EAI Endorsed Transactions
on Scalable Information Systems

Research Article EAI.EU

Digital Investigation of Network Traffic Using Machine Learning

Saswati Chatterjee[1,*], Suneeta Satpathy[2] and Arpita Nibedita[3]

[1]Faculty of Emerging Technology, Sri Sri University, Cuttack, Odisha, India
[2]Center for AI & ML, SOA University, Bhubaneswar, Odisha, India
[3]CSE, Trident Academy of Technology, Bhubaneswar, Odisha, India

Abstract

In this study, an intelligent system that can gather and process network packets is built. Machine learning techniques are used to create a traffic classifier that divides packets into hazardous and non-malicious categories. The system utilizing resources was previously classified using a number of conventional techniques; however, this strategy adds machine learning., a study area that is currently active and has so far yielded promising results. The major aims of this paper are to monitor traffic, analyze incursions, and control them. The flow of data collection is used to develop a traffic classification system based on features of observed internet packets. This classification will aid IT managers in recognizing the vague assault that is becoming more common in the IT industry The suggested methods described in this research help gather network data and detect which threat was launched in a specific network to distinguish between malicious and benign packets. This paper's major goal is to create a proactive system for detecting network attacks using classifiers based on machine learning that can recognize new packets and distinguish between hostile and benign network packets using rules from the KDD dataset. The algorithm is trained to employ the characteristics of the NSL-KDD dataset.

Keywords: KDD, Hybrid Machine learning, Network forensics, DDoS

Received on 17 July 2023, accepted on 09 September 2023, published on 03 October 2023

doi: 10.4108/eetsis.4055

1. Introduction

The dataset grows in size as a result of the heavy network traffic and the varied sources from which the information is gathered. As a result, when dealing with a huge dataset, data analysis is challenging. Presently, the detection of normal and aberrant patterns relies heavily on data mining techniques. because it can handle big datasets and uncover hidden information. The clustering algorithm, association algorithm, mining algorithm, and classification algorithm are only a few of the many algorithms that make up a data mining algorithm. A classification technique that may be applied to spot patterns maintains the identification of pertinent data structures. Data linkages and statistical or predictive models of the data, etc.

The task of classifying each dataset instance is called classification. By applying classification algorithms to the data set that has been gathered, it is possible to distinguish between malicious and legitimate traffic. This kind of categorization is essential for network monitoring systems and security incidents.

Subsequently, it became necessary to detect network activities using clearly specified port numbers. For instance, port 80 is used for HTTP communication while port 25 is used for SMTP connection. Yet, due to applications' use of dynamically changing port numbers and the rapidly expanding internet, classifying traffic based on ports has become a laborious task. Network traffic classification based on ports led to the adoption of payload-based inspection. This categorization can be carried out with respectable accuracy after the payload can be obtained and thoroughly inspected. The payload-based categorization has some limitations in terms of speed and resource utilization, although it has good accuracy. Although some writers in the research community presented automatic processes for the derivation of payload

*Corresponding author. Email: cshiva68@gmail.com

traits and demonstrated some encouraging findings, these methods still have their own drawbacks. It uses a lot of processing power and relies on it for the methods it covers.

But it will take up less memory and processing time if we merely examine the first few bytes of the payload. Because of the constant increase in network data brought on by technological improvements, researchers have begun using machine learning techniques based on features to classify the data. The categorization model is produced by machine learning-based algorithms using a huge data collection and deduced features. Also, it is becoming more important for machine learning-based classifications to take into account statistically based features of network traffic, by taking into account machine learning (ML) based techniques and basing it on these identified feature data, a decent traffic classifier may be produced. Despite ML classifiers' encouragingly high efficiency and accuracy, accuracy is typically lower than that of payload-based classifiers.

The rest of the paper is embodied as section 2 represents the related work, section 3 defines the datasets, section 4 elaborates the proposed classification algorithm, section 5 defines the proposed work In Section 6 the result analysis The research is then concluded in section 7.

2. Related Work

Intrusion detection system research has been conducted extensively [1][2][3][4][5] According to researchers [6] NIDS can be categorized as anomaly-based and signature-based. By monitoring network activity and comparing it to the established trends of identified assaults or attacks, a signature-based surveillance system has the advantage of identifying existing assaults with an elevated rate of detection. However, according to scientists [7], it does a poor job of detecting fresh attacks or even variations of established assaults. An assault will be viewed as any deviation from these patterns of typical behaviour by an anomaly-based identification system. Because it can identify unknown threats, the anomaly-based technique has an edge in intrusion detection. Since the functioning of ML is in reality in accordance with the premise of various methodologies by autonomous knowledge to boost efficiency, it has been determined to apply different theories as well as approaches for attack detection [8]. In order to characterize and categorize hostile cyber actions directed at Web systems, an ML-based methodology was put out [9][10][11] It was discovered that supervised learning approaches accurately identify between attack Web sessions. One of the most well-liked research topics in this area right now is the main concern of attack detection. In order to identify communications utilizing the Domain Generation Algorithm, [12] introduced and explained the development of a fraudulent domain name recognition system based on machine learning is therefore possible to draw the conclusion that ML does in fact provide cybersecurity insights and assist in the growth of various strategies. Identifying the attack has traditionally been viewed as an issue of categorization where receiving network data is classified into an appropriate category, claim [13][14] There have been numerous studies on these techniques, including support vector machines [15] decision trees [16]; SVM is becoming more prominent among the aforementioned methods due to its potential outcomes SVM is a kind of operational algorithms for the attacking problem when compared to other approaches. The expansion of a reliable, successful, and flexible intrusion attack has therefore been seen as the preferred option The proper employment of SVM has received a lot of attention recently, and numerous researchers have looked into compared between this kind of network, and can able to find out the correct performance measure. Moreover, SVM was contrasted with different classifiers and showed that it outperformed the other approaches by a wide margin. SVM is an appropriate method for identifying network intruders and its utility has been shown in numerous circumstances. SVM is the intrusion detection technique employed in our analysis in light of these. However, using SVM alone might not be sufficient to further enhance detection performance [Hackers are constantly looking for ways to break into the network and compromise systems. The repercussions are severe and may jeopardize the privacy of any host device linked to the network as well as passwords, bank login information, social security numbers, and medical records. Monitoring network data from anywhere is a major concern in the field of cybersecurity because many networks in use today feature remote access devices. Typically, the consumer will rely on the network of the service provider to secure the security of their equipment. The scopes of other works that describe how they capture MITM attack traffic are frequently quite constrained. To the best of our knowledge, several of these experiments use closed, artificial network environments for the deployment of their attacks. Their traffic frequently does not reflect all of the typical traffic kinds encountered on real networks.

Traffic must cover the majority of typical network usage in order to be deemed representative. Web browsing, file transfers, server contact, and multimedia streaming are examples of this, although they are not the only ones. Other studies oversimplify or concentrate on a specific network activity, such as conducting File Transfer Protocol (FTP) downloads for the attacker to intercept, as the type of data the attack targets. Several researchers have utilized the NSL-KDD and KDD99 datasets as examples in their work. The majority of hybrid IDS systems train each planned model separately before simply averaging their results to produce final results. The results of the assessment show how effective it is at producing a large enough number of detections with high levels of accuracy (96.1%) and a false positive rate (3.3%). The Naive Bayes technique [17] produced a detection rate of 95% using 41 common features from the KDD99 dataset after eliminating 90% of the instances of the standard features.

3. Dataset

The NSL-KDD dataset was deployed in this study to illustrate the superiority of the suggested methodology. The

KDD99[18] redundant training and test datasets, which have 78% and 75% of repeated records, respectively, are quite large. The outcome of the establishment of an evaluation for much more accurate detection accuracy can be severely impacted by redundant data sets. The new NSL-KDD datasets are the outcome of the necessary KDD99 change. Table 1 displays results on the decrease of redundant data in the KDD train and test sets, respectively, while Table 2 displays the specific distinctions between KDD99 with attack names and attack classes discovered in NSL-KDD

Table 1. Results of repeated records in the KDD Set

	Original data Records	Distinct data Records	Decrease Rate
Attacks	3,925,650	262,178	93.32%
Normal	972,781	812,814	16.44%
Total	4,898,431	1,074,992	78.05%

Table 2. NSL-KDD datasets contain four attack categories with relevant attack names.

Attack Type	**Attack Name**
Denial of Service (DoS)	back, land, Neptune, pod, smurf, teardrop
Remote to Local (R2L)	Guess password, ftp write, imap, phf, multihop, warezmaster, wareclient, spy
User to Root (U2R)	buffer overflow, loadmodule, perl, rootkit.
Probing	satan, ipsweep, nmap, portsweep.

The following four broad categories were used by the NSL-KDD dataset to group the various attacks:

(i) **Denial of Service (DOS):** These are used to restrict approved users from using hosts' or networks' services. The major objective was to fully consume memory resources, which prevents users from accessing a system or network and makes it hard to fulfil legitimate network requests.

(ii) **Remote to Local** This one involves trying to get into an existing account from a different host. These sorts of assaults are associated with, the victim forwarding data packets to the target computer through the network without being able to see the computer's security flaws or having the same rights that a local user would have.

(iii) **User to Root (U2R):** These are instances of system exploitation where an attacker logs in with restricted user privileges or regular user privileges and attempts to use system bugs to achieve privilege, for instance, via Perl, a rootkit, etc.

(iv) **Probe:** Attacks known as "probes" scan computer networks or systems in an effort to gather information or identify vulnerabilities. Identification of the systems in a network that possesses known flaws that could be later used to compromise the system is the aim of this data collection.

4. Proposed Classification Algorithm

Based on the actions of cybercriminals, attack intention analysis deduces the purpose of an attack. In addition to giving more information about the cybercrime's proof and the attacker's conduct, which facilitates the identification of the offender.

When an attack is expected and the attacker's intended target is known, the attack intention is identified. Determining the underlying goals of current cyberattacks is becoming more and more difficult due to their complexity. Even specialists have trouble figuring out the entryway. An attacker employs tools to hide or camouflage his patterns from his victims and follows a logical set of steps to complete his purpose. Pattern recognition is more difficult in network systems with a variety of attack tactics. For instance, false positive or false negative reading errors are the main problem in IDS, especially in misuse-based and anomaly-based detection, as described by [19].

The alternate method builds the attack automatically using a graph from the checking model The strength of this paradigm, according to Qin and Lee [20][21][22] is its effectiveness in evaluating protocols. This paradigm has restricted scalability but is also more reliable than other approaches like simulation or theorem methodologies. The researchers presented a method for creating attack path graphs in order to identify invasive intentions. They arrived at the conclusion that the suggested strategy is inadequate and only catches the initial phases of attack intention. The researchers also found that the attack's vulnerability is dependent on its goal, demonstrating that the method is inadequate for large volumes of data.

The limitations of security sensors and network monitoring technologies, in summary, make attack observation imprecise and challenging to comprehend [23] The cloud-based [24] attack trees can be analyzed to forecast a group of attack libraries that are related to the attack graphs and represented by a collection of graphs. This method's manual implementation.

4.1 Network forensics' detection of intentions

K – Nearest Neighbor (KNN)

It is an approach for dividing a dataset into sets according to whether an attack was malicious or not, using the closest mean. Realistic forecasts may only be made with past training. In this case, the K-nearest neighbor technique was used to identify the K neighbors that needed to be classed. The likelihood of each attribute in the K neighbors was then determined as the attribute's weight.
Euclidean distance can be measured as

$$\text{Distance (A1 A2)} = \sqrt{\sum_{i=1}^{n}(a1i - a2\,i)\hat{}\ 2} \quad (1)$$

Support Vector Machine (SVM)

The SVM is a margin-based method of classification that maximizes class separation while adhering to the idea of structural risk reduction. As a result, SVM has excellent generalization abilities and is resistant to overfitting problems. SVM has the ability to do novelty detection. Each support vector's weight w from x can be calculated as follows:

$$X = \sum_{i=1}^{m} y^i * w^j * Sim\,(x, v^i) \quad (2)$$

Where, (x, v^i) is the similarity between x and v^i

Naïve Bayesian Classifier (NBC)

The assault packet type is identified using a Naive Bayesian technique by computing the posterior probability. Depending on the feature, Naive Bayesian is utilized to categorize various anomalies. Mathematical expressions of this sort of classifier are represented in the subsequent straightforward manner:

$$\text{Gain (D | B)} = \text{H(D)} - \sum_{v \in values} \frac{|D_v|}{D} \quad (3)$$

5. Proposed Work

A number of different approaches are combined in the ensemble learning approach in order to learn with greater precision than each of the systems. Similar to this, reached the executed outcomes by combining the output of the classifiers using various attack approaches. An ensemble technique brings a group of students together for data processing.

To combat this DDoS attack, here we generate a proposed framework The design of the suggested model is viewed in Figure 1.

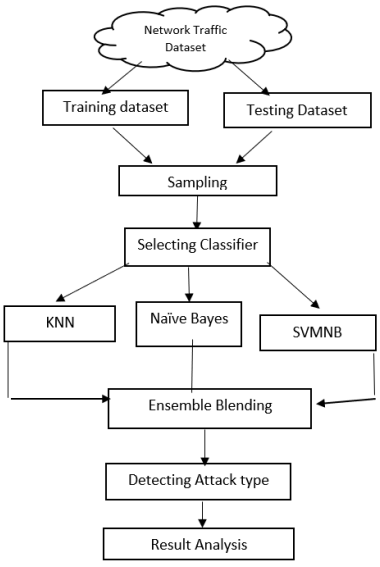

Figure 1. Proposed System Architecture

Figure 1 describes the proposed system. We filter the dataset using a number of cascade classifiers built on the characteristics of the machine learning techniques. The SVM algorithm can yield more accuracy when employed as a binary classifier since it can show greater accuracy on huge datasets. Figure 1 illustrates how the ensemble blending detects the attack based on the classifier. The Naive Bayesian classifier (NBC) and KNN are next trained to determine the type of attack using the filtered dataset from the earlier stage. The RMSE has also been used to detect DDoS assaults.

6. Experimental Analysis & Evaluation

Machine learning modules for Python were used to implement the model. The framework is split into two parts, the first of which involves training the SVM and the second of which involves training the Naive Bayesian classification method. Performance analysis was done using an approach based on percentages (70% training dataset and 30% testing dataset). A 10-fold cross-validation approach was deployed to assess the second phase of the Bays algorithm's effectiveness.
The true positive rate (TPR) and the false positive rate (FPR) for various threshold values can be estimated in order to assess our suggested methodology. We consider the following metrics while evaluating the framework's efficiency. The confusion matrix also known as the contingency table, is shown in Table 3 and contains a list of the classification findings. The number of persons who were reported as true positives when they weren't is shown in the True Positive box in the upper left corner. The number of samples that were falsely labeled as positive when they were actually negative is shown in the False-positive lower right cell. False-negative counts the number of people who were wrongly included in the sample as true.

Table 3. Confusion Matrix

	Condition Positive	Condition Negative
Predicted Condition Positive	True Positive	False Positive
Predicted Condition Negative	False Positive	True Positive

The receiver operating characteristic curve (or ROC curve) is obtained by manipulating the true positive rate for numerous cut points against the false-positive rate [25] Because any gain in sensitivity would be followed by a fall in specificity, ROC makes the trade-off between normal and attack evident. The test would be conducted with more accuracy the closer the curve maintains both the top and left borders of the ROC space. In Table 4 the accuracy analysis is described.

Table 4. Analysis of accuracy for Dataset 1 attack packet detection

Classification Algorithm	Recall	Precision	RMSE	Accuracy
KNN	0.945120	0.992026	3.886	0.96900
NBC	0.973680	0.947904	0.005	0.960619
SVM	0.963492	0.965959	0.513	0.964724
SVMNB	0.966094	0.985876	0.523	0.975883

In the suggested hybrid approach, we employ the SVM in the initial stage to more accurately identify attack packets in the network than other techniques. Table 4 displays the values for the metrics for Dataset 1, whereas Table 5 displays the performance metrics for Dataset 2. As seen in Tables 4 and 5, SVM provides superior accuracy than NBC and KNN algorithms.

For the experimentation dataset, certain occurrences are chosen. dataset1 and dataset 2 are two subsets. from the original dataset, and dataset 2.

Table 5. Analysis of accuracy for dataset 2 attack t packet detection

Classification Algorithm	Recall	Precision	RMSE	Accuracy
KNN	0.854452	0.853477	0.75	0.853964
NBC	0.979729	0.620542	0.003	0.759825
SVM	0.859259	0.661912	0.413	0.747784
SVMNB	0.904228	0.754500	0.423	0.82606

Figure 2 displays the proposed method's accuracy in identifying anomalies.

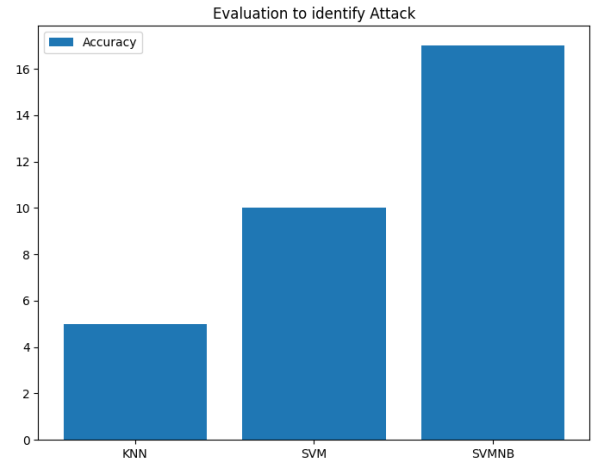

Figure 2. Accuracy in identifying attacks

7. Conclusion

Over the past year, numerous researchers have been working on anomaly identification in a variety of domains. In this research, we emphasized the importance of identifying anomalies in network packets. We developed a machine learning-based approach to network anomaly identification. We showed the current methods for identifying abnormal users in networks after sequentially defining anomaly detection. Crucially, our method shows promise for real-world Network Forensic applications since it can find abnormal patterns that the trained model missed. Detecting distributed denial-of-service attacks is the trickiest difficulty in cloud computing. DDoS attacks, which bombard the target with a lot of traffic, could bring the system down. Attackers carry out undetectable application-layer DDoS attacks by pretending to be actual customers.

In this work, we suggested an effective system for Attack packet detection and kind of attack detection simultaneously in order to address some issues from earlier research. This system was based on the provided feature list. The approach takes into account proactively capturing network packets and subjecting them to an attack packet classification algorithm based on SVM. Also, we thoroughly compared our suggested strategy to various approaches that are already in use using both real and simulated data.

References

[1] Aburomman AA, Reaz MBI. A survey of intrusion detection systems based onensemble and hybrid classifiers. Comput. Secur. 2017;65:135–52. doi:10.1016/j.cose.2016.11.004.

[2] Fernandes G, Rodrigues JJPC, Carvalho LF, Al-Muhtadi JF, Proença ML. A comprehensive survey on network anomaly

detection. Telecommun. Syst. 2019;70:447–89. doi:10.1007/s11235-018-0475-8.
[3] Liao H-J, Lin C-HR, Lin Y-C, Tung K-Y. Intrusion detection system: a comprehensive review. J. Netw. Comput. Appl. 2013;36(1):16–24. doi: 10.1016/j.jnca.2012.09.004.
[4] Patcha A, Park J-M. An overview of anomaly detection techniques: existing solutions and latest technological trends. Comput. Netw. 2007;51(12):3448–70. doi: 10.1016/j.comnet.2007.02.001
[5] Wu SX, Banzhaf W. The use of computational intelligence in intrusion detection systems: a review. Appl. Soft Comput. 2010;10(1):1–35. doi: 10.1016/j.asoc.2009.06.019.
[6] Mishra P, Varadharajan V, Tupakula U, Pilli ES. A detailed investigation and analysis of using machine learning techniques for intrusion detection. IEEE Commun.Surv. Tutor. 2019;21(1):686–728. doi:10.1109/COMST.2018.2847722.
[7] Moustafa N, Creech G, Slay J. Big data analytics for intrusion detection system: Statistical decision-making using finite Dirichlet mixture models. In: Data analytics and Decision Support for Cybersecurity. Springer; 2017. p. 127–56. doi:10.1007/978-3-319-59439-2_5.
[8] Fang W, Tan X, Wilbur D. Application of intrusion detection technology in network safety based on machine learning. Saf. Sci. 2020; 124:104604. doi: 10.1016/j.ssci.2020.104604
[9] Lopez-Martin M, Carro B, Sanchez-Esguevillas A. Application of deep reinforcement learning to intrusion detection for 18 computers & security 103 (2021) 102158 supervised problems. Expert Syst. Appl. 2020;141:112963. doi:10.1016/j.eswa.2019.112963.
[10] Li Y, Xia J, Zhang S, Yan J, Ai X, Dai K. An efficient intrusion detection system based on support vector machines and gradually feature removal method. Expert Syst. Appl. 2012;39(1):424–30. doi:10.1016/j.eswa.2011.07.032.
[11] Goseva-Popstojanova K, Anastasovski G, Dimitrijevikj A, Pantev R, Miller B. Characterization and classification of malicious web traffic. Comput. Secur.2014;42:92–115. doi:10.1016/j.cose.2014.01.006.
[12] Almashhdani AO, Kaiiali M, Carlin D, Sezer S. MaldomDetector: a system for detecting algorithmically generated domain names with machine learning. Comput. Secur. 2020;93:101787. doi:10.1016/j.cose.2020.101787.
[13] Ahmed M, Mahmood AN, Hu J. A survey of network anomaly detection techniques. J. Netw. Comput. Appl. 2016;60:19–31.
[14] Kumar G, Thakur K, Ayyagari MR. Mlesidss: machine learning-based ensembles for intrusion detection systems–areview. J. Supercomput. 2020. doi:10.1007/s11227-020-03196-z.
[15] Velliangiri S. A hybrid BGWO with KPCA for intrusion detection. J. Exp. Theor. Artif.Intell. 2020;32(1):165–80. doi:10.1080/0952813X.2019.1647558.
[16] G. Kim, S. Lee and S. Kim, A novel hybrid intrusion detection method integrating anomaly detection with misuse detection, Expert Systems with Applications. 41 (2014) 1690-1700.
[17] M. Panda and M. R. Patra, Network intrusion detection using nave Bayes, International Journal of Computer Science and Network Security. 7(12) (2007) 258- 263.
[18] KDD Cup'99 Data, http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html
[19] V.Bolon-Canedo, N.Sanchez-Marono, A.Alonso-Betanzos, "An ensemble of filters and classifiers for microarray data classification", journal of Pattern Recognition 45,2012, pp: 531– 539.
[20] P. Singh and V. Ranga, "Attack and intrusion detection in cloud computing using an ensemble learning approach," International Journal of Information Technology, vol. 13, no. 2, pp. 565–571, 2021.
[21] J. Shroff, R. Walambe, S. K. Singh, and K. Kotecha, "Enhanced security against volumetric DDoS attacks using adversarial machine learning," Wireless Communications and Mobile Computing, vol. 2022, Article ID 5757164, 10 pages, 2022.
[22] Sheeraz Ahmed, Zahoor Ali Khan, Syed Muhammad Mohsin, Shahid Latif, Sheraz Aslam, Hana Mujlid, Muhammad Adil, Zeeshan Najam, "Effective and Efficient DDoS Attack Detection Using Deep Learning Algorithm, Multi-Layer Perceptron", Future Internet, vol.15, no.2, pp.76, 2023
[23] Samantaray, M., Satapathy, S., Lenka, A. (2022). A Systematic Study on Network Attacks and Intrusion Detection System. In: Skala, V., Singh, T.P., Choudhury, T., Tomar, R., Abul Bashar, M. (eds) Machine Intelligence and Data Science Applications. Lecture Notes on Data Engineering and Communications Technologies, vol 132. Springer, Singapore. https://doi.org/10.1007/978-981-19-2347-0_16
[24] S. Potluri, M. Mangla, S. Satpathy and S. N. Mohanty, "Detection and Prevention Mechanisms for DDoS Attack in Cloud Computing Environment," 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), Kharagpur, India, 2020, pp. 1-6, doi: 10.1109/ICCCNT49239.2020.9225396.
[25] Ashraf Uddin M, Stranieri A, Gondal I, Balasubramanian V (2020) Dynamically recommending repositories for health data: a machine learning model. In: Proceedings of the Australasian Computer Science Week Multiconference. ACM. Pp 1–10. https://dl.acm.org/doi/abs/10.1145/3373017.3373041.