Industrial Engineering Journal
ISSN: 0970-2555
Volume : 51, Issue 8, August : 2022

ADVANCED INTRUSION DETECTION SYSTEMS LEVERAGING MACHINE LEARNING TECHNIQUES

[#1]**KANDUKURI CHANDRASENA CHARY,** ***Research Scholar,***
[#2]**Dr. SATISH NARAYAN GUJAR,** ***Supervisor***
Department of Computer Science and Engineering,
School of Engineering and Technology,
UNIVERSITY OF TECHNOLOGY, JAIPUR, RAJASTHAN.

ABSTRACT: The internet has had a huge impact on the world. In reality, it makes it easier to keep communication with people in one's social networks and contact them as needed. The transmission of personal and professional information carries several risks that affect both individuals and enterprises. The security of our information is constantly jeopardized due to the essential role that the internet plays in our everyday lives. Implementing an intrusion detection system (IDS) is required to protect internet users from potentially harmful network attacks. An intrusion detection system (IDS) is a device that detects and categorizes unauthorized network activity. Notifications are sent when an event occurs. This examination will focus mostly on the NB, SVM, and KNN algorithms, as well as machine learning. Initially, these procedures will be implemented to identify the ideal level of accuracy utilizing the USNW NB 15 DATASET. The ideal technique is implemented in the next phase of our database processing, depending on the outcome of the first phase. To assess the model's functionality, we will run it on two different datasets. The effectiveness of the suggested technique is examined using the NSL-KDD and UNSW-NB15 datasets.
Index terms: machine learning, algorithms NB, SVM and KNN

1. INTRODUCTION

Today, there are far more computers than in the past. Many people believe that their desktop computers, laptops, tablets, and smartphones are essential parts of their everyday lives. The major goal is to ensure the security of data collected online. Individuals use intrusion detection systems (IDS) to ensure the security of data transmitted across networks.

An intrusion detection system (IDS) is a hardware or software component that monitors network or system behavior for malicious activities or policy breaches. Afterwards, the management system creates reports. An intrusion monitoring system is unquestionably important, which is why creating a high-quality model is critical. In this field, machine learning has proven useful by effectively identifying any unexpected occurrences that occur within the system's processes. To perform properly, an intrusion detection system (IDS) must be highly proficient in recognizing harmful network traffic. Accuracy has a considerable impact on categorization systems' efficacy. This paper describes a novel technique to using an intrusion detection system (IDS) that improves the accuracy and usefulness of detecting hostile network activity. The dataset will be trained using three different classification algorithms in the first phase. Following this part, the dataset will be trained with more accurate tree techniques. The final section will provide a list of concerns that should be investigated further in the future.

2. DATASET DESCRIPTIONS

Researchers can access a large quantity of public records using the internet. After reading the literature, it was clear that a large percentage of the publications were written decades ago and would be of little use in finding new threats. The

KDD98 and KDD'99 collections are among those that could be used.

According to Slay N. M., the UNSW-NB15 dataset was developed in 2015 at the Australian Centre for Cybersecurity's cyber range facility. The data can be downloaded in CSV or other formats. We chose not to use the original CSV files because they contained a large number of records (about 2.5 million) and were divided into four separate files.

The updated CSV files, which contain 82,332 items and 175,341 transactions, provide the study data for our training and testing sets. The collection contains 47 features. These features include numeric, nominal, and categorical data kinds. This binary dataset comprises many different classes that have been correctly assigned to their corresponding categories. Each assault was assigned to the training and testing sets, as shown in Table 1.

Table 1 is a thorough inventory of all UNSW-NB15 files.

Table1.The UNSW-NB15 Datasets collection.

DATASET	CLASS	TRAIN-SET	TEST-SET
UNSW-NB15	NORMAL	56 000	37 000
	GENERIC	40 000	18 871
	EXPLOITS	33 393	11 132
	FUZZERS	18 184	6 062
	DOS	12 264	4 089
	RECONNAISS ANCE	10 491	3 496
	ANALYSIS	2 000	677
	BACKDOOR	1 746	583
	SHELLCODE	1 133	378
	WORMS	130	44
	TOTAL	175 341	82 332

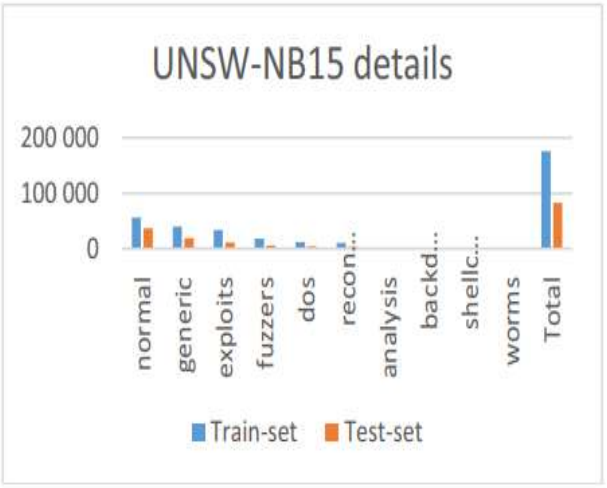

Fig.1. Specifics of the UNSW-NB15

NSL-KDD

The KDD'99 dataset is unreliable for detecting network flaws due to its age and the inclusion of duplicate data. The problem has been fixed in the upgraded version of KDD'99, known as NSL-KDD. The NSL-KDD training set has 125,973 data points, whereas the testing set has 22,544 data points. The dataset includes 41 factors. They are individually identified and may have a numeric, binary, or nominal value. The information includes four types of attacks: unauthorized local access (U2R), denial of service (DoS), illegal remote access (R2L), and probing. There is also a group that everyone in the class considers to be normal. Each assault was assigned to either the training or testing set, as shown in Table 2.

Table2.The datasets of NSL-KDD.

Dataset	Class	Train-set	Test-set
NSL-KDD	normal	67 343	9,711
	dos	45 927	7 458
	probe	11 656	2 421
	r2l	995	2 754
	u2r	52	200
	Total	125 973	22 544

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 51, Issue 8, August : 2022

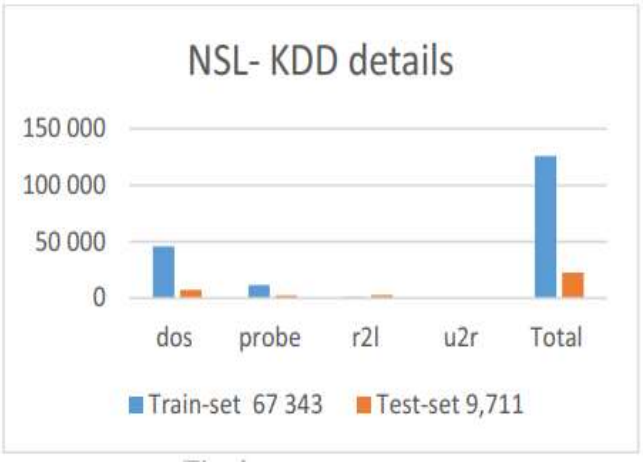

Fig.2.Specifics regarding NSL-KDD

3. RELATEDWORKS

Several approaches have been used to evaluate breach detection systems. The traditional methods of data management are facing issues in the setting of "Big Data." This has resulted in the formation of a significant number of researchers working on a big data-based intrusion detection system that is both quick and precise. This section focused on using machine learning to detect huge data intrusions. Ferhat and his coworkers implemented cluster machine learning. The authors used the k-Means technique from Spark's machine learning toolkit to distinguish network intrusions from normal data flow. The proposed method combines training and testing for the KDD Cup 1999. The authors of this concept identified important qualities without using a feature selection technique.

According to Peng et al., intruder detection systems should use PCA and Mini Batch K-means clustering. The dimensionality of a dataset is reduced using principal component analysis. Next, the data is grouped using the micro batch K-means++ algorithm. The proposed model was assessed using all of the KDDCup1999 data.

Peng et al. used machine learning to organize their data. The authors offer a cloud-compatible intrusion detection system (IDS). This IDS uses decision trees to efficiently evaluate massive amounts of data. Researchers recommend using preprocessing to identify phrases in the dataset. Following that, the data is standardized to ensure the accuracy of both the input and the identification. We compared K-Nearest Neighbors (KNN) and Naïve Bayesian algorithms for Intrusion Detection Systems (IDS) to the decision tree technique. The method was shown to be effective in experiments using the KDDCUP99 dataset.

Belouch et al. tested the performance of SVM, Naïve Bayes, Decision Tree, and Random Forest classification algorithms in Intrusion Detection Systems using Apache Spark. Use the UNSW-NB15 dataset to determine accuracy, prediction time, and training time.

Manzoor and Morgan's real-time intrusion detection system uses Apache Storm and SVM. The authors used C-SVM and lib SVM to categorize intrusion detection. KDD 99 was used to evaluate the proposed methodology. Furthermore, numerous research used attribute selection approaches. Randhika and Vimalkumar showed a complicated big data strategy for detecting smart grid flaws. The system uses methods such as Naïve Bayes, SVMs, DTs, Random Forests, and neural networks. The PCA feature selection approach is also used. Principal Component Analysis (PCA) reduces the number of dimensions, whereas correlation is used to select features. The proposed approach would improve classification accuracy while shortening the time required for attack prediction. The Synchro Phasor dataset was created to have both training and testing goals. The approach is evaluated using recall, accuracy, specificity, and FPR.

Dahiya and Srivastava created a spark-based intrusion detection system that is both precise and efficient. The suggested system uses seven categorization algorithms, including Naïve Bayes, REP TREE, Random Tree, Random Forest, Random Committee, Bagging, and Randomizable Filtered. The features were reduced using linear discriminant analysis (LDA) and canonical correlation analysis (CCA). Each classifier was tested on two UNSW-NB 15 datasets in the proposed study. The experiments carried out using the proposed method revealed that the LDA + random tree strategy was more efficient and

effective. The AUROC for dataset 1 is 99.1, and for dataset 2, it is 97.4. The AUROC score for our model was 99.55. This accelerates and improves our workflow.

Wang et al. built Hongbing SP-PCA-SVM with Spark's aid. The approach parallelizes SVM and PCA. PCA is a technique for extracting characteristics from data and analyzing them for aggregation in order to reduce the number of dimensions. KDD99 was used to train and assess the suggested approach.

Giura and Wang (2012) provide an extensible paradigm and approach for identifying Advanced Persistent Threats (APTs) within any company. Concepts are organized in the assault pyramid in a similar way to an attack tree. The assault is supposed to take place at the pyramid's summit, but nearby planes can also be used. "Planes" or "habitats" refer to the physical world, network, human, and program. The pyramid assault path begins with information collection, then moves on to operations, exploitation, delivery, reconnaissance, and exfiltration before reaching the target. The broader attack pyramid reveals entry points and weak points leading to the target over multiple layers.

Mirsky et al. (2018) created a network intrusion detection system (NIDS) that comes pre-configured for deployment. This Network Intrusion Detection System (NIDS) may quickly detect online local network intrusions that aren't being monitored. The authors are aware that Kitsune improves upon prior techniques that used artificial neural networks (ANNs) to detect network disruptions. Numerous limits must be considered. An annotated dataset is necessary for a successful launch. Furthermore, the bulk of supervised learning algorithms require a fast CPU to train the model, and they only address known security flaws. The newly trained models cannot be used until the company's network intrusion monitoring system has been upgraded. These issues encourage Kitsune to evolve into an open, lightweight ANN-based NIDS that uses machine learning. Currently, the goal is to finish work on network devices. The Kitsune framework consists of a feature extractor, a feature mapping tool, as well as a packet capture and parser. Feature extractors capture data, such as channel information. The feature mapper provides the anomaly detector with a more comprehensive set of features. Finally, the packet capturer and processor handle data, including packet meta information. KitNET, Kitsune's principal problem-solving instrument, uses autoencoders to discern between normal and aberrant traffic patterns. The online feature extraction method is both fast and exact, as it monitors and analyzes all network channel properties. This approach uses a hash table to store damped incremental statistics.

Milajerdi et al. (2019) proposed HOLMES for recognizing the behaviors of APT campaigns. They provide a detection signal by combining audit data from Windows ETW and Linux audited hosts with warnings from intrusion detection systems (IDSs). Their tactics often focus on the links between cyberattacks and problematic information processes, such as files and programs. The system's design permits alerts that resemble the "death chain," a series of Advanced Persistent Threat (APT) attacks. The cyber death chain serves as an example of the APT lifetime. Reconnaissance begins the lifespan, which ends with data extraction.

Nagaraja created a way to detect P2P botnets. Initially, botmasters shared network control with P2P connections to help manage interruptions while remaining anonymous. To leverage on this feature, the researchers apply Markovian diffusion processes to graph model network flows. P2P botnets differ from conventional network exchanges because of their graph topology. The goal is to identify communication and transportation patterns.

According to Yedidia et al. (2003), Oprea et al. (2015) used belief propagation to detect advanced persistent threats (APTs) early on. Advanced Persistent Threats (APTs) are cyberattacks that are persistent and target a specific organization. It uses modern software and runs silently in order to absorb. We identified Advanced Persistent Threats (APTs) by looking at common infection

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 51, Issue 8, August : 2022

patterns. The attacker connects to command and control (C&C) servers, bypasses firewalls using HTTP(S) protocols, quickly logs in to multiple domains while concealing their identity via redirection, and uses attack-related domains that share hosts, location/IP space, and access time. The goal was to discover atypical host communications that could indicate an Advanced Persistent Threat.

4. CLASSIFICATION ALGORITHMS

The obtained data can be utilized for a variety of purposes, including market research, monitoring production operations, and going forward with scientific inquiries. In the field of machine learning, classification algorithms are extremely significant. Their job is to organize unidentified data into distinct groups. The study employed the following methods:

Machine learning methods are referred to as support vector machines (SVMs). People believe that the Support Vector Machine (SVM) method is one of the most effective ways to use machine learning to classify data. It provides a quick and straightforward approach to make predictions. The categorization approach includes leveraging support vectors from a data source to categorize data points according to a hyperplane.

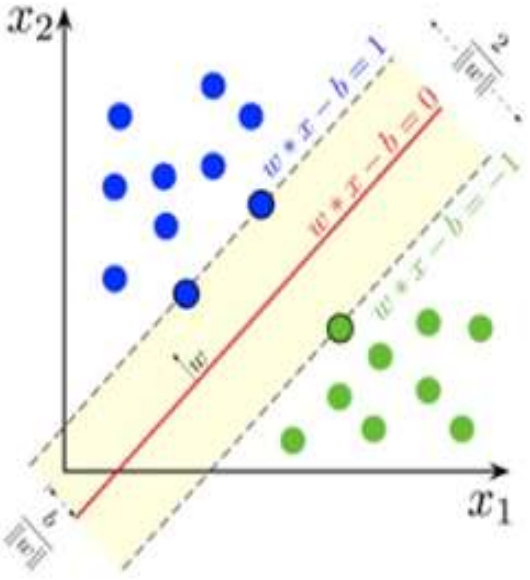

Fig.3.SVM

The K-Nearest Neighbor (KNN) approach is a safe way to partition data into multiple groups. One intriguing aspect of this is that it may be utilized for both regression and classification tasks.

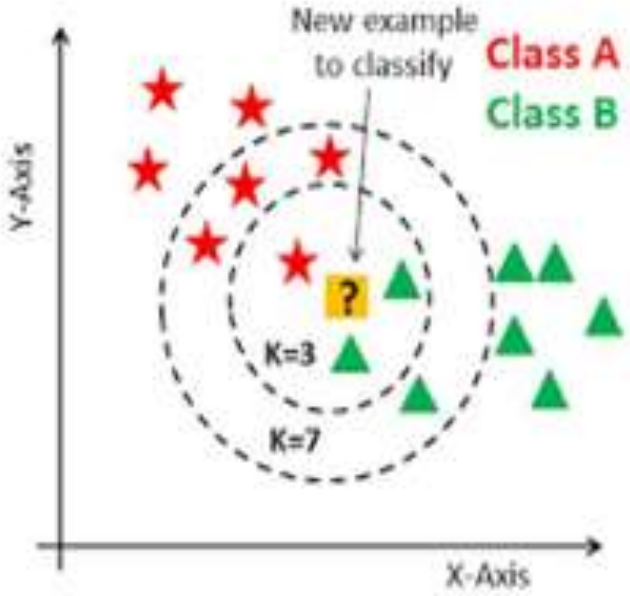

Fig.4.Specifics regarding NSL-KDD

Naïve Bayes (NB) classifies data by determining the likelihood of each model belonging to a specific category. The fundamental concepts underlying the Bayes theorem are presented below. It stems from the belief that changing the values of attributes in the same class does not affect the values of other attributes. This theory is known as "conditional independence of class."

$$P(H/X) = P(X/H).P(H)/P(X) \quad (1)$$

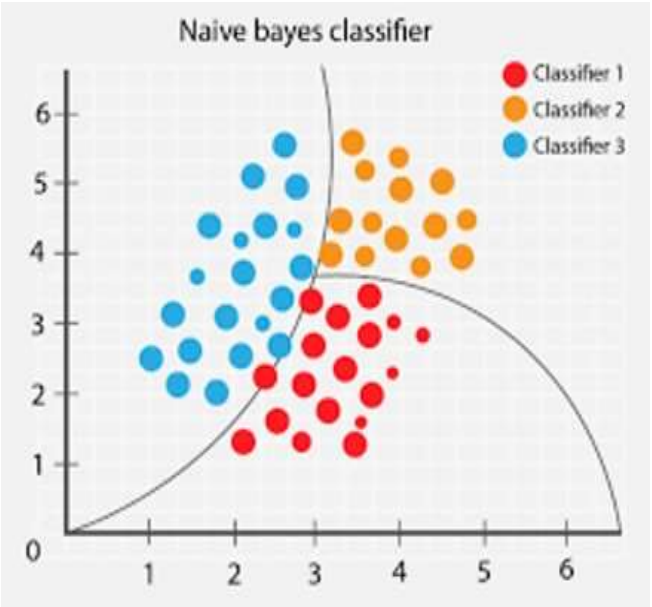

Fig.5.NaïveBayes

6. METHODOLOGY

Industrial Engineering Journal
ISSN: 0970-2555
Volume : 51, Issue 8, August : 2022

We compared the accuracy of SVM, KNN, and Naïve Bayes classification algorithms on a specific dataset. Using the class attribute, we partition the raw dataset of 19 different types of attacks into five groups. Normal, Dos, Probe, R2L, and U2R are the groups that were assigned. Points of Data

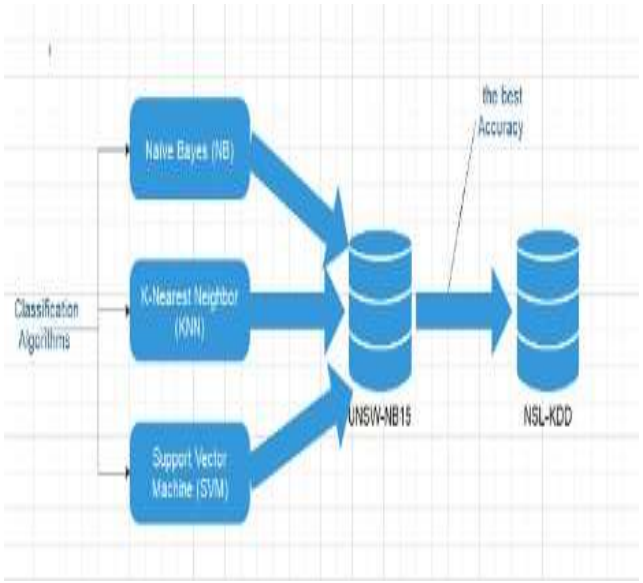

Fig.6.Testing procedures and procedures

Table3.Attacks' accuracy rate, as measured by the UNSW-NB15

	Accuracy
KNN (k=3)	93.3333%
NB	95.55555%
SVM	97.77777%

Fig. 7. A comparison of the performance of different classifiers on the UNSW-NB15

Table4.(NSL-KDD) measures the accuracy rate of attacks.

	Accuracy Rate on NSL-KDD
SVM	97,29
Naïve Bayes	67,26

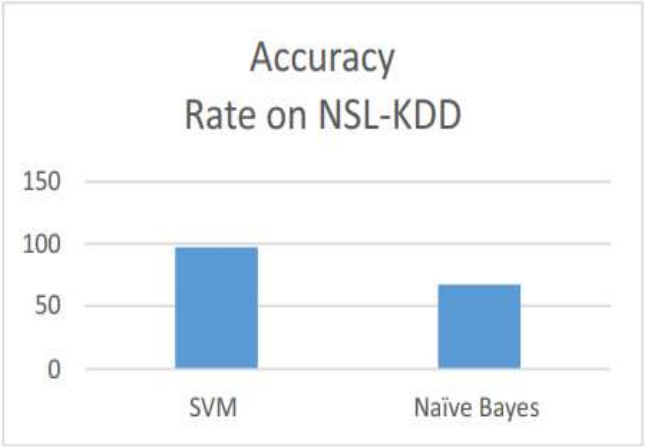

Fig. 8. A comparison of the performance of different classifiers on NSLKDD

There were significant variances in the sizes and types of SVM algorithms, yet when employed on a similar sample, they consistently demonstrated higher accuracy.

7. CONCLUSION

The original information in this search was processed using three methods: Support Vector Machines (SVM), Naive Bayes (NB), and K-Nearest Neighbors (KNN). Each employs a three-neighborhood system. The KNN technique is abandoned after producing poor results, and the two other algorithms are utilized to manage the secondary database. The Support Vector Machine (SVM) always works effectively, regardless of the threats it faces or the size of the database. The following are enhancements that will make this model perform better. A defense system will also be created around it, with real-time testing taking place.

REFERENCES

1. Tchakoucht TA, Ezziyyani M. Building a fast intrusion detection system for high-speed-

networks: probe and DoS attacks detection. Procedia Comput Sci. 2018;127:521–30.

2. M. Belouch, S. El Hadaj, and M. Idhammad. A two-stage classifier approach using reptree algorithm for network intrusion detection. International Journal of Advanced Computer Science and Applications, 8(6), pp.389- 394 (2017)
3. Zuech R, Khoshgoftaar TM, Wald R. Intrusion detection and big heterogeneous data: a survey. J Big Data. 2015;2:3.
4. M. Belouch, S. El Hadaj, & M. Idhammad. Performance evaluation of intrusion detection based on machine learning using Apache Spark. Procedia Computer Science, 127, 1-6,(2018).
5. Sahasrabuddhe A, et al. Survey on intrusion detection system using data mining techniques. Int Res J Eng Technol. 2017;4(5):1780–4.
6. N. Moustafa, N. (2017). Designing an online and reliable statistical anomaly detection framework for dealing with large high-speed network traffic (Doctoral dissertation, University of New South Wales, Canberra, Australia). (2017)
7. Dali L, et al. A survey of intrusion detection system. In: 2nd world symposium on web applications and networking (WSWAN). Piscataway: IEEE; 2015. p. 1–6.
8. W. Richert, L. P. Coelho, “Building Machine Learning Systems with Python”, Packt Publishing Ltd., ISBN 978-1-78216-140-0
9. M. Bkassiny, Y. Li, and S. K. Jayaweera, “A survey on machine learning techniques in cognitive radios,” IEEE Communications Surveys & Tutorials, vol. 15, no. 3, pp. 1136–1159, 2012.
10. Scarfone K, Mell P. Guide to intrusion detection and prevention systems (idps). NIST Spec Publ. 2007;2007(800):94. 6. Debar H. An introduction to intrusion-detection systems. In: Proceedings of Connect, 2000. 2000.
11. A. Iftikhar, M. Basheri, M. Javed Iqbal, A. Raheem; “Performance Comparison of Support Vector Machine, Random Forest, and Extreme Learning Machine for Intrusion Detection”, IEEE ACCESS, Survivability Strategies for Emerging Wireless Networks, 6 ,pp.33789-33795, (2018).