Network Intrusion detection by using SMO-SVM

Alesh Kumar Sharma¹, Satyam Maheswari²

¹MTech Scholar, Department of Computer Application, SATI Vidisha, Madhya Pradesh.
E-mail:eraleshsharma2006@gmail.com

²Assistant Professor, Department of Computer Application, SATI Vidisha, Madhya Pradesh.
E-mail:satyam.vds@gmail.com

Abstract
As network attacks have increased in number and severity over the past few years, intrusion detection system (IDS) is increasingly becoming a critical component to secure the network. Due to large volumes of security audit data as well as complex and dynamic properties of intrusion behaviors, optimizing performance of IDS becomes an important open problem that is receiving more and more attention from the research community. Intrusion poses a serious security risk in a network environment. The ever growing new intrusion types pose a serious problem for their detection. In this paper, a new intrusion detection method based on Principle Component Analysis (PCA) and SMO-SVM with low overhead and high efficiency is presented. System call data and command sequences data are used as information sources to validate the proposed method. The frequencies of individual system calls in a trace and individual commands in a data block are computed and then data column vectors which represent the traces and blocks of the data are formed as data input. PCA is applied to reduce the high dimensional data vectors and distance between a vector and its projection onto the subspace reduced is used for anomaly detection. Experimental results show that the proposed method is promising in terms of detection accuracy, computational expense and implementation for real-time intrusion detection.

Keywords: Network Intrusion Detection, Principal Component Analysis (PCA), SMO-SVM, Detection Rate and False Positive Rates.

I. INTRODUCTION

With the tremendous growth of network-based services and sensitive information on networks, network security is becoming more and more importance than ever before. Intrusion detection techniques are the last line of defenses against computer attacks behind secure network architecture design, firewalls, and personal screening. In spite of the plethora of intrusion prevention techniques available, attacks against computer systems are still successful. There for, intrusion detection systems (IDSs) play a vital role in network security. Symantec in a recent report[1] uncovered that the number of fishing attacks targeted at stealing confidential information such as credit card numbers, passwords, and other financial information are on the rise, going from 9 million attacks in June2004 to over 33 millions in less than a year.

One solution to this is the use of network intrusion detection systems (NIDS), which detect attacks by observing various network activities. It is therefore crucial that such systems are accurate in identifying attacks, quick to train and generate as few false positives as possible. This paper presents the scope and status of our research in anomaly detection. This paper gives a study of Principle Component Analysis (PCA) via SMO-SVM technique for identifying novel network intrusion detections. We present experimental results on KDDCup’99 data set. Experimental results have demonstrated that our method is much more efficient in the detection of network intrusions. Section 2 describes IDS in general. Section 3 presents an overview of frequently occurring network attacks, and section 4 discusses related research done so far. Section 5 describes our proposed method and section 6 presents the experimental results. Finally, section 7 provides the concluding remarks and future scope of the work.

II. INTRUSION DETECTION

An Intrusion Detection System (IDS) inspects the activities in a system for suspicious behaviour or patterns that may indicate system attack or misuse. There are two main categories of intrusion detection techniques; Anomaly detection and Misuse detection [2]. The former analyses the information gathered and compares it to a defined baseline of what is seen as “normal” service behaviour, so it has the ability to learn how to detect network attacks that are currently unknown. Misuse Detection is based on signatures for known attacks, so it is only as good as the database of attack signatures that it uses for comparison. Misuse detection has low false positive rate, but cannot detect novel attacks. However, anomaly detection can detect unknown attacks, but has high false positive rate.

Many types of data can be used for anomaly detection, such as Unix commands, audit events, keystroke, system calls, and network packages, etc. Early studies [2, 3] on anomaly detection mainly focus on learning normal system or user behaviors from monitored system log or accounting log data. Examples of the information derived from these logs are: CPU usage, time of login, duration of user session, names of files accessed, etc. In recent years, many research
in anomaly detection focus on learning normal program behavior. Forrest et al. introduced a simple anomaly detection method based on monitoring the system calls issued by active, privileged processes [4]. This work was extended by various methods. Lee et al. used data mining approach to study a sample of system call data to characterize sequences occurring in normal data by a small set of rules [5]. Warrender et al. proposed Hidden Markov Model (HMM) method for modeling and evaluating invisible events based on system calls [6].

In practice, a protected computer system could produce massive data streams, for example, during the experiments of capturing the system calls on the sendmail, only 112 messages produced a combined trace length of over 1.5 million system calls [4].

Therefore, processing the high dimensional audit data in real time for online intrusion detection would be computationally expensive.

Principal Component Analysis (PCA, also called Karhunen-Loeve transform) is one of the most wildly used dimension reduction techniques for data analysis and compression in practice. In this paper, we discuss a novel intrusion detection method based on PCA, by which intrusion detection can be employed in a lower dimensional subspace and the computational complexity can be significantly reduced. Two types of data are used to verify the proposed method and the testing results show that the method is efficient and effective.

### Training Phase
- **Data set**
- **Rule set**
- **Network Traffic Data**

### Classification Phase
- **Belief Assignment**
- **Data Fusion**
- **Decision making**
- **State**

Fig. 1 Intrusion detection identification

### III. NETWORKING ATTACKS

The simulated attacks were classified, according to the actions and goals of the attacker. Each attack type falls into one of the following four main categories [3]:
- **Denial of Service (DoS)** attacks have the goal of limiting or denying services provided to the user, computer or network. A common tactic is to severely overload the targeted system. (e.g. apache, smurf, Neptune, Ping of death, back, mailbomb, udpstorm, SYNflood, etc.).
- **Probing or Surveillance** attacks have the goal of gaining knowledge of the existence or configuration of a computer system or network. Port Scans or sweeping of a given IP-address range typically fall in this category. (e.g. saint, portsweep, mscan, nmap, etc.).
- **User-to-Root (U2R)** attacks have the goal of gaining root or super-user access on a particular computer or system on which the attacker previously had user level access. These are attempts by a non-privileged user to gain administrative privileges (e.g. Perl, xterm, etc.).
- **Remote-to-Local (R2L)** attack is an attack in which a user sends packets to a machine over the internet, which the user does not have access to in order to expose the machine vulnerabilities and exploit privileges which a local user would have on the computer (e.g. xclock, dictionary, guest_password, phf, sendmail, xsnoop, etc.).

### IV. RELATED WORK

The problem of huge network traffic data size and the invisibility of intrusive patterns which normally are hidden among the irrelevant and redundant features have posed a great challenge in the domain of intrusion detection [8]. One way to address this issue is to reduce these input features in order to disclose the hidden significant features. Thus, an accurate classification can be achieved. Besides identifying significant features that can represent intrusive patterns, the choice of classifier can also influence the accuracy and classification of an attack. The literature suggests that hybrid or assembling multiple classifiers can improve the accuracy of detection [7] [9]. According to Chebrolu et al. [9], an important advantage for combining redundant and complementary classifiers is to increase robustness, accuracy and better overall generalization. Mukkamala et al. [10] demonstrated the use of ensemble classifiers gave the best accuracy for each category of attack patterns. Ensemble methods aim at improving the predictive performance of a given statistical learning or model fitting technique. The general principle of ensemble methods is to construct a linear combination of some model fitting method, instead of using a single fit of the method. In designing a classifier, the first step is to carefully construct different connectional models to achieve best generalization performance for classifiers. Chebrolu et al. [9] proposed CART-BN approach, where CART performed best for Normal, Probe and U2R and the ensemble approach worked best for R2L and DoS.

Meanwhile, Abraham et al. [11] illustrated that ensemble Decision Tree was suitable for Normal, LGP for Probe, DoS and R2L and Fuzzy classifier was for R2L. Abraham et al. [12] also demonstrated the ability of their proposed ensemble structure in modeling lightweight distributed IDS. Meanwhile, Mukkamala et al. [7] proposed three variants of Neural Networks, SVM and MARS as components in their IDS. This combining approach has demonstrated better performance when compared to single classifier approach.

Here, we have chosen two machine learning techniques to develop our classifiers and they are: Principle Component Analysis and SMO-SVM.

---

**All Rights Reserved © 2012 IJARCSEE**
V. THE PROPOSED INTRUSION DETECTION METHOD BASED ON PCA VIA SMO-SVM

A. Principal Component Analysis

PCA is a useful statistical technique that has found application in fields such as face recognition and image compression, and is a common technique for finding patterns in data of high dimension. The entire subject of statistics is based on around the idea that you have this big set of data, and you want to analyze that set terms of the relationships between the individual points in that set [14].

The goal of PCA is to reduce the dimensionality of the data while retaining as much as possible of the variation present in the original dataset. It is a way of identifying patterns in data, and expressing the data in such a way as to highlight their similarities and differences [15].

B. SMO-SVM Algorithm

The math model of support vector classifier can be considered as a quadratic problem with some constraints. Traditional algorithms always involve matrix operation so the expenses of saving and computing are large; especially the large scale data sets. The data sets we chose for our intrusion detection experiments are large, so after analysis and comparison we choose Sequential Minimal Optimization (SMO) algorithm [16, 17]. It was brought up by Platt in 1998. Its foundation thought is to decompose a large quadratic problem to a serial of minimal quadratic problems.

The SMO algorithm is derived by taking the idea of the decomposition method to its extreme and optimizing a minimal subset of just two points at each iteration. The power of this technique resides in the fact that the optimization problem for two data points admits an analytical solution, eliminating the need to use an iterative quadratic programme optimizer as part of the algorithm[18].

The requirement that the \( \sum_{i=1}^{m} \alpha_i y_i = 0 \) is enforced throughout the iterations implies that the smallest number of multipliers that can be optimized at each step is 2: whenever one multiplier is updated, at least one other multiplier needs to be adjusted in order to keep the condition true.

At each step SMO chooses two elements \( \alpha_i \) and \( \alpha_j \) to jointly optimize, finds the optimal values for those two parameters given that all the others are fixed, and updates the \( \alpha \) vector accordingly. The choice of the two points is determined by a heuristic, while the optimization of the two multipliers is performed analytically. Despite needing more iterations to converge, each iteration uses so few operations that the algorithm exhibits an overall speed-up of some orders of magnitude. Besides convergence time, other important features of the algorithm are that it does not need to store the kernel matrix in memory, since no matrix operations are involved, that it does not use other packages, and that it is fairly easy to implement.

![Figure 2: The proposed Intrusion Detection System](image)

Notice that since standard SMO does not use a cached kernel matrix, its introduction could be used to obtain a further speed-up, at the expense of increased space complexity.
Dataset Description
The data set used was the KDD Cup 1999 Data [13], which contained a wide variety of intrusions simulated in a military network environment. It consisted of approximately 4,900,000 data instances, each of which is a vector of extracted feature values from a connection record obtained from the raw network data gathered during the simulated intrusions. The simulated attacks fell in one of the following four categories: DOS-Denial of Service (e.g. a syn flood), R2L- Unauthorized access from a remote machine (e.g. password guessing), U2R- Unauthorized access to superuser or root functions (e.g. a buffer overflow attack), Probing-surveillance and other probing for vulnerabilities (e.g. port scanning).

C. Feature Selection
The feature selection included the basic features of an individual TCP connection such as its duration, protocol type, number of bytes transferred, and the flag indicating the normal or error status of the connection. Other features of an individual connection obtained using some domain knowledge, and included the number of file creation operation, number of failed login attempts. In total, there were 41 features, with most of them taking on continuous values.

D. Normalization
Since our algorithm is designed to be general, it must be able to create clusters given a dataset from an arbitrary distribution. A problem with typical data is that different features are on different scales. This cause bias toward some features over other features. To solve this problem, we convert the data instances to a standard form based on the training dataset’s distribution. That is, we make the assumption that the training dataset accurately reflects the range and deviation of feature values of the entire distribution. Then, we can normalize all data instances to a fixed range of our choosing, and hard code the cluster width based on this fixed range.

VI. EXPERIMENT AND RESULTS
The data set used to perform the experiment was taken from KDD Cup ’99, which is widely accepted as a benchmark dataset.
A KDD Cup’99 distribution records as DataSet_1 by class type is summarized in Table 1. The behavior of data for Intrusion Detection System (IDS) can be categorized as in Table 2.

<table>
<thead>
<tr>
<th>Class</th>
<th>No. of Samples</th>
<th>Sample percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal</td>
<td>10680</td>
<td>62.587%</td>
</tr>
<tr>
<td>Probe</td>
<td>4633</td>
<td>27.150%</td>
</tr>
<tr>
<td>DoS</td>
<td>599</td>
<td>3.510%</td>
</tr>
<tr>
<td>U2R</td>
<td>95</td>
<td>0.556%</td>
</tr>
<tr>
<td>R2L</td>
<td>1057</td>
<td>6.194%</td>
</tr>
<tr>
<td>Total</td>
<td>17064</td>
<td>100%</td>
</tr>
</tbody>
</table>

Table 1: Sample distribution of the DataSet_1

<table>
<thead>
<tr>
<th>Actual Classification</th>
<th>Predicted Normal</th>
<th>Predicted Attack</th>
</tr>
</thead>
<tbody>
<tr>
<td>Normal</td>
<td>TN</td>
<td>FP</td>
</tr>
<tr>
<td>Intrusions (attacks)</td>
<td>FN</td>
<td>TP</td>
</tr>
</tbody>
</table>

Table 2: Behavior of data

Intrusion detection system require high detection rate and low false alarm rate, thus the performance of an Network intrusion detection system (NIDS) can usually be evaluated in terms of detection rate and false alarm as below:

\[
\text{Detection Rate} = \frac{TP}{TP + FP}
\]

\[
\text{False Alarm} = \frac{FP}{FP + TN}
\]

Where,
FN is False Negative,
TN is True Negative,
TP is True Positive, and
FP is False Positive

A series of experiments was conducted using SOM using PCA [20] as a single classifier and SMO-SVM using PCA approach with the benchmark dataset, KDD-Cup’99[19]. All data was normalized and some features have been changed before the implementation to obtain a better output. The results are implemented by five category classes Normal, Probe, DoS, U2R & R2L.

A. Results analysis for Dataset_1
Table 3 show the confusion matrix of PCA via SMO-SVM classification. A “Confusion Matrix” is sometimes used to represent the result of cross validation, as shown in Table 3. The Advantage of using this matrix is that it not only tells us how many got misclassified but also what misclassifications occurred. PCA technique applied on DataSet_2 of KDD CUP 99 Dataset and 12 features reduces out of 41 features.
### Table 3: Classification result for PCA via SMO-SVM using Cross-Validation

<table>
<thead>
<tr>
<th></th>
<th>Actual DoS</th>
<th>Predicted DoS</th>
<th>Actual U2R</th>
<th>Predicted U2R</th>
<th>Actual R2L</th>
<th>Predicted R2L</th>
<th>Actual Probe</th>
<th>Predicted Probe</th>
<th>Actual Normal</th>
<th>Predicted Normal</th>
</tr>
</thead>
<tbody>
<tr>
<td>DoS</td>
<td>4437</td>
<td>0</td>
<td>0</td>
<td>61</td>
<td>125</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>U2R</td>
<td>37</td>
<td>14</td>
<td>0</td>
<td>24</td>
<td>20</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>R2L</td>
<td>37</td>
<td>2</td>
<td>3</td>
<td>1</td>
<td>556</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Probe</td>
<td>35</td>
<td>0</td>
<td>0</td>
<td>1018</td>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Normal</td>
<td>149</td>
<td>0</td>
<td>1</td>
<td>24</td>
<td>10506</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 3: Classification result for PCA via SMO-SVM using Cross-Validation

### Table 4: Summary of overall measurement using Dataset_1

<table>
<thead>
<tr>
<th></th>
<th>PCA via SMO-SVM</th>
<th>PCA via SOM</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>DR</td>
<td>FAR</td>
</tr>
<tr>
<td>DoS</td>
<td>96</td>
<td>2.1</td>
</tr>
<tr>
<td>U2R</td>
<td>14.7</td>
<td>0</td>
</tr>
<tr>
<td>R2L</td>
<td>0.5</td>
<td>0</td>
</tr>
<tr>
<td>Probe</td>
<td>96.3</td>
<td>0.7</td>
</tr>
<tr>
<td>Normal</td>
<td>98.4</td>
<td>11</td>
</tr>
</tbody>
</table>

Table 4: Summary of overall measurement using Dataset_1

### Figure 3: Detection Rate comparison graph by using DataSet_1

### Figure 4: False Alarm Rate comparison graph by using DataSet_1

The experimental results for a PCA via SOM classification and the proposed PCA via SMO-SVM classification are summarized in Table 4. The Table 4 representing measurement in terms of detection rate and false alarm on the Dataset_1. From Table 4, the PCA via SOM has produced a slightly high detection rate but with high false alarm rates in U2R and R2L. In contrast, **PCA via SMO-SVM** records high detection rate with low false alarm rates in Normal, DoS and Probing.

### VII. CONCLUSION AND FUTURE WORK

Our research work based on network intrusion detection system, we found that Most of the existing IDs use all 41 features in the network to evaluate and look for intrusive pattern some of these features are redundant and irrelevant. The drawback of this approach is time-consuming detection process and degrading the performance of ID system. To solve this problem we proposed an algorithm based on PCA (Principal Component Analysis) and SMO-SVM that uses Principal Component Analysis as a Features reduction algorithm. The goal of PCA is to reduce the dimensionality of the data while retaining as much as possible of the variation present in the original dataset and trained SMO-SVM to identify any kind of new attacks. Tests and comparison are done on KDD CUP 99 dataset. The test data contains 4 kinds of different attacks in addition to normal system call.

Our experimental results showed that the proposed model gives better and robust representation of data as it was able to reduce features resulting in a 70.73% data reduction. Meantime it significantly reduce a number of computer resources, both memory and CPU time, required to detect an attack. This shows that our proposed algorithm is reliable in network intrusion detection. Currently in some cases the detection rate reduces when we apply the dimension reduction techniques. In future, we will continue on our research of improving detection performance of both normal and malicious activities.
REFERENCES


[14] Lindsay I Smith A tutorial on Principal Components Analysis February 26,2002


