Machine Learning based Intrusion Detection System using Feature Engineering and Data Mining Techniques

Mahpara Tanveer
Mehran University of Engineering and Technology,
Jamshoro,Pakistan
mahparaaslam@yahoo.com

Tanveer Hassan
Mehran University of Engineering and Technology,
Jamshoro,Pakistan
tanveerhassan880@hotmail.com

, Farzana Rauf Abro
Mehran University of Engineering and Technology,
Jamshoro,Pakistan
farzana.abro@faculty.muet.edu.pk

Faheem Yar Khuhawar
Mehran University of Engineering and Technology,
Jamshoro,Pakistan
faheem.khuhawar@faculty.muet.edu.pk

***Abstract*— In today's world, where advancement in technology and Internet is flourishing by leaps and bounds, it is also getting common amongst people of every age and category which includes the mischievous ones too with wrong intension of attacking the system. It is extremely important to build a system with the best security measures. Intrusion Detection System (IDS) is the most effective way to secure any system and to detect any doubtful activity. IDS is commonly available everywhere, with that being said the false alarm rate which effects the accuracy of an IDS, is still a major concern. In this work, experiments are carried out to tackle threats using various Machine Learning (ML) algorithms, and their outcome is analyzed. The results suggest an improved accuracy if ML based IDS is employed using feature engineering along with data mining techniques**

Keywords*— *Intrusion Detection System, Machine Learning, Feature Extraction, Data Mining

I. INTRODUCTION

In today's modern era, cyber security is of importance while designing any network. Internet is a great source of knowledge of everything good, however it has also given an equal opportunity to the criminal minded people to learn several ways to harm any valuable data or system [1]. The heterogeneous nature of networks along with the complex underlying networking has made cyber security a lot more challenging than ever before.

Intrusion is the set of attempts that are made to compromise the confidentiality, integrity and availability of resources of a system at any computing platform. Anti-phishing working group has released a report which states about 227,000 detections of malware occur on daily basis, which are also linked to more than 20 million new detection of malware attacks daily [2].

IDS is the combination of software and hardware, designed to monitor traffic for any malicious activity and alarm the system or admin. To detect this malicious activity, IDS can either use database of signatures that contain pattern of malicious activities to be matched [3]. A drawback of Signature based IDS is if the attacker's signature is new, it would not be able to detect it. In contrast, Anomaly based IDS release the alarm if the traffic behavior is different from normal traffic pattern. A drawback of Anomaly based IDS is that it requires extensive training and are expensive computationally. Accuracy remains the main issue in both, in this experiment we have come up with some techniques which can help in achieving higher accuracy. IDS can be classified into different types such as Network based IDS (NIDS) [4] and Host based IDS (HIDS) [3]. Regardless the type, there are four main steps to design an IDS as shown in Fig. 1.

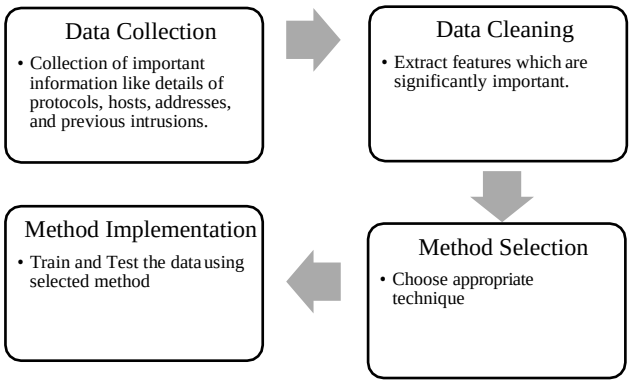


Fig. 1. Steps involved to design an IDS

II. DATASET DESCRIPTION

A. KDD-99 Cup Dataset

In 1988, at Lincoln labs, the TCP dumped data was captured for 7 weeks which resulted in 4.8 million records [2]. This dataset is multi-variant and is widely used to train algorithms. Attributes used in this dataset are integer and categorial in nature. It contains 42 attributes and 4 different patterns of attacks plus the normal traffic pattern. These 4 different attack patterns are as follows,

- Denial of Service (DoS): This category aims to interrupt and freeze the normal functioning of the system [5]. Attack types can be Smurf, Pod.
- Probing: This type of attack aims to gather information about the target / victim and use vulnerabilities like backdoor, poor password, open

ISBN-978-969-23372-1-2

ports to harm the system [6]. For example, port scanning.

- User to Root (U2R): The aim is to gain access to the root from a normal user login [7], [8]. For example, Buffer overflow attack
- Remote to Local (R2L): The aim is to gain unauthorized local access to the victim's machine from a remote machine. For example, like guessing password to gain access.

B. *NSL_KDD Dataset*

KDD Cup 99 dataset has quite a lot of issues such as redundancy, skewed and non-stationary nature. Researchers proposed a new modified version of this dataset which is called NSL-KDD dataset. It consists of only selected records to deal with the drawbacks of KDD Cup 99 [9]. However, the dataset still faces some problems when it comes to recognize attacks having low footprint [10]. Fig. 2 shows the data distribution of NSL-KDD dataset, where there are non-overlapping samples in Training and Testing Set

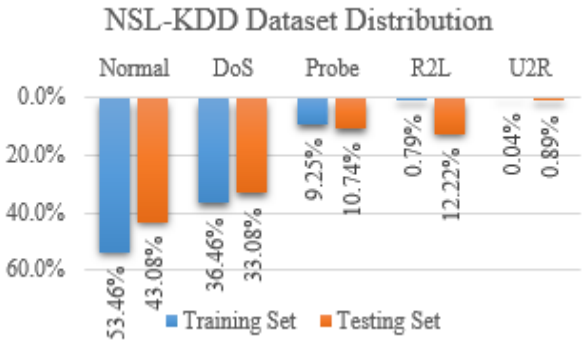


Fig. 2. Distribution of Attacks in Training and Testing Set of NSL-KDD Dataset

III. METHODOLOGY

IDS requires a constant human supervision in order to work perfectly, however rapid growth in technology and increasing rate of zero-day attacks (new attacks) has made it difficult for humans to take prompt and accurate actions [11].

In this work, we have utilized various Machine Learning (ML) algorithms to train and test over benchmarked datasets. The performance of it is compared in mitigating all four types of attacks. Several researchers [12], [13], [14], have stated that most of the published work experimented using only 10% of training set as the full dataset has 4,898,431 records in total with 22 attack types that are unlabeled. The labelled dataset is only 10% of the full dataset.

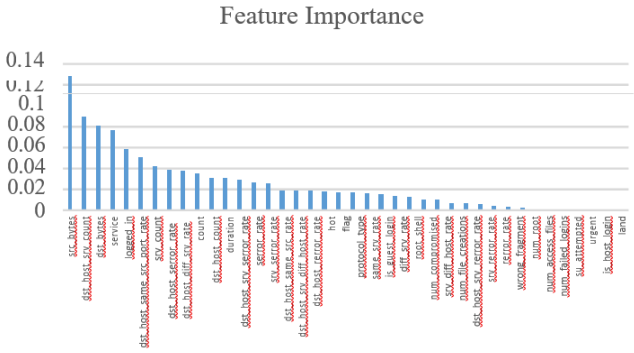
A. *Data collection, labelling and analysis*

A relevant data is collected from raw NSL-KDD dataset [15]. Later, the data is labelled, and attack type is distributed properly along with the analysis of dimension for training and testing set.

B. *Feature Engineering using Data Mining Technique*

All categorical attributes are converted to binary via one- hot encoding and label encoding technique. Later, Recursive Feature Elimination (RFE) method is used to eliminate features having least impact on prediction. In other words, features important to recognize the attacker are selected. RFE method has valuable impact on the performance of ML based algorithms.

Data mining techniques involve generation of new information from a huge bank of information. In this case, it is used to build 4 different datasets based on 4 different attack types. This process involves careful selection of features as wrong selection may make the model take longer time to predict [16]. Fig. 3 shows the importance of each



feature in descending order.

Fig. 3. Distribution of Importance of Features in Descending order

C. *Machine Learning*

Various best supervised ML algorithms are explored as follows.

- Random Forest: In this, the number of trees is directly proportional to the results it can achieve. The greater the number of trees, the more accurate the results gets. The method involves recognizing the root node and later splitting the nodes connected to the features randomly. Random Forest is created and utilized to predict the results [17].
- K-Nearest Neighbor: This is a non-parametric method used to solve classification problem. There are no assumptions for data distribution which is underlying. It utilizes the "K" number of neighbors as core deciding factor by calculating the distance between input and test data according to the following formula,

$$d = \frac{\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}}{n} \quad (1)$$

$$d = |x - y| = \sqrt{\sum_{i=1} |x_i - y_i|^2}$$

- Support Vector Machine (SVM): The trained models of SVM can efficiently perform both linear and non-linear classification. It is computationally practical and is used in Network based IDS [18].

- Ensemble Learning: It is utilized to combine the results of all the algorithms for the purpose of improving the outcome. This method is similar to Ada-Boost based algorithm, where weak classifiers are combined to build a strong classifier. The outcome suggest that ensemble learning predicts better than Ada-Boost [19].
- Artificial Neural Network: A model based on connected units or nodes that process information similar to the biological nervous system in brain.

D. Metrics Analysis

- The predictions or outcome of ML based IDS are represented in the form of confusion matrix or error matrix, which has 4 parameters, described as follows.
- True Positive (TP): It occurs when there is an attack, and the model recognizes the attack.
- True Negative (TN): It occurs when there is no suspicious activity or attack, and the model predicts nothing.
- False Positive (FP): It occurs when there is no cyber-attack, and the model predicts there is one.
- False Negative (FN): It occurs when there is a cyber-attack, and the model predicts nothing.

E. Performance Measurement

Performance of deployed ML algorithms is measured via parameters such as Accuracy, Precision, Recall and F1-score.

- Accuracy is measure of correctly predicted outcomes and is represented as the ratio of correct predictions to total number of predictions.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (2)$$

- *Precision* is defined as the count of TP over the count of both positives (TP+FP).

$$Precision = \frac{TP}{TP + FP} \quad (3)$$

- Recall represents the ratio of correctly predicted results to the correctly predicted outcome (TP) and wrongly predicted negatives (FN).

$$Recall = \frac{TP}{TP + FN} \quad (4)$$

IV. EXPERIMENTS

Random Forest based IDS was trained using data mining techniques to utilize 13 best features of each category. We visualize the performance of Random Forest based IDS using confusion matrix as shown in TABLE I.

TABLE I. CONFUSION MATRIX OF RANDOM FOREST ALGORITHM AGAINST DOS ATTACK USING TESTING SET

		ACTUAL CLASS	
		Attack	***No Attack***
PREDICTED CLASS	***Attack***	9671	40
	No Attack	44	7416

The rows in the confusion matrix represent the prediction and the column represent the actual class. The results from confusion matrix suggest that TP = 9671 and TN = 7416, which means the model correctly predict DoS attack. Similarly, FP = 40 and FN = 44 shows that the model inaccurately predicted the DoS attack

Using the results from confusion matrix, the performance parameter "Accuracy" is calculated to be around 99%. The summarized results of "Accuracy" for various ML based IDS are shown in TABLE III. The high accuracy rate shows that the model is robust against False Alarms. To further explore the extend of IDS, Artificial Neural Network (ANN) based IDS was designed and developed. Similar, procedure was adopted to assess its performance. The results of ANN based IDS against training set turned out to be 98.48%. Whereas the prediction performance against the testing set turned out to be 98.49%.

TABLE II. PREDICTED RESULTS APPYING RANDOM FOREST ALGORITHM

ALGORITHM	**ATTACK**	**PARAMETER**	**VALUE**
Random Forest	***DoS***	Accuracy	0.99773 (+/- 0.00224)
		Precision	0.99826 (+/- 0.00379)
		Recall	0.99638 (+/- 0.00416)
		F-measure	0.99718 (+/- 0.00293)
	Probe	Accuracy	0.99357 (+/- 0.00465)
		Precision	0.99141 (+/- 0.00544)
		Recall	0.98673 (+/- 0.01054)
		F-measure	0.98964 (+/- 0.00697)
	U2R	Accuracy	0.99724 (+/- 0.00304)
		Precision	0.96673 (+/- 0.10915)
		Recall	0.82474 (+/- 0.22009)
		F-measure	0.85368 (+/- 0.12596)
	R2L	Accuracy	0.97928 (+/- 0.00538)
		Precision	0.97376 (+/- 0.01311)
		Recall	0.96551 (+/- 0.01201)
		F-measure	0.96913 (+/- 0.00994)

TABLE II shows the readings of different performance parameters such as accuracy, precision, recall and F-score using only Random Forest algorithm on testing test. These different parameters give us a deep analysis in terms of understanding.

TABLE III. ACCURACY TABLE OF DIFFERENT ATTACK TYPES

ALGORITHM	**ATTACK TYPE**	**ACCURACY**
Random Forest	***DoS***	99.773%
	U2R	99.724%
	R2L	97.928%
	Probe	99.357%

K-Nearest Neighbor	***DoS***	99.715%
	U2R	99.703%
	R2L	96.737%
	Probe	99.077%
Support Vector Machine	***DoS***	99.371%
	U2R	99.632%
	R2L	99.632%
	Probe	98.450%
Ensemble Learning	***DoS***	99.785%
	U2R	97.198%
	R2L	99.724%
	Probe	99.242%

TABLE III gives the accuracy percentages achieved using four different algorithms on testing set. Use of different techniques as well as merging these different techniques can definitely have a positive impact on accuracy of the designed model.

V. CONCLUSION AND FUTURE WORK

Experimental work suggests that merging different techniques can help achieve high accuracy of ML based IDS. Techniques such as Feature Engineering, Data Mining and Recursive Feature Elimination have played a pivotal role in achieving high accuracy rate. It was assumed that Artificial Neural Network based IDS would perform best, however the outcome suggests otherwise. The reason for this behavior can be complex tweakable training parameters such as activation function, learning algorithms and weights. In future, we intend to analyze further and design intelligent self-adaptive IDS due highly variable traffic patterns

REFERENCES

[1] Vusal Aliyev, "Using honeypots to study skill level of attackers based on the exploited vulnerabilities in the network," Chalmers University of Technology, Göteborg, Sweden, 2020.

[2] "2000 DARPA Intrusion Detection Scenario Specific Datasets | MIT Lincoln Laboratory," Lincoln Laboratory, Massachusetts Institute of Technology, Jul. 2000. https://www.ll.mit.edu/r- d/datasets/2000-darpa-intrusion-detection-scenario-specific- datasets (accessed Nov. 14, 2020).

[3] A. P. Singh and M. D. Singh, "Analysis of Host-Based and Network-Based Intrusion Detection System," I.J. Comput. Netw. Inf. Secur., vol. 6, no. 8, pp. 41–47, 2014, doi: 10.5815/ijcnis.2014.08.06.

[4] Megha Gupta, "Article: Hybrid Intrusion Detection System: Technology and Development," Int. J. Comput. Appl., vol. 115, no. 9, pp. 5–8, 2015, doi: 10.5120/20177-2384.

[5] M. Abliz and T. F. Znati, "Defeating DDoS using productive puzzles," in 2015 International Conference on Information Systems Security and Privacy (ICISSP), Feb. 2015, pp. 114–123, [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7509 986.

[6] V. Shmatikov and M.-H. Wang, "Security against Probe- Response Attacks in Collaborative Intrusion Detection," in Proceedings of the 2007 Workshop on Large Scale Attack Defense, 2007, pp. 129–136, doi: 10.1145/1352664.1352673.

[7] A. Alharbi, S. Alhaidari, and M. Zohdy, "Denial-of-Service, Probing, User to Root (U2R) & Remote to User (R2L) Attack Detection using Hidden Markov Models," Int. J. Comput. Inf. Technol., vol. 7, no. 5, pp. 204–210, 2018, [Online]. Available: https://www.ijcit.com/archives/volume7/issue5/.

[8] S. Paliwal and R. Gupta, "Denial-of-Service, Probing & Remote to User (R2L) Attack Detection using Genetic Algorithm," Int. J. Comput. Appl., vol. 60, no. 19, pp. 57–62, 2012, doi: 10.5120/9813-4306.

[9] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications, Jul. 2009, pp. 1–6, doi: 10.1109/CISDA.2009.5356528.

[10] J. McHugh, "Testing Intrusion Detection Systems: A Critique of the 1998 and 1999 DARPA Intrusion Detection System Evaluations as Performed by Lincoln Laboratory," ACM Trans. Inf. Syst. Secur., vol. 3, no. 4, pp. 262–294, Nov. 2000, doi: 10.1145/382912.382923.

[11] TrendMicro, "A Trend Micro White Paper | Addressing Big Data Security Challenges: The Right Tools for Smart Protection," 2012. [Online].

[12] A. H. Sung and S. Mukkamala, "Identifying important features for intrusion detection using support vector machines and neural networks," in 2003 Symposium on Applications and the Internet, 2003. Proceedings., Jan. 2003, pp. 209–216, doi: 10.1109/SAINT.2003.1183050.

[13] Chi Hoon Lee, Jin Wook Chung, and Sung Woo Shin, "Network Intrusion Detection Through Genetic Feature Selection," in Seventh ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing (SNPD'06), Jun. 2006, pp. 109–114, doi: 10.1109/SNPD-SAWN.2006.52.

[14] H. G. Kayacik, A. N. Zincir-Heywood, and M. I. Heywood, "Selecting Features for Intrusion Detection: {A} Feature Relevance Analysis on {KDD} 99," 2005.

[15] K. Chumachenko, "Machine Learning Methods for Malware Detection and Classification," Kaakkois-Suomen ammattikorkeakoulu, 2017.

[16] O. Jimenez-del-Toro et al., "Chapter 10 - Analysis of Histopathology Images: From Traditional Machine Learning to Deep Learning," in Biomedical Texture Analysis, A. Depeursinge, O. S. Al-Kadi, and J. R. Mitchell, Eds. Academic Press, 2017, pp. 281–314.

[17] Google, "Machine Learning Crash Course | Google Developers." https://developers.google.com/machine-learning/crash-course/ (accessed Nov. 22, 2020).

[18] E. Hodo, X. J. A. Bellekens, A. W. Hamilton, C. Tachtatzis, and R. C. Atkinson, "Shallow and Deep Networks Intrusion Detection System: {A} Taxonomy and Survey," CoRR, vol. abs/1701.0, 2017, [Online].

[19] W. Hu, W. Hu, and S. Maybank, "AdaBoost-Based Algorithm for Network Intrusion Detection," IEEE Trans. Syst. Man, Cybern. Part B, vol. 38, no. 2, pp. 577–583, Apr. 2008, doi: 10.1109/TSMCB.2007.914695.