ISSN: 2278 – 1323
International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)
Volume 2 Issue 11, November 2013

Evaluation of Network Intrusion Detection System using PCA and NBA

Aradhana Srivastav, Pankaj Kumar, Rajkumar Goel

***Abstract* — The growing age of Information technology and telecommunication, it becomes very necessary to provide a higher level of security to computer network and private information from intruders, who play a very essential role to ruin a system and hack the private information. Therefore, Intrusion Detection System (IDS) is still an interesting area of research in computer network and its related security. IDS is a collection of applications, which is used to detect the intrusions by collecting information from different networks and systems, to analyze symptoms of intrusions.**

In this paper, Principal Component Analysis (PCA) and Network Behavior Analysis (NBA) with KDD cup 99 dataset are used to detect the new attack as well as existing attacks or intrusions. The objective of PCA is to dimension reduction of data set and NBA is used for analyze the behavior of network, while KDD cup99 dataset is used for evaluation of training and testing dataset.

The main objective of this paper is evaluation of malicious data set, and to find out the intrusions. On the basis of evaluation result, we are defining the type of intrusions and also show the experimental result and analysis.

***Index Terms*— KDD cup99 data set, IDS, NIDS, NBA, and PCA**

I. INTRODUCTION

Security is becoming a major challenge in network security. Nowadays Intrusion detection systems (IDS) are very important for every information technology. Various researchers have been done lots of research work in the field of Intrusion Detection System (IDS) using Principal Component Analysis (PCA) and Network Behavior Analysis (NBA). But perfect IDS have still not been found and it stays a hot and challenging area in computer security research.

Keeping these things in mind we are trying to develop an Intrusion Detection System (IDS) using principle component analysis having the characteristics of anomaly-based detection and signature-based detection. Thus, for working in this direction we have studied different research papers regarding intrusion detection system. In this area, a lot of research works is done but either with individually signature-based IDS or Anomaly-based IDS. It is not sufficient to provide security over network or within organization.

Thus, using different components like Neural Network (NN), PCA, NBA etc., we have developed a model for intrusion detection in the area of computer networks security.

In 1987, the concept of intrusion detection systems was first introduced by James Anderson [11] [12] and then intrusion detection model was developed by Dorothy Denning [2].

In the subsequent years, an ever-increasing number of research prototypes are exploring. Intrusion detection has become a mature industry and a proven technology. Nearly all of the easy problems have been solved [19] [21] [20]. However, approaches used in intrusion detection such as those relying on statistical techniques, mobile agents, neural networks, artificial immunity, etc. are essentially based on the observation of events and their analysis. Therefore, data collections are constituted as the first step for most intrusion detection systems. Nowadays, these data are generally characterized by their elevated volume, which make it difficult to be analysed. In fact, most current intrusion detection methods cannot process large amounts of audit data for real-time operations and it seems better to have a new information content of user behaviours, emphasizing the significant features.

IDSs detect computer network behaviour as normal or abnormal but cannot identify the type of attacks. Our model is designed to give a solution for these letter problems, mainly the reduction of huge amount of audited data and identification of both attack types and normal user profiles, and thus to improve detection rates.

An Intrusion Detection System plays as a key technique to detect the intrusions in the area of information technology. An IDS is the process of detecting the events occurring in a network or computer system and analyse them for symptoms of intrusion. These types of unwanted symptoms are as called intrusion.

Intrusions are which compromise with the integrity, confidentiality, availability of the system and to get the private information or hack the system from the system or networks. Intrusions are created by attackers to access the system from the internet. In today's world, the Internet is an important part of our life. People cannot think of a single moment without the existence of the Internet. With the increasing involvement of the Internet in our daily life, it is very important to make it secure. This raises important issues with regards to security.

An Intrusion Detection System (IDS) inspects all inbound and outbound network activity and identifies suspicious patterns that may indicate a network or system attack from someone attempting to break into or compromise a system [17]. Figure 1 shows the taxonomy of IDS. IDS have two main detection techniques.

Anomaly-based Detection: it is a system for detecting computer intrusions and misuse by monitoring system behaviour and classifying it as either *normal* or *anomalous.*

Signature-based Detection: it is a system for detecting computer intrusions and misuse by *predefined signatures* or *patterns* in database.

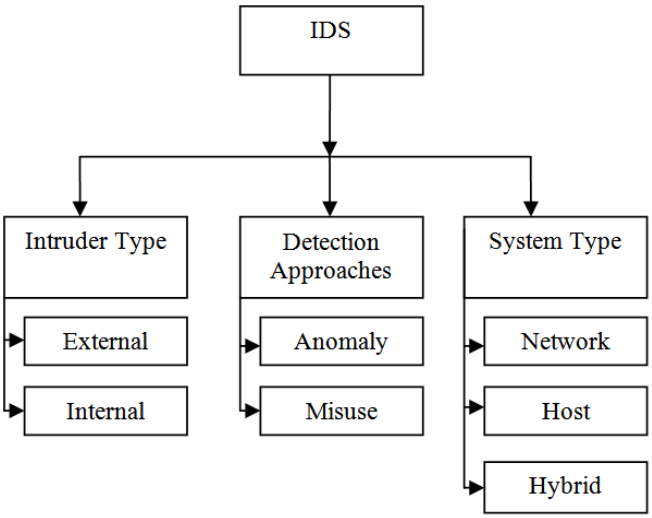

Figure 1: Taxonomy of IDS

Among the vast variety of techniques which have been used for the IDS, we are using the combination of PCA and Network behaviour Analysis (NBA) and KDD99 dataset applications have the ability to reduce the data set.

PCA: Principle Component Analysis is a way to identify patterns in data and expressing the data in such a way to highlight their similarity and differences. The main advantage of PCA to reduce the dimension and can compress the data without much loss of information. The working principle of PCA is shown in Figure 2 and describes as:

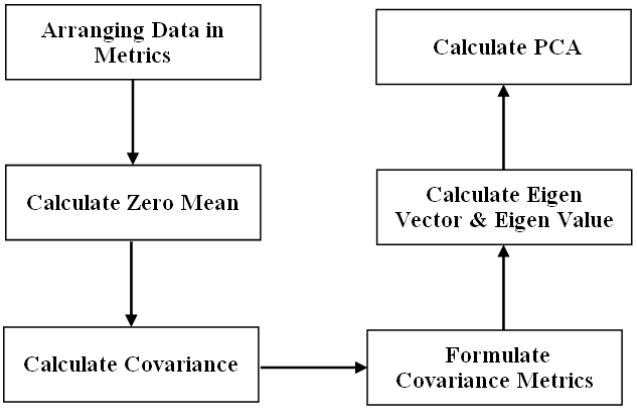

Figure 2: Working of PCA

We arrange the data in metrics form. After arranging the data in matrices form, we calculate the zero mean value (i.e. average value) of metrics and then calculate covariance, by which we can formulate covariance metrics for Eigen value and Eigen vector and then we apply PCA to get better result which can be shown in mathematical equation in following manner.

Given a set of observations or connections or data X_1, X_2……. X_n, where each observation is represented by a vector of length m, the data set is thus represented by a matrix $X_{m\times n}$ in Equation 1.

$$X_{m\times n} = \begin{bmatrix} x_{11} & \cdots & x_{11} \\ \vdots & \ddots & \vdots \\ x_{m1} & \cdots & x_{mn} \end{bmatrix} = [X_1, X_2, X_3 \ldots, X_n] \ldots (1)$$

The average observations (μ) are calculated as which is shown in Equation 2.

$$\mu = \begin{pmatrix} \bar{x}_1 \\ \vdots \\ \bar{x}_m \end{pmatrix} = \begin{pmatrix} \frac{1}{n}\sum_{i=1}^{n} X_{1_i} \\ \vdots \\ \frac{1}{n}\sum_{i=1}^{n} X_{m_i} \end{pmatrix} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2)$$

The standard deviation (Ø) is calculated to subtract equation 2 into equation 1 and defined in Equation 3.

$$\emptyset_i = X_i - \mu \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (3)$$

The sample covariance of matrix (V) of the data set is defined using the equation 3 which is shown in Equation 4.

$$V_{m\times m} = \frac{1}{n}\sum_{i=1}^{n} \phi_i \phi_i^t$$

$$V_{m\times m} = \frac{1}{n}\sum_{k=0}^{n} (X_i - \mu)(X_i - \mu)^t \ldots (4)$$

Where ϕ^t, is the transpose of matrix ϕ.
The covariance measures the degree of the linear relationship between two observation and data in dataset.
Eigenvector is corresponding to the highest Eigen values. Component analysis is defined by λ and no. of Eigen vectors is defined by g, and S, is the predefined ratio of variation in the new updated sub-space, which is total variation in the original space of dataset as described in Equation 5.we use the following equation:

$$\frac{\lambda_1 + \lambda_2 + \lambda_3 + \cdots + \lambda_g}{\sum_{i=1}^{m} \lambda_i} \geq S \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (5)$$

Finally, principle component g is calculated, which is required for next phase.

NBA: Network Behaviour Analysis is used to measure the normal flow of traffic in the network. Basically, NBA is always required whenever we talk about anomaly-based intrusion detection system to find out suspicious activity in the system. NBA works similar to Network-Based IDS however the difference between both is that Network-Based IDS are placed at the boundary between two networks and are responsible for monitoring a particular network segments. However, NBA detects an attack by monitoring network traffic for any unusual flows or sometimes they detect for any policy or rule violation.

NBA, detecting efficiency is varied with network behaviour, by which they are easily capable of detecting unknown attacks. Anomaly-based methodology they are capable of detecting those attacks which have some effects to the network.

KDD Dataset: KDD data set used for detection purpose in intrusion detection system, which was held in conjunction with KDD-99. It is offline database has different kinds of data set like- normal data and malicious data. This database contains a standard set of data to be audited, which includes a wide variety of intrusions simulated in a military network environment [12].

The remainder of this paper is organized as follows: Section II discusses about the related work on intrusion detection system and different types of approaches and tools. Section III discusses about the proposed methodologies. Section IV discusses about result and evaluation according the methodology Section V describes the conclusion and future work and in last section VI describe the references.

II. RELATED WORK

The concept of intrusion detection systems was first introduced by James Anderson [11] [12]. However, the field did not take off until 1987 when Dorothy Denning published an intrusion detection model [2]. The model detects the intrusion and suspicious activities of a system. In this model, Dening described that how we can found an intrusion in a system. IDS can be classified as:

Host Based IDS: In a host based IDS the host operating system or the application logs in the audit information. These audit information includes events like the use of identification and authentication mechanisms (logins etc.), file opens and program executions, admin activities etc. This audit is then analysed to detect trails of intrusion. Host-based IDS can monitor multiple computers simultaneously [22].

Network Based IDS: They reside on computer or any other appliance which is connected to an organization's network and there it looks for signs of attacks. In an organization's network they are installed at specific place from where it can watch the movement of traffic coming in and going out and whenever a predefined condition occurs it takes an action and notifies the appropriate administrator. It yields many more false-positive readings than host-based IDS [23].

*Host Intrusion Detection System***:** Host Intrusion Detection Systems (HIDS) are basically run on different types of hosts or devices in the network. HIDS mainly monitors the inbound and outbound packets from the device only and alert the administrator if any type of suspicious activity is detected. HIDS analyse network traffic and system-specific settings such as software calls, local security policy, local log audits, and more. A HIDS must be installed on each machine and requires configuration specific to that operating system.

Dorothy E. Denning [2] defined An Intrusion Detection Model of a real-time intrusion-detection expert system capable of detecting break-ins, penetrations, and other forms of computer abuse is described. The model is based on the hypothesis that security violations can be detected by monitoring a system's audit records for abnormal patterns of system usage. The model includes profiles for representing the behaviour of subjects with respect to objects in terms of metrics and statistical models, and rules for acquiring knowledge about this behaviour from audit records and for detecting anomalous behaviour. The model is independent of any particular system, application environment, system vulnerability, or type of intrusion, thereby providing a framework for a general-purpose intrusion detection expert system. The model was divided into six main components [2]:

a) *Subjects*: Initiators of activity on a target system normally users.
b) *Objects*: Resources managed by the system-files, commands, devices, etc.
c) *Audit records*: Generated by the target system in response to actions performed or attempted by subjects on objects-user login, command execution, file access, etc.
d) *Profiles*: Structures that characterize the behavior of subjects with respect to objects in terms of statistical metrics and models of observed activity. Profiles are automatically generated and initialized from templates.
e) *Anomaly records*: Generated when abnormal behavior is detected.
f) *Activity rules*: Actions taken when some condition is satisfied, which update profiles, detect abnormal behavior, relate anomalies to suspected intrusions, and produce reports.

Models works like rule based pattern matching. Whenever an audit record is generated, it matches against profile and apply related rule to update profile check for abnormal behaviour, and report anomalies detected. The model does not contain any special features for dealing with complex actions that exploit a known or suspected security flaw in the target system; indeed, it has no knowledge of the target system's security mechanisms or its deficiencies. Although a flaw-based detection mechanism may have some value, it would be considerably more complex and would be unable to cope with intrusions that exploit deficiencies that are not suspected or with personnel-related vulnerabilities. By detecting the intrusion, however, the security officer may be better able to locate vulnerabilities

J. Ryan, M. J. Lin and R. Miikkulainen [13] defines intrusion detection model using Neural Networks on signature-based detection. They gave NNID based an elegant solution to offline monitoring utilizing these user profiles. A back propagation neural network called NNID was trained in the identification task and tested on a system of 10 users. The concept behind this is that whenever a user used the system, user leaves a 'print', a neural network is used to identify each user and if a user's behaviour does not match with print, then an alert is send to system administrator of a possible security breech.

The model is based on these three ideas. NNID is a back propagation neural network trained to identify users based on what commands they use during a day. And at the end of the day security administrator checks the behaviour of user against user's normal behaviour, if it founds different than normal then an investigation is launched. The NNID model is implemented in a UNIX environment and consists of keeping logs of the commands executed, forming command histograms for each user, and learning the users" profiles from these histograms. NNID provides an elegant solution to off-line monitoring utilizing these user profiles.

There was possibility that this model does not work well as the number of user's will increase. Because with more users, the network have to make finer distinction and it would be difficult to maintain the same level of false alarms and number of false alarm will increase. This model was easy to train but it was expensive to run because system operates off-line on daily logs. The system was best for offline detection but not for real time intrusion detection. The system was trained in the identification task and tested experimentally on a system of 10 users. The system was 96% accurate in detecting unusual activity, with 7% false alarm rate. These results suggest that learning user profiles is an effective way for detecting intrusions.

W. Lee and S. Stolfo [16] defined an Intrusion Detection System (IDS) on Anomaly-based detection to protect

network from outsider attacks. The fundamental of this model is quite different. Here they take a data-centric point of view and consider intrusion detection as a data analysis process. The system sends a notification or takes an action should an intruder attempt to go past the security mechanism such as authentication, or firewall. Anomaly detection is about finding the normal usage patterns from the audit data, whereas misuse detection is about encoding and matching the intrusion patterns using the audit data. They used Classification, Link Analysis and Sequence Analysis for the data mining. They developed a systematic framework for designing, developing and evaluating intrusion detection systems. Specifically, the framework consists of a set of environment-independent guidelines and programs that can assist a system administrator or security officer:

- Select appropriate system features from audit data to build models for intrusion detection.
- Architect a hierarchical detector system from component detectors.
- Update and deploy new detection systems as needed.

Authors supply training data containing pre-labelled 'normal' and 'abnormal' sequences and used a sliding window to scan the normal traces and create a list of unique sequences of system calls. They call this list the "normal" list. Next, they scan each of the intrusion traces. For each sequence of system calls, they first look it up in the normal list. If an exact match can be found then the sequence is labelled as "normal". Otherwise it is labelled as "abnormal".

They performed experiment on send mail call data and on tcpdump data. And they use RIPPER [32], a rule learning program to train data. The following learning tasks were formulated to induce the rule sets for normal and abnormal system call sequences:

- Each record has n positional attributes, p_1, p_2... p_n, one for each of the system calls in a sequence of length n; plus a class label, "normal" or "abnormal".
- The training data is composed of normal sequences taken from 80% of the normal traces, plus the abnormal sequences from 2 traces of the sscp attacks, 1 trace of the syslog-local attack, and 1 trace of the syslog-remote attack.
- The testing data includes both normal and abnormal traces not used in the training data.

RIPPER outputs a set of if-then rules for the "minority" classes, and a default "true" rule for the remaining class. The following exemplar RIPPER rules were generated from the system call data:

Normal:	p_2=104, p_7=112
Normal:	p_6=19, p_7=105
Abnormal:	True

First line says that if p_2 is 104 and p_7 is 112 then the sequence is 'normal'. Second line says that if p_6 is 19 and p_7 is 105 then the sequence is 'normal'. Last line says that if none of the above, the sequence is 'abnormal'.

Using this approach normal behavior of a program execution can be established and used to detect its anomalous usage. The weakness of the model may be that the recorded (rote learned) normal sequence database may be too specific as it contains about ~1,500 entries. Same experiment is done on tcpdump data and found that if collected data set is not designed for security purpose then it cannot be used directly to build detection model. After seeing the output of both experiments we can conclude that in this paper authors proposed a systemic framework that employs data mining techniques for intrusion detection. This framework consists of classification, association rules, and frequency episodes programs, which can be used to (automatically) construct detection models. The experiments on send mail system call data and network tcpdump data demonstrated the effectiveness of classification models in detecting anomalies. The accuracy of the detection models depends on sufficient training data and the right feature set. Authors suggested that the association rules and frequent episodes algorithms can be used to compute the consistent patterns from audit data. These frequent patterns form an abstract summary of an audit trail, and therefore can be used to: guide the audit data gathering process; provide help for feature selection; and discover patterns of intrusions.

M. Moradi and M. Zulkernine [19] defined, Neural Network Based System for Intrusion Detection and Classification of Attacks which works on Multi Layer Perceptron (MLP) is used for intrusion detection based on an off-line analysis approach. While most of the previous studies have focused on classification of records in one of the two general classes - normal and attack, in this author aims to solve a multi class problem in which the type of attack is also detected by the neural network. Different neural network structures are analysed to find the optimal neural network with regards to the number of hidden layers. An early stopping validation method is also applied in the training phase to increase the generalization capability of the neural network. Author is used soft computing to remove the problem of rule based IDS. Soft computing is a general term for describing a set of optimization and processing techniques that are tolerant of imprecision and uncertainty. The principal constituents of soft computing techniques are Fuzzy Logic (FL), Artificial Neural Networks (ANNs), Probabilistic Reasoning (PR), and Genetic Algorithms (GAs). The idea behind the application of soft computing techniques and particularly ANNs in implementing IDSs is to include an intelligent agent in the system that is capable of disclosing the latent patterns in abnormal and normal connection audit records, and to generalize the patterns to new (and slightly different) connection records of the same class.

According to this paper Different structures of MLP are examined to find a minimal architecture that is reasonably capable of classification of network connection records. The results show that even an MLP with a single layer of hidden neurons can generate satisfactory classification results. Because the generalization capability of the IDS is critically important, the training procedure of the neural networks is carried out using a validation method that increases the generalization capability of the final neural network.

In this study author solve multi class problem and they used MATLAB Neural Network Toolbox to implement of the MLP networks. Using this tool one can define specifications like number of layers, number of neurons in each layer, activation functions of neurons in different layers, and number of training epochs. Then the training feature vectors and the corresponding desired outputs can be fed to the neural network to begin training. Authors used Error back- propagation algorithm for training and had 35 input neurons (equal to the dimension of the feature vector) and

three output neurons (equal to the number of classes). Number of the hidden layers and neurons in each were parameters used for the optimization of the architecture of the neural network.

Author applied the early stopping validation method which increased the generalization capability of the neural network and at the same time decreased the training time. It should be mentioned that the long training time of the neural network was mostly due to the huge number of training vectors of computation facilities and results were quite different as the number of layers increased. A two layer neural network was also successfully used for the classification of connection records. Although the classification results were slightly better in the three layer network, application of a less complicated neural network was more computationally and memory wise efficient. Author perform experiments using different number of layers and found that the designed system is capable of classifying records with about 91% accuracy with two hidden layers of neurons in the neural network and 87% accuracy with one hidden layer.

S. Mukkamala, G. Janoski and A. Sung [27] defined Intrusion Detection Using Neural Networks and Support Vector Machines on signature-based detection. The key ideas are to discover useful patterns or features that describe user behaviour on a system, and use the set of relevant features to build classifiers that can recognize anomalies and known intrusions.They compare the performance of neural networks based, and support vector machine based, systems for intrusion detection. In this paper the neural networks and support vector machines are trained with normal user activity and attack patterns. The data they used originated from MIT's Lincoln Labs. It was developed for KDD competition by DARPA and is considered a standard benchmark for intrusion detection evaluations and detects both anomalies and misuse. The approach is to train the neural networks or support vector machines to learn the normal behaviour and attack patterns; then significant deviations from normal behaviour are flagged as attacks.

Here IDSs maintains a set of historical profiles or recorded profiles for users, matches an audit record with fitting profile, updates the profile when it found needed, and generate alert when anomalies found. Authors first implement both model and then compare the result of both models output and found both the neural networks and SVMs deliver highly accurate results greater than 95% accuracy on testing set and show compatible level of performance. The training time for SVMs is significantly shorter, an advantage that becomes rather important in situations where retraining needs to be done quickly e.g., when new attack patterns are discovered. The running time of SVMs is also notably shorter. On the other hand, SVMs can only make binary classifications, which is a severe disadvantage where the intrusion detection system requires multiple-class identifications e.g., all 22 different types of attacks need to be differentiated.

At the end of paper author concluded that SVMs have great potential to be used in place of neural networks due to its scalability (large data sets and large number of features in patterns can easily overwhelm neural networks) and faster training and running time. On the other hand, neural networks have already proven to be useful in many IDSs, and are especially suited for multi-category classifications.

Sodiya and Akinwale [26] defined Intrusion Detection Systems (IDS) as systems that can detect external as well as internal both type of attack on network and undertake some measures to eliminate them. Intrusion detection system not only detects intrusion but also can detect new attack and send information to security for taking appropriate action. Intrusion-detection system sends attack information to related person via e-mail, pager or mobile messages.

Intrusion-detection system detects an intrusion by deviation from normal behaviour, from attack signature and they examine such features like:

- Network traffic
- Login failure
- Defragmentation

File activity or even user location for signs of attacks [10].

Li Yang, Bi. Fang, Y. Chen, Li. Guo [33] defines an Intrusion Detection Model Based on Feature Selection and Maximum Entropy Model. In this model, firstly they identify important input feature and eliminate useless input features from the dataset to simplify it and after that Maximum Entropy (ME) model is used to learn and detect intrusion from the selected input features. To extracting the needed and important feature from data set author used Information Gain and Chi-Square approach. The most advantage of lightweight model is that by means of feature selection, it can greatly reduce the redundant and least important features for intrusion detection, therefore reduce the computational cost in the process of intrusion detection. Maximum Entropy model is proved to be a good classifier when provided enough input features; it's very effective in the field of intrusion detection.

An intrusion is a successful violation of a network's security policy [34].

Li Tian, W. Jianwen [17] defined a Network Intrusion Detection System based on Improve K-means Clustering Algorithm. Author gives a new model of anomaly intrusion detection based on clustering algorithm. Because of the k-means algorithm's shortcomings about dependence and complexity, the author gives an improved clustering algorithm through studying on the traditional means clustering algorithm. The new algorithm learns the strong points from the k-medoids and improved relations trilateral triangle theorem. K-means algorithm put the similar data in the same cluster, the dissimilar ones in different cluster, and marks these data, then divides the data of the network into different clusters, and estimates the abnormality of the network data according to the mark of the cluster but it does not give globally optimal solution. So author proposed a type of k-means algorithm based on the k-medoids cyclic method and the improved triangle trilateral relations theorem, which improves the k-means algorithm from reduces makes the improvement to the initial cluster centre dependence and the algorithm time expenses. This improved algorithm removes the limitation of original K-means Cluster Algorithm i.e. when the original k-means algorithm handles the value type, the selection of initial cluster central point affect the algorithm deeply. The algorithm adopts the gradient solution algorithm, whose direction is carries on along the direction that the energy is small, but the result is very possibly the partial optimum value not the global optimum. When the data quantity is large, the time consumption of the computation distance is large accordingly. Therefore an algorithm is

needed to improve the algorithm dependence on initial cluster central point and algorithm time expenses.

When clustering algorithm is used in developing model, it suffers from very high fault detection rate while it maintains the lower false drop rate. But result shows that the use of enhanced algorithm in the network intrusion detection system could increase the fault detection rate of anomalous detection and decrease the fault drop rate well.

In the subsequent years, an ever-increasing number of research prototypes were explored because intrusion detection has become a mature industry and a proven technology, nearly all of the easy problems related to IDS have been solved [19][20][21]. However, approaches used in intrusion detection such as those relying on statistical techniques, mobile agents, neural networks, artificial immunity, etc. are essentially based on the observation of events and their analysis. Therefore, data collection constitutes the first step for most intrusion detection systems. Nowadays, these data are generally characterized by their elevated volume, which make it difficult to be analysed. In fact, most current intrusion detection methods cannot process large amounts of audit data for real-time operations and it seems better to have a new information content of user behaviours, emphasizing the significant features.

S.E Smaha [25] defined a new practical approach of Haystack Intrusion detection System Model b for detection of intrusion in multi-user Air Force computer system, which was based on anomaly detection. It reduces voluminous system audit trails to short summaries of user behaviours, anomalous events, and security incidents. The main reason for designing the model was to help the System Security Officer (SSO) detect and investigate intrusion by insiders. It reports on anomalous event in each day audit trail's file, and check user activity against predefined security constraints and models of typical user behaviour.

It was the initial model of intrusion detection model; it reflects a particular understanding of its underlying problem domain of intrusion detection and its intended security-conscious military environment. The main approach was same as given by Dorothy E. Denning's model presented [2]. Here they did change in model that fit for available data on target machine and the security requirements of users. There were so many problems existed at that time like how do they test an intrusion detection system and measures effectiveness of system. Reason being, test cases were generated by software developer according to their own knowledge of system weaknesses.

III. PROPOSED METHODOLOGY

From the study of research survey, most current anomaly IDS detects computer network behaviour as normal or abnormal but still cannot identify the type of attacks. So, to provide the better security and effective, efficient results we are using a latest technology.

In this paper, a new technique is proposed to combine PCA and NBA with KDD dataset which is efficient to detect all types of intrusion. The relationship among PCA, NBA and KDD Dataset is shown in figure 3.

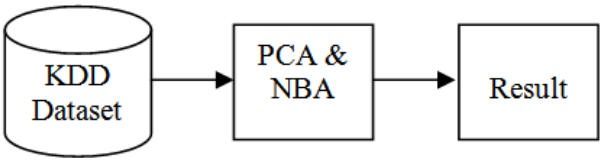

Figure 3: Process of PCA & NBA with KDD Dataset

Our methodology is divided into two parts:

- Training
- Testing

In first stage, we will train the system so that, it can be able to detect attacks in second stage i.e. testing.

The working process of proposed methodology is shown in figure 4.

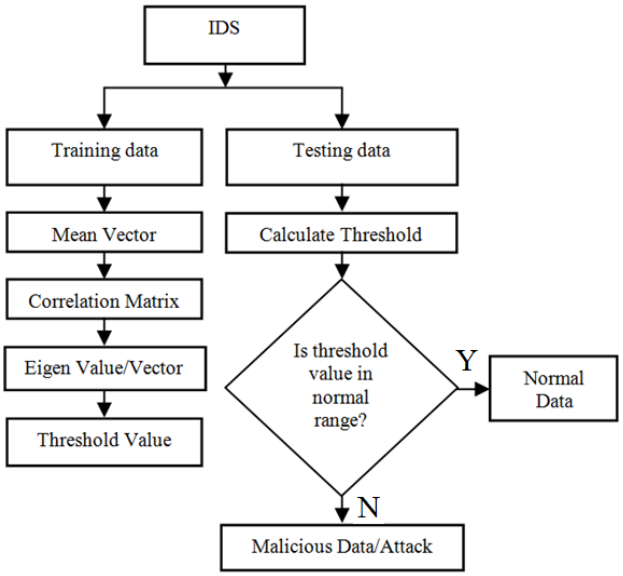

Figure 4: Proposed Methodology

The intrusion detection system monitors and collects data from a target system. After that proposed system processes whole data and correlates the collected information. If it founds any intrusion then it takes appropriate action. Response from proposed system is generally sent to auto response system or security staff for automatic or manual appropriate response action.

Thus, we will develop a model for intrusion detection in computer networks based on principal component analysis (PCA) where each network connection is transformed into an input data vector. PCA is then employed to reduce the high dimensional data vectors and thus, detection is handled in a low dimensional space with high efficiency and low use of system resources. The data from these experiments reveal that we can get better security by employing Anomaly-based IDS with Principal Component Analysis, Network Behaviour Analysis (NBA) and KDD data set. We will train our system to give the security from the intrusion and test it.

The main objective of proposed system is not only to prevent the existing attack but also to detect new attack as soon as possible.

For evolution, proposed system is divided into two basic stages:

a) First stage is training
b) Second stage is testing

First stage (Training): This is learning stage during which our system will construct a model using normal behavior of the network and by taking some attacks' behaviors. And it follows the given derivation:
First we collect the training data from the KDD dataset which, we want to secure, as an m × n matrix. Where n represents the total number of connection or data and m represents the number of features (dimensions). So if Γ is a vector that corresponds to behavior of a certain data and its dimensions, then we can write:

$$\Gamma = \begin{pmatrix} x_1 \\ \vdots \\ x_m \end{pmatrix} \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (6)$$

The training data is taken as input and a mean vector (μ) of the whole sample of data is computed:

$$\mu = \begin{pmatrix} \bar{x}_1 \\ \vdots \\ \bar{x}_m \end{pmatrix} \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (7)$$

Then standard deviation Ø is calculated from the mean vector using the given equation no. 8:

$$\phi_i = \Gamma_i - \mu \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (8)$$

After that a correlation matrix ($V_{m\times n}$) is computed from the training data, where $Ø^t$ is transpose of standard deviation.

$$V_{m\times n} = \frac{1}{n}\sum_{i=1}^{n} \phi_i \phi_i^t \dots\dots\dots\dots\dots\dots\dots (9)$$

Then, Eigen analysis is done on correlation matrix to extract the pair of Eigen values/eigen vectors. These pairs are used to make up the set of Principal Components that will use in online analysis. Here we will select only major Principal components and we will discard the remaining components. And only major principal components will used in system where eigenvector corresponding to the highest Eigen values. Component analysis is defined by λ and no. of Eigen vectors is defined by g, we use the following equation:

$$\frac{\lambda_1 + \lambda_2 + \dots + \lambda_g}{\sum_{i=1}^{m} \lambda_i} \geq s \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (10)$$

Then a new matrix U= m×g is created by using these eigenvectors that constitutes his columns. This matrix corresponds to the new subspace W_g. Because we extract the most significant principal components using PCA, that makes our method a dimensionality reduction method because now only a subset of most important principal components are used to classify any new data.

Now we will calculate the corresponding distribution in new subspace or compressed data which is defined by chosen principal components, for each known data, by projecting their vectors into the new compressed data (y_i) denoted in new subspace W_g.

$$y_{i\ (i=1..n)} = U^t \phi_i \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (11)$$

We define a reference vector for each data class on the baisis of euation 11 as follows:

$$Y^k(k = 1..p) = \frac{1}{q}\sum_{i=1}^{q} Y_i^k \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (12)$$

Where q= number of data (n).

At the end of this stage, we need to specify threshold Θ for each known data class. Each threshold Θ represent an interval [$Θ_{min}$, $Θ_{max}$], learned from training data.

$$\theta_{max}^k = max\left\|Y_i^k - Y^k\right\|^2 \dots\dots\dots\dots\dots\dots\dots\dots\dots (13)$$

These threshold will work as a key element to determining the new captured profile whether it is belong to normal class or not.
The same approach is used on different known attacks. As a result, we obtain:
μ: mean vector of the known attacks sample;
U: the matrix that corresponds to the new subspace $W_{'g'}$ representing the known attacks.
Second stage (Testing): In this stage we will check that whether the audited profile is normal or not, if not then it belongs to which type of attack class and last but not the least we will discover new type of attack if audited profile does not belong to any class. So, here we will give a new data vector Γi (a test network connection) for testing:
We project it onto the subspace representing the normal behaviors:

$$y_i = U^t \phi_i \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (14)$$

Where Y_i is the representation of Γ_i in the new sub-space $W_{'g'}$

$$\phi_i = \Gamma_i - \mu \dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots\dots (15)$$

After that, we will compare the obtained feature vector Yi with those representing the normal behaviors using the squared Euclidean distance ε .

$$\epsilon_{max}^k = max\left\|Y_i^k - Y^k\right\|^2 \dots\dots\dots\dots\dots\dots\dots\dots\dots (16)$$

After that, this user profile is classified according to the following equation:

$$If\ \epsilon_k \in \left[\theta_{min}^k, \theta_{max}^k\right]\ then\ \Gamma_i \in C^k \dots\dots\dots\dots\dots\dots (17)$$

Where Euclidean distance (ε) is taken arises to find out the thresholds value (θ).Else Γ_i is anomalous so, if the audited data does not belong to any known data class, it is then considered as anomalous (attack). Though, the type of newly discovered attack is not yet known.

IV. EVALUATION AND RESULT

We are proposing a new system in which we are combing both signature-based and anomaly-based IDS to detect the new attacks by applying PCA and NBA with the help of KDD dataset.

PCA is used for dimension reduction of data, which work on Eigen vector and Eigen values and if there is any new data

found then NBA is used to measure the data flow so they can get the suspicious data and can get better result.

Here, a tabular form is shown to get the accuracy and result of proposed system. We show the analysis and evaluation of KDD data set in to our offline system. We count total number of malicious data of different types of attacks. After that according to each attack, we are trained our system and then we test it to find out the intrusion reduction rate to get better result and accuracy. In our system one can easily understand the Identification Rate (IR) of each attack by which one can evaluate the accuracy of the system and can get the required goal.

Table 1: Result

Attack type	Number in Total	Number for training	Number for testing	IR (%)
Nmap	257	200	57	100.00
portsweep	1040	901	139	100.00
satan	1734	1433	301	99.01
smurf	253260	253149	111	100.00
teardrop	977	820	157	100.00
warezclient	1019	886	133	100.00
bufferoverflow	49	44	5	100.00
guesspassword	62	48	14	100.00

V. CONCLUSION & FUTURE WORK

We have developed an intrusion detection system using principal component analysis to secure network from attacks. We used machine learning technique of dimensionality reduction. For the dimensionality reduction we used principal component analysis. By using PCA we designed a model and implemented it. Our system learns the behaviour of connection at training time over training data and at the time of testing it identify known attacks as well as it also identifies new type of attacks.

We proposed a novel method for intrusion identification. Intrusions are detected based on normal behaviour for anomaly detection while the individual type of attack or a new attack is identified based on the behaviour of each type of attack for intrusion identification.

As the volume of related information is explosively increasing nowadays, processing large amounts of audit data for real-time intrusion detection and identification is crucial for quick response against attacks. It is thus an effective model to process a high quantity of audit data in real-time with low overhead and it is suitable for real-time intrusion identification.

Extensive experiments are conducted to test our model and to compare with the results of other methods reported in the recent literature. Since in previous studies researcher trained and test their model with selected number of connections according to their convenience but in our study we used testing and training data connection in bulk. In spite of that our model is very much promising in terms of detection accuracy and computational efficiency for real-time intrusion detection in comparison to previous given systems. The model is also effective to identify most individual known attacks as well as new attacks.

Our proposed model provides a robust and best representation of data using the PCA, NBA and KDD dataset to remove the limitations from existing system. Using this model we can reduce data, approximately 60% in training time because we put some data from total amount of data and 80% in testing time because we test our trained system and finally we get almost 99% accuracy. Our proposed model not only reduces data but also decrease time in intrusion detection and increase efficiency and accuracy.

For the future work, we will develop an online self-adaptive intrusion identification model for updating each individual attack database dynamically and automatically and thus improving the identification rates. Our proposed model is basically work for offline system because we are using offline KDD data set 99. In future it can be extended using online data set so that, it can be directly detect the system from malicious codes and viruses and can be provide better security and more better efficiency.

REFERENCES

[1] IDS Web Site: http://www.webopedia.com/TERM/I/intrusion_detection_system.html. [31/10/2013]
[2] D. E. Denning. “An Intrusion-Detection Model”. IEEE transactions on software engineering, Volume: 13 Issue: 2, February 1987.
[3] H. Deber, M. Becker and D. Siboni, “A Neural Network Component for an Intrusion Detection System”, Research in Security and Privacy, Proceedings. IEEE Computer Society Symposium on, pp. 240-250, 1992.
[4] E. Hooper, “An Intelligent Intrusion Detection and Response System Using Hybrid Ward Hierarchical Clustering Analysis”, International Conference on Multimedia and Ubiquitous Engineering, in IEEE, pp. 1187-1192, 2007.
[5] G. Xin and Li. Y. jie, “A new Intrusion Prevention Attack System Model based on Immune Principle”, International Conference on e-Business and Information System Security (EBISS), in IEEE, pp. 1-4, 2010.
[6] H. Jiawei, M. Kamber, “Data Mining Concepts and Techniques”, Machinery Industry Press, Beijing, pp. 232-235, 2001
[7] C. Herringshaw, “Detecting Attacks on Networks”, Digital Object Identifier, Volume: 30 Issue: 12, pp. 16-17, 1999
[8] K. Hwang, M. Cai, Y. Chen, M. Qin , “Hybrid Intrusion Detection with Weighted Signature Generation over Anomalous Internet Episodes”, IEEE Transactions on Dependable Computing, Volume: 4 Issue: 1, pp. 41-55, 2007
[9] I.T. Jolliffe, “Principal Component Analysis”, 2nd Edition, Springer-Verlag, NY, 2002
[10] I. T. Jolliffe, “Principle Component Analysis”, 2nd Edition, Springer-Verlag, NY, 2002.
[11] J.P. Anderson, “Computer security technology planning study”. Technical Report, ESDTR-73-51, United States Air Force, Electronic Systems Division, October 1972.
[12] J.P. Anderson, “Computer Security Threat Monitoring and Surveillance”. Technical Report, James P. Anderson Company, Fort Washington, Pennsylvania, April 1980.
[13] J. Ryan, M. J. Lin, R. Miikkulainen, “Intrusion Detection with Neural Networks”, Technical Report, AAAI, pp. 72-77, 1997.
[14] J. Beale (2007), “Snort: IDS and IPS Toolkit”, available: http://ebookee.org/Snort-IDS-and-IPS-Toolkit-Jay-Beale-s-Open-Source-Security-Repost-_412527.html.
[15] J. Shlens, “A Tutorial on Principal Component Analysis”. Version 3.01, April 2009.
[16] W. Lee, S. Stolfo, “Data mining approaches for intrusion detection”. Proceedings of the 1998 USENIX Security symposium, 1998.
[17] Li. Tian and W.Jianwen, “Research on Network Intrusion Detection System Based on Improved K-means Clustering Algorithm”, Computer Science-Technology and Applications, IEEE Computer Society, pp. 76-79 2009.
[18] M. Roesch (2009), “Snort User Manual 2.8.5”: http://www.snort.org/assets/125/snort_manual-2_8_5_1.pdf
[19] M. Moradi and M. Zulkernine, “A Neural Network Based System for Intrusion Detection and Classification of Attacks”, Proc. of the 2004 IEEE International Conference on Advances in Intelligent

Systems-Theory and Applications, pp. 148:1-6, Luxembourg, November 2004.

[20] M. Singh and S. S. Sodhi, "Distributed Intrusion Detection using Aglet Mobile Agent Technology", Proc. Of Conference on Challenges & Opportunities in Information Technology (COIT-2007), Mandi Gobindgarh, March 2007.

[21] M. Glickman, Justin Balthrop and Stephanie Forrest, "A Machine Learning Evaluation of an Artificial Immune System", Evolutionary Computation Conference, Volume: 13, pp. 179 – 212, June 2005.

[22] L. Mechtri, D. Tolba, N. Ghoualmi, "Intrusion Detection Using Principal Component Analysis", Engineering Systems Management and Its Applications (ICESMA), pp. 1-6, 2010.

[23] R. Karthick, P. Vipul, B. Ravindran, "Science Adaptive Network Intrusion Detection System using a Hybrid Approach", Fourth International Conference on Communication Systems and Networks (COMSNETS), in IEEE, pp. 1-7, 2012 .

[24] L. K. Ronald, R.D. Vines, "Cloud Security: A Comprehensive Guide to Secure Cloud Computing", e-book published by Wiley Publishing, Inc., pp. 61-169, 2010.

[25] S.E. Smaha, "Haystack: an intrusion detection system", Aerospace Computer Security Applications Conference, pp. 37-44, 1988.

[26] A. Sodiya, A. Akinwale, "A new two - tiered strategy to intrusion detection", Information Management and Computer Security, Volume: 12 Issue: 1, pp. 27-44, 2004.

[27] S. Mukkamala, G. Janoski, A. Sung, "Intrusion Detection Using Neural Networks and Support Vector Machines", IEEE Transaction 2002.

[28] The third international knowledge discovery and data mining tools competition dataset (1999), "KDD99-Cup":http://kdi.ics.uci.edu/databases/kddcup99/kddcup99.html

[29] Tcpreplay Pcap editing & replay tools for *NIX (2010). :http://tcpreplay.synfin.net/wiki/manual#a3.xOnlineManual

[30] V. Paxson, "Bro: A system for detecting network intruders in real-time", In Proceedings of the 7th USENIX Security Symposium, San Antonio, TX, 1998.

[31] W. Lee, S. J. Stolfo, and K. Mok, "Data mining in work flow environments: Experiences in intrusion detection", In Proceedings of the 1999 Conference on Knowledge Discovery and Data Mining (KDD-99), 1999.

[32] W. W. Cohen, "Fast effective rule induction", In Machine Learning the 12th International Conference, Lake Taho, CA, 1995. Morgan Kaufmann.

[33] Yang Li, Bin-Xing Fang, You Chen and Li Guo, "A Lightweight Intrusion Detection Model Based on Feature Selection and Maximum Entropy Model", National Grand Fundamental Research, 2006.

[34] J. Zhou, A .Carlson, and M. Bishop., "Verify Results of Network Intrusion Alerts Using Lightweight Protocol Analysis", Proceedings of the 21st Annual Computer Security and Applications Conference (ACSAC), 2005

[35] S. Lakhina, S. Joseph and B. Verma, "Feature Reduction using Principle Component Analysis for Effective Anomaly-Based Intrusion Detection on NSL-KDD", International Journal of Science and Engineering, vol.6, no. 2, pp.1790-1799,2010.

[36] V.Kumar, O. P. Sangwan, "Development and assessment of Intrusion Detection System using Machine Learning Algorithm" ICNICT, November 2012.