Intrusion Classification using SVM Classifier Technique

Jayateerth V Vadavi
Associate Professor
Department of CSE,
SDMCET Dharwad.

Varsha S Betur
PG Student
Department of CSE,
SDMCET Dharwad

Dr Umakant Kulkarini
Associate Professor
Department of CSE,
SDMCET Dharwad

Abstract: With rapid expansion and the rising complexity of network infrastructures and progression of attacks, discovering and preventing network abuses is becomingmuch more strategic to ensure an adequate degree of protection from both internal and external attacks.The malicious abusers tryvarious techniques like sniffing unencrypted or clear text traffic,password cracking etc. to utilize the system vulnerabilities and compromise critical systems during communication in networking system.Intrusion Detection System (IDS) is a network security system for detecting attacks on computer network. Many unsupervised and supervised learning approaches from field of pattern recognition and machine learning have been utilized to enhance the efficiency of IDS.Most of data mining and bio-informatics application require processing of large data. A hugequantity of resources have been consumed in Intrusion-Detection-Systems (IDS) and several machine learning algorithms like decision tree, genetic algorithm, Support vector machines ,Artificial Neural Network and hybrid intelligent system are explored to build an IDS. In proposed method machine learning classifier such as SVM is used to detect fraud attacks.

Key Words: *KDD Dataset, SVM Classifier and IDS*

I. INTRODUCTION

Internet is a worldwide public network. With the enlargement of Internet and its possible, therehas been succeeding change in business replica of organization across world.More and to takebenefit of novel business system popularly calledmore persons were getting associated to Internet daily basisas e-Business. Internetworkconnectivity has nowbecome very essentialfeature of today's e-business.The malicious abusers are usesvariousmethods like sniffingunencrypted,Password cracking, or clear the text traffic etc. to utilize system vulnerabilities declaredabove and cooperation critical system. Thus, there requirebeingfew ofsecurity to organization personalcapital from Internet withof insideusers as survey says that 80% of the attacks are happen from inside users for everydetail that know the system more than an stranger know and access todata is easier for insider.

Various associations across globeorganizethe firewalls to defend their private meshfrom Public network. Excluding, when this comes to protectingthe Private network from Internet using firewalls, no network is hundred percentprotected. This is because;the business needsseveral accesses to be arranged on internal systems tothe Internet users. The firewall gives security by admitting only particular services throughit. The firewall applies a policy for disallowing or allowing connections based onorganizational business needs and defence policy. The firewall protects group from malicious attack from Internet by dropping associations fromunknown source.

In this research, we are resulting in new test andtrain sets that consist of only chosenrecords from KDD dataset whichdoes not bear from the problems. Further, the number of data inthe test and train sets is limited. Thisbenefit makes it sensible to run experiments on using complete datasetwithout small portion. Therefore, theevaluation outcomes of various research workswill be comparable and consistent.

II. LITERATURE SURVEY

ChiragModi et.al [1] introduced review on various Intrusion-Detection-System (IDS) in clouds and suggests IDS position in Cloud structural design to attain desired security in next generation network. Intrusion to attacks can be mainly Insider attack, Root attacks by user,Flooding attack, backdoor channel attacks, Port Scanning. Anomaly Detection,Signature based Detection, Artificial-Neural-Network (ANN) based IDS, Association Rule based IDS,Fuzzy Logic based IDS, and Support-Vector-Machine (SVM) based IDS, Hybrid Techniques, Genetic-Algorithm (GA) based on IDS, and Host based on Intrusion Detection Systems (HIDS). The proposed system finally recognizedsome security challenge that require to be addressed by cloud study community before cloud can turn into a trusted andsecure platform for delivery of prospect Internet of Things.

Wei-Chao Lin et.al [2] proposes the Intrusion-detection-system (IDS) identifiesvariouscategories of malicious mesh traffic and computer which does notidentified by firewall of conventional.The Propose systemdepictsnew feature method like cluster-center and nearest-neighbor (CANN) method. In this method, two distances can besummed and measured, the first is based on distance between individual data model and its cluster center, and second distance is between data which is its closest neighbor in similar cluster. Then, this novel and 1D distance based on feature is utilized to signify each information sample for intrusion detection using a k-Nearest Neighbor (k-NN) classifier. The results are based on KDD-Cup 99 dataset depicts that the CANN classifier do not only executesbest or same to k-NN and support-vector-machines are trained as well tested by the unique feature depiction in terms of accuracyclassification, false alarms and detection rates.

Shih-Wei Lin et.al [3] developed the System of Intrusion-detection (IDS) to observe the attacks in networks or computer. An intelligent method with attribute selection also decision rules which can applied to IDScan be proposed. The key plan is to take benefit of SVM (Support Vector Machine), DT (Decision Tree), and SA (Simulated Annealing). In this system, SVM and SA can discover the finest selected attributes to raise accuracy of anomaly intrusion detection. By examining the databyemploying KDD'99 dataset,SAand DT can attain decision rules for novel attacks alsoboost the accuracy rate of classification. In addition, good parameter setting for SVMand DT are repeatedly adjusted through SA. The proposed system,results other existing methods and successful in identifying anomaly intrusion detection.

Ugo Fiore et.al [4] proposed Boltzmann machine due to rapid increase and extending complexity of mesh infrastructure and attack evolution. Preventing and identifyingthe mesh abuses wasidentifying more and more plan to ensure thetolerable degree of shield from bothinternaland external menaces. Anattractiveattribute of an effective model for anomaly detection is ability to settle in to generalize and change to its action to manyvarious network surroundings. They discoveredefficiency of detection system based on machine learning, by Discriminative Restricted of Boltzmann Machine to unite the open power of generative samplesby better classification accuracy capability to infer section of its information from in full training dataset.

RanaAamirRazaAshfaq et.al [5] proposes fuzziness based on semi-supervised learning method by using unlabeled examplessupport with supervised learning method to boost the classifier recital for IDSs. A SLFN (single hidden layer feed-forward neural network) is trained to result a fuzzy association vector, and example categorization (mid,low and high fuzziness types) on unlabeled modelsare performed byquantity of fuzzy. This classifier can retrained followingincorporate each typeindividually into original set of training. The results are using this model of IDS on NSL-KDD dataset explain that unlabeled sample which belongs to high and low fuzziness categories make mainassistances to enhanceperformance of classifier is compared to existing methods such as naive bayes, random forests,support-vector-machine etc.

III. METHODOLOGY

The anticipated method is shows in Fig. 1, is segregated into two phases training and testing. In training KDD cup 1999 dataset is used as input that is generated using a simulation of a military network. In military KDD dataset, network attacks fall into one of four categories: Denial-of-Service (Dos): invader tries to avoid legitimate users from using service. KDD dataset is arranged in structured form in pre-processing stage then features of KDD dataset are extracted, trained using machine-learning technique called SVM and stored in knowledgebase. In testing user KDD data is extracted and classified using SVM classifier to detect type of attack. Remote-to-Local (r2l): Attacker has no account on victim machine, hence tries to get access. User-to-Root (u2r): Attacker has local access to victim machine and tries to gain super user privileges.

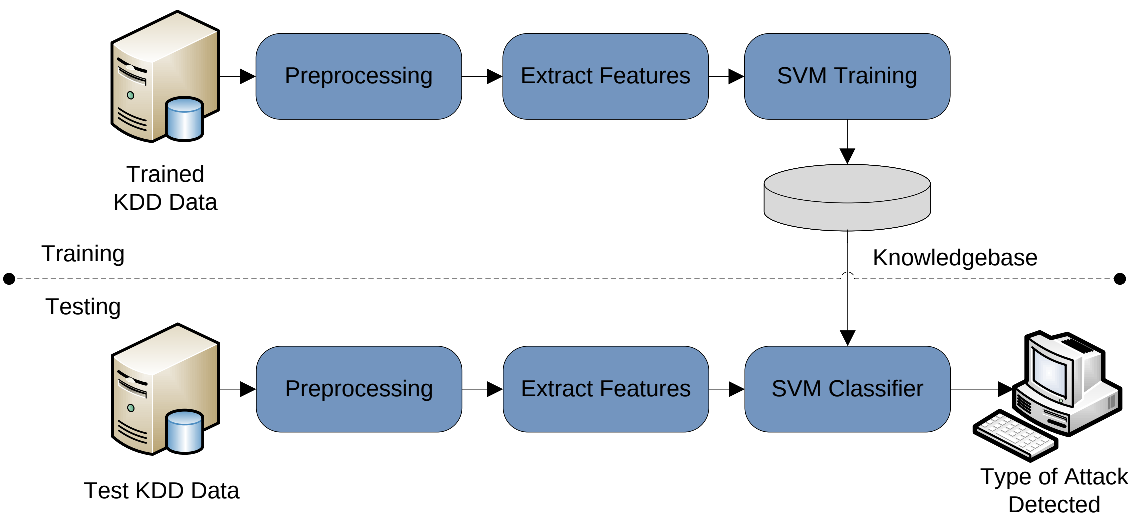

Fig. 1: System Architecture of Proposed Approach

3.1 Preprocessing

Here we are cleaning the dataset if there is any unwanted data. Usually, dataset may have the missing value redundant value and duplicate value. In this proposed system we are employing the KDD dataset. KDD'99 has been one most wildly used database for assessment of anomaly exposure techniques. This data set can be arranged by Stolfo and is created based on data detained in DARPA'98 IDS assessment program. The KDD train dataset have approximately 4,900,000 single connection vector each of which has 41 features and is labeled as whether normal or attack, with precisely one explicit attack type. We are considering the two attacks U2R and R2L.

User to Root Attack (U2R): it is the category of develop in which an attacker to initiates with access to usual account on scheme and which is proficient to develop a few susceptibility to get root access to system. Remote-to-Local-Attack (R2L): it happens when an attacker who hascapability to transfer the packets to device over mesh but who don't have an account on that machine proposes some susceptibility to get local access as of that system.

3.2 TF-IDFFeature Extraction Method

The TF-IDF is elegant in its simplicity. Given a query is composed set of words w_i, we calculate w_i, d for each w_i for every data d ϵ D. In easier way, this can be made by running through data collection also keeping a running sum of $f_{w,d}$ and $f_{w,D}$. Once done, we can estimate$w_{i\ d}$ according to mathematical framework obtained before. Once all w_i, dís are found, we return a set D* containing documents d such that we maximize the following equation:

$$\Sigma_i w_{i,d} \quad (1)$$

Either user or system can randomlyresolve the size of D* prior to initiating the query. Also, data are revisited in a declining order according to above equation. This is implementing method of TF-IDF.

3.3 SVM Classifier

The support-vector-machine (SVM) is a training method for learning regression and classification rules from statistics, for example SVM can be exploited to learn radial-basis-function (RBF), polynomial and multi-layer-perceptron (MLP) classifiers.SVMs mainly based on structural hazardminimization principle, can narrated to regularization theory. This principle integrates capacity control to avoidover fitting and thus is a fractional solution to the bias-variance trade-off dilemma.

If training data are linearly separable then there exists a pair (w,b) such that

$$w^T x_i + b \geq 1, for\ all\ x_i \in P$$
$$w^T x_i + b \leq -1, for\ all\ x_i \in N \quad (2)$$

with the decision rule given by

$$f_{w,b}(x) = sgn(w^T x + b) \quad (3)$$

'w' represents the weight vector and b the bias (or − b is termed the threshold). The inequality constraints (4) can be merged to present

$$y_i(w^T x_i + b) \geq 1, for\ all\ x_i \in P \cup N \quad (4)$$

Without losing of generality pair (w,b) can be rescaled such that

$$\min_{i=1,\ldots l} |w^T x_i + b| = 1 \ (5)$$

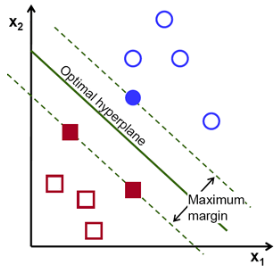

Fig. 2: SVM hyper plane

This constraint defines the set of canonical hyper-planes on N $^{\Re.}$ to restrict expressiveness of hypothesis space, the SVM searches for simplest solution that differentiate data correctly.The learning problem is hence reformulated as:

minimize$||W^2|| = W^T W$ subject to the constraints of linear separability (6).

This is corresponding tomaximizing distance, normal to hyper plane, between convex hulls of two classes; this distance is called the margin. The optimization is now a convex quadratic programming (QP) problem

$$\min_{w,b} imize\ \Phi(w) = \frac{1}{2}||w||^2$$
$$subject\ to\ y_i(w^T x_i + b) \geq 1, i = 1, \ldots l \quad (6)$$

This problem has a global optimum; thus the problem of many local optima in the case of training e.g. a neural network is avoided. The overflow diagram of this proposed system is shows in Fig. 3.

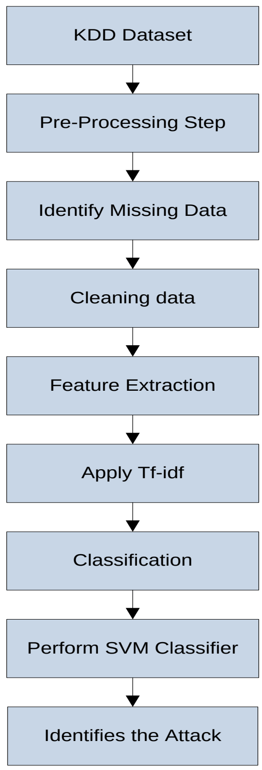

Fig. 3: Flow Chart of Proposed System

Algorithm 1: Proposed System

Input: Data
Output: Categories the attack

Begin

Step.1: Start
Step.2: Train the all KDD Dataset
Step.3: Test one single Dataset
Step.4: Pre-process the data by cleaning the data and identifying the data
Step.5: Extract features by using TF-IDF technique
Step.6: Classify the Attacks by SVM Classifier
Step.7: Output Result: Identifies the attacks
Step.8: Stop

End

IV. RESULTS

Intermediate result of the proposed system briefly summarized in this section. As previously mentioned entire module is divided as training and testing. During the training 500 records per attack is used to train the algorithm.Knowledge base is created by at training. During testing multiple case is studied by using different samples. i.e.Consider

1. Test Case 1: Considering first network sample

Input:

0,tcp,private,REJ,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,219,16, 0,0,1,1,0.07,0.07,0,255,16,0.06,0.09,0,0,0,0,1,1,neptune,21.

Output:

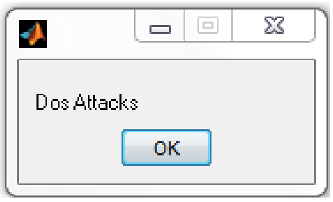

Fig 4: Classification Output

2. Test Case 2: Considering second sample

Input:

169,tcp,telnet,SF,1567,2857,0,0,0,3,0,1,4,1,0,0,1,0,0,0,0,0, 1,1,0,0,0,0,1,0,0,1,1,1,0,1,0,0,0,0,0,buffer_overflow,3.

Output:

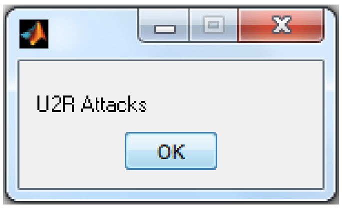

Fig 5: Classification Output

As per above data system performance is tested with multiple input samples. As per KDD dataset it includes four types' network attacks, so proposed system is designed with detection of network attack and its normal characteristics also. From the above it is proven designed model is simple and effective for network fraud detection.

V. CONCLUSION

There are numerous approaches to detect the attacks in an Intrusion-Detection-System. Each of the approaches has its own advantages and disadvantages. Thus a judicious method has to be made while selecting a mode to implement attack detection in an intrusion-detection-system. In proposed method machine learning techniques are employed to detect fraud attacks in networking system. Major focus is given to enhance the detection of malicious attack using SVM classifier.

REFERENCES

[1] ChiragModi, Dhiren Patel, Hiren Patel, BhaveshBorisaniya, Avi Patel And MuttukrishnanRajarajan, "A Survey of Intrusion Detection Techniques In Cloud", Journal of Network and Computer Applications, Elsevier, Vol. 36, No. 1, pp. 42-57, 2013.

[2] Wei-Chao Lin, Shih-Wen Ke and Chih-Fong Tsai, "CANN: An Intrusion Detection System Based on Combining Cluster Centers and Nearest Neighbours", Knowledge-Based Systems, Elsevier, Vol. 78, pp. 13-21, 2015.

[3] Shih-Wei Lin, Kuo-Ching Ying, Chou-Yuan Lee and Zne-Jung Lee, "An Intelligent Algorithm With Feature Selection and Decision Rules Applied to Anomaly Intrusion Detection", Applied Soft Computing, Elsevier, Vol. 12, No. 10, pp. 3285-3290, 2012.

[4] Ugo Fiore, Francesco Palmieri, Aniello Castiglione and Alfredo De Santis, "Network Anomaly Detection with Restricted Boltzmann Machine", Neurocomputing, Elsevier, Vol. 122, pp. 13-23, 2013.

[5] RanaAamirRazaAshfaq, Xi-Zhao Wang, Joshua Zhexue Huang, Haider Abbas and Yu-Lin He, "Fuzziness Based Semi-Supervised Learning Approach for Intrusion Detection System", Information Sciences, Elsevier, Vol. 378, pp. 484-497. 2017.

[6] SumaiyaThaseenIkram and Aswani Kumar Cherukuri , "Intrusion Detection Model Using Fusion of Chi-Square Feature Selection and Multi Class SVM", Journal of King Saud University-Computer And Information Sciences, Elsevier, Vol. 29, No. 4, pp. 462-472, 2017.

[7] AsafShabtai, Uri Kanonov, Yuval Elovici, ChananGlezer and Yael WeissAndromaly, "A Behavioral Malware Detection Framework for Android Devices", Journal of Intelligent Information Systems, Springer, Vol. 38, No. 1, pp. 161-190, 2012.

[8] SannasiGanapathy, Kanagasabai Kulothungan, Sannasy Muthurajkumar, Muthusamy Vijayalakshmi, PalanichamyYogesh and ArputharajKannan, "Intelligent Feature Selection and Classification Techniques for Intrusion Detection In Networks: A Survey", EURASIP Journal on Wireless Communications and Networking, Springer, No. 1, pp. 271. 2013.

[9] Hamed Haddad Pajouh, GholamhosseinDastghaibyfard and SattarHashemi, "Two-Tier Network Anomaly Detection Model:

A Machine Learning Approach", Journal of Intelligent Information Systems, Springer, Vol. 48, No.1, pp. 61-74, 2017.

[10] Carlos A. Catania, Facundo Bromberg and Carlos GarcIaGarino, "An Autonomous Labeling Approach To Support Vector Machines Algorithms for Network Traffic Anomaly Detection", Expert Systems with Applications, Elsevier, Vol. 39, No. 2, pp. 1822-1829, 2012.

[11] Shabtai, Asaf, Uri Kanonov, Yuval Elovici, ChananGlezer, and Yael Weiss, ""Andromaly": a behavioral malware detection framework for android devices", Journal of Intelligent Information Systems, Vol. 38, No. 1, pp. 161-190, 2012

[12] Ganapathy, Sannasi, Kanagasabai Kulothungan, Sannasy Muthurajkumar, Muthusamy Vijayalakshmi, PalanichamyYogesh, and Arputharaj Kannan. "Intelligent feature selection and classification techniques for intrusion detection in networks: a survey." EURASIP Journal on Wireless Communications and Networking , No. 1, pp. 271, 2013

[13] Catania, Carlos A., Facundo Bromberg, and Carlos GarcíaGarino. "An autonomous labeling approach to support vector machines algorithms for network traffic anomaly detection." Expert Systems with Applications, Vol. 39, No. 2, pp. 1822-1829, 2012

[14] Louvieris, Panos, Natalie Clewley, and Xiaohui Liu, "Effects-based feature identification for network intrusion detection", Neurocomputing, Vol. 121, pp. 265-273, 2013

[15] Eesa, Adel Sabry, ZeynepOrman, and Adnan MohsinAbdulazeezBrifcani, "A novel feature-selection approach based on the cuttlefish optimization algorithm for intrusion detection systems", Expert Systems with Applications, Vol. 42, No. 5, pp. 2670-2679, 2015

[16] Narudin, F.A., Feizollah, A., Anuar, N.B. and Gani, A, "Evaluation of machine learning classifiers for mobile malware detection", Soft Computing, Vol. 20, No. 1, pp.343-357, 2016.