A Framework for Simulation of Intrusion Detection System using Support Vector Machine

D.P.Gaikwad
AISSMS College of Engineering
University of Pune, Pune, India

R.C.Thool,Ph.D
SGGS Institute of Engineering and Technology
SRT Marathwada University, Nanded, India

ABSTRACT

An intrusion compromises the security and the value of a computer system in network. Legitimate users find it difficult to access network services due to the network attacks as they intentionally occupy or sabotage network resources and services. The intrusion detection system defends the critical computer system and networks from cyber-attacks. Various techniques of machine learning are applied to intrusion detection system. In this paper, a framework for simulation of intrusion detection system is described. The radial basis kernel based support vector machine is used to simulate the intrusion detection system. The major research goal regarding the SVM is to improve the speed in training and testing by determining the best kernel for a given data. Out of the various parameters of the packet only few important normalized parameters are used which will result in improving speed of training the SVM and high detection rate. The KDDCUP'99 dataset is used to train and test the system. The experimental results show that the detection rate of the system is 88.27% with good speed. Furthermore, two applications of framework are described to show how the system can be used to generate pattern of attack for testing the system and how the system prevent downloading of large PDF files from server by unauthorized user.

General Terms

Computer Network Security, Intrusion Detection, Packet Classification.

Keywords

SVM, Kernel, Normalization, MMH, KKT, FTP server.

1. INTRODUCTION

The set of action that can be threatens the confidentially, integrity, or availability of resources is called as intrusion. The extensive growth of the Internet and increasing availability of tools for intruding and attacking network have prompted intrusion detection to become a critical component of network administration. Most of the intrusion detection system do not provide a complete solution and are limiting. Most of the organizations are becoming vulnerable to potential cyber threads such as network intrusions. For safe network transaction and network security, intrusion detection system, firewalls, authentication, and other hardware and software are needed. The role of our system is to attempt to trap an opponent's presence on a compromised network. In literature survey, we observed that the most of the paper on IDS are on signature based or anomaly based. The hybrid intrusion detection system comprises signature and anomaly based approaches. In this paper, we attempt to implement the anomaly based intrusion detection system. The pattern based intrusion detection method is attempted for some application of this system. The proposed system is able to detect all types of attacks using anomaly based intrusion detection system and it detect Neptune, Teardrop, Warezclient and Snurf attack using pattern based intrusion detection technique. The system is not capable to detect all types of attacks using pattern based intrusion detection. The system is required to improve the signature or pattern based intrusion detection technique. The work can be extended to implement the complete hybrid system.

The rest of the paper is organized as follows: In Section II, we provide a brief description of methodologies and techniques of intrusion detection system. Section III deals literature survey. The detailed description of Support Vector Machine and normalization techniques is given in Section IV. Over all experimental setups of proposed system is presented in Section V. Experimental Results and analysis are presented in Section VI. Discussions on the additional applications of simulator are given in Section VII. The paper is concluded in Section VIII with future plan for extending this research.

2. METHODOLOGIES AND TECHNIQUES OF IDS

2.1. Methodologies of Intrusion Detection

There are three primary methodologies of Intrusion detection system such as signature-based, anomaly-based, and stateful protocol analysis.

2.1.1 Signature Based Intrusion Detection

Signature based detection is the simplest detection method. A signature of packet is a pattern or string that corresponds to a known threat. It is patterns of well-known attacks. Signature-based detection is the process of comparing pattern against observed events to identify possible incidents. It is effective for detecting known threats but it is not effective for detecting unknown threats. It is also incapable to track and understand the state of complex communications. It cannot detect the multiple events because it lacks the ability of remembering previous request [1] [2].

2.1.2 Anomaly Based Detection

Anomaly Based Detection is profile based detection which presents the normal behavior of user, hosts, network connections, or applications. The computer or network profiles can be generated by monitoring the characteristics of typical activity over a period of time. The current profile is matched with normal profile to identify significant deviations. Anomaly-based detection is the process of comparing definitions of what activity is considered normal against observed events to identify significant deviations. The Anomaly based intrusion detection system uses statistical methods to compare the characteristics of current activity to thresholds related to the profile. Profiles for anomaly-based detection can either be static or dynamic. It can be very effective at detecting previously unknown threats and produces many false positives [1] [2].

2.1.3 Stateful Protocol Analysis Based Intrusion Detection

Stateful Protocol Analysis Based Intrusion Detection is protocol profile based. Anomaly based detection uses user, host, or network specific profiles. It is relies on vendor developed universal profiles that specify how particular protocols should and should not be used. Stateful protocol analysis is the process of comparing predetermined profiles of generally accepted definitions of benign protocol activity for each protocol state against observed events to identify deviations. It is capable to understand and track the state of transport, network, and application protocols that have a notion of state. It can identify unexpected sequences of commands and perform authentication [1] [2].

2.2 Techniques of Intrusion Detection

There are two main techniques of Network Intrusion detection system as described below [1].

2.2.1 Anomaly Detection

The purpose of anomaly detection is to analyze the network and decide what is normal and what is not. Statistical measures are applied to events and it is decided whether they match the model of statistic of “normal”. After that it will generate reports and alerts for the events that are outside the probability window of normal.

2.2.2 Misuse Detection

The aim of misuse detection is that it must have prior knowledge of what an attack constitutes and then detect it. It cannot detect something, of which it does not have prior information. It requires constant updates with the new rules and is easier to deceive. Hence, anomaly detection is preferred over misuse detection [2]. Anomaly detection system can detect new attacks but sometimes a legitimate behavior can be wrongly considered to be an attack leading to false positive. The false positive and false negative rates cannot reduce in real time system.

3. LITERATURE SURVEY

Zonghua Zhang et al. [3] have modified the traditional SVM, RSVM and One class SVM based on the method of Online SVM. The proposed work is based on text processing model. The tf-idf text processing model have been used witch consider the time information and the correlation between the processes. The proposed modified algorithm outperforms conventional SVMs, RSVM in term of number of support vector and training time. The system can extend by applying of online training with unsupervised and incremental learning methods. **Yang Yi, Jiansheng Wu and Wei Xu** [4] have proposed RS-ISVM which is combined with modified U-RBF kernel function. They have embedded the mean and mean square difference values of feature attributes in RBF kernel function. The reserved set strategy is applied which keeps samples those are more likely to the support vectors. The concentric circle method is suggested by authors to select samples to shorten the training time. **Yu-xin and WU Mu-qing** [5] have proposed a new intrusion detection technology to improve the classification accuracy and reduce the training time. The proposed work combines multiclass support vector machine and feature extraction technology. The intrusion detection system has two phases. The first phase is used to project the original training data into kernel fisher discriminant analysis space. In second phase, fuzzy clustering technology is used to cluster the projected data. Authors have concluded that suitable feature extraction technology can improve the overall performance of intrusion detection and increase the efficiency of detection. **Weijun li and Zhenyu Liu [6]** have developed the intrusion detection system using the min-max normalization method. The intrusion data of KDD99 is normalized before passing to the SVM. They found that normalization can help to speed up calculation and get a good performance of classifier. They have analyzed and simulated some of the normalization methods in this paper. Their conclusion is that min-max normalization has better accuracy and good performance in speed. **Carlos A. Catania, Facundo Bromberg and Carlos García Garino** [7] have proposed robust SbSVM approach for intrusion detection. An autonomous labeling approach for dealing with situations where class distribution does not present the imbalance required for SVM algorithms. The autonomous labeling process is made by SNORT. **Yinhui Li, Jingbo Xia et al.** [8] have proposed a pipeline IDS via a series of machine learning strategies. The compact data set is used by clustering redundant data in KDD Cup dataset. They have proposed the Gradual Feature Removal (GFR) feature reduction method to reduce the feature dimension from 41 to 19 to seize the key features from dataset. This method owns reasonable property in precise feature selection and shows advantage in the experimental result. **Taeshik Shon and Jongsub Moon** [9] have proposed the real time intrusion detection system using Enhanced SVM, which combines soft margin SVM using supervised learning and one-class SVM approach using the unsupervised learning. Authors have applied the one-class SVM approach using unsupervised learning for detecting anomalies. It is difficult to use the one-class SVM in the real world, due to its high false positive rate. The supplement components, such as, field selection using GA, packet filtering using PTF, packet profiling using SOFM and packet flow-based data preprocessing are used. They have suggested that the use of data mining and data clustering methods help for more advanced profiling of normal packet. **Shi-Jinn Horng et al.**

[10] have proposed an SVM-based intrusion detection system with BIRCH hierarchical clustering for data preprocessing. The BIRCH hierarchical clustering algorithm is used to produce a reduced training dataset from the KDD Cup 1999. The proposed algorithm eliminates unimportant features to shorten the training time. The proposed system has better performance in the detection of DoS and Probe attacks. **Kamran Shaf and Hussein A. Abbass** [11] presented UCSSE framework for real time extraction of general rules using Supervised Learning Classifier. The algorithm used in this frame work automatically identifies and extract signatures for normal and intrusion activities. The UCSSE framework has hybrid detection capabilities **Muamer N. Mohammed and Norrozila Sulaiman** [12] has proposed intrusion detection system for wireless local network using support vector machine. Authors have suggested that the good intrusion detection system should perform with a high precision, a high recall, a lower false positive rate and a lower false negative rate. A poor intrusion detection system may have a high precision but a high false negative rate.

4. SUPPORT VECTOR MACHINE AND NORMALIZATION TECHNIQUES

4.1. Theoretical Background of Support Vector machine

The support vector machine is machine learning algorithm which is presented by Vladimir Vapnik and colleagues Bernhard Boser Isabelle Guyon. It is a new method used for classification of both the linear and nonlinear data. It searches the linear optimal separating Hyperplane which is called a decision boundary separating the tuples of one class from another class. The SVM finds this hyperplane using support vectors from training tuples and margin defined by support vectors. It is highly accurate and has ability to model complex nonlinear decision boundaries. The support vector machine provides a compact description of learned model. It is like multilayer perceptrons and radial basis function network which can be used for nonlinear regression pattern classification. The statistical learning approach is used to learn the SVM. It does not incorporate problem domain and provides a good generalization. The theoretical background of support vector machine is described in this section.

Let data D be ($\mathbf{X}_1$, Y_1)… ($\mathbf{X}_{|D|}$, $Y_{|D|}$), where $\mathbf{X}_i$ is the set of training tuples associated with Y_i class labels. The value of Yi may be either +1 or -1 (i.e Y_i € {+1,-1}. The 2-D data are linearly separable because a straight line can be drawn to separate all of the tuples of class +1 from all of the tuples of class -1.We can draw infinite number of separating lines to separate two classes. It searches hyperplane with largest margin called as Maximum Marginal Hyperplane which is more accurate than hyperplane with small margin. The MMH is linear class boundary which one that will have minimum classification error on unseen test data. The associated margin gives the largest separation between classes. The separating hyperplane can be written as below [13].

$$W.X + b = 0 \tag{1}$$

Where W = {w_1, w_2… w_n} is a weight vector and b a scalar (bias). For 2-D of tuples X={x_1, x_2} where x_1 and x_2 are the values of attribute. If we think of b as an additional weight vector w0, the above hyperplane can be rewritten as below.

$$W_0 + W_1X_1 + W_2X_2 = 0 \tag{2}$$

Following condition satisfies for the point that lies above the hyperplane.

$$W_{0+}W_1X_{1+}W_2X_2 > 0 \tag{3}$$

And any point that lies below the separating hyperplane satisfies

$$W_{0+}W_1X_{1+}W_2X_2 < 0 \tag{4}$$

The weights can adjusted so that the hyperplane defining the sides of the margin can be written as

$$H1\colon W_{0+}W_1X_{1+}W_2X_2 \geq 0\ , for\ y_{i=} + 1 \quad \text{And}$$

$$H2\colon W_{0+}W_1X_{1+}W_2X_2 \leq 1, for\ y_{i=} - 1 \tag{5}$$

Any tuple that falls on or above H1 belongs to class +1 and any tuple that falls on or below e H2 belongs to class -1. By combing two inequalities the following equation can be rewritten as.

$$Y_i(W_{0+}W_1X_{1+}W_2X_2) \geq 1, for\ all\ i \tag{6}$$

Any training tuples that fall on hyperplane H_1 or H_2 are support vectors. The support vectors give most information regarding the classification, but are most difficult to classify. The size of the maximum margin can obtain by the formulae. The distance from the separating hyperplane to any point on H_1 is 1/||W||, where the ||W|| is the Euclidean norm of W, that is$\sqrt{W.W}$. This is equal to the distance from H_2 to separating hyperplane. Therefore the maximal size of maximum margin is 2/||W||. We can find the MMH and Support Vector by using some Mathematics tricks. The equation, $Y_i(W_{0+}W_1X_{1+}W_2X_2) \geq 1$, which is called as a Constrained Quadratic Optimization problem. The formula $Y_i(W_{0+}W_1X_{1+}W_2X_2) \geq 1$, for all I, can be rewritten using Lagrangian formulization and Karush-Kuhn-Tucker (KKT) condition to find the MMH and support vector for classifying the small dataset.

For large data, special and more efficient algorithm for training SVMs can be used. Once we found the support vector and MMH, we can train support vector machine. The MMH is linear class boundary. By using the trained support vector machine we can test it using the following Lagrangian equation.

$$\mathrm{d}(\mathrm{X}^T) = \mathrm{b}_0 + \sum_{l=1}^{\mathrm{l}} \left(\alpha_i Y_i \mathrm{X}_\mathrm{i}\ \mathrm{X}^T\right) \tag{7}$$

Where Y_i is a class label of support vector Xi; X^T is the test tuple; α_i is Lagrangian multipliers and b_0 are numeric parameter determined automatically by optimization and l is the number of support vectors. Given a test tuple, X^T, we plug it into equation and then check to see the sign of the result. This tells us on which the side the hyperplane the test tuple falls. If the sign is positive, then X^T falls on or above the MMH, and so the SVM predicts that X^T belongs to class +1. If the result is negative it belongs to class -1.

Nonlinearly separable data cannot classify using the linear SVM, because we cannot draw the straight line between the classes. We can extend the linear SVMs to obtain the nonlinear SVMs in two steps. In the first step, we transform the original input data into a higher dimensional space using nonlinear mapping function. In second step, the linear separating hyperplane is searched in the new space. The choosing of the mapping function and computation cost are some problem with this transformation. For classification of the test vector X^T, we have to compute its dot product with every one of the support vector (Xi).The Lagrangian formula contains the product of X_i and X^T for finding the MMH and support vector when the data is nonlinear. In training, we have to compute a similar dot product several times in order to find the MMH and support vectors. The training tuple appear only in the form of dot products, Φ (x_i). Φ (X_u), where Φ(x) is nonlinear mapping function applied to transform the training tuples. These computations are very expensive. Instead of computing the dot product on the transformed data tuples, we can use mathematics trick called as kernel. Applying the kernel function, K (X_i, Xj), to the lower dimensioned original input data is mathematically equivalent to dot products. That is, K (X_i, X_j) = Φ (x_i). Φ (x_j). In other words, we can replace Φ (x_i). Φ (x_j) with K (X_i, Xj) in the training algorithm. After applying this trick, we can then proceed to find a maximal separating hyperplane. The procedure is same as described as above, although it involves placing a user determined upper bound C, on the Lagrange multiplier $α_i$. These properties of kernel are used to replace the dot product of X_i support vector and X^T test vector. The basic main kernel functions are as given below. Each of these results in a different nonlinear classifier in input space.
Polynomial kernel of degree h:

$$\mathrm{K}(\mathrm{X_i}, \mathrm{X_j}) = (X_i . X_j + 1)^h \qquad (8)$$

Gaussian Radial Basis function kernel:

$$\mathrm{K}(\mathrm{X_i}, \mathrm{X_j}) = -\frac{1}{2\sigma} \left(e^{-(X_i - X_j)^2}\right) \qquad (9)$$

Sigmoid Kernel:

$$tanh = (\kappa \mathrm{x_I} - \mathrm{Xj} - \delta), \text{for som } \kappa > 0 \qquad (10)$$

The SVM with the Gaussian radial basis function is same as Radial basis function neural network and SVM with sigmoid kernel is same as multilayer perceptron with no hidden layer. The SVM training always finds a global solution, unlike neural networks such as backpropogation, where many local minima usually exist. Figure1 display the architecture of a support vector Machine. The complexity of conventional approach is controlled by keeping the number of hidden layer small [14]. The support vector machine offer a solution to the design of learning machine by controlling model complexity independently of dimensionality.

4.2 Methods of Data Normalization

The normalization is very important part of preprocessing of training data set in intrusion detection. The normalization of training dataset can help speed up the learning phase of classifier. Some features have very large difference between the maximum and minimum value in normal and attack. The complicated normalization algorithm or methods will require more processing time. The choice of fast and effective normalization algorithm is to be required. Following normalization algorithm is available to implement the normalization process [3][6].

4.2.1 Zero-Mean normalization method

This method is based on the mean and standard deviation. Standard deviation is calculated by following formula.

$$X_{std} = \sqrt{\left[\left[\frac{1}{(n-1)}\right] \sum_{n=1}^{n} (\mathrm{X_i} - \mathrm{X_{mean}})^{\ 2}\right]} \qquad (11)$$

$\mathrm{X_{mean}}$ is mean of data and the normalization equation is as follow.

$$Xi' = \frac{X_i - X_{mean}}{X_{std}} \qquad (12)$$

This method is used when we have outliers that have great effect on the range of the data or when we do not know the actual minimum and maximum of input data.

4.2.2 Sigmoidal normalization method

This method used to transforms the input data nonlinearly into the range -1 to 1.

$$X_i' = \frac{1 - e^{-a}}{1 + e^{-a}} \qquad (13)$$

Where a is

$$a = \frac{\mathrm{X_i} - \mathrm{X_{mean}}}{\mathrm{X_{std}}}$$

In this method outlier points are compressed along the tails of the sigmoidal function. It can be used when you have outlier data points. It prevents the most commonly occurring values from being compressed into essentially the same values without losing the ability to represent very large outlier values.

4.2.3. Softmax Normalization

This method reaches softly toward its maximum and minimum value. The output range of this transformation is from 0 to 1 and it assures that no value lies outside this range. This method is more or less linear in the middle range.

$$X_i' = \frac{1}{1+e^{-a}} \tag{14}$$

Where a is

$$a = \frac{\mathrm{X_i - X_{mean}}}{\mathrm{X_{std}}}$$

4.2.4 Decimal Scaling methods

The output of this method is in range between -1 to 1. It normalizes the data by moving the decimal point of values. The movement of decimal point is depends on the maximum absolute value.

$$X_i' = \frac{X_i}{10^j} \tag{15}$$

Where, j is the smallest integer such that MAX ($|X_i'|$) <1. This Scaling is useful when attributes values are greater than 1 in absolute value.

4.2.5 Max Normalization

The normalization equation of this method is given as follow.

$$X_i = \frac{X_i}{X_{max}} \quad ,X_{max} = \max_{1\le i\le N} X_i \tag{16}$$

In this method the normalization value range may be large when the maximal positive value is too small and the minimum negative value is too small.

4.2.6 Min-Max Normalization

This method performs a linear transformation on the original dataset into the specified interval ($\mathrm{New_{min,}}$ $\mathrm{New_{max}}$).

$$X_i = New_{min} + (New_{max} - New_{min}) * [\frac{X_i - X_{min}}{X_{max} - X_{min}}] \tag{18}$$

$$\mathrm{X_{max}} = \max_{1\le i\le N} X_i \ ,X_{min} = \min_{1\le i\le N} X_i$$

The scaling of data x from ($\mathrm{X_{min}}$, $\mathrm{X_{max}}$) to ($\mathrm{New_{min,}}$ $\mathrm{New_{max)}}$ is done using min-max method. It preserves the all relationships of the data values. The Zero-Mean Sigmoidal and Softmax normalization require much more time to calculate the mean and standard deviation. The overflow problems are also worse in these methods. Max and Scaling normalization are same in computation time. Max normalization has simple rules and adjustable range and has better performance than Zero Mean method [3].

5. OVER ALL FRAMEWORK OF PROPOSED SIMULATOR.

5.1 Flow of the system

SVM in intrusion detection system has many advantages as mentioned above. Thus, simulator uses features of SVM for detecting the intrusions as well as will also attempt to block the packets. KDDCUP'99 dataset is used as a training dataset. It is very large data set which encompasses a wide variety of data types. It is used to enable researchers in knowledge discovery and data mining to scale existing and future data analysis algorithms to very large and complex data sets. The figure 2 shows the flow of the algorithm which involved different processes of the system.

5.2 Experiment methods

The proposed system is a Network based Intrusion Detection System (NIDS) is developed using Support Vector Machine (SVM) (supervised learning).An attempt has been made to reduce or eliminate the limitations which are present in the existing methodologies for intrusion detection. This system has been implemented using Java (jdk 1.7). Following are the steps which are followed system to operate precisely.

5.2.1. Login

The system has been given added security using authentication. The user of the system is a network administrator. If the user fails to log into the system, he/she will not be given access to the functionalities of the system which are explained further.

5.2.2. Importing data for preprocessing and normalization

In this stage, we have to import the dataset which has to be given for the training of Support Vector Machine Model. The dataset used in our system is KDD Dataset. KDDCUP'99 dataset is used as a training input. This dataset is used to enable researchers in knowledge discovery and data mining to scale existing and future data analysis algorithms to very large and complex data sets. The dataset is not used for training purpose. Only a part of the Dataset has been used as the training data. Large margin classifiers are known to be sensitive to the way features are scaled. Therefore it is essential to normalize either the data or the kernel itself. The accuracy of an SVM can severely degrade if the data is not normalized. This system contains continues feature values which are measured in a different scale and has a different range of possible values. Normalization places the values of numeric attributes on the same scale and prevents attributes with a large original scale from biasing the solution. Normalization minimizes overflows and underflows errors. Normalization can also bring the numerical attributes to the same scale (0, 1) as the exploded categorical data.

The Min-Max method have implemented for normalization of the dataset of the KDDCUP'99 for training SVM. The KDD Dataset contains string as well as numerical values which cannot be understood by SVM. These string values are hence converted to numerical values. All the numerical double values are big and hence normalization is done to get the double values in the range of 0 to 1.After normalization, values are then converted to double values. The ranges of all the values from KDD Dataset are from the range 0 to 1 and this is necessary to improve the efficiency of SVM.

5.2.3 Training and plotting the vector space

Once the normalization is completed, the SVM is we trained for the normalized values. The normalized dataset is divided into two sets namely, training set and testing set. The user is supposed to provide with the training and testing percentage. Depending on the user provided percentages, the normalized set is divided. The selection of dataset records for training and testing is done randomly. After the division, the training is done and then the actual vector space will be plotted. The SVM classifier is a non-linear classifier and it classifies the dataset records into +1 and 0 regions depending on whether it is an intrusion or not an intrusion respectively. A SVM training algorithm builds a model that assigns new examples into one category or the other. A set of features that describes one case is called a vector. SVM modeling is used to find the optimal cases with one category of the target variable which are on hyper plane that separates clusters of vector in such a way that one side of the plane and cases with the other category are on the other side of the plane. This is called as Vector Space. For training, we are using an open source SVM package called Libsvm. The kernel used for training is Gaussian RBF (Radial Basis Function) Kernel. RBF is a reasonable choice because it nonlinearly maps samples into a higher dimensional space. It can handle the case when the relation between class labels and attributes is nonlinear. SVM can easily solve input complexity problems using RBF kernel.

5.2.4 Cross Validation and Prediction

The percentage of testing data from KDDCUP'99 is decided before training the SVM. The feature extraction method applied on the testing dataset. Feature extraction is an attribute reduction process. The feature extraction process produces a much smaller and richer set of attributes. Feature extraction projects a data set with higher dimensionality onto a smaller number of dimensions. Once the features are extracted they are given to the SVM. These features are compared with vector space and according to which side they lie SVM takes the decision whether the packet is intrusion or not. Accordingly it will block or allow the packets. The testing procedure is used to cross check the class labels which are known with the ones predicted by the classifier constructed by this system. The prediction accuracy obtained from the "unknown" set more precisely reflects the performance on classifying an independent data set. It is also called cross-validation. This module shows us how accurately or precisely the constructed classifier predicts the class labels (0 or +1).

6. EXPERIMENTAL RESULTS

The experiments are conducted to verify the effectiveness of proposed system. Firstly, the subset obtained in Section 5.2.2 is randomly divided into two subsets, each subset contains both the data of normal and abnormal class, one is as the training set, and the other is as the test set. Secondly, randomly select data sets from the testing and training subset, each set contains 145-150 normal and attacks samples. Each subset is tested with trained SVM. It is found that the accuracy of the system is 88.27% for each testing set. The accuracy of the system is shown by system itself in message form. We can also plot the classification accuracy in form of 3D Bar Chat or Pie Chart

.

7. ADDITIONAL APPLICATIONS OF THE SIMULATOR

Two applications have developed in order to demonstrate the use of SVM for IDPS namely Attack and FTP client and server application.

7.1 Attack Application

This application has been developed to send different attack patterns to the machine having the SVM IDPS running on it. Taking various features into consideration SVM decides whether it is an intrusion or not. The features that classifier is considering are Source address, Destination address, Duration, Protocol type, Service, Source bytes, Destination bytes, Count and 4 more fields. We can send segregated attack patterns, normal packet patterns or we can send an attack having all various attack patterns and normal packet patterns mixed. As this system is using anomaly based detection whenever new attacks are developed they can be detected by the system as opposed to misuse based detection technique. Whenever an attack pattern is sent to the system having IDPS, it detects the attack pattern and prevents it. Figure 3 shows the options to generate the different patterns of attack and detection of the same attack. It also sets on an alarm which will alert the network administrator even if he is not present near the system.

7.2 FTP Client and Server

This application is just used to demonstrate the correctness and working of the SVM classifier depending on or considering a selected feature. Here only one feature is considered that is the size of the file to be downloaded or uploaded as opposed to the features which considered in the attack application. FTP Server should have the IDPS running on it and FTP Client can upload as well as download files from the FTP Server. In case of download, the IDPS on FTP Server allows or denies the access to the file depending upon the size of the file. If the IDPS is trained for a particular size it will allow the Client to download the file of that size. If not, it will deny access to Client and it will set on an alarm to alert the network administrator. Similarly in the case of client uploading files, the FTP Server denies any file which is being uploaded by the Client. After the FTP Server is trained for the size of the file which was being uploaded, the FTP Server accepts the file of trained size from the Client. Log file are text files that will be automatically created by the system whenever the packets are analyzed and the decision is taken. The network administrator is supposed to train the system and he can see and maintain the logs of the IDPS system.

8. CONCLUSION

In this paper a framework for simulation of intrusion detection is presented. The Min-Max method of normalization is used to normalize the training and testing dataset. The system provides the facility to divide normalized dataset into two sets as training and testing sets as per user's choice in percentage. The RBF kernel based SVM is used to classify the KDDCUP'99 dataset in term of normal and abnormal classes. The attempt is made to develop the hybrid intrusion detection. Using this system, user can generate the attack on him for testing the system. The signature or pattern based intrusion detection module is used to detect this type of novel attack. The proposed Simulator is used to block unauthorized user and who try to access the large sized PDF file from the server. The entire simulator is very user friendly, self-explanatory and hybrid. It has the facility to plot the 3D Bar Chart and Pie Chart automatically. User need not to use the other external software for same. In the presented work an efficient intrusion detection system have designed for wired network. The system is demonstrating reactions to different attacks like Teardrop attack, Neptune attack, Smurf attack, Warezclient attack. The classification accuracy of the system is 88.27%.The implementation language java has given wide portability as java is platform independent. The simulator described in this paper is off line. The simulator does not work for real time packet analysis. Due to promising experimental results and application we plan to extend the simulator for real time intrusion detection.

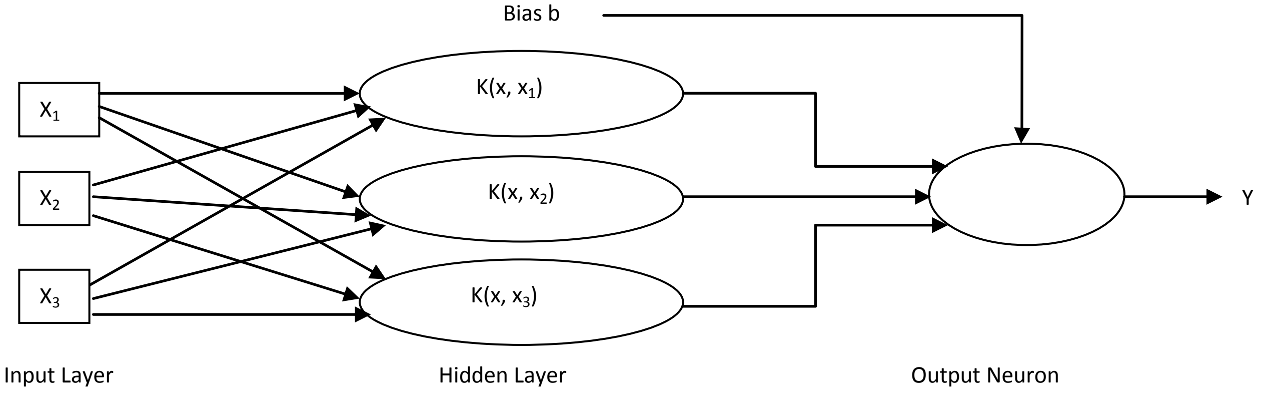


Fig 1: General Architecture of Support Vector Machine.

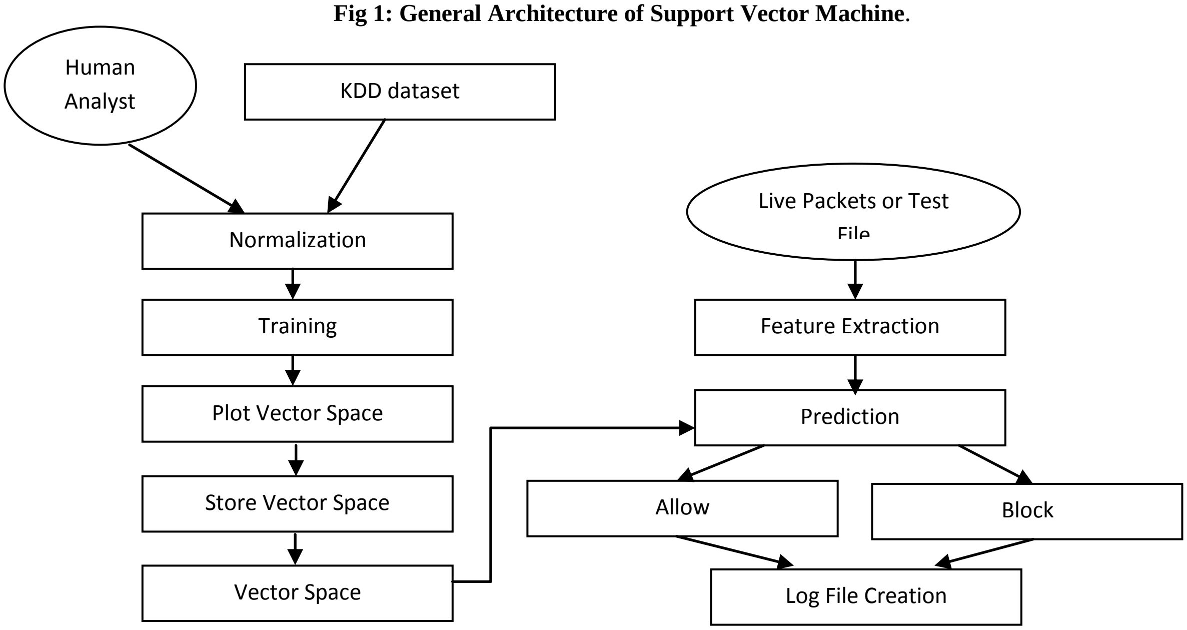
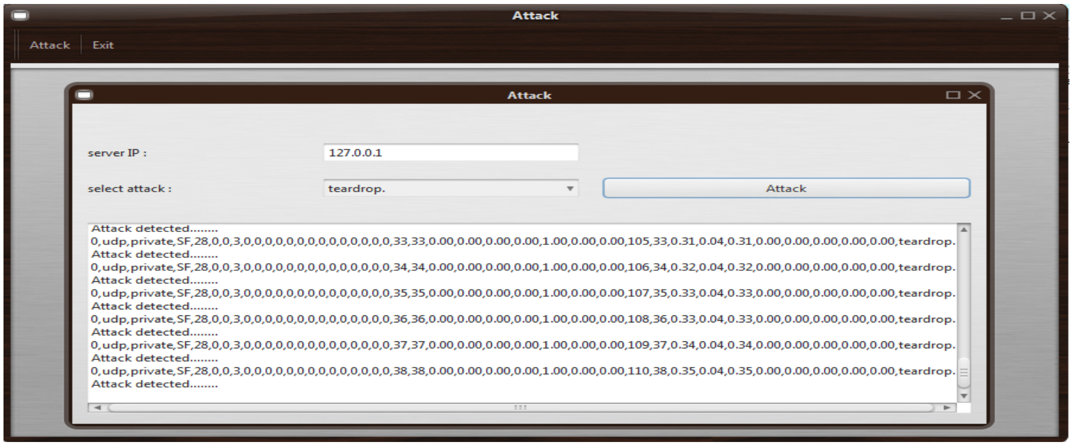


Fig 2: Flowchart of Proposed Intrusion Detection System Simulator.



server IP : 127.0.0.1

select attack : teardrop.

```
Attack detected........
0,udp,private,SF,28,0,0,3,0,0,0,0,0,0,0,0,0,0,0,0,0,0,33,33,0.00,0.00,0.00,0.00,1.00,0.00,0.00,105,33,0.31,0.04,0.31,0.00,0.00,0.00,0.00,0.00,teardrop.
Attack detected........
0,udp,private,SF,28,0,0,3,0,0,0,0,0,0,0,0,0,0,0,0,0,0,34,34,0.00,0.00,0.00,0.00,1.00,0.00,0.00,106,34,0.32,0.04,0.32,0.00,0.00,0.00,0.00,0.00,teardrop.
Attack detected........
0,udp,private,SF,28,0,0,3,0,0,0,0,0,0,0,0,0,0,0,0,0,0,35,35,0.00,0.00,0.00,0.00,1.00,0.00,0.00,107,35,0.33,0.04,0.33,0.00,0.00,0.00,0.00,0.00,teardrop.
Attack detected........
0,udp,private,SF,28,0,0,3,0,0,0,0,0,0,0,0,0,0,0,0,0,0,36,36,0.00,0.00,0.00,0.00,1.00,0.00,0.00,108,36,0.33,0.04,0.33,0.00,0.00,0.00,0.00,0.00,teardrop.
Attack detected........
0,udp,private,SF,28,0,0,3,0,0,0,0,0,0,0,0,0,0,0,0,0,0,37,37,0.00,0.00,0.00,0.00,1.00,0.00,0.00,109,37,0.34,0.04,0.34,0.00,0.00,0.00,0.00,0.00,teardrop.
Attack detected........
0,udp,private,SF,28,0,0,3,0,0,0,0,0,0,0,0,0,0,0,0,0,0,38,38,0.00,0.00,0.00,0.00,1.00,0.00,0.00,110,38,0.35,0.04,0.35,0.00,0.00,0.00,0.00,0.00,teardrop.
Attack detected........
```

Fig 3: The Snapshot to show the details of detected attack packets

9. REFERENCES

[1] Rafeeq Ur Rehman. Intrusion Detection Systems with Snort Advanced IDS Techniques Using Snort, Apache, MySQL, PHP, and ACID. Prentice Hall PTR Upper Saddle River, New Jersey (Report).

[2] D.P.Gaikwad and R.C.Thool 2010. A Survey on Architecture Taxonomy and Products of Intrusion Detection System. Proc. of the International Conference on Computer Applications (ICCA) 2010, DOI: 10.3850/978-981-08-7304-2_0382.

[3] Zonghua Zhang and Hong Shen 2005. Application of online-training SVMs for real-time intrusion detection with different considerations. Elsevier B.V, Computer Communications 28(2005), pp.1428–1442.

[4] Yang Yi, Jiansheng Wu and Wei Xu 2012.Incremental SVM based on reserved set for network intrusion detection. Journal of Expert Systems with Applications,DOI:10.1016/j.eswa.2010.12.141.

[5] WE1 Yu-xin and WU Mu-qing 2008. KFDA and clustering based multiclass SVM for Intrusion Detection. The Journal of China Universities of posts and Telecommunications, Volume 15, Issue 1, March 2008.

[6] Weijun li1and Zhenyu Liu 2011. A method of SVM with Normalization in Intrusion Detection. Elsevier, Procedia Environmental Sciences 11 (2011), pp. 256 – 262,DOI:10.1016/j.proenv. 2011.12.040.

[7] Carlos A. Catania, Facundo Bromberg and Carlos García Garino 2011. An autonomous labeling approach to support vector machines algorithms for network traffic anomaly detection. Elsevier, International Journal of Expert Systems with Applications 39 (2012), pp.1822–1829, DOI:10.1016/j.eswa.2011.08.068.

[8] Yinhui Li, Jingbo Xia, Silan Zhang, Jiakai Yan, Xiaochuan Ai and Kuobin Dai 2011. An efficient intrusion detection system based on support vector machines and gradually features removal method. Expert Systems with Applications 39,(2012),pp.424–430,DOI:10.1016/j.eswa.2011.07.032.

[9] Taeshik Shon and Jongsub Moon 2007. A hybrid machine learning approach to network anomaly detection. Elsevier, Information Sciences 177 (2007), pp. 3799–3821, DOI:10.1016/j.ins.2007.03.025

[10] Shi-Jinn Horng, Ming-Yang Su, Yuan-Hsin Chen, Tzong-Wann Kao, Rong-Jian Chen, Jui-Lin Lai and Citra Dwi Perkasa 2011. A novel intrusion detection system based on hierarchical clustering and support vector machines. Elsevier, International Journal of Expert Systems with Applications 38 (2011), pp.306–313, DOI: 10. 1016/j.eswa.2010.06.066.

[11] Kamran Shaf and Hussein A. Abbass 2009. An adaptive genetic-based signature learning system for intrusion detection. Elsevier, International Journal of Expert Systems with Applications 36 (2009), pp.12036–12043, DOI:10.1016/j.eswa.2009.03.036.

[12] Muamer N. Mohammed and Norrozila Sulaiman 2012. Intrusion Detection System Based on SVM for WLAN. Elsevier, Procedia Technology 1 (2012), pp.313–317, DOI: 10.1016/j.protcy.2012.02.066.

[13] Jiawei Hans and Micheline Kamber. Data Mining: Concepts and Techniques. Elsevier, Second Edition.

[14] Simon Haykin. Neural Network: A Comprehensive Foundation, Prentice Hall, Second Edition.