A Comprehensive Analysis of Machine Learning Techniques for Diagnosing DDOS Intrusions Using Network Traffic Data

Sujan Tumbaraguddi, *MSc Data Science Student, Coventry University, Coventry, England*, and RAMANA Kulanthaivelu, *MSc Data Science Student, Coventry University, Coventry, England*

***Abstract*— Distributed Denial of Service (DDOS) attack detection and mitigation have become essential for maintaining network security due to the rise in cyber threats. In this research, we provide an in-depth analysis of machine learning techniques used to use network traffic data to identify DDOS breaches. We test a range of classification techniques, such as Random Forest, Naive Bayes, Logistic Regression, K-Nearest Neighbors (KNN), and Decision Tree, in an effort to create models that can accurately differentiate between typical network activity and unusual activity suggestive of denial-of-service (DDOS) attacks. The outcomes of the experiment show how well these algorithms function at precisely detecting DDoS intrusions, improving network security and reducing the risk of cyberattacks.**

Index Terms*— *Machine Learning; Cybersecurity; Intrusion Detection; Network Traffic Analysis; Random Forest; Logistic Regression; Naive Bayes; Support Vector Machine (SVM); Decision Tree; Anomaly Detection; DDoS Attacks; Feature Extraction; Data Preprocessing; Model Evaluation; Accuracy Assessment

I. INTRODUCTION

The proliferation of internet-connected devices and the exponential growth of network traffic have significantly increased the risk of cyber threats, with DDOS attacks posing a significant challenge to network security. DDOS attacks aim to disrupt network services by inundating target systems with a deluge of traffic, rendering them inaccessible to legitimate users. Detecting and mitigating DDOS intrusions are paramount for safeguarding network infrastructure and maintaining service availability.

Machine learning techniques offer a promising approach to address the complexities of identifying DDOS attacks in network traffic data. By analyzing patterns and anomalies in network traffic, machine learning models can effectively distinguish between normal network behavior and malicious activities associated with DDOS attacks. This paper presents a comprehensive analysis of machine learning methods applied to diagnose DDOS intrusions, aiming to provide insights into their performance and effectiveness in enhancing network security.

II. LITERATURE REVIEW

A logistic regression model was presented in "Detection of Application Layer DDoS Attack by Modelling User Behavior Using Logistic Regression" (Yadav and S. Selvakumar) to model typical user browsing behavior and identify application layer DDOS attack traffic. The best features were chosen and added to the feature set to create distinguishing factors that would identify attackers apart from normal users. Testing was done in a Testbed environment using both regular and malicious traffic from web server logs. According to the results, the approach successfully separates attack traffic from regular traffic, averaging a 98.64% detection rate with a 1.41% false positive rate.

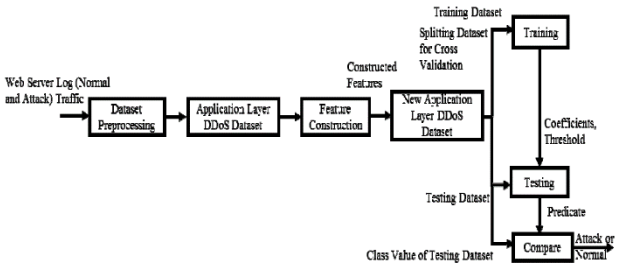


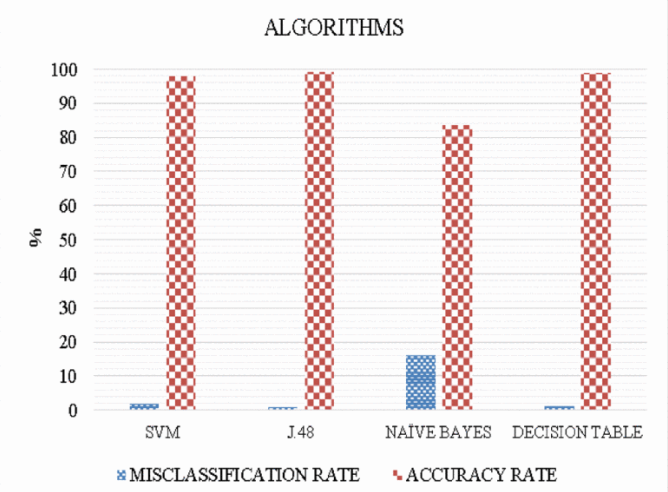
Figure 1: Block schematic diagram of proposed method (Yadav and S. Selvakumar (n.d.))

An anomaly-based statistical pattern identification method for application layer DDoS attack detection is presented in this paper. The procedure is provided in block graphic form. Preprocessing web server logs yields a dataset that includes both legitimate and malicious traffic. On this dataset, feature extraction is done, and then new attributes are created. A new dataset designed exclusively for application layer DDoS attacks is produced by integrating these properties. To avoid overfitting, the dataset is divided into subgroups for testing and training using tenfold cross-validation. During the evaluation phase, the algorithm is tested on incoming traffic and taught to recognize application layer DDOS assaults.

In their study on intrusion detection, Mehmood and Rais (2016) explored the application of supervised machine learning algorithms. These included support vector machines (SVMs), naïve Bayes classifiers, J.48 decision trees, and decision tables, selected for their high efficiency. The algorithms were trained and assessed using the widely-used KDD99 dataset, recognized

as a benchmark for evaluating detection model performance.

Figure 2: Overall comparison of different supervised algorithms Mehmood and Rais (2016)



This study examines machine learning techniques including SVM, naive Bayes, J.48, and decision tables for anomaly detection, using the KDD99 dataset. Each algorithm demonstrates varied performance across different classes, with no single algorithm achieving high True Positive Rates (TPR) consistently. However, the J.48 decision tree exhibits the highest overall accuracy and lowest misclassification rate compared to others. This could be attributed to the decision tree's capability to perform well with redundant features. Future research will explore combining this approach with other feature selection methods.

Almulla (2022) explores cyber-attack detection in network traffic using machine learning, aiming to differentiate between normal and abnormal network data. The study employs the Support Vector Machine (SVM) Algorithm for collaborative learning, which identifies trends and clusters characteristics based on commonality. Given the unpredictable nature of threats, early detection is crucial to mitigate potential damage. Visualizations and statistical tests are utilized to validate hypotheses and assess relationships among data components. Modelling techniques, including LSVM, C5, Random Trees, Neural Networks, Logistic Regression, and Trees-AS, are employed, with C5 identified as the most suitable based on parameters like accuracy, sensitivity, and specificity. Enhancing real-world accuracy entails collecting real-time network data for analysis, enabling the detection of unauthorized access. Future research could explore additional modelling techniques to expand comparative analyses of model performance.

III. Problem and Dataset Description

The dataset used in this study, taken from the knowledge discovery in databases (kdd) repository, was rigorously selected to provide detailed insights into network connections, with a focus on distributed denial of service (ddos) intrusions. It consists of 42 numerical and category elements that address various aspects of network traffic. Numeric characteristics include metrics like connection duration, bytes exchanged, error rates, and counts of certain events, whereas categorical features include protocol kinds, service details, connection flags, and attack class indicators. The introduction of the 'attack_class' column as the target variable allows network connections to be classified as normal or anomalous, improving proactive detection and mitigation of ddos attacks. By applying machine learning models to this structured dataset, cybersecurity experts and organisations obtain essential tools for fortifying their networks against ddos attacks, protecting vital resources and assuring continuous operations and data integrity. This dataset's precise insights into network traffic patterns are critical in strengthening network infrastructure resilience against malicious attacks, ultimately improving organisations' overall security posture in the face of emerging cyber threats.

No	Feature	Type	Categorical Range Value
1	id	Omit	
2	duration	Numeric	
3	protocol_type	Categorical	
4	service	Categorical	
5	flag	Categorical	
6	src_bytes	Numeric	
7	dst_bytes	Numeric	
8	land	Categorical	0,1
9	wrong_fragment	Numeric	
10	urgent	Numeric	
11	hot	Numeric	
12	num_failed_logins	Numeric	
13	logged_in	Categorical	0,1
14	num_compromised	Numeric	
15	root_shell	Categorical	0,1
16	su_attempted	Numeric	
17	num_root	Numeric	
18	num_file_creations	Numeric	
19	num_shells	Categorical	0,1,2,5
20	num_access_files	Categorical	0,1,2,3,4
21	num_outbound_cmds	Omit	
22	is_host_login	Categorical	0,1
23	is_guest_login	Categorical	0,1
24	count	Numeric	
25	srv_count	Numeric	
26	serror_rate	Numeric	
27	srv_serror_rate	Numeric	
28	rerror_rate	Numeric	
29	srv_rerror_rate	Numeric	
30	same_srv_rate	Numeric	
31	diff_srv_rate	Numeric	
32	srv_diff_host_rate	Numeric	
33	dst_host_count	Numeric	
34	dst_host_srv_count	Numeric	
35	dst_host_same_srv_rate	Numeric	
36	dst_host_diff_srv_rate	Numeric	
37	dst_host_same_src_port_rate	Numeric	
38	dst_host_srv_diff_host_rate	Numeric	
39	dst_host_serror_rate	Numeric	
40	dst_host_srv_serror_rate	Numeric	
41	dst_host_rerror_rate	Numeric	
42	dst_host_srv_rerror_rate	Numeric	
43	attack_class	attack_class	attack_class

TABLE I

IV. Methods

In this section, we outline the machine learning methods employed to address the problem of diagnosing DDOS intrusions using network traffic data. We investigate the following algorithms:

A. Random Forest

Random Forest is renowned for its robustness, particularly in managing large datasets with high dimensionality, like the network traffic data from the Knowledge Discovery in Databases repository. With its ability to handle various network connection parameters such as protocol type, service, flag, source bytes, destination bytes, and attack class, Random Forest proves to be a suitable choice for analyzing and categorizing complex data.

B. Logistic Regression

Logistic Regression, a statistical tool, efficiently identifies DDoS incursions by analysing network traffic data with 43 features. Logistic Regression uses features such protocol type, service, flag, source bytes, destination bytes, and attack class to estimate the likelihood of each class based on the linear combination of feature values.

C. Naive Bayes

Naive Bayes, a probabilistic classifier, is ideal for detecting DDoS incursions using network traffic data from the KDD repository. Using features such as protocol type, service, flag, source bytes, destination bytes, and attack class, Naive Bayes predicts the likelihood of each class given the feature values. Its simplicity and efficiency make it an invaluable tool for accurately categorising network connections.

D. K-Nearest Neighbors (KNN)

K-Nearest Neighbours (KNN), a non-parametric classification technique, is capable of detecting DDoS attacks using the dataset's 43 features. KNN assigns a class label to each instance based on the most prevalent class among its k nearest neighbours. Its flexibility and simplicity make it ideal for accurately identifying network connections in the context of DDoS intrusion detection.

E. Decision Tree

To diagnose DDoS incursions, the 43 features of the dataset are quickly analysed using Decision Tree, a common classification technique. By splitting the data recursively based on feature values, Decision Tree generates a tree-like structure in which each internal node represents a feature and each leaf node represents a class label. Its straightforward style and ability to handle both numerical and categorical data make it a viable option for DDoS intrusion detection.

The subsequent subsections provide the implementation details of each algorithm.

IV. Experimental Setup

In this section, we detail the experimental setup conducted to prepare the dataset for machine learning analysis. The following steps were performed:

A. Data Loading

The dataset was loaded from the provided CSV file using the pandas library in Python.

B. Data Cleaning

The 'ID' column, which served as an identifier with no analytical value, was manually removed. The irrelevant 'num_outbound_cmds' column, which consistently contained 0, was removed from the dataset. Removal improved dataset efficiency for machine learning tasks because the feature lacked variability and provided no predictive value. This ensured data integrity. The removal of extraneous features streamlined the dataset for efficient machine learning analysis, increasing accuracy in diagnosing DDoS intrusions from network traffic data.

C.Feature Encoding:

LabelEncoder encoded categorical columns ('protocol_type', 'service', 'flag', 'land', 'logged_in', 'root_shell', 'num_shells', 'num_access_files', 'is_host_login', 'is_guest_login', and 'attack_class') into numerical labels. Each categorical value was assigned a unique numerical representation, which aided machine learning algorithms' comprehension. This transformation speeds up data preprocessing, improving model training efficiency and analysis. Numerical representation enables algorithms to efficiently process categorical data, which contributes to accurate predictive modelling.

D. Data Splitting

The dataset was divided into features (X) and the target variable (y), with X representing the independent variables and y representing the dependent variable. This separation allows for supervised learning by isolating input features from the target variable. Such segmentation allows machine learning algorithms to comprehend the link between features (X) and target variables (y), which is critical for predictive modelling and performance evaluation.

E. Train-Test Split

The train_test_split method separated the dataset into training and testing sets. This split divided the data into four subsets: x_train (training features), x_test (testing features), y_train (training target variable), and y_test. With a test size of 20% and a random state of 42, this division provided enough data for both training and evaluating machine learning models, allowing for rigorous performance assessment.

V. Results

TABLE II

Algorithm	Test Accuracy	Precision	Recall	F1-score
Naive Bayes	52.22%	0.539	0.522	0.374
Logistic Regression	84.22%	0.842	0.842	0.842
K-Nearest Neighbors	99.02%	0.99	0.99	0.99
Random Forest	99.57%	0.996	0.996	0.996
Decision Tree	99.52%	0.995	0.995	0.995

The table above presents the experimental outcomes of employing various machine learning algorithms for diagnosing DDOS intrusions using network traffic data. Here's an overview of the findings:

Random Forest: Achieved the highest accuracy of 99.57%, showcasing robust performance with the specified parameters.

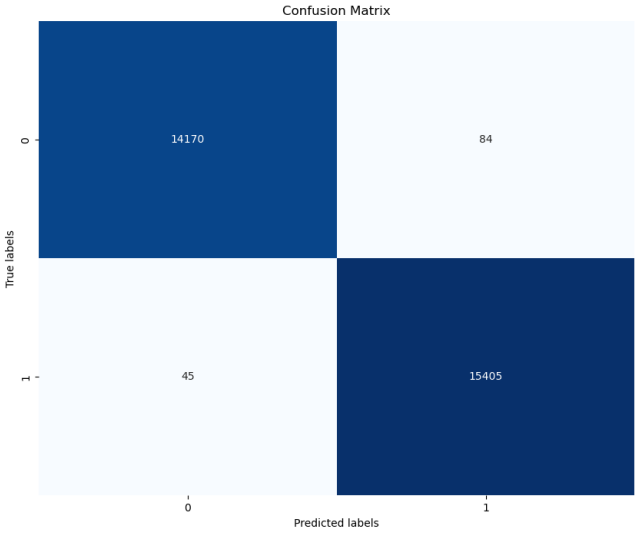


Figure 3: Random Forest Confusion matrix

Naive Bayes: Demonstrated the lowest accuracy of 52.26%, indicating comparatively weaker performance among the tested algorithms.

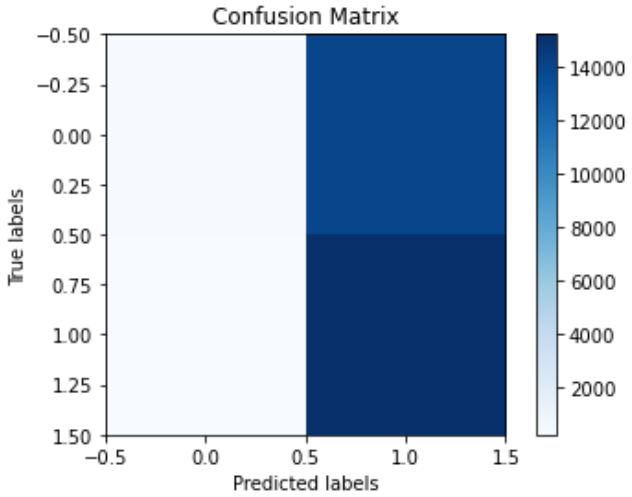


Figure 4: Naive Bayes Confusion matrix

Logistic Regression: Displayed moderate performance with an accuracy of 84.22%. The accuracy exhibited slight variations with different regularization strengths.

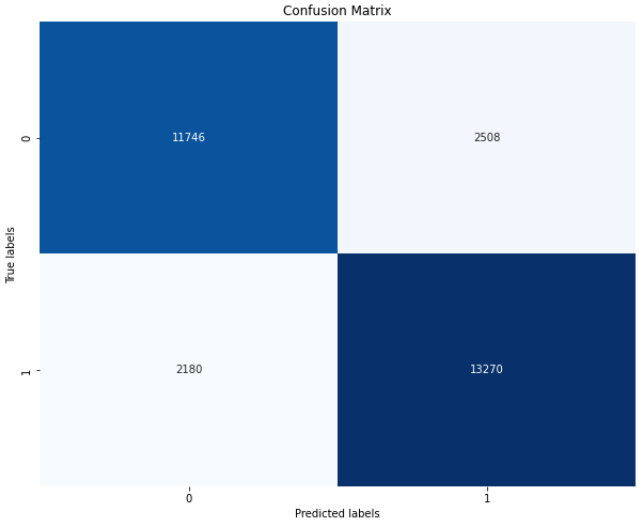


Figure 5: Logistic Regression Confusion matrix

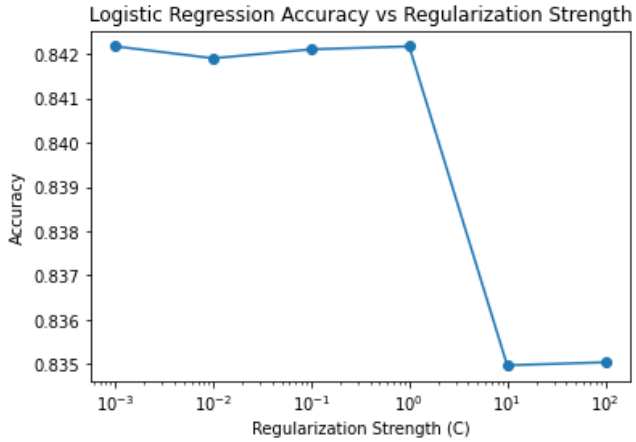


Figure 6

K-Nearest Neighbors (KNN): Attained an accuracy of 99.02%, with differing accuracies observed for various values of K.

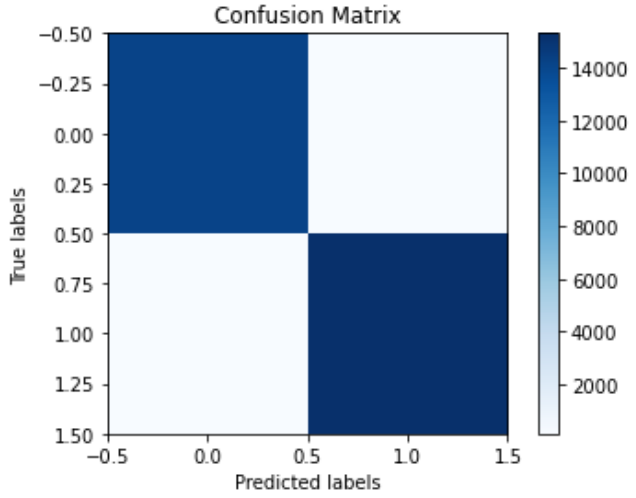


Figure 7: K-Nearest Neighbors (KNN) Confusion matrix

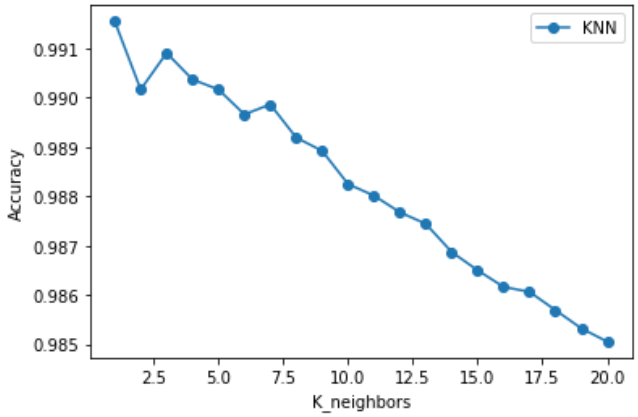


Figure 8: KNN Accuracy for neighbours

Decision Tree: Recorded an accuracy of 99.53%, slightly below Random Forest.

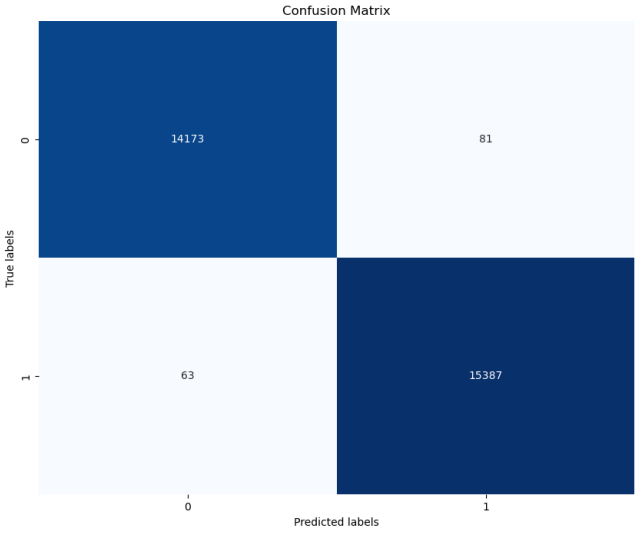


Figure 9: Decision Tree Confusion matrix

Analyzing these results reveals that Random Forest and Decision Tree models performed exceptionally well, surpassing the accuracy of other models. Although Logistic Regression showed lower accuracy compared to ensemble methods, it still presented commendable performance. However, Naive Bayes exhibited suboptimal results, indicating potential limitations in handling the dataset's complexity.

Various factors, including parameter tuning, feature selection, and dataset characteristics, influenced the algorithms' performance. Further optimization and fine-tuning could enhance model accuracy in diagnosing DDOS intrusions effectively. While accuracy serves as a vital metric for evaluating model performance, comprehensive assessment incorporating precision, recall, and F1-score is essential, particularly in imbalanced datasets.

In conclusion, ensemble methods such as Random Forest and Decision Tree demonstrate effectiveness in diagnosing DDOS intrusions. However, logistic regression and naive Bayes may require additional refinement or alternative methodologies to bolster their performance in this specific context.

VI. Discussion

In this study, we used network traffic data to assess how well different machine learning algorithms performed when used to diagnose DDOS incursions. With an accuracy of 52.22%, Naive Bayes had the lowest accuracy of all the algorithms examined. With an accuracy of 84.22%, Logistic Regression revealed a substantial improvement; its regularisation strength had a minimal effect on accuracy. The K-Nearest Neighbours algorithm (KNN) yielded an impressive 99.02% accuracy rate; one neighbour is the ideal number. The models of Random Forest and Decision Tree fared better than any other algorithm, with respective accuracy of 99.57% and 99.52%. These findings demonstrate how well ensemble learning methods perform when dealing with challenging classification challenges.

VII. Conclusion

We conclude that, with over 99% accuracy rates, ensemble techniques such as Random Forest and Decision Tree are highly effective in diagnosing DDOS incursions using network traffic data. On the other hand, the accuracy of the naive Bayes and logistic regression models was significantly lower, indicating that they were unable to handle the dataset's complexity.

In order to stay up with the rapidly changing landscape of cyber threats, it will be imperative to give ongoing machine learning model adaptation and refining top priority going forward. To strengthen overall defence strategies, efforts should also be directed towards improving model interpretability and incorporating complementing security measures. In the end, even while machine learning algorithms provide insightful information, human knowledge in cybersecurity operations should still be supplemented by these algorithms, not replaced.

References

Almulla, K. (2022). “Cyber-attack detection in network traffic using machine learning” (2022). . Rochester Institute of Technology.

Chou, D., & Jiang, M. (2022). A Survey on Data-driven Network Intrusion Detection. ACM Computing Surveys, 54(9), 1–36. https://doi.org/10.1145/3472753

Churcher, A., Ullah, R., Ahmad, J., ur Rehman, S., Masood, F., Gogate, M., Alqahtani, F., Nour, B., & Buchanan, W. J. (2021). An Experimental Analysis of Attack Classification Using Machine Learning in IoT Networks. Sensors, 21(2), 446. https://doi.org/10.3390/s21020446

Dash, S. (2023, February 7). Understanding the ROC and AUC Intuitively. Medium. https://medium.com/@shaileydash/understanding-the-roc-and-auc-intuitively-31ca96445c02

Geurts, P., Ernst, D., & Wehenkel, L. (2006). Extremely randomized trees. Machine Learning, 63(1), 3–42. https://doi.org/10.1007/s10994-006-6226-1

Joaquín Gaspar Medina-Arco, Magán-Carrión, R., Rafael Alejandro Rodríguez-Gómez, & García-Teodoro, P. (2024). Methodology for the Detection of Contaminated Training Datasets for Machine Learning-Based Network Intrusion-Detection Systems. Sensors, 24(2), 479–479. https://doi.org/10.3390/s24020479

Kato, K., & Klyuev, V. (2014). An Intelligent DDoS Attack Detection System Using Packet Analysis and Support Vector Machine. International Journal of Intelligent Computing Research, 5(3), 464–471. https://doi.org/10.20533/ijicr.2042.4655.2014.0060

KDD-Datasets. (n.d.). Www.kaggle.com. Retrieved March 24, 2024, from https://www.kaggle.com/datasets/rasoulrahimii/kdddatasets?select=Test.csv

LaValley, M. P. (2008). Logistic Regression. Circulation, 117(18), 2395–2399. https://doi.org/10.1161/circulationaha.106.682658

M. Bhuyan, D. Bhattacharyya, & J. Kalita. (2015). Towards Generating Real-life Datasets for Network Intrusion Detection. Semantic Scholar. https://www.semanticscholar.org/paper/Towards-Generating-Real-life-Datasets-for-Network-Bhuyan-Bhattacharyya/e35e943f71f3ab95889055f718388fb49daf7a31

Mehmood, T., & Rais, H. B. M. (2016). Machine learning algorithms in context of intrusion detection. 2016 3rd International Conference on Computer and Information Sciences (ICCOINS). https://doi.org/10.1109/iccoins.2016.7783243

Quinlan, J. R. (1996). Learning decision tree classifiers. ACM Computing Surveys, 28(1), 71–72. https://doi.org/10.1145/234313.234346

Scarfone, K., & Mell, P. (2007, February 20). Guide to Intrusion Detection and Prevention Systems (IDPS). Csrc.nist.gov. https://csrc.nist.gov/pubs/sp/800/94/final

Scarfone, K., & Mell, P. (2010). Intrusion Detection and Prevention Systems. Handbook of Information and Communication Security, 177–192. https://doi.org/10.1007/978-3-642-04117-4_9

Staudemeyer, R. C., & Omlin, C. W. (2014). Extracting salient features for network intrusion detection using machine learning methods. South African Computer Journal, 52. https://doi.org/10.18489/sacj.v52i0.200

Yadav , S., & S. Selvakumar . (n.d.). Detection of Application Layer DDoS Attack by Modeling User Behavior Using Logistic regression . https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7359289