ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

IJARCCE

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 7, Issue 6, June 2018

A Machine Learning Approach for Intrusion Detection System

Gaurav Agrawal[1], **Shivank Kumar Soni**[2], **Chetan Agrawal**[3]

Research Scholar in CSE Dept, RITS, Bhopal, India[1]

Asst. Prof. in CSE Dept, RITS, Bhopal, India[2]

Asst. Prof. in CSE Dept, RITS, Bhopal, India[3]

Abstract: Intrusion detection system is a derived barrier of resistance which observes the standard actions of the client for any unspecified or unbalanced act, either inside the network or inside the Host. Intrusion detection systems elevate alarms for anomaly recognition in addition to misuse recognition. It could be applied as a federal as well as distributed setup. It fundamentally observes the internet log for network actions and application, structure and data server logs for host related actions. The rationale of this work is to represent an inventive scheme that presents outcomes of suitably classified and wrongly categorized as fractions and the attributes selected. During this research we enlightened the method "A Machine Learning Approach for Intrusion Detection System" which is advised to develop the fitness of discovery of intrusion pertaining variety of Machine learning algorithms on KDDCUP99 data set. During the experimentation we make use of Adaboost, JRip, NaiveBayes and Random Tree classifiers to classify the variety of attacks from the KDDCUP99 data set. The implementation outcomes study of proposed algorithm exhibit that the used machine learning algorithms offers maximum Receiver Operating Characteristics (ROC) to 99.9 %.

Keywords: Classification, Data Mining, NIDS, Cyber Security, Kdd Cup 99, Machine Learning**.**

I. INTRODUCTION

Intrusion Detection is described as the problem of identifying individuals who are using a computer system without authorization (i.e., 'crackers') and those who have legitimate approach to the system but are abusing their privileges (i.e., the 'insider threat') [1]. An **Intrusion Detection System** (IDS) is a computer program that attempts to perform intrusion detection by either misuse or anomaly detection, or a combination of techniques. IDS should preferably perform its effort in real time [1]. With the emergence of intrusion detection systems as a more common feature in the cyber security domain, IDSs should not be believed to be a cure-all solution to the network security problem. In the cyber security arena, intrusion detection is one of the favorite and most active research areas. Still there are many shortcomings in the modern day intrusion detection systems. An intrusion detection system cannot compensate for a weak identification and authentication mechanisms. It requires human intervention to further investigate the attacks. It cannot address the problems in the quality and integrity of information the system provides. Most of the modern IDSs contemplate mainly on string matching and other forms of signature identification mechanisms to detect the classes of attacks. These mechanisms rely on previously-encountered attack signatures for the analysis. We can say that they are only as accurate as the information they rely on. They still lack the intelligence and decision making power to efficiently predict the threats. Network-based intrusion detection systems are more vulnerable to overload as they cannot analyze all the traffic on a busy network. It only takes a single vulnerability on one machine to allow an adversary to gain entry and wreak havoc on the entire network. Hence the response time of an intrusion detection system should be close to zero. This research addresses the problem of reducing the number of features and correctly identifying relevant features from a set of collected data for an anomaly-based intrusion detection system while maintaining integrity of the data. Data acquired for an intrusion detection system frequently originates from multiple sources such as system activity logs, content of data packets and headers, system calls, memory and disk approach activities, and other information. Intrusion detection systems may also share these logs among other network devices for collaboration in a distributed manner. Reducing the amount of data to that which is relevant requires categorizing the information from the logs into parameters, also referred to as dimensions. In a data set of network traffic, attacks are identified by the selection of features that represent particular activities. This implies that not all attacks are found by the same selection of features in all cases. Research conducted by [2] using the KDD CUP 99 data outcomes in a various set of attributes for which each of the four major attack types. Without reducing the number of features, detecting attack patterns within the data is more difficult for rule generation, forecasting, or classification. One of the problems is that not all of the features are important [3]. Identifying and eliminating redundant and irrelevant features within the data, while maintaining the integrity of the corpus, outcomes in features which succinctly describe the activity recorded. Reducing the number of features pertinent to intrusion detection analysis provides better data manageability, lowers computing resource requirements, and usually better outcomes.

DOI 10.17148/IJARCCE.2018.7613

The rationale of this work is to signify an innovative scheme that suitably identifies appropriate attacks from an intrusion data set that diminishes the quantity of data wanted for abnormal activity recognition while preserving the truthfulness of the data set. By sinking the unnecessary features, unrelated features, and noise, improved outcomes might be achieved in the analysis of the data for discovering abnormal activities. The probable outcomes of our work included the following objectives:

1. Schemes to categorize related features and diminish the number of characteristics chosen from a source of network traffic data exclusive of modifying the attributes of the data illustration.
2. Evaluate outcomes of suitably classified and inaccurately classified as fractions, and the characteristics selected.

II. INTRUSION DETECTION SYSTEM

To distinguish intruders, evolving Intrusion Detection Systems (IDSs) is the mainly admirable resolution to defend systems and networks. Hence the endeavor of IDS is not merely to categorize intruders but as well to scrutinize the attack of intruders. A particular system of securing information and resources from prohibited approach, injurious and denial of utilization is to be constructed. For all system, the defense perception is to be prepared based on the expected accomplishment. Primarily safety is concerned with the following features in a computer organization.

- ***Confidentiality:*** data is to be accessed only by allowed users.
- ***Integrity:*** data must persist unchanged by damaging or malevolent efforts.
- ***Availability:*** computer is liable to function without decrease of approach and grant resources to authorized clients when they desire it.

Especially an intrusion is described as a set of occurrences which are strange and sudden to the client, which negotiate the security of a computer organization. It could be made from outside area or inside area of the organization. Formerly in 1980‟s P. Anderson has described intrusion as the range of illicit strength to access data, cheat data, or making the computer organization insecure. Intrusion Detection System (IDS) was economically endorsed in the year 1990. Since then various designs were proposed to adapt intrusion detection systems [4].

It performs similar to an intruder alarm and discovers any variety of contravention and produces alarms similar to audible, visual and as well messages similar to e-mail. On the complete, IDS is principally demoralized for preventing imperfect actions that may assault or abuse the organization by discovering attacks through providing preferable maintain for security organization and also provide useful information concerning intrusion. But formation of IDS should own small false alarms while task of the detection of attacks. IDSs have become defensive methods everywhere in existing networks. There is no thorough and expert methodology proposed in verifying the potency of these organizations.

III. LITERATURE REVIEW

This chapter starts with literature survey and explores previous work approved by various researchers in the domain of attack classification of KDD cup dataset in current years. We present brief descriptions of the Data Mining and Machine Leaning included in the studies that we have done.

Authors of research work [5] stated an Aho-Corasick algorithm based on parallel string matching for recognition of intrusion. The balance Space utilization among homogenous Finite State Machine (FSM) for every string matcher and a finest set of bit location clusters are established and the objective patterns are sorted by Binary Reflected Grey Code (BRGC) which diminishes the bit transmissions and are consumed for recognition of intrusions.

Work of [6] has examined the feature selection of network traffic and the impacts on the detection rates. The KDDCUP 99 dataset is exerted as experimental dataset. The detection rates are found by choosing the various combinations of these feature groups. The ineffectiveness of the approach is also shown in finding anomalies by looking at the host based features within the shorter time interval of 2 seconds.

In research work [7] authors have acknowledged a novel process for HNIDS via taking two stage strategies with weight balancing model. In the online stage, the network packets are detained and divide according to the nature of protocol, then intrusion are discovered by every sensor. In the offline, training dataset is exerted to construct model, which could identify intrusion. It calculates the SMOTE over sampling process, AdaBoost and random forests algorithm.

Authors of [8] have researched with Conditional Random Fields and Layered Approach to tackle two concerns namely precision and Recall. The proposed system based on Layered Conditional Random fields outperforms other well distinguished process for instance the decision trees and the NaiveBayes. The improvement in attack detection is very high, particularly, for the U2R attacks (34.8% improvement) and the R2L attacks (34.5% improvement).

DOI 10.17148/IJARCCE.2018.7613

Authors of [9] have focused on the exercise of weight of network protocol and modeled a weight founded anomaly detector which could effectively discover outliers of network servers. It expands these researches by pertain a novel noise decreased Fuzzy Support Vector Machine to enhance the recognition rate. The novel process known as PAYL-FSVM employs reform error based fuzzy membership function to decrease the noise of the data and to resolve the sharp boundary difficulty. The outcome of noisy data still receives part in reducing the precision.

Authors in [10] have developed a C4.5 Decision Tree algorithm and converted it into rules. The rules are exerted to detect the intrusions from the normal data. The network behavior is analyzed and classified as normal or misuse. The complete processing of the network data is found to be an overhead in this case.

Xiaodan Wang et al [11] have proposed Decision Tree based Support Vector Machine. The feature space of the Support Vector Machines is divided based on the decision tree structure. The structure of the tree is closely related to the accomplishment. An innovative reparability measure is described based on the distribution of the training samples in the feature space. This measure is exerted in the formation of the Decision Tree. The accomplishment is improved than the individual usage of Decision Tree or Support Vector Machines.

Fariba Haddadi et al [12] have represented the two layer feed forward NN for detection of intrusions. Early stopping strategy is exerted in training to overcome the matter of over-fitting. DARPA dataset is exerted for the experiments. The pre-processed data is converted in the range [-1, 1] and given to the NN for classification of Intrusions.

Demidova and Ternovoy [13] have demonstrated the use of Neural Networks for detecting network attacks. The Back-Prorogation Neural Network is exerted to find the attacks in the network traffic. The detection rate is enhanced whereas the false alarm rate is also very high.

AI Islam and Sabarina [14] have devoted research efforts to model the detection system utilizing Recurrent Neural Networks (RNN) which detects the flooding attacks such as DoS and DDoS attacks. Several index terms like Denial-of-service, Distributed-Denial-of-Service, IP spoofing, Flood attack, Zombie, RNN Ensemble are described and they are exerted in detection rate of attacks but the detection of innovative attacks is found to be very low.

Intelligent intrusion detection Hierarchical Neuro-Fuzzy Classifier is exerted Principal Component Analysis (PCA) to reduce the features and Fuzzy-C Means Clustering is exerted to create the Fuzzy rules. *kddcup 99* data is exerted for evaluation of the experiments. Genetic Algorithm is exerted in optimizing the outcomes of the detection model.

IV. PROPOSED WORK

In this chapter we will explain our approach **"A Machine Learning Approach for Intrusion Detection System"** which is proposed to enhance the competence of recognition of intrusion employing different WEKA classifiers on processed *KDDCUP99* dataset. A WEKA 3.8.1 Tool is employed for the rationale Outcome analysis [56]. During the experiment we employed Adaboost, JRip, NaiveBayes and Random Tree classifiers to categorize the different attacks from the processed *KDDCUP99*. The WEKA Classifiers are calculating experimental evaluation outcomes on the basis of following parameters i.e. precision, recall, f-measures and ROC Curve Area.

We apply processed KDDcup99 Dataset on WEKA tool, it takes its 70% part for training rationale (the % of dataset training could be variable) and 30% part for testing rationale.

On the training part (i.e. 66% of KDDcup99 Dataset) apply the preprocessing procedures according to classifier used at that time of execution (in our experiment these are Adaboost, J48, JRip, NaiveBayes, Randomtree) than train the classifier and generated a trained classifiers for the detection rationale.

The remaining 34% of data set is used for the rationale of testing, it will preprocess by WEKA Classifier and then applied on trained classifier which will further classify them into their attack categories (i.e. DoS, Probe, R2L, U2R, Normal as shown in table 4.1).

Proposed Algorithm

Step1: Pre-Processing- The dataset training and testing is separated into the individual attack label. By defaults KDDCup99 dataset is arrangement of 5 attack categories that are DOS, R2L, U2R, Probe and Normal however in our proposed work KDDCup99 dataset is processed as mentioned 5 attack categories. The attacks in KDDCup99 training dataset and attacks in KDDCup99 testing dataset are shown in the Table 4.1. The Number of samples in the kddcup99 dataset and distribution of attacks is shown in Table 4.2

DOI 10.17148/IJARCCE.2018.7613

Table 4.1: Attacks Present In the Kddcup'99 Datasets

Attack Name	Attacks in KDDCup99 Training set	Additional attacks in KDDCup99 Test set
DoS	Back, Neptune, smurf, teardrop, land, pod.	apache2, mailbomb, processtable.
Probe	Satan, portsweep, ipsweep, nmap.	Mscan, saint.
R2L	warezmaster, arezclient, ftpwrite, guesspassword, imap, multihop, phf, spy	Sendmail, named, snmpgetattack, nmpguess, xlock, snoop, worm.
U2R	Rootkit, bufferoverflow, loadmodule,perl.	httptunnel, ps, sqlattack

Table 4.2: Number of Samples in the Kddcup99 Test Set and Distribution of Attacks

Attack Category	Number of Samples	Distribution of Attacks in %
Normal	60589	19.48
DoS	229853	73.90
R2L	16179	5.20
U2R	228	0.07
Probe	4165	1.4
Total	**311014**	**100**

Step 2: Dataset Training - The KDDCup99 dataset in ARFF file Format is employed for the rationale of experimentation study. The KDDCup99 dataset training is an assortment of 494,020 records. All dataset tuple is a solo attached vector expressed through 41 feature values and precisely one tag of either 'normal' or an 'attack' is given. The size of KDDCup99 is 51MB of which 70% is used for training.

Step 3: Dataset Testing - KDDCup99 dataset testing is discovered for the experimental study of proposed system. The dataset testing is separated into the individual attack. By defaults KDDCup99 dataset is arranged of five attack categories that are DOS, R2L, U2R, Probe and Normal. The size of KDDCup99 Test dataset is 45 MB of which 30% is used for testing.

Step 4: Classification- Processed KDDCup99 dataset is tested with the various WEKA classifiers like Adaboost, J48, JRip, NaiveBayes, and Random Tree.

V. RESULT ANALYSIS

Following Evaluation Parameters are used to carried out the experimental study of proposed method

1. **True Positive (TP) / Recall :**

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

2. **False Positive (FP):**

$$\text{FP} = \frac{\text{FP}}{\text{TN} + \text{FP}}$$

3. **True Negative (TN):**

$$\text{TN} = \frac{\text{TN}}{\text{TN} + \text{FP}}$$

4. **False Negative (FN):**

$$\text{FN} = \frac{\text{FN}}{\text{FN} + \text{TP}}$$

5. **Precision:**

$$\text{Accuracy} = \text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

DOI 10.17148/IJARCCE.2018.7613

6. **F-Measure: -**

$$\mathrm{F-measure} = \frac{2 * \mathrm{Precision} * \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}}$$

7. **ROC: -** Receiver operating characteristics (ROC) plans are supportive for systematizing classifiers and visualizing their outcome. Receiver Operating Characteristic (ROC), or ROC curve, is a graph plot that exhibits the outcome of a binary classifier technique as its intolerance threshold is various.

5.2 Experimental Results :

The Experiment Outcomes of the Adaboost, NaiveBayes, JRip and Random Tree classifiers is mentioned in following tables.

Table 6.1 Shows the outcomes of True Positive Parameter and its comparison with various machine learning algorithms we used i.e. Adaboost, NaiveBayes, JRip and Random Tree classifiers

Table 6.1: Results of True Positive

Class	NaiveBayes	JRip	Random Tree	Adaboost
DoS	0.793	1	1	1
u2r	0.712	0.864	0.848	0.818
Probe	0.983	0.982	0.991	0.994
r2l	0.966	0.75	0.83	0.831
Normal	0.68	0.952	0.945	0.948
Weighted Avg.	**0.782**	**0.977**	**0.98**	**0.981**

Table 6.2 Shows the outcomes of False Positive Parameter and its comparison with various machine learning algorithms we used i.e. Adaboost, NaiveBayes, JRip and Random Tree classifiers

Table 6.2: Results of False Positive

Class	NaiveBayes	JRip	Random Tree	Adaboost
Dos	0.01	0.011	0.001	0
u2r	0.003	0	0	0
Probe	0.138	0	0	0
r2l	0.075	0.01	0.011	0.011
Normal	0.005	0.013	0.011	0.011
Weighted Avg.		**0.011**	**0.003**	**0.003**

Table 6.3 Shows the outcomes of Precision Parameter and its comparison with various machine learning algorithms we used i.e. Adaboost, NaiveBayes, JRip and Random Tree classifiers

Table 6.3: Results of Precision

Class	NaiveBayes	JRip	Random Tree	Adaboost
Dos	0.995	0.996	1	1
u2r	0.131	0.934	0.918	0.947
Probe	0.087	0.985	0.981	0.99
r2l	0.411	0.808	0.804	0.81
Normal	0.971	0.946	0.954	0.955
Weighted Avg.	**0.948**	**0.977**	**0.98**	**0.981**

Table 6.4 Shows the outcomes of Recall Parameter and its comparison with various machine learning algorithms we used i.e. Adaboost, NaiveBayes, JRip and Random Tree classifiers

Table 6.4: Results of Recall

Class	NaiveBayes	JRip	Random Tree	Adaboost
Dos	0.793	1	1	1
u2r	0.712	0.864	0.848	0.818
Probe	0.983	0.982	0.991	0.994
r2l	0.966	0.75	0.83	0.831
Normal	0.68	0.952	0.945	0.948
Weighted Avg.	**0.782**	**0.977**	**0.98**	**0.981**

Table 6.5 Shows the outcomes of F-Measure Parameter and its comparison with various machine learning algorithms we used i.e. Adaboost, NaiveBayes, JRip and Random Tree classifiers

Table 6.5: Results of F-measure

Class	NaiveBayes	JRip	Random Tree	Adaboost
Dos	0.883	0.998	1	1
u2r	0.221	0.898	0.882	0.878
Probe	0.159	0.984	0.986	0.992
r2l	0.576	0.778	0.817	0.821
Normal	0.8	0.949	0.95	0.951
Weighted Avg.	**0.841**	**0.977**	**0.98**	**0.981**

Table 6.6 Shows the outcomes of ROC Parameter and its comparison with various machine learning algorithms we used i.e. Adaboost, NaiveBayes, JRip and Random Tree classifiers

Table 6.5: Results of ROC

Class	NaiveBayes	JRip	Random Tree	Adaboost
Dos	0.987	0.994	1	1
u2r	0.997	0.954	0.932	0.986
Probe	0.994	0.996	0.995	1
r2l	0.976	0.971	0.992	0.995
Normal	0.977	0.998	0.997	0.999
Weighted Avg.	**0.985**	**0.994**	**0.999**	**0.999**

Since it might be noticed that conclusion of NaiveBayes Classifier is minor middling. For U2R and R2L attack is it’s fewer than 41% for precision. The execution outcomes of implied algorithm display that the implied machine learning algorithms recommends greatest classification Receiver Operating Characteristics (ROC) up to 99.9 %.

VI. CONCLUSION

Provoked by the limitations of preceding methods to discover Intrusion is a current research issue, for instance high false positive discovery ratios and poor recognition accomplishment on unusual but hazardous classes of network attacks, an innovative machine learning framework is established that influence innovative scheme to intrusion recognition. The proposed scheme **“A Machine Learning Approach for Intrusion Detection System”** is proper for handing out huge multiclass intrusion detection datasets such as the *KDDCUP99 etc*. The deed of this algorithm is evaluated to with the typical machine learning classifiers i.e. Adaboost, JRip, NaiveBayes and Random Tree for the rationale of categorization. Classifiers are assessed based on parameters like True Positive, False Positive, Precision, recall, f-measures and ROC Curve Area completion criterion’s. A WEKA 3.8.1 tool is exerted for the rationale of investigational study. It is scrutinized that Adaboost is the superlative classifier amongst all exerted classifiers through the testing. The completions of the all classifiers are considered with other characteristic machine learning Algorithms. The execution outcomes of implied algorithm exhibit that the implied machine learning algorithms offers highest classification Receiver operating characteristics (ROC) up to 99.9 %. In future this job might be expanded in order to comprise more classifiers and might in addition accomplish characteristic selection to develop classification precision & usefulness. In order to experiment the precision of this scheme in real-time, a network might be exploited which is capable to initiate normal real time intrusions with several packets and various network circumstances.

VII. REFERENCES

[1.] Balasubramaniyan, Jai Sundar, Jose Omar Garcia-Fernandez, David Isacoff, Eugene Spafford, and Diego Zamboni. "An architecture for intrusion detection using autonomous agents." In Computer security applications conference, 1998. Proceedings. 14th annual, pp. 13-24. IEEE, 1998

[2.] Lima, Christiane Ferreira Lemos, Francisco M. de Assis, and Cleonilson Protásio de Souza. "An empirical investigation of attribute selection techniques based on shannon, renyi and tsallis entropies for network intrusion detection." American Journal of Intelligent Systems 2, no. 5 (2012): 111-117.

[3.] Velayutham, C., and K. Thangavel. "A novel entropy based unsupervised feature selection algorithm using rough set theory." In Advances in Engineering, Science and Management (ICAESM), 2012 International Conference on, pp. 156-161. IEEE, 2012.

[4.] Liao, Hung-Jen, Chun-Hung Richard Lin, Ying-Chih Lin, and Kuang-Yuan Tung. "Intrusion detection system: A comprehensive review." Journal of Network and Computer Applications 36, no. 1 (2013): 16-24.

[5.] Hyunjin Kim, Hyejeong Hong, Hong-Sik kim and Sungho Kang. “A Memory-Efficient Parallel String Matching for Intrusion Detection Systems”, IEEE communication letters, pp. 1004-1006, 2009

[6.] Ma, Wanli, Dat Tran, and Dharmendra Sharma. "A study on the feature selection of network traffic for intrusion detection purpose." In Intelligence and Security Informatics, 2008. ISI 2008. IEEE International Conference on, pp. 245-247. IEEE, 2008.

[7.] Yueai, Zhao, and Chen Junjie. "Application of Unbalanced Data Approach to Network Intrusion Detection." In Database Technology and Applications, 2009 First International Workshop on, pp. 140-143. IEEE, 2009.

[8.] Gupta, K.K., Nath, B. and Kotagiri, R. “Layered Approach Using Conditional Random Fields for Intrusion Detection”, IEEE Trans.Dependable and Secure Computing, Vol. 7, No. 1, pp. 35 - 49, 2010.

[9.] Guiling Zhang, Yong Zhen Ke, Liankun Sun and Wei Xin Liu. “An Improvement of Payload-based Intrusion Detection Using Fuzzy Support Vector Machine”, in Proc. of the International Conference on Information Security, pp. 1-4, 2010

DOI 10.17148/IJARCCE.2018.7613

[10.] Juan Wang, Qiren Yang and Dasen Ren. “An Intrusion Detection Algorithm Based on Decision Tree Technology”, in Proc. of the International Conference on Information Processing, pp. 333-335, 2009.

[11.] Xiaodan Wang, Zhaohui Shi, Chongming Wu and Wei Wang. “An Improved Algorithm for Decision-Tree-Based SVM”, in Proc. of the International Conference on Information Security, pp. 4234-4238, 2006.

[12.] Haddadi, F., Khanchi, S., Shetabi, M. and Derhami, V. “Intrusion Detection and Attack Classification Using Feed-Forward Neural Network”, in Proc. of the International Conference on Computer and Network Technology, pp. 262-266, 2010

[13.] Demidova, Y. and Ternovoy, M. “Neural Network Approach of Attacks Detection in the Network Traffic”, in Proc. of the International Conference on CAD Systems in Microelectronics, pp. 128-129, 2007

[14.] AI Islam, A.B.M.A. and Sabrina, T. “Detection of various Denial of Service and Distributed Denial of Service Attacks using RNN Ensemble”, in Proc. of the twelfth International Conference on Computers and Information Technology, pp. 603-608, 2009.

DOI 10.17148/IJARCCE.2018.7613