Survey Paper of Network Intrusion Detection based on Machine Learning Algorithm

Sanjeev Joshi [1], Prof. Suresh. S. Gawande[2], Prof. Satyarth Tiwari[3]
M. Tech. Scholar, Department of Electronics and Communication, Bhabha Engineering Research Institute, Bhopal[1]
Guide, Department of Electronics and Communication, Bhabha Engineering Research Institute, Bhopal[2]
Co-guide, Department of Electronics and Communication, Bhabha Engineering Research Institute, Bhopal[3]

Abstract- These days, intrusion detection system (IDS) is the most arising pattern in our general public. This basically screen network traffic and will alarm the organization chairman of any unordinary action. IDS System work by one or the other searching for marks of known assaults or deviations of typical movement. While there are a few detriments of IDS, for example, low recognition rate and high bogus caution rate. Intrusion detection is the process of analyzing the network packets to identify if the packet is legitimate or anomalous. The major challenges involved in this domain includes the huge volume of data for training and the fast and streaming data that is to be provided for the prediction process. Further, the intrinsic data imbalance contained in the domain presents more challenges to the intrusion detection model.

Keywords- IDS, Imbalance, Machine Learning

I. INTRODUCTION

One of the major concerns in computer system security is to prevent unautorized access to such system. So, to prevent such unauthorized access to the system, there is need for detection and prevention system. Such suspicious and unauthorized users generally named as "Intruders". Here they are stopped to admittance any part of computer system. The recognition process is used to determine if fewone will try to intrude in the target system, if it is successful, and also to find the activity logs [1]. Although you do not consider too confidential communications, strangers are unlikely to read emails, use computer systems to attack or disturb other systems, send fake emails or emails from a computer system, or check personal information on your computer system which contain information such as financial statements, account details, etc. Invaders are also called attackers, crackers or hackers. You may not be interested in the identity of the owner of the target system. They have always taken control of the computer system to launch attacks on other desired computer systems. Usually, attackers take control of target systems, such as government or financial systems, which allow them to hide it and their actual position, and thus to easily launch attacks. Once a computer system are connected to the Internet, it does not matter if few secret activities are performed by or simply playing games while chatting with friends and the system are also targeted. For intruders, you may be able to take care of all our activities on our system. It is possible to violate system information, reformat the hard disk and cause any kind of damage. To protect the system, it is too unfortunate that intruders constantly discover new vulnerabilities, also known as "loopholes" [2, 3]. These vulnerabilities must be exploited on computer system or system software. The difficulty in software, it is difficult to carefully test the security of computer systems. It is up to the user to get and install the file fixes, but to configure the software for safer operation. There are also software applications with predefined custom settings that allow other users to admittance the computer system unless the settings are changed to be safer. Examples include chat programs that allow external users to execute commands on their computer systems that allow fewone to implement destructive programs that run when the user is clicked. This probably would not allow a stranger to review important documents [4]. Likewise, it may be appropriate to keep tasks on the computer system confidential, whether it is monitoring our documents or running other applications. In addition, users must ensure that the information entered on the computer system remains intact and available when necessary. The possibility of intentional abuse of our computer system by Internet intruders could lead to security breaches [5]. There are even more risk that can be encountered, even if users are not connected to the Internet, such as hard disk errors, theft, power outages and so on. The bad news is that it may not be planned. The good news is that few common measures can be taken here to reduce the likelihood of being affected by the most common threats. Few of these steps help manage intentional and accidental risks. Before we learn what, we can do to protect our computer system or home network, let's examine few of these risks for convenience.

II. LITERATURE REVIEW

Abhinav Singhal et al. [1], this paper outlines an approach to build an Intrusion detection system for a

network interface device. This research work has developed a hybrid intrusion detection system which involves various machine learning techniques along with inference detection for a comparative analysis. It is explained in 2 phases: Training (Model Training and Inference Network Building) and Detection phase (Working phase). This aims to solve all the current real-life problem that exists in machine learning algorithms as machine learning techniques are stiff they have their respective classification region outside which they cease to work properly. This paper aims to provide the best working machine learning technique out of the many used. The machine learning techniques used in comparative analysis are Decision Tree, Naïve Bayes, K-Nearest Neighbors (KNN) and Support Vector Machines (SVM) along with NSLKDD dataset for testing and training of our Network Intrusion Detection Model. The accuracy recorded for Decision Tree,
Naïve Bayes, K-Nearest Neighbors (KNN) and Support Vector Machines (SVM) respectively when tested independently are 98.088%, 82.971%, 95.75%, 81.971% and when tested with inference detection model are 98.554%, 66.687%, 97.605%, 93.914%. Therefore, it can be concluded that our inference detection model helps in improving certain factors which are not detected using conventional machine learning techniques.

Lau lin et al. [2], in imbalanced organization traffic, pernicious digital assaults can regularly stow away in a lot of typical information. It displays a serious level of covertness and jumbling in the internet, making it hard for Network Intrusion Detection System (NIDS) to guarantee the precision and practicality of discovery. This paper explores AI and profound learning for interruption recognition in imbalanced organization traffic. It proposes an original Difficult Set Sampling Technique (DSSTE) calculation to handle the class awkwardness issue. To start with, utilize the Edited Nearest Neighbor (ENN) calculation to separate the imbalanced preparing set into the troublesome set and the simple set. Then, utilize the KMeans calculation to pack the larger part tests in the troublesome set to diminish the larger part. Zoom in and out the minority tests' persistent characteristics in the troublesome set integrate new examples to expand the minority number.

A. Raghavan et al. [3], successful and proficient malware recognition is at the bleeding edge of examination into building secure computerized frameworks. Similarly as with numerous different fields, malware location research has seen a sensational expansion in the utilization of AI calculations. One AI strategy that has been utilized broadly in the field of example matching overall—and malware identification specifically—is covered up Markov models (HMMs). Gee preparing depends on a slope climb, and thus we can frequently work on a model via preparing on numerous occasions with various beginning qualities. In this exploration, we think about helped HMMs (utilizing AdaBoost) to HMMs prepared with different arbitrary restarts, with regards to malware identification. These procedures are applied to an assortment of testing malware datasets. We observe that irregular restarts perform shockingly well in contrast with helping. Just in the most troublesome "cold beginning" situations (where preparing information is seriously restricted) does helping seem to offer adequate improvement to legitimize its higher computational expense in the scoring stage.

Zhiyou Zhang et al. [4], in this paper, aimed at detection of internal intruders in HIDS. Commonly used login ids and passwords may be shared along with co-workers for professional purposes, which can be tampered or used by the attackers as a means of intrusion into the system details. The user was monitored and System Calls (SC) was extracted and the habitual SC pattern based on the habits of the user was taken into account and the profile of the user was stabilized. The forensic technique and other data mining techniques were applied at SC level host IDS to spot the internal attacks. Along with the user login credentials the forensic technique was applied to investigate the computer usage fashion against the collected user profile pattern and thereby check the identity of the user.

Afreen Bhumgara et al. [5], in this paper, With the decision rate threshold of 0.9, the system was able to perform with an accuracy rate of 94%. Nokia Research Center researchers modeled HIDS for mobile devices. The limitation include that each protocol state consume resources for tracing and testing, and its inability to guess the attacks resembling benign protocol. Access control fills in as the cutting edge of resistance against interruptions, bolstering both confidentiality and integrity parameters. Intrusion detection is the process of progressively observing the events occurring in a PC or network, examining them for indications of conceivable episodes and often interdicting the unapproved access. A state transition diagram can be constructed for the sequence of events, but not for the complex forms and hence the attacks having complex behavior which cannot be modeled as the state transition diagram will go unnoticed by the system.

Ritumbhira Uikey et al. [6], in this paper, along with various protection mechanisms accompanied with mobiles they felt an urge for attack monitor methods as a second line of defense. The framework was designed, taking into consideration the privacy of the mobile user in creating the user profile. The framework had a major share with the host-based intrusion detection in line with the network-based detection system, as researchers felt that mobile requires the monitoring system at both ends. The

framework included data collection and IDS modules, the former entrusted with responsibility of monitoring the operating system activities, calculating the system measurements and the data collection at the application level and the later feeding on the collected and pre-processed data performs the actual intrusion detection.

Aditya Phadke et al. [7], targeted Advanced Persistent Threat attacks, by analyzing the 30 behavioral pattern of the host user through a 83-dimensional vector, each attribute representing one manner of the user. In order to form the database, they collected 8.7 million features from 4000 malicious and normal programs through the Virtual Machine (VM) environment. The system was designed in such way that frequency of occurrence of each behavior is calculated for each process. C4.5 decision tree was used to build a classifier for the collected information, and each new instance was analyzed against the tree to be segregated as malicious or normal instance. The model had a false positive rate of 5.8% and a false negative rate of 2.0%.

S. Sivantham et al. [8], in this paper, represented a novel HIDS aimed at discovering unknown malware codes. The collection of previous malware codes was taken as repository and each new sequence of behavior was compared with the repository to identify new malware code. Applied rule-based IDS to tackle the DDoS attacks in which the resources are made unavailable for the user when they are required. The utmost capacity of each of the middle-ware layer was fixed and set of rules were formed to detect the DDoS attacks. The system produced an alert when the count of the requests to a particular resource exceeded a particular threshold and concepts from learning automata were employed to avoid further attacks.

Azar Abid Salih et al. [9], in this paper, applied Machine Learning techniques, namely Naive Bayes, a Bayesian Network and Artificial Neural Network, to perform supervised learning of the malicious code. They gathered 323 features for training the classifiers. The detection rate for a specific set of worms was over 98%. Though HIDS were able to perform better by centering the user profile collected across, it prompted the challenge and there was a lack of information of the user Centralized reporting wasn't feasible with HIDS. They consumed the host details and resources which may violate the privacy issues of the user and may dispute with the already existing security protocols.

Lukman Hakim et al. [10], in this paper, centered over detecting intrusion in Routing Protocol for Low Power and Lossy Networks (RPL) attacks. The operations of the RPL were converted into finite state machines through which the network was monitored and any malicious activity was detected. The research was further extended by wherein the simulation trace files were used to model the finite state machines to observe the RPL attacks. The model was further converted into a set of rules to monitor the data transferred between the network nodes. The drawback of the work was that the True Positive Rate was even able to reach 100% but the False Positive Rate was not that low ranging between 0 to 6.78%. Over that it also had an overhead of 6.3% in terms of energy when compared to normal RPL network.

T. Sree Kala et al. [11], in this paper, designed an IDS for cloud environment based on state transition methodology. A Hidden Markov Model which builds a model where the behavior of user observed over long time period is marked as states and relevant transitions between them was used in their research. They developed three profiles, namely, low, middle and high, based on the matching of the probability of the user to the baseline profile. They give input as VM SC and bring about a diagram of state transition, wherein each transition represents the probability of targeting the next state and the probability of creating next system call. This model is tested against DoS attacks with 100% detection rate but with a poor false positive rate of 5.66%.

III. INTRUSION DETECTION SYSTEM

IDS detects malicious activity in computer systems and performs forensics after the attack is complete. Check network resources to detect intrusions and attacks that have not been blocked by preventive techniques (firewall, router packet filtering, proxy server). Intrusion is an attempt to compromise the confidentiality, integrity or availability of a system. Intrusion detection systems can be regarded as a rough analogy with true intruder detectors. Misuse based IDS (as shown in Figure 1) is designed to detect violations of predefined security policies. But things get complicated both with the introduction of possible harmful behaviors that cannot be predetermined [9-11]. An example would be a developer in a company that transfers large amounts of data in a short period of time. This may be a potential data leakage problem, but it may not be detected by the admittance policy because it is allowed to transfer files [12]. For this specific reason, the detection of statistical anomalies has been introduced, in which a profile of a user or a system is created and deviations from the profile are reported. While both kinds of systems are independently useful, a hybrid of the two can reduce but not eliminate the individual disadvantages. An important factor that defines the kind of implementation inherited from IDS is the source of audit data. The two primary sources are host-based protocols used by host-based IDSs and data packets that exist on a network that are exploited by network IDSs. Host protocols can be kernel logs, application logs, or device-related logs [11].

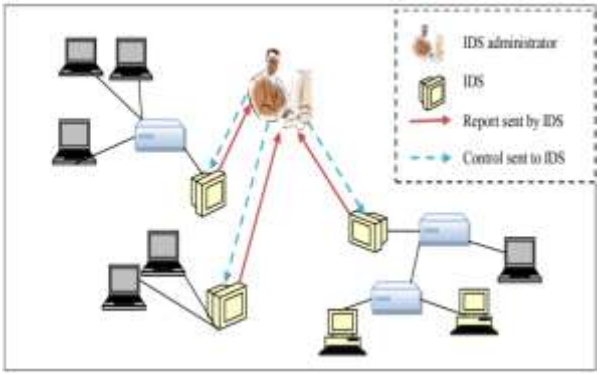

Figure 1: Intrusion Detection System

There are several problems with IDS based on host and network IDS. They include:

- Heterogeneous operating systems make the enumeration of system-specific detection parameters extremely long for any system.
- Increasing the number of critical nodes in the network increases performance.
- Performance degradation in the host system due to additional security activities, such as B. Registration.
- Difficulty in detecting attacks at the network level.
- Host with insufficient computing power to offer a complete host-based IDS.

In contrast, network-based intrusion detection systems can have a central system with a network connection to passively monitor network traffic. They have no impact on system performance and can easily detect network-level attacks when installed at the edge of the network. Network-based ID implementation is too simple [13]. Hostbased IDs in a critical performance-sensitive host network must be carefully selected so as not to unduly restrict the performance of each system.

IV. ML ALGORITHM

Supervised learning is two stage forms, in the initial step: a model is fabricate depicting a foreordained arrangement of information classes or ideas. The model developed by investigating database tuples portrayed by traits. Each tuple is expected to have a place with a predefined class, as dictated by one of the qualities, called to have a place with a reclassified class, as controlled by one of the traits called the class name characteristic. The information tuple are dissected to fabricate the model all things considered from the preparation dataset.

Learning

The main property of an ML is its capability to learn. Learning or preparing is a procedure by methods for which a neural system adjusts to a boost by making legitimate parameter modifications, bringing about the generation of wanted reaction. Learning in an ML is chiefly ordered into two classes as [9].

- Supervised learning
- Unsupervised learning

Supervised Learning

Regulated learning is two stage forms, in the initial step: a model is fabricated depicting a foreordained arrangement of information classes or ideas. The model developed by investigating database tuples portrayed by traits. Each tuple is expected to have a place with a predefined class, as dictated by one of the qualities, called to have a place with a reclassified class, as controlled by one of the traits called the class name characteristic. The information tuple are dissected to fabricate the model all things considered from the preparation dataset.

Unsupervised learning

It is the kind of learning in which the class mark of each preparation test isn't knows, and the number or set of classes to be scholarly may not be known ahead of time. The prerequisite for having a named reaction variable in preparing information from the administered learning system may not be fulfilled in a few circumstances.

Data mining field is a highly efficient techniques like association rule learning. Data mining performs the interesting machine-learning algorithms like inductive-rule learning with the construction of decision trees to development of large databases process. Data mining techniques are employed in large interesting organizations and data investigations. Many data mining approaches use classification related methods for identification of useful information from continuous data streams.

Nearest Neighbors Algorithm

The Nearest Neighbor (NN) rule differentiates the classification of unknown data point because of closest neighbor whose class is known. The nearest neighbor is calculated based on estimation of k that represents how many nearest neighbors are taken to characterize the data point class. It utilizes more than one closest neighbor to find out the class where the given data point belong termed as KNN. The data samples are required in memory at run time called as memory-based technique. The training points are allocated weights based on their distances from the sample data point. However, the computational complexity and memory requirements remained key issue. For addressing the memory utilization problem, size of data gets minimized. The repeated patterns without additional data are removed from the training data set.

Naive Bayes Classifier

Naive Bayes Classifier technique is functioned based on Bayesian theorem. The designed technique is used when dimensionality of input is high. Bayesian Classifier is used for computing the possible output depending on the input. It is feasible to add new raw data at runtime. A Naive Bayes classifier represents presence (or absence) of a

feature (attribute) of class that is unrelated to presence (or absence) of any other feature when class variable is known. Naïve Bayesian Classification Algorithm was introduced by Shinde S.B and Amrit Priyadarshi (2015) that denotes statistical method and supervised learning method for classification. Naive Bayesian Algorithm is used to predict the heart disease. Raw hospital dataset is employed. After that, the data gets preprocessed and transformed. Finally by using the designed data mining algorithm, heart disease was predicted and accuracy was computed.

Support Vector Machine

SVM are used in many applications like medical, military for classification purpose. SVM are employed for classification, regression or ranking function. SVM depends on statistical learning theory and structural risk minimization principal. SVM determines the location of decision boundaries called hyper plane for optimal separation of classes as described in figure 1.4. Margin maximization through creating largest distance between separating hyper plane and instances on either side are employed to minimize upper bound on expected generalization error. Classification accuracy of SVM not depends on dimension of classified entities. The data analysis in SVM is based on convex quadratic programming. It is expensive as quadratic programming methods need large matrix operations and time consuming numerical computations.

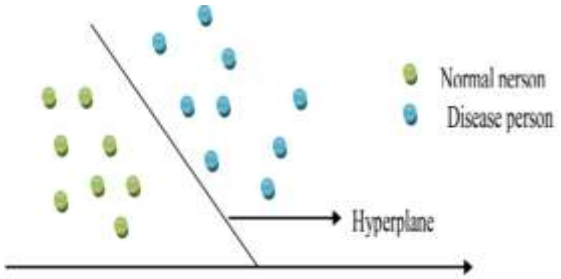

Figure 2: Support Vector Classification

V. METHODOLOGY

In distinction, network-based intrusion detection systems will have a central system with a network association to passively monitor network traffic. They need no impact on system performance and might simply observe network-level attacks once put in at the edge of the network. Network-based ID implementation is too easy. Hostbased IDs in an exceedingly important performance-sensitive host network should be rigorously designated therefore as not to unduly limit the performance of each system.

In this section, the brief explanation of overall process of proposed methodology is described.

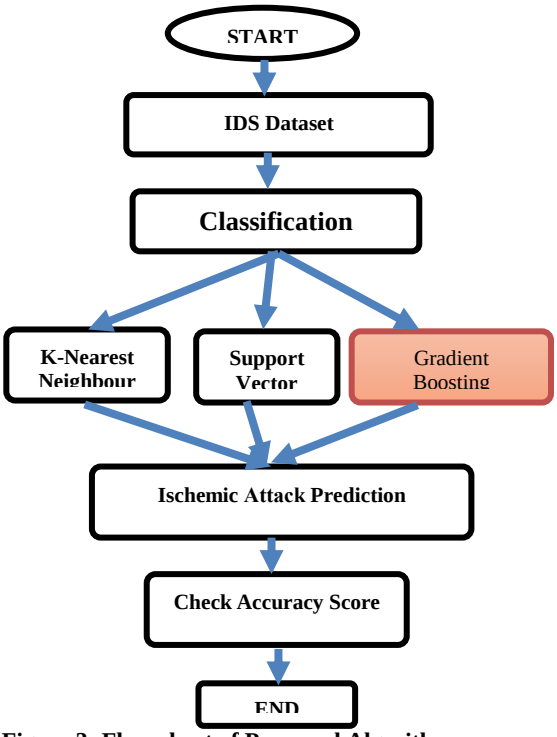

Figure 3: Flow chart of Proposed Algorithm

- Dataset was divided into two datasets (70%/30%, training/testing) to avoid any bias in training and testing. Of the data, 70% was used to train the ML model, and the remaining 30% was used for testing the performance of the proposed activity classification system.
- Precision provides a measure of how accurate your model is in predicting the actual positives out of the total positives predicted by your system.
- Recall provides the number of actual positives captured by our model by classifying these as true positive.
- F-measure can provide a balance between precision and recall, and it is preferred over accuracy where data is unbalanced.

VI. CONCLUSION

In the previous few decades, internet technology has extended its application space in many various domains in our life like banking operations, on-line auctions, electronic commerce applications, social networking, and on-line application / registration, etc. However, because of the weakness of computer systems' security, numerous electronic networks have typically been intruded by the hackers particularly with denial-of-service or distributed denial-of- service attacks. Anomaly detection principally depends on the illustration of the conventional behavior of the users, hosts and network connections. It is extremely laborious to notice abnormal requests within the network.

Therefore, machine learning algorithms are used as a versatile and powerful technique.

REFERENCES

[1] Abhinav Singhal, Akash Maan, Daksh Chaudhary and Dinesh Vishwakarma, "A Hybrid Machine Learning and Data Mining Based Approach to Network Intrusion Detection", Proceedings of the International Conference on Artificial Intelligence and Smart Systems, IEEE 2021.

[2] Lan Liu, Pengcheng Wang , Jun Lin, and Langzhou Liu, "Intrusion Detection of Imbalanced Network Traffic Based on Machine Learning and Deep Learning", IEEE Access 2020.

[3] A. Raghavan, F. D. Troia, and M. Stamp, ``Hidden Markov models with random restarts versus boosting for malware detection," *J. Comput. Virol. Hacking Techn.*, vol. 15, no. 2, pp. 97107, Jun. 2019.

[4] Zhiyou Zhang and Peishang Pan "A hybrid intrusion detection method based on improved fuzzy C-Means and SVM", IEEE International Conference on Communication Information System and Computer Engineer (CISCE), pp. no. 210-214, Haikou, China 2019.

[5] Afreen Bhumgara and Anand Pitale, "Detection of Network Intrusion Using Hybrid Intelligent System", IEEE International Conferences on Advances in Information Technology, pp. no. 167-172, Chikmagalur, India 2019.

[6] Ritumbhira Uikey and Dr. Manari Cyanchandani "Survey on Classification Techniques Applied to Intrusion Detection System and its Comparative Analysis", IEEE 4th International Conference on Communication $ Electronics System (ICCES), pp. no. 459-466, Coimbatore, India 2019.

[7] Aditya Phadke, Mohit Kulkarni, Pranav Bhawalkar and Rashmi Bhattad "A Review of Machine Learning Methodologies for Network Intrusion Detection", IEEE 3rd National Conference on Computing Methodologies and Communication (ICCMC), pp. no. 703-709, Erode, India 2019.

[8] S. Sivantham, R.Abirami and R.Gowsalya "Comapring in Anomaly Based Intrusion Detection System for Networks", IEEE International conference on Vision towards Emerging Trends in Communication and Networking (ViTECon), pp. no. 289-293, Coimbatore, India 2019.

[9] Azar Abid Salih and Maiwan Bahjat Abdulrazaq "Combining Best Features selection Using Three Classifiers in Intrusion Detection System", IEEE International Conference on Advanced science and Engineering (ICOASE), pp. no. 453-459, Zakho - Duhok, Iraq 2019.

[10] Lukman Hakim and Rahilla Fatma Novriandi "Influence Analysis of Feature Selection to Network Intrusion Detection System Performance Using NSL-KDD Dataset", IEEE International Conference on Computer Science, Information Technology, and Electrical Engineering (ICOMITEE), pp. no. 330-336, Jember, Indonesia 2019.

[11] T. Sree Kala and A. Christy, "An Intrusion Detection System Using Opposition Based Particle Swayam Optimization Algorithm and PNN", IEEE International Conference on Machine Learning, Big Data, Cloud and Parallel Computing, pp. no. 564-569, Coimbatore, India 2019.

[12] Xiaoyan Wang and Hanwen Wang "A High Performance Intrusion Detection Method Based on Combining Supervised and Unsupervised Learning", IEEE Smart World, Ubiquitous Intelligence $ Computing Advanced $ Trusted Computing, Scalable Computing, Internet of People and Smart City Innovations, pp. no. 889-897, Guangzhou, China 2018.

[13] P. Singh and M. Venkatesan, "Hybrid Approach for Intrusion Detection System", IEEE International Conference on Current Trends Towards Converging Technologies (ICCTCT), pp. no. 654-659, Coimbatore, India 2018.

[14] M. Tavallaee, E. Bagheri, W, Lu and A. Ghorbani, "A Detailed Analysis of the KDD CUP 99 dataset", IEEE Symposium on Computational Intelligence for Security and Defense Applications, pp. no. 892-899, Ottawa, India 2018.

[15] Karuna S. Bhosale and Assoc. prof. Maria, "Data Mining Based Advanced Algorithm for Intrusion Detection in Communication Networks", IEEE International Conference on Computational Techniques, Electronics & Mechanical System (CTEMS), Belgaum, India 2018.