [8], [12]. The results of the research prove that the appropriate preprocessing and feature selection and algorithm optimization results in a higher intrusion detection performance of machine learning. The Smart Network Security System created during this research provides a viable, scalable, and flexible framework which fits the current needs of cybersecurity, as it fits a number of gaps in the previous IDS studies. Critical analysis of the identified studies included in this paper has identified the following research directions: Deep Learning Models can also be integrated to create hybrid learning frameworks, thereby enabling deep learners to enhance their writing skills

1) Deep Learning Models Integration Deep learning models can be used to create hybrid learning systems so that deep learners can improve their writing abilities. The existing models use the classical machine learning algorithms. Future models could use deep learning models like Convolutional Neural Networks (CNN), LSTMs, Autoencoders, or Transformer-based models to learn intricate spatial and temporal attack patterns [11].

2) Learning Online and Incremental Learning.

The current system is based on the offline-trained model which is static. The adoption of online learning methods would enable the IDS to meet the changing attack vectors through constant updating of the IDS depending on the current traffic dynamics.

3) Cloud and Large Scale Deployment.
Implementation of the IDS on the cloud systems like AWS or Azure.

References

[1] C. Kruegel and G. Vigna, "Anomaly detection of web-based attacks," ACM CCS, pp. 251–261, 2003.
[2] H. Debar, M. Dacier, and A. Wespi, "A revised taxonomy for intrusion-detection systems," Annales des Télécommunications, vol. 55, no. 7, pp. 361–378, 2000.
[3] D. Barbara et al., "ADAM: Detecting intrusions by data mining," in Proc.
IEEE Workshop on Information Assurance and Security, 2001.
[4] A. Lazarevic, L. Ertoz, V. Kumar, A. Ozgur and J. Srivastava, "A comparative study of anomaly detection schemes in network intrusion detection," in Proc. SIAM Int. Conf. Data Mining, 2003.
W. Lee and S. Stolfo, "A framework for constructing features and models for intrusion detection systems," ACM Trans. Information and System Security, vol. 3, no. 4, pp. 227–261, 2000.
[5] N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems," in Proc. MilCIS, 2015, pp. 1–6.
[6] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 dataset," in Proc. IEEE CISDA, pp. 1–6, 2009. L. Breiman, "Random forests," Machine Learning, vol. 45, no. 1,
pp. 5–32, 2001.
[7] J. Han, M. Kamber, and J. Pei, Data Mining: Concepts and Techniques, 3rd ed. Morgan Kaufmann, 2011.
[8] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in Proc. 22nd ACM SIGKDD, pp. 785–794, 2016.
[9] A. Javaid, Q. Niyaz, W. Sun, and M. Alam, "A deep learning approach for network intrusion detection," in Proc. 9th EAI BIONETICS, 2016.
[10] N. Shone, T. N. Ngoc, V. D. Phai, and Q. Shi, "A deep learning approach to network intrusion detection," IEEE Trans. Emerging Topics in Computational Intelligence, vol. 2, no. 1, pp. 41–50, 2018.
[11] Z. Ding, B. Guo, and S. Liu, "Intrusion detection based on deep feature extraction and random forest," IEEE Access, vol. 7, pp. 155842–155859, 2019.
[12] I. Sharafaldin, A. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization," in Proc. ICISSP, 2018.
[13] M. Ahmed, A. N. Mahmood, and J. Hu, "A survey of network anomaly detection techniques," Journal of Network and Computer Applications, vol. 60, pp. 19–31, 2016.
[14] S. Garcia, M. Grill, J. Strobl, and A. Zunino, "An empirical comparison of botnet detection methods," Computers & Security, vol. 45, pp. 100–123, 2014.
[15] R Dubey, "An empirical study of intrusion detection system using feature reduction based on evolutionary algorithms and swarm intelligence methods", International Journal of Applied Engineering Research, 2017. pp. 8884-8889.
[16] D Rathore, A Jain, "Design Hybrid method for intrusion detection using Ensemble cluster classification and SOM network", International Journal of Advanced Computer Research, 2012.