Intrusion Detection System Using Ensemble Approach A Machine Learning Technique

Mohammed Shoaib Lingadhalli
Department of Computer Science and Engineering,, Vishveshvaraya Technological University

Mrigyank Shekhar
Department of Computer Science and Engineering,, Vishveshvaraya Technological University

Mrinalini Ghosh
Department of Computer Science and Engineering,, Vishveshvaraya Technological University

Kanchan Purohit
Assistant Professor
Department of Computer Science and Engineering,, Vishveshvaraya Technological University

ABSTRACT

In this digital era, intrusion detection is gaining it's extensive importance. However, detection of intrusion is not only the concern but identifying the type of attack on the system within a short span of time has also become a major challenge for the generation. Proceeding with the developments in Machine Learning (ML) a hybrid model is developed which evaluates the degree of intrusion depending upon optimal features through which model is trained. A hybrid model is developed by ensembling Random Forest, Support Vector Machines (SVM) and K Nearest Neighbour (KNN) classifiers. Dataset used is NSL-KDD, it consists of 41 feature values among which 13 features are selected. Results of proposed model unveiled that the ensembled technique had a remarkable impact on computational complexity, time complexity and accuracy.

Keywords: Intrusion, Machine Learning, Ensembling, Random Forest, SVM, KNN.

1. INTRODUCTION

The growth of computer networks and it's applications has increased the risks of intrusion. Intrusion is referred as, an attacker enters the system and loads vicious packets into the user system so that one can corrupt, sneak or alter any privileged information. Intrusion happens either on the server or directly on user system as an outcome of system weaknesses like cross site scripting, security misconfiguration and program errors. Attackers are generally attracted to global networks which consists of several parallel online services that are handled by billions of servers. Such networks are prone to attacks and increased risks of these networks is the major reason for necessity of intelligent intrusion detection systems[1,3,7].

Intrusion Detection Systems(IDS) can be classified as signature and anomaly based intrusion detection systems. Signature based IDS can be referred as misuse based IDS. It is most basic and yet defective approach, the main idea here is to map the activities based on available knowledge to detect intrusions. Instructions are identified depending upon various system vulnerabilities and familiar signatures. But it fails to recognize the new attacks on the system. Contrarily, anomaly based IDS first

learns the activity of machine and if the machine deviates from the expected behaviour then it is declared as suspicious [4].

An intruder follows below mentioned steps in order to attack on the system[1]:

- Collecting Information - It involves extracting all the necessary information about the target host who will be attacked. It is possible by executing various network commands such as “whois” and “nslookup”.
- Probe and Scanning - Scanning the target host thoroughly so that information about any unprotected port can obtained where intrusion can take place.
- R2L access - Remote to Local access is obtaining local access of the target host by launching attacks on the open ports. Access can be obtained by launching various attacks such as password dictionary attack, ftp-write attack and xlock attack.
- U2R access - Intruder takes advantage of system’s defenselessness to gain the root access as he already holds the user access. Perl, fdformat, xterm are some of the attacks which can performed to gain root access.
- Launching attacks - The final step to attack the target host and tamper or obtain the confidential information which is supposed goal.

Therefore, it is essential to build an efficient intrusion detection system which protects the networks system by sensing various attacks. A hybrid classification based model is proposed. Then, NSL-KDD dataset’s feature dimensions are reduced by implementing feature selection. Later, using the developments in the field of machine learning an intrusion detection model is designed which detects the system attacks. The developed model requires extraction of features, reduction in dimensions that narrows down the extracted features, and selected features. Apparently all the features are mixed hence transformation features are used for feature extraction. Classification criteria serves as key for necessary feature extraction.

2. LITERATURE SURVEY

D. P. Gaikward [1] Came up with ML techniques to implement IDS. Base classifier here is partial decision tree because of it’s simplicity. Selection of required features is done using genetic algorithm which reduces the 41 feature dimension space to 13 feature dimension space to improve system accuracy. The developed intrusion detection system was tested for time required for building the model, true positive, false positive, false positive and system accuracy. The modeling building time was very high and the false positives accuracy observed was more

A. Das [2] Proposed model uses a feature extraction module (FEM) that describes the details about the network which are useful in upcoming stages. FPGA architecture is used for intrusion detection. Existing integrated-circuit models were outdated and their drawbacks were remarkably handled in the proposed model. The developed network intrusion detection system uses PCA as classifier. The results can be classified based on extensive pipelining and hardware parallelism.

Sunil Nilkanth Pawar [3] Rule generation is done using chromosomes with appropriate features. Every rule is defined with a function called effective fitness function. Chromosomes consists of variable number of rules. Less number of chromosomes are required for detection of an effective intrusion because every chromosome itself is a complete solution to the given problem. Proposed model is tested by Defense Advanced Research Project Agency and observed that the computational time is very less..

Eduardo De la Hoz [4] Proposes a model where selection of features is done using PCA along with the Fisher's discriminant ratio. Incoming packets can be segregated as normal and malicious packets with the help of probabilistic self-organizing maps. Methodologies adapted to detect the network intrusions are classification approaches, statistical techniques and probabilistic self-organizing maps. The developed model showed better accuracy compared to the existing models

F. Hosseinpour [5] A distributed framework is designed for infiltration detection by utilizing genetic algorithm to enhance the secondary immune response. After training the detectors using negative algorithm they are distributed to each host as an agent of main IDS engine. Therefore, the amount of load on the central engine is reduced and performance of the system is increased. The uniqueness of this is the distribution of both of primary and memory cell detectors to each host while using the genetic algorithm for evolution of the memory cells.

Fangjun Kuang [6] In this paper kernel principal component analysis is used for feature reduction and also reducing the training time. A multi-layer support vector machine is used as basic classifier. N-RBF here is used for removal generated due to differences in features, it also helps in reducing the training time. Performance of the developed model is found to be very high.

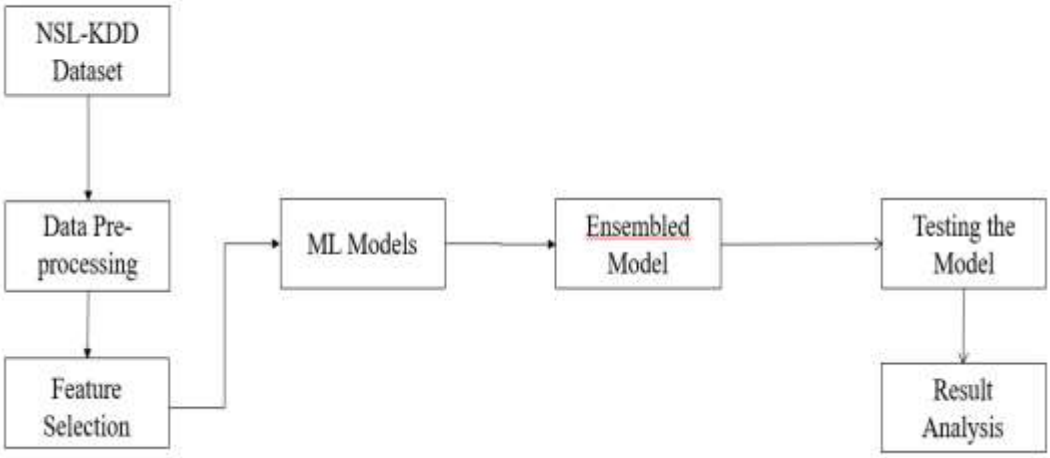
S.A Aljawarneh [7] SRE mechanisms were used to obtain accurate signatures. The implementation was accomplished with the help of Needleman-Wunsch algorithm which was inadequate to manage invariant parts and distance restrictions of polymorphic worms. Consecutively, an Enhanced Contiguous Substring Reward (ECSR) algorithm was further developed to improve the the results

Iftikhar Ahmad et al. [8] Proposed model performed feature selection based on eigen values using PCA algorithm. Instead of considering the existing approaches a new genetic principal component approach is developed for choosing the features whose eigen values if highest. SVM is the classifier used.

L. Dhanabal [9] Proposes a model by using various classifiers and trained it through the NSL-KDD training dataset. Futher model was tested for 6 subset feature. After testing all the classifiers, accuracy of all the models was depicted. It was found that C4.5 classifier had the best results among other classifiers considered.

Ujwala Ravale [10] Proposes a model where, classification methods like RBF kernel function of SVM and K-Means clustering a data mining technique are combined. These techniques are used reduce the number of attributes related with every data point. But accuracy of the proposed model didn't meet the expectations.

3. EXPERIMENT



The above figure shows the system architecture for building an hybrid model using various machine learning classifier models

Functionality Overview of Proposed Model

1.Choosing a proper dataset such as NSL-KDD dataset. NSL-KDD training dataset is used for training purposes and NSL-KDD testing dataset is used for testing purposes.

2.Preprocessing Phase: The dataset is preprocessed to reduce or eliminate the noise from the data.Categorical values are converted into numerical features. The dataset is normalized in order to convert the data values within a range.

3.Building the classifiers based on SVM, KNN & Random Forest classifiers.

4.Using the best classifier to select 13 features among all the features of NSL-KDD dataset.

5.Building the hybrid model using the SVM,KNN,Random Forest classifiers using Bagging approach.

6.Developing the model and evaluating its accuracy and results.

1. Dataset Description

NSL-KDD is the standard dataset taken for training and testing. Training set has 125973 entries and test set has 22544 entries with 41 features.

The NSL-KDD dataset is a updated version of KDD dataset.

- Redundant records were removed in order for the classifier to give impartial results.
- Sufficient number of records have been provided in the dataset for both training and testing purposes.

Apart from normal data, the NSL-KDD dataset records contain 39 different attack types. These can be classified into four different types of attacks.

Denial of Service Attack (DoS): This attack occurs when a legitimate user is denied access to the system. Examples are 'neptune', 'teardrop', 'ping of death' , 'pod' , 'back' , 'mail bomb','smurf' and 'land'.

User-to-root attack (U2R): This attacks occurs when the attacker already possess the user access and tries to gain the root access forcefully. Examples are 'buffer overflow', 'load-module', 'rootkit' and 'perl'.

Remote-to-local attack (R2L): This attack occurs when an intruder successfully gains the user permissions of target host. Examples are 'phf' , 'warezclient', 'warezmaster', 'spy', 'ftp-write' , 'imap','multihop' and 'guess passwd'.

Probe: This attack type of attack occurs when an intrude dodges the current firewalls of the system and gains the necessary information. Examples are 'nmap' , 'isweep' , 'satan' and 'portsweep'.

Therefore, five classes have been considered namely Normal class, DoS class which consists of 10 types of attacks, Probe class which has 6 different types of attacks, R2L class that includes 16 types of attacks and U2R class which has 7 different types of attacks.

The attacks are therefore labeled in respective classes. The dataset is further divided into four parts for training and testing purposes.

data_DoS_train → all training data entries having normal label as well as Dos.

data_Probe_train → all training data entries having normal label as well as Probe.

data_U2R_train → all training data entries having normal label as well U2R.

data_R2L_train → all training data entries having normal label as well as R2L.

data_DoS_test → all testing data entries having normal label as well as Dos.

data_Probe_ test → all testing data entries having normal label as well as Probe.

data_U2R_ test → all testing data entries having normal label as well U2R.

data_R2L_ test → all testing data entries having normal label as well as R2L.

2. Preprocessing Phase

(i) To work well with the classifiers, we need numerical features since most of the machine learning algorithms do not work well with non numerical features. Therefore we need to convert categorical features like protocol_type, service, flag to numerical features. We can assign these features numerical values using Label Encoding and One Hot Encoding.

(ii) Labelling of attacks into different classes. Normal is labelled as class 0, DoS is labelled as class 1, Probe is labelled as class 2, U2R is labelled as class 3, and R2L as class 4.

(iii) The four classes of attacks class 1,2,3,4 are mixed with class 0 to form four different training and testing datasets.

(iv)Feature Selection is done to select 13 features to reduce computational complexity. Filter or Wrapper approach can be used for this purpose.

(v)Feature Scaling is done in the range{0,1} to make the feature values comparable.

3.Training

The four classes of datasets are used to train SVM, KNN, Random Forest binary classifiers. The ensembled model is then trained using Bootstrapping approach.

Data set type	No of data samples					
	Records	Normal	DoS	Probe	U2R	R2L
NSL-KDD Train	125973	67343	45927	11656	52	995
	%	53.46	36.45	9.25	0.04	0.79
NSL-KDD Test	22543	9711	7458	2421	200	2754
	%	43.08	33.08	10.74	0.89	12.22

4. RESULT AND DISCUSSION

The three classifiers and further the ensembled model is tested using the four classes of testing datasets.

The performance of the models is evaluated using confusion matrix generated which gives the score for accuracy , precision, recall and f-score.

SVM Results

Class name	Accuracy	Precision	Recall	F-Measure
DoS	99.371%	99.107%	99.450%	99.278%
Probe	98.450%	96.907%	98.365%	97.613%
U2R	96.793%	94.854%	96.264%	95.529%
R2L	99.652%	91.988%	83.981%	85.918%

KNN Results

Class name	Accuracy	Precision	Recall	F-Measure
DoS	99.715%	99.678%	99.665%	99.672%
Probe	99.077%	98.606%	98.508%	98.553%
U2R	96.737%	95.311%	95.484%	95.389%
R2L	99.703%	93.282%	84.835%	87.754%

Random Forest Results

Class name	Accuracy	Precision	Recall	F-Measure
DoS	99.802%	99.879%	99.665%	99.785%
Probe	99.662%	99.721%	99.349%	99.378%
U2R	99.765%	96.396%	83.432%	91.707%
R2L	98.079%	97.390%	97.019%	97.256%

Ensemble Model Results

Class name	Accuracy	Precision	Recall	F-Measure
DoS	99.802%	99.852%	99.705%	99.772%
Probe	99.283%	98.765%	98.952%	98.992%
U2R	97.213%	95.811%	96.433%	96.029%
R2L	99.765%	94.515%	86.144%	90.643%

5. CONCLUSION

An intrusion detection model was developed to detect and classify attacks on a system. Using the 13 out of 42 feature of NSL-KDD data set. The resulting model provides an efficient and effective approach to detect and classify 4 different types of attacks namely, Denial of Service(DOS) , Probe and Scanning, Root to Local(R2L) and User to Root(U2R) attacks.The ensemble model based on SVM, KNN and Random Forest, which was trained on the training data set performed with the best accuracy rate in comparison to the SVM, KNN and Random Forest individually. An interface is designed for the users based on the proposed detection model so that an individual can detect the intrusion and also classify the type of attack attempted. As many of the existing systems mainly focus only on building the models, proposed system acts an extension to these works by providing an interface to these models. The proposed model can detect intrusion with 99.26% accuracy. The result produced has better performance and accuracy compared to the other existing approach of detecting intrusion. Different datasets containing new records related to traffic and intrusion can be used to explore further improvements. The concepts of neural networks can be explored to further improve the understanding of IDS to enhance its performance. Deep learning techniques can also be used to increase the accuracy of the system..

6. REFERENCES

1. D.P. Gaikward, Ravindra c Thool, Intrusion detection system using bagging with partial decision tree base classifier, in: Proceeding of International Conference on Advanced in Computing, Communication and Control, sICAC3(15, in: Procedia Computer Science, vol. 49, Elsevier, 2015, pp. 92–98
2. A. Das, D. Nguyen, J. Zambreno, G. Memik, A. Choudhary, An FPGA-based network intrusion detection architecture, IEEE Trans. Inf. Forensics Secur. 3 (1) (2008) 118–132.
3. Sunil Nilkanth Pawar, Rajankumar Sadashivrao Bichkar, Genetic algorithm with variable length chromosomes for network intrusion detection, Int. J. Autom. Comput. 12 (3) (2015) 337–342.
4. Eduardo De la Hoz, Emiro De La Hoz, Andreı´s Ortiz, Julio Ortega, Beatriz Prieto, PCA Filtering and Probabilistic SOM for Network Intrusion Detection, vol. 164, Special Issue: Advances in Computational Intelligence in Elsevier—Neurocomputing, 2015, pp. 71–81.
5. F. Hosseinpour, A. Meulenberg, S. Ramadass, P. Vahdani, Z. Moghaddasi, Distributed agent based model for intrusion detection system based on artificial immune system, Int. J. Digit. Content Technol. Appl. 7 (2013) 206–214.
6. Fangjun Kuang, Siyang Zhang, Zhong Jin, Weihong Xu, A novel SVM by combining kernel principal component analysis and improved chaotic particle swarm optimization for intrusion detection, Soft Comput. 19 (2015) 1187-1199.
7. S.A. Aljawarneh, R.A. Moftah, A.M. Maatuk, Investigations of automatic methods for detecting the polymorphic worms signatures, Futur. Gener. Comput. Syst. 60 (2016) 67–77.
8. I. Ahmad, M. Hussain, A. Alghamdi, A. Alelaiwi, Enhancing SVM performance in intrusion detection using optimal feature subset selection based on genetic principal components, Neural Comput. 24 (2014) 1671–1682.
9. L. Dhanabal, S.P. Shantharajah, A study on NSL-KDD dataset for intrusion detection system based on classification algorithms, International Journal of Advanced Research in Computer and Communication Engineering 4 (2015) 446–452.
10. Ujwala Ravale, Nilesh Marathe, Puja Padiya, Feature selection based hybrid anomaly intrusion detection system using K means and RBF kernel function, in: Proceeding of International Conference on Advanced Computing Technologies and Applications, ICACTA-2015, Procedia Computer Science, vol. 45, Elsevier, 2015, pp. 428–435.