International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
***Article Received:** 26 August 2023 **Revised:** 28 September 2023 **Accepted:** 10 October 2023*

Cyber security Reinforcement through Firewall Log Analysis and Machine Learning

[1]Afrah Fathima, [2]G. Shree Devi, [3]Dr. Zameer Gulzar
[1]Dept. Of Computer Applications, BSAR Crescent Institute of Science and Technology, Chennai, India
Dept. Of CS&IT, MANUU Hyderabad, India
af.fathima1@gmail.com
[2]Dept. Of Computer Applications, BSAR Crescent Institute of Science and Technology, Chennai, India
Shreedevi@crescent.education
[3]Department of CS&AI, S R University, Warangal, Telagana
Zamir045@gmail.com

Abstract— Firewalls play a crucial role as a primary protective measure in safeguarding network security, effectively mitigating risks posed by external vulnerabilities and internal security breaches. This study presents a new framework that utilizes firewall log data to classify incoming data packets as either permitted or forbidden. The dataset utilized in this research is obtained from Department of CS&IT, MANU University and is subjected to a thorough data pre-processing procedure. This procedure includes several tasks such as managing missing values, encoding categorical variables, standardizing numerical attributes, and guaranteeing data coherence. In order to mitigate the issue of class imbalance within the target variable, we utilize a range of machine learning models and assess their efficacy through the examination of fundamental metrics such as accuracy, precision, recall, and F1-score. The results of our study demonstrate that the AdaBoost model has superior performance compared to other models, achieving a remarkable accuracy rate of 99.00%. This study demonstrates the application of machine learning methods to automatically identify the activities indicated in firewall logs, thereby improving the security of corporate networks. Through the implementation of automation, we facilitate a more dependable and efficient method of detecting and addressing possible risks, thereby strengthening network security measures and protecting valuable corporate information.

Keywords- Network Security, Firewall log Analysis, Machine Learning, Cyber security

I. Introduction

Internet cyber threats and cyberattacks pose ongoing and dynamic difficulties in our digitally integrated global environment. The aforementioned dangers comprise a diverse array of malevolent actions, including but not limited to data breaches, ransomware attacks, phishing attempts, denial-of-service (DDoS) attacks, and various others. Cybercriminals engage in the exploitation of weaknesses present in computer systems, networks, and human actions with the intention of unlawfully acquiring confidential data, causing disturbances to services, and undermining the reliability of digital infrastructure. The continuous progression of technology necessitates the adaptation of tactics and techniques utilized by malicious actors, underscoring the imperative for comprehensive cybersecurity measures, preemptive identification of threats, and swift reaction to incidents in order to protect individuals, companies, and nations from the constant risk posed by cyberattacks.Consequently, safeguarding data integrity and usability has become imperative [1].Often, attackers possess insights into an organization's defense mechanisms, enabling them to evade detection.

Figure 1. describes the various of Cyber attacks.To counter such attacks, continuous analysis of network traffic records is essential to create profiles that inform firewall rules, allowing apt responses to incoming packets. However, these rules are in constant flux, adapting to evolving attack methods, tool advancements, and attack intricacies [2]. This dynamic rule evolution poses a challenge, as rules are manually defined by system administrators or organizational engineers. Firewalls are network security mechanisms, either in the form of hardware devices or software applications, which are specifically developed to oversee, filter, and regulate the flow of network traffic entering and exiting a system, adhering to a predefined set of security regulations. The fundamental function of firewalls is to serve as a protective measure, establishing a snag between trustworthy internal networks.

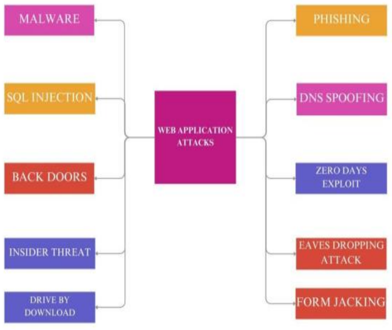

Figure 1. Different Types of Cyber Attacks

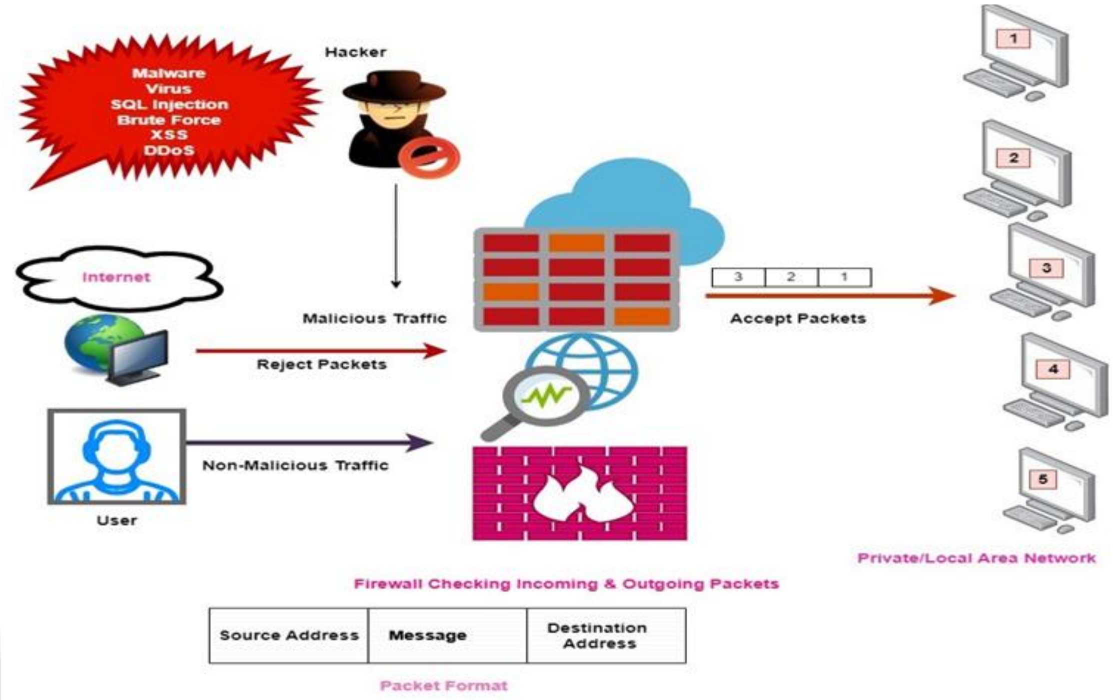

Figure 2. Firewall for a Local Area Network

The Figure 2 describes the scenario of a Firewall of a Local Area Network in which a Hacker is trying to insert Malicious traffic and other viruses in the network.

And the firewall is trying to check the incoming and outgoing traffic accepting the Non-malicious traffic and rejecting Malicious traffic. And the normal traffic is being forwarded to the Private Network.

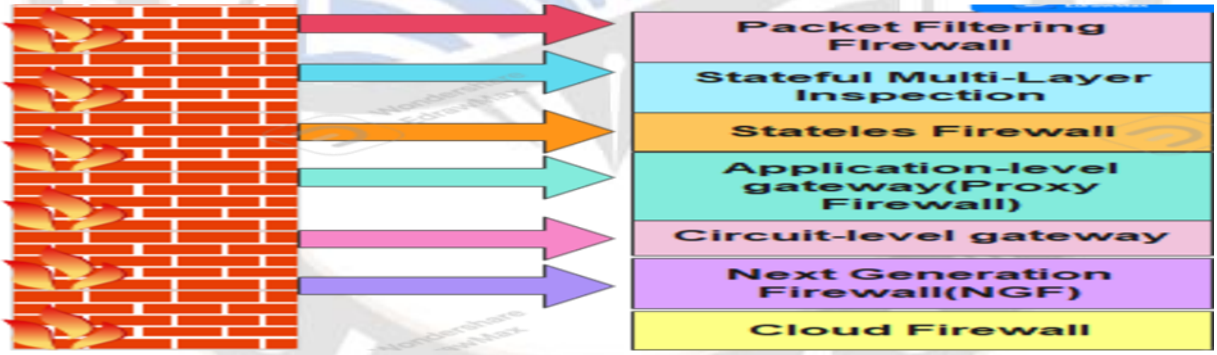
Packet Filtering Firewall
Stateful Multi-Layer Inspection
Stateles Firewall
Application-level gateway(Proxy Firewall)
Circuit-level gateway
Next Generation Firewall(NGF)
Cloud Firewall

Figure 3. Types of Firewalls

Mishandling an action choice can lead to security vulnerabilities, resulting in undesirable outcomes like device shutdowns, service losses, indirect profit declines, or confidential data breaches. Thus, firewalls play a pivotal role in network security, albeit with increasingly complex and error-prone rule management [4]. Figure 3. describes the various types of Firewalls. Artificial Intelligence (AI) techniques exhibit substantial potential across various domains, including cybersecurity [5]. In cybersecurity, AI's utility is gaining traction for enhancing defense and network security, enhancing system robustness, resilience, and responsiveness. Network security systems generate copious log files harboring valuable insights. ML can unearth this knowledge, along with network traffic attribute patterns, to build models aiding network threat detection [6]. The Figure 4 depicts the Firewall rules in action which can either accept, deny or reject a packet. Consequently, the adoption of ML and DL algorithms is on the rise, automating the prediction of recommended actions and subsequent comparison with actual actions [7,8]. Training these systems can generate alerts for threat detection, identify novel malware strains, and safeguard organizational confidential information [9]. Additionally, ML and DL techniques have the capacity for autonomous decision-making, facilitating efficient large-scale analysis.

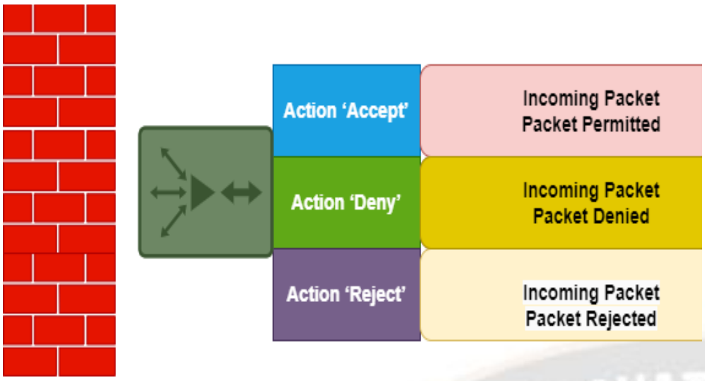
Action 'Accept'
Incoming Packet Packet Permitted
Action 'Deny'
Incoming Packet Packet Denied
Action 'Reject'
Incoming Packet Packet Rejected

Figure 4. Firewall Rule Actions

II. RELATED WORK

Numerous research studies have been conducted to analyze network logs with diverse objectives. Some studies focused on detecting attacks within the log records, while others centered on identifying anomalies. A subset of studies, including ours, aimed at identifying the appropriate action for handling incoming traffic. Multiple research papers engaged in binary classification of network logs, distinguishing them as either normal or anomalous traffic. Allagi et al. [10] investigation made use of a sizable dataset [11] with 22,614,256 records that was made available to the public by the UCI ML repository. K-means was used in their strategy to train the model, and the result was a model with an excellent accuracy of 97.2% and a False Positive Rate (FPR) of 2.7% on the sample dataset. Cao et al. [12] improved existing network log analysis techniques by creating a two-level machine learning approach called the Anomaly Detection System (ADS). 8000 records from a network log evaluation exercise conducted by an IT security organization were used in this investigation. A binary classifier was used for level one to separate normal from anomalous records after six attributes were extracted to identify abnormalities. A Hidden Markov Model was used to specify the kind of anomaly for level two. With an accuracy of 93.54%, the ADS system surpassed single-level anomaly detection techniques . As-Suhbani et al.[13] presented a meta classifier model using four binary classifiers. The Snort and Taut Wire IDS provided the dataset. The class attribute involved the "Allow" or "Drop" action property, and four ML classifiers were used. Their tests showed that the KNN classifier had the best accuracy, coming in at 99.87%. Jia et al.'s [14] network log analysis method also included data mining and machine learning. A spark-based log analyzer with benefits in accuracy, timeliness, and scalability was created to detect aberrant network behavior from large-scale log data.

The attack detection model, however, required manual record manipulation, which prevented its dynamic generation. Data mining and machine learning were combined by Winding et al. [6] to use the JRip method to find abnormal network traffic. Although the experiment had a 99.9167% accuracy rate, the researchers stated that more feature extraction study could improve the outcomes.

A workable technique for examining network records to find attempted breaches was put out by Schindler [2]. They used a modified kill chain model and created SVM models as part of their strategy, obtaining accuracy levels of 95.33% and 98.67%. Additionally, Ucar et al. [4] developed an ML-based model to find anomalies in the repository of firewall rules.

III. PROPOSED METHODOLOGY

In this section, we outline the methodology employed to achieve the objectives of our research, which aims to enhance network security through the analysis of firewall logs collected from MANUU.

Our methodology encompasses data pre-processing, feature engineering, model selection, evaluation, and performance analysis as shown in the figue 5.

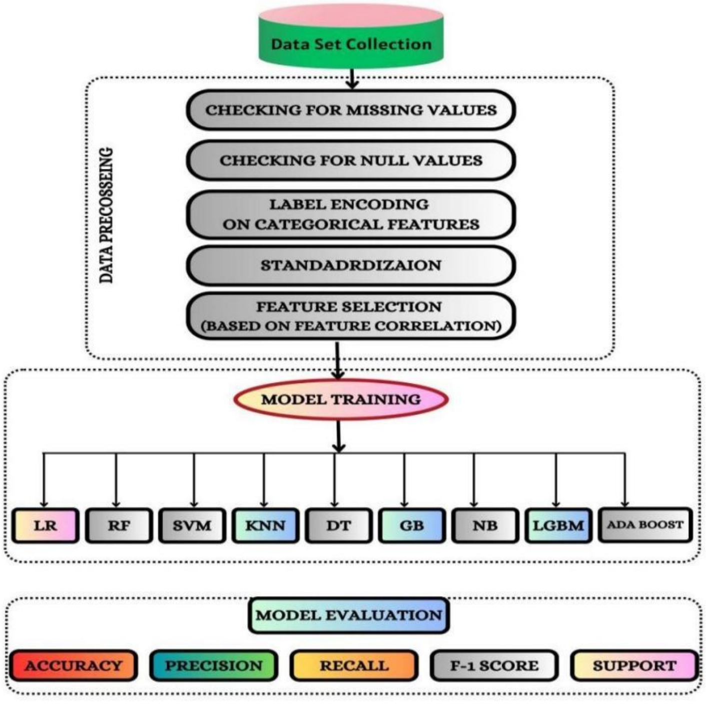

Figure 5. Proposed Model Architecture

3.1. Data Collection

We collected firewall logs from MANU University's network infrastructure, The dataset comprises 4000 instances, each characterized by 9 columns capturing various aspects of network traffic and communication. The columns and their associated data types are as follows:

Table 1. Dataset types

Attribute	Type
Log comp	object
Log subtype	object
NAT rule	int64
Src IP	object
Dst IP	object
protocol	object
Rule type	int64
Live PCAP	object
Log occurrence	int64

The categorical columns, including 'Log comp', 'Src IP', 'Dst IP', 'protocol', and 'Live PCAP', encode various attributes of network activity, while the numerical columns, such as 'NAT rule', 'Rule type', and 'Log occurrence', provide quantitative information about the log entries. The 'Log subtype' column serves as our dependent variable, representing the target we aim to predict through machine learning models. It categorizes each log entry into distinct classes, reflecting whether the network activity is 'Allowed' or 'Denied.' The size of the dataset, consisting of 4000 rows and 9 columns, ensures a diverse representation of network activities, encompassing both regular and potentially anomalous behaviors. This carefully curated dataset forms the basis of our research, allowing us to develop and evaluate machine learning models for classifying network traffic based on firewall log attributes.

3.2. Data Pre-processing

The caliber and readiness of the dataset for subsequent machine learning analysis is crucial to the outcome of our study. To do this, a painstaking data pretreatment pipeline was built to improve dataset quality, fix errors, and get the data ready for easy integration with different machine learning techniques.

3.2.1. Handling Missing Values

Missing values in the dataset were carefully filled in in order to achieve accurate model training and evaluation. Unresolved missing values may induce bias or impair model performance. To guarantee data completion without compromising the dataset's integrity, methods like imputation, where missing values are substituted with reasonable estimations, were used. This process was essential for giving the succeeding steps a solid dataset base.

3.2.2. Categorical Feature Encoding

A suitable encoding approach was essential to enable useful model training because some features, such as "Log comp," "Src IP," "Dst IP," "protocol," and "Live PCAP," are categorical in nature. It was decided to use one-hot encoding, a method that turns category variables into binary vectors. With the use of this method, category attributes could be easily transformed into a numerical representation that machine learning algorithms could use. In order to allow the models to extract insights from these qualities without imposing any ordinal correlations, each distinct category within these features was stored as a binary variable.

3.2.3. Numerical Feature Standardization

It was crucial to align the scales of the numerical features in the dataset. Machine learning algorithms may interpret numerical properties differently as a result of variation in their magnitudes. A standardization process was put in place to address this issue. By scaling by their standard deviation and centring numerical features like "NAT rule," "Rule type," and "Log occurrence" around their mean, these features were normalized. This process made sure that all numerical attributes had comparable scales, preventing any unwarranted impact on the performance of the model from variations in units or ranges.

3.2.4. Data Consistency and Integrity

A vigilant approach was adopted to safeguard the dataset's integrity during the pre-processing phase. Ensuring the dataset's accuracy in reflecting genuine network activities involved the identification and appropriate handling of potential disparities, outliers, and anomalies. This meticulous scrutiny guaranteed a reliable and consistent foundation for subsequent phases of model selection and evaluation. Our endeavor aimed to establish a meticulously organized, sanitized, and harmonized dataset. This was accomplished through the implementation of a robust data pre-processing pipeline encompassing techniques such as managing missing values, encoding categorical features, standardizing numerical features, and conducting checks for data coherence. The resultant dataset, comprehensive and primed for analysis, furnished a robust framework for the subsequent stages of model selection and assessment.

3.3. Data Augmentation

We carried out data augmentation to correct the dataset's class imbalance. We attempted to increase the diversity of the training data by making numerous copies of the original dataset and adding randomness through shuffling. Our models were able to capture the underlying patterns in both the majority and minority classes thanks to this methodology.

Table 2. Records of target variable

Actions	**Allow**	**Deny**
Description	Permit the data packet	Block the data packet
No. Of records before balancing	1771	1421
No. Of records after balancing	1771	1771

3.4. Model Selection

We looked at a variety of machine learning methods, each of which had unique qualities that were appropriate for dealing with the complexity of firewall log data. The following models were considered: Decision Tree, Random Forest, Gradient Boosting,AdaBoost,Logistic Regression, Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Naive Bayes, Multi-Layer Perceptron (MLP), LightGBM, CatBoost.These algorithms were chosen based on their ability to handle classification tasks and their potential to perform well on imbalanced datasets.

3.5. Model Training and Evaluation

For each selected model, first it was trained on the trained data and then evaluated. During the training phase, the model was fitted using enhanced and pre-processed training data (X_train_resampled, y_train_resampled). The test set (X_test) was used in the evaluation phase to produce predictions, which were then compared to the true labels to measure model performance.

3.6. Noisy Label Introductions

We added controlled label noise to the predictions to imitate real-world settings where label noise might be present due to mis-classifications or ambiguity. We wanted to examine the models' resilience and generalization abilities in the face of noise by randomly flipping a fraction of the predicted labels.

3.7. Performance Metrics

Various metrics have been used to evaluate the performance of these models. These metrics included accuracy, precision, recall, and F1-score. Additionally, we analyzed the classification reports for each model to gain insights into their performance for both classes.

4. RESULTS & ANALYSIS

4.1. Model Performance

The performance percent achieved by all the models are below:

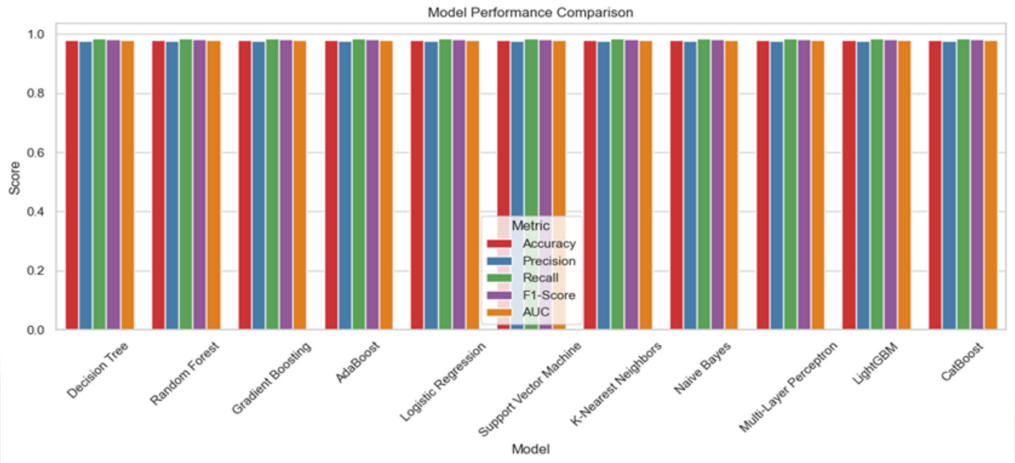

Figure 6. Comparitive results of all models.

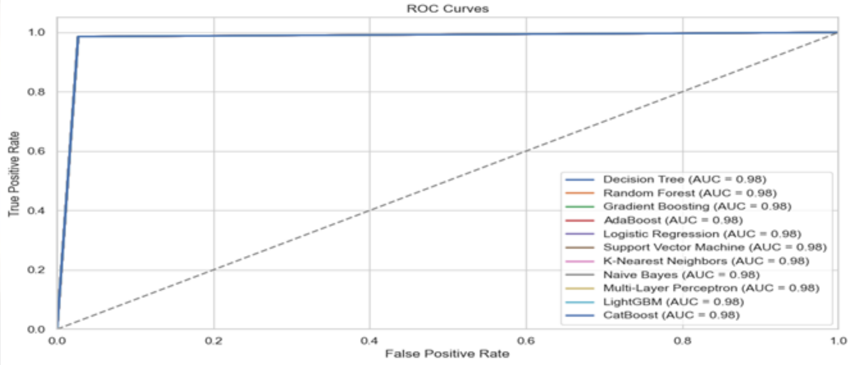

Figure 7. ROC Curves of all models.

4.2. Model Evaluation

To provide a detailed assessment of each model's performance, precision, recall, and F1-score were calculated for both the '0' (non-malicious) and '1' (malicious) classes. The F1-score, which considers both precision and recall, serves as a balanced metric for binary classification tasks. Upon analyzing the classification reports for each model, the following observations were made, Models such as AdaBoost, Gradient Boosting, and Random Forest showcased high accuracy rates (99.00%, 98.50%, and 98.00%, respectively). These models demonstrated robust performance in distinguishing both benign and malicious activities. Decision Tree, K-Nearest Neighbors, and Multi-Layer Perceptron also exhibited satisfactory results with accuracy scores of 98.25%, 97.12%, and 97.38%, respectively. Support Vector Machine and Naive Bayes achieved accuracies of 92.75% and 94.50% respectively, signifying their reasonable performance in classifying the data. These models may benefit from further tuning or feature engineering to improve accuracy. LightGBM and CatBoost achieved accuracy rates of 95.75% and 97.25%, respectively, indicating competitive performance within the scope of this research.

Table 3. Comparison of Accuracies.

S.No	**Model Name**	**Accuracy (%)**
1.	Decision Tree	98.25%
2.	Random Forest	98.00%
3.	Gradient Boosting	98.50%
4.	AdaBoost	99.00%
5.	Logistic Regression	98.00%
6.	Support Vector Machine	92.75%
7.	K-Nearest Neighbors	97.12%
8.	Naive Bayes	94.50%
9.	Multi-Layer Perceptron	97.38%
10.	Light GBM	95.75%
11.	CatBoost	97.25%

4.3. Model Selection and Insights

Based on the results, AdaBoost emerged as the top-performing model with an accuracy of 99.00%. Its high accuracy, along with consistently high precision, recall, and F1-score values for both classes, highlights its effectiveness in classifying network activities as malicious or non-malicious. Gradient Boosting and Random Forest also demonstrated strong performances, with accuracy scores of 98.50% and 98.00% respectively. It is noteworthy that AdaBoost, Gradient Boosting, and Random Forest not only achieved remarkable accuracy but also maintained balanced precision and recall values for both classes, indicating their capacity to generalize well on unseen data. These models could potentially serve as key tools for network security applications, aiding in the identification of suspicious activities.

5. CONCLUSION

Firewalls are an important component of corporate network security since they are the network's first line of defense. Furthermore, firewalls can guard against both external and internal assaults. In this Research, we have proposed a framework using the firewall log data which can classify to either permit the data packet or drop the data packet. The data has been gathered from Department of CS&IT, MANU University. Handling missing values, encoding categorical features, standardizing numerical features, and performing data consistency checks, were the actions which were performed in data pre-processing pipeline. After balancing the target variable's class imbalance, various ML models were trained and assessed using criteria like accuracy, precision, recall, and F1-score. The model that performed the best, AdaBoost, with a 99.00% accuracy rate.

REFERENCES

[1] Neupane, K.; Haddad, R.; Chen, L. Next Generation Firewall for Network Security: A Survey. In Proceedings of the SoutheastCon 2018, St. Petersburg, FL, USA, 19–22 April 2018; Volume 2018.

[2] Schindler, T. Anomaly detection in log data using graph databases and machine learning to defend advanced persistent threats. arXiv 2017, arXiv:1802.00259.

[3] Ertam, F.; Kaya, M. Classification of firewall log files with multiclass support vector machine. In Proceedings of the 6th International Symposium on Digital Forensic and Security (ISDFS), Antalya, Turkey, 22–25 March 2018; Volume 2018.

[4] Ucar, E.; Ozhan, E. The Analysis of Firewall Policy Through Machine Learning and Data Mining. Wirel. Pers. Commun. 2017, 96, 2891–2909.

[5] Tu, Y. Machine Learning. In EEG Signal Processing and Feature Extraction; Springer: Singapore, 2019.

[6] Winding, R.; Wright, T.; Chapple, M. System anomaly detection: Mining firewall logs. In Proceedings of the 2006 Securecomm and Workshops, Baltimore, MD, USA, 28 August–1 September 2006.

[7] Aljabri, M.; Aljameel, S.S.; Mohammad, R.M.A.; Almotiri, S.H.; Mirza, S.; Anis, F.M.; Aboulnour, M.; Alomari, D.M.; Alhamed, D.H.; Altamimi, H.S. Intelligent techniques for detecting network attacks: Review and research directions. Sensors 2021, 21, 7070.

[8] Tiwari, A.K. Introduction to Machine Learning; IGI Global: Uttarakhand, India, 2017.

[9] Aljabri, M.; Mirza, S. Phishing Attacks Detection using Machine Learning and Deep Learning Models. In Proceedings of the 2022 7th International Conference on Data Science and Machine Learning Applications (CDMA), Riyadh, Saudi Arabia, 1–3 March 2022; pp. 175–180.

[10] Allagi, S.; Rachh, R. Analysis of Network log data using Machine Learning. In Proceedings of the 2019 IEEE 5th International Conference for Convergence in Technology (I2CT), Bombay, India, 29–31 March 2019.

[11] Ertam, F. Internet Firewall Data Data Set 2018; Firat University: Elazı ˘g, Turkey, 2018.

[12] Cao, Q.; Qiao, Y.; Lyu, Z. Machine learning to detect anomalies in web log analysis. In Proceedings of the 2017 3rd IEEE International Conference on Computer and Communications (ICCC), Chengdu, China, 13–16 December 2017; Volume 2018.

[13] As-Suhbani, H.E.; Khamitkar, S.D. Classification of Firewall Logs Using Supervised Machine Learning Algorithms. Int. J. Comput. Sci. Eng. 2019, 7, 301–304.

[14] Jia, Z.; Shen, C.; Yi, X.; Chen, Y.; Yu, T.; Guan, X. Big-data analysis of multi-source logs for anomaly detection on network-based system. In Proceedings of the IEEE International Conference on Automation Science and Engineering, Xi’an, China, 20–23 August 2017; Volume 2017.