STATISTICAL TECHNIQUES IN ANOMALY INTRUSION DETECTION SYSTEM

Hari Om & Tanmoy Hazra
Department of Computer Science & Engineering, Indian School of Mines, Dhanbad, India

ABSTRACT
In this paper, we analyze an anomaly based intrusion detection system (IDS) for outlier detection in hardware profile using statistical techniques: Chi-square distribution, Gaussian mixture distribution and Principal component analysis. Anomaly detection based methods can detect new intrusions but they suffer from false alarms. Host based Intrusion Detection Systems (HIDSs) use anomaly detection to identify malicious attacks i.e. intrusion. The features are shown by large set of dimensions and the system becomes extremely slow during processing this huge amount of data (especially, host based). We show the comparative results using three different approaches: Principal Component Analysis (PCA), Chi-square distribution and cluster with Gaussian mixture distribution. We get good results using these techniques.

KEYWORDS: Principal Component Analysis, Outlier, Mahalanobis Distance, Confusion Matrix, Anomaly Detection, Gaussian mixture distribution, Chi-square distribution, Expectation maximization algorithm.

I. INTRODUCTION
The process of monitoring events happens in a computer system or network and properly analyzing them and signals for intrusions is called as intrusion detection. An intrusion is defined as an attack in a network or system by an intruder/attacker that hampers the security goals such as integrity, confidentiality, authentication i.e. violation of the security policies of a system. An intrusion detection system (IDS) is a program that analyzes what happens/has happened during an execution and tries to find out indications if the computer has been misused. There are various types of attacks: external attacks, internal penetrations, and misfeasors. An intruder tries to gain unauthorized access to a particular user’s system or network. With rapid advancement of computers in past few years, their security has become an important issue. Host based intrusion detection systems (HIDSs) are used to monitor and analyze a mysterious activity of the system. The HIDSs can be classified into two types: anomaly detection that is based on finding abnormal data by statistical measurement and misuse detection that is based on predefined signature. Anomaly detection is used to capture the changes in behavior that deviates from the normal behavior. These methods take training data as input to build normal system behavior models. Alarms are raised when any activity deviates from the normal model. These models may be generated using statistical analysis, data mining algorithms, genetic algorithms, artificial neural networks, fuzzy logic, rough set, etc. Anomaly detection methods may alarm for normal activity (false positive) or may not alarm in attacks.
(false negative). Nowadays, the numbers of new attacks are increasing and the variations of known attacks cannot be recognized by misuse detection. Here, we develop an intrusion detection system using the Chi-square distribution and Gaussian mixture distribution that are used to detect outlier data and find out the results using three different approaches.

Outlier detection is the most important issue in data analysis. The outliers describe abnormal data behavior, i.e. data which deviate from the natural data. The cut-off value or threshold which divides anomalous and non-anomalous data numerically is often the basis for important decisions. There are many methods which discuss about univariate outlier detection. They are based on efficient location, data quantiles, and diversity of data estimation. Their major disadvantage is that their predefined rules are not dependent upon the size of sample data. In most of the decision rules, outliers are recognized even for normal data. At least no differentiation is shown between outliers and extreme data. The backbone for multivariate outlier detection is nothing but Mahalanobis distance. The robust estimation of attributes in the Mahalanobis distance and the contrast with a critical value of the Chi-square distribution is the commonly used method for multivariate outlier detection. The values larger than critical value are not necessarily outliers; they can still belong to the data distribution [39]. To differentiate between outliers, and extremes of a distribution, Garrett introduces the concept of Chi-square plot. It draws the robust Mahalanobis distances empirical distribution against the Chi-square distribution [38]. The rest of the paper is organized as follows. Section 2 discusses the related work and section 3 discusses the proposed work.

II. RELATED WORK

Many designs have been developed for intrusion detection [1-6]. Shyu has developed a network based intrusion predictive model using principal component analysis (PCA) and Chi-square distribution for KDD1999 dataset [1]. Denning describes an intrusion detection model that is capable of detecting break-ins, penetrations and other types of computer attacks. This model is based on the hypothesis that the security violations can be detected by monitoring audit records of the system for abnormal patterns of system usage [2]. Errors in multivariate data have been detected using PCA [3]. Ye discusses an anomaly detection technique based on a Chi-Square statistic into information systems that achieves 100% detection rate [4]. Puketza discusses methodologies to test an intrusion detection system and gets satisfactory result in the course of testing IDSs [6]. He further finds experimentally that the performance of Chi-square distribution is better than that of the Hotelling’s $T^2$ [14]. In [34], the PCA methodology is discussed to detect intrusion. Filzmoser applies the Chi-square method to detect multivariate outlier in exploration geochemistry [17]. Garrett has made a tool for multivariate outlier detection [18]. Gascon et al. have shown a comprehensive statistical analysis in relation to the vulnerability disclosure time, updates of vulnerability detection systems (VDS), software patching releases and publication of exploits [22]. Davis and Clark have shown a trend toward deeper packet inspection to construct more relevant features through targeted content parsing [23]. Jin et al. discuss direct utilization of the covariance matrices of sequential samples to detect multiple network attacks [24]. Yeung and Ding show in their experimental results that the dynamic modeling approach is better than the static modeling approach for the system call datasets, while the dynamic modeling approach is worse for the shell command datasets [25]. Hussein and Zulkernine present a framework to develop components with intrusion detection capabilities [26]. Wang et al. present several cross frequency attribute weights to model user and program behaviors for anomaly intrusion detection [27]. Chen et al. discuss an efficient filtering scheme that can reduce system workload and only 0.3% of the original traffic volume is required to examine for anomaly [28]. Casas et al. present an unsupervised network intrusion detection system that is capable of detecting unknown network attacks without using any kind of signatures, labeled traffic, or training [29]. Trajlovi´c discusses variance estimation and ranking methods for stochastic processes modeled by Gaussian mixture distributions. It is shown that the variance estimate from a Gaussian mixture distribution has the same properties as a variance estimate from a single Gaussian distribution based on a reduced number of samples [31]. Dasgupta presents provably the first correct algorithm for learning a mixture of Gaussians, which is very
simple and returns the true centers of the Gaussians to within the precision specified by the user, with high probability and has linear complexity in the dimension of data and polynomial in the number of Gaussians [32]. Dempster et al. present a general approach to iterative computation of maximum-likelihood estimator for incomplete data since each iteration of the algorithm consists of an expectation step followed by a maximization step [33]. Chandola et al. have presented a brief survey of anomaly detection [35]. Teodoro et al. have discussed anomaly detection methodologies and different type of problems [36]. In next section, we discuss statistical techniques for intrusion detection.

III. STATISTICAL TECHNIQUES IN INTRUSION DETECTION

In this work, we discuss three approaches, which are based on PCA, Chi-square distribution and Gaussian mixture distribution and then discuss their comparative performances for host based intrusion detection system. We may mention that these techniques have been discussed in literature for network based intrusion detection systems, not for host based systems. In [34], the PCA has been used for outlier detection in hardware profile. We now discuss PCA, Chi-square distribution, and Gaussian mixture distribution each for host based intrusion detection system.

3.1 Principal Component Analysis

The principal component analysis (PCA) is a common technique to find the natures in a high dimension data by reducing its dimensions without losing the information contained in it. It produces a set of principal components which are orthonormal eigenvalue/eigenvector pairs. It projects a new set of axes that best fit the data. In our proposed scheme, these set of axes represent the normal feature data. Outlier detection occurs by mapping the data on to these normal axes and calculating the distances from these axes. If the distance is greater than a certain threshold or cutoff, we may confirm there is an attack i.e. outlier detection. Principal components are particular linear combinations of \(m\) random variables: \(X_1, X_2, \ldots, X_m\) that have two important properties:

- a. \(X_1, X_2, \ldots, X_m\) are uncorrelated and sorted in descending order.
- b. Total variance, \(X\), of \(X_1, X_2, \ldots, X_m\) is given by

\[
X = \sum_{i=1}^{m} X_i
\]  

These variables are found from eigenanalysis of the correlation matrix of the original variables \(X_1, X_2, \ldots, X_m\). The values of correlation matrix and covariance are not same [9-10],[15].

Let the original data, \(X\), be an \(n \times m\) data matrix of \(n\) observations with each observation consisting of \(m\) fields (dimensions) \(X_1, X_2, \ldots, X_m\) and \(R\) be a \(m \times m\) correlation matrix of \(X_1, X_2, \ldots, X_m\). The eigenvalues of \(R\) are the roots of the following polynomial equation:

\[
|R - \lambda I| = 0
\]

Each eigenvalue of \(R\) has a corresponding non-zero vector \(e\), called an eigenvector,

\[
Re = \lambda e
\]

If \((\lambda_1, e_1), (\lambda_2, e_2), \ldots, (\lambda_m, e_m)\) are the \(m\) eigenvalue-eigenvector pairs of the correlation matrix \(R\), the \(i\)th principal component is given by

\[
y_i = e_i^T(x - \bar{x})
\]

\[
= e_{i1}(x_1 - \bar{x}_1) + e_{i2}(x_2 - \bar{x}_2) + \ldots + e_{im}(x_m - \bar{x}_m), \quad i = 1, 2, \ldots, m.
\]

where
eigen values are ordered as \(\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_m \geq 0\),
\(e_i = (e_{i1}, e_{i2}, \ldots, e_{im})\) is \(i\)th eigenvector,
\(x = (x_1, x_2, \ldots, x_m)^T\) is transpose of observation data
\(\bar{x} = (x_1, x_2, \ldots, x_m)^T\) is transpose of sample mean vector of the observation data.

For any attribute \(x_i\), let the observed data be \(a_1, a_2, \ldots, a_n\), then we have
\[ \bar{x}_i = \frac{1}{n} \sum_{i=1}^{n} a_i \]

The principal components calculated from the covariance matrix are usually different from the principal components generated from the correlation matrix. When some values are much larger than others, their corresponding eigenvalues have larger weights. One of these techniques is known as the Mahalanobis distance

\[ d^2 (x, y) = (x - y)^T R^{-1} (x - y) \quad (3) \]

where \( R^{-1} \) is the sample correlation matrix, \( x \) and \( y \) are vectors and here \( X^T \) represents the transpose of \( X \).

Using the correlation matrix, the relationships between the data fields are exhibited more effectively. There are two main issues in using PCA, calculation of distance and interpretation of the set of principal components.

Each eigenvalue of a principal component corresponds to the variation amount it bounds. The larger eigenvalues are more significant and correspond to their eigenvectors. The principal components are sorted in descending order. Eigenvectors of the principal components exhibit axes which best fit a data sample. Points which lie at a far distance from these axes are assumed to show abnormal behavior and they can be easily recognized. Using a threshold value, the data generated by normal system with Mahalanobis distance greater than the threshold or cutoff is considered an outlier and, here, it is an intrusion but sometimes it alerts the user as intrusion if the data is on the threshold boundary.

Consider the sample principal components, \( y_1, y_2, \ldots, y_m \) of an observation \( X \) where

\[ y_i = e_i^T (X - x), \quad i = 1, 2, \ldots, m \]

The sum of squares of the partial principal component scores is equal to the principal component score:

\[ \sum_{i=1}^{m} y_i^2/\lambda_i = y_1^2/\lambda_1 + y_2^2/\lambda_2 + \ldots + y_m^2/\lambda_m \quad (4) \]

equates to the Mahalanobis distance of any observation \( X \) from the mean of the normal sample dataset [15]. Major principal component score is used to detect extreme deviations with high values on the original dataset. The minor principal component score is used to detect some attacks which may not use the same correlation model. As a result, two thresholds are needed to detect attacks. \( q \) is a subset of major principal components and \( r \) is a subset of minor principal components. The major principal component score threshold or cutoff is denoted \( T_q \) while the minor principal component score threshold or cutoff is referred to as \( T_r \). \( q \) is most significant principal components and \( r \) is least significant principal components. An attack occurs for any observation dataset \( X \) if:

\[ \sum_{i=1}^{q} y_i^2/\lambda_i > T_q \quad \text{or} \quad \sum_{i=m-r+1}^{m} y_i^2/\lambda_i > T_r \quad (5) \]

### 3.2 Chi-square distribution

The Chi-square distribution is a statistical distribution. It has a minimum value 0, but no bound in maximum value. The peak of the curve is to the right of 0, and slowly decreases. The mean and the standard deviation of the Chi square distribution are respectively the degree of freedom and double of the degrees of freedom. There is a different Chi-square distribution for each degree of freedom. This means that the Chi square distribution covers more regions with a peak farther to the right, for higher than for smaller degrees of freedom. For any specific significance level, the critical region yields a larger Chi-square value, the higher the degree of freedom [37]. The multivariate data size is represented by the covariance matrix. Mahalanobis distance is a popular distance measurement technique which involves this covariance matrix in the calculation. For \( m \)-dimensional multivariate sample \( x_i (i = 1, \ldots, n) \), the Mahalanobis distance (MD) is represented as

\[ \text{MD}_i = \left( (x_i - t)^T C^{-1} (x_i - t) \right)^{1/2}, \quad \text{for} \; i = 1, \ldots, n \]
Where $C$ and $t$ are respectively the estimated covariance matrix and multivariate arithmetic mean. The values are approximately Chi-square distributed with $p$ degrees of freedom ($X^2_p$) for multivariate Gaussian distributed data. Observations which have a large Mahalanobis distance can be considered as multivariate outliers. That’s why; we apply the concept of quantiles. This method has many drawbacks. An efficient technique estimates Mahalanobis distances to measure outliers efficiently. The minimum covariance determinant estimator is most commonly used in such problems. Using efficient location estimators in the above equation yields robust distances. Let the squared robust distance for an observation is higher than $X^2_{p,0.995}$, it can be considered as an outlier. The Chi-square graph is obtained by the squared Mahalanobis distances. It can be time consuming for big data sets, and also to some cases it is subjective [38].

3.3 Gaussian Mixture Distribution

Let the covariance matrix $C$ is non-singular. The probability density function of $X$ can be represented as:

$$|2\Pi C|^{1/2} \exp(-1/2 (x-\mu)^\top C^{-1} (x-\mu))$$

(6)

Here $\mu$ is the mean value, $C$ is covariance matrix and $|$ represents the determinant. Singular covariance matrix is also possible in multivariate Gaussian distributions. The above equation cannot be applied for the probability distribution function in that case. Let us consider here the covariance matrices are non-singular. Let an N components mixture model, each $z_m$ is between 1 and N. Maximizing the sum is

$$\sum_m \sum_o p(o|x_m;\theta_o)[\log p(o;\theta) + \log p(x_m|o;\theta)]$$

(7)

From the above mixture model equation, we get $p(o;\theta) = t_o$ and $p(x_m|o;\theta) = f(x_m;\theta_o)$

Maximizing sum equation can be represented as

$$E = \sum_m \sum_o p(m|x_m;\theta)[\log t_o + \log f(x_m;\theta_o)]$$

(8)

In the E-step- membership degree, we apply Bayes theorem:

$$W_{mo} = p(o|x_m;\theta_o) = f(x_m;\theta_o).t_o/\sum_o f(x_m;\theta_o).t_o$$

(9)

In the M-step- weighted maximum likelihood,

$$\sum_m \sum_o W_{mo} \log t_o = \sum_o h_o \log t_o$$

(10)

Where $h_o = \sum_m W_{mo}, \sum_o t_o = 1$

Applying a Lagrange multiplier, its solution is written as

$$t_o = h_o/\sum h_n$$

(11)

Maximizing

$$\sum_m \sum_o W_{mo} \log f(x_m;\theta_o)$$

(12)

This can be partitioned into $N$ different maximizations, each of the following form

$$g_o = \arg\max_x \sum_m W_{mo} \log f(x_m|g)$$

(13)

[40]

The above discussion is based on topic discussed in [40]. The cluster is used to automatically estimate the parameters of a Gaussian mixture model from sample dataset. This method is similar to normal clustering method except that it permits cluster parameters to accurately estimate even when the clusters overlap.

IV. EXPERIMENT

To carry out the experiments, the performance logs need to be generated. The steps for generating the performance logs are as follows.

- On the start menu, go to settings, then Control Panel. Double click Administrative Tools and double click Computer Management.
- Explore performance Logs and Alerts by right click Counter Logs, and then click New Log Settings.
- Type a name for the counter log and then click OK.
- Click Add Counters.
- In the Performance object box, select a performance object that need be monitored.
- Counters added for experiment.
• On the General tab under Sample data every, sampling interval of 15 seconds is configured.
• On the Log Files tab log files properties is configured as Comma delimited files that can be viewed later in reporting tools such as Microsoft Excel.

After the performance log has been generated for each day, the log is divided into 4 groups, and the average values for each column of the table are calculated. After finding the average values, the values are maintained in another table. These values are used as our normal data set.

In the meanwhile, for one day the system is left to work when the graphics driver, audio driver, and USB driver have been disabled. This generates the logs for system performance that have been considered as intruded data [16]. We have taken the same number and the same type of attributes in our experiments. For our experiment, we have taken the normal dataset and the testing dataset i.e. mixture dataset (normal and intrusion) have been shown in Tables 1 and 2, respectively.

### Table 1: Normal dataset with some selective attributes

<table>
<thead>
<tr>
<th>Committed byte in use</th>
<th>Available Mbytes</th>
<th>Cache faults/sec</th>
<th>Page faults/sec</th>
<th>Page writes/sec</th>
<th>Page op/sec</th>
<th>Pool Non-paged Allocs</th>
<th>Pool Paged Allocs</th>
<th>System driver total byte</th>
<th>Write copies/sec</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.82441</td>
<td>8508</td>
<td>172.57</td>
<td>2519</td>
<td>101.6124</td>
<td>671</td>
<td>295.3630</td>
<td>973</td>
<td>0.06857</td>
<td>4005</td>
</tr>
<tr>
<td>3.64145</td>
<td>3572</td>
<td>173.68</td>
<td>8485</td>
<td>57.15040</td>
<td>864</td>
<td>256.0914</td>
<td>499</td>
<td>0.24534</td>
<td>2241</td>
</tr>
<tr>
<td>5.11114</td>
<td>4928</td>
<td>168.01</td>
<td>9718</td>
<td>79.47708</td>
<td>971</td>
<td>334.4868</td>
<td>715</td>
<td>0.14372</td>
<td>3968</td>
</tr>
<tr>
<td>5.63854</td>
<td>6946</td>
<td>165.26</td>
<td>9128</td>
<td>36.17087</td>
<td>94</td>
<td>162.9619</td>
<td>512</td>
<td>0.03686</td>
<td>9153</td>
</tr>
<tr>
<td>4.57840</td>
<td>4615</td>
<td>170.97</td>
<td>1429</td>
<td>51.11998</td>
<td>798</td>
<td>280.3527</td>
<td>608</td>
<td>0.15858</td>
<td>5618</td>
</tr>
<tr>
<td>4.77965</td>
<td>7979</td>
<td>1696</td>
<td>49.93331</td>
<td>807</td>
<td>128.9745</td>
<td>175</td>
<td>0.16840</td>
<td>9088</td>
<td>21694</td>
</tr>
<tr>
<td>6.23580</td>
<td>037</td>
<td>1640.77</td>
<td>8065</td>
<td>94.47212</td>
<td>975</td>
<td>308.6162</td>
<td>728</td>
<td>0.13230</td>
<td>9061</td>
</tr>
<tr>
<td>5.61765</td>
<td>1849</td>
<td>1648.30</td>
<td>9971</td>
<td>46.02698</td>
<td>896</td>
<td>219.4003</td>
<td>424</td>
<td>0.01555</td>
<td>0526</td>
</tr>
<tr>
<td>4.11134</td>
<td>8119</td>
<td>1720.17</td>
<td>0732</td>
<td>132.1210</td>
<td>622</td>
<td>525.6392</td>
<td>525</td>
<td>0.06567</td>
<td>2177</td>
</tr>
<tr>
<td>9.08474</td>
<td>6823</td>
<td>1619.29</td>
<td>0476</td>
<td>68.32869</td>
<td>399</td>
<td>458.0278</td>
<td>67</td>
<td>0.60135</td>
<td>9321</td>
</tr>
<tr>
<td>10.70498</td>
<td>4185</td>
<td>1580.08</td>
<td>8785</td>
<td>76.97290</td>
<td>274</td>
<td>512.5241</td>
<td>61</td>
<td>1.16544</td>
<td>1987</td>
</tr>
<tr>
<td>10.84210</td>
<td>5109</td>
<td>1563.67</td>
<td>4455</td>
<td>102.9971</td>
<td>76</td>
<td>469.8837</td>
<td>442</td>
<td>0.46537</td>
<td>3406</td>
</tr>
<tr>
<td>8.92087</td>
<td>0039</td>
<td>1650.93</td>
<td>6255</td>
<td>35.11830</td>
<td>951</td>
<td>189.8569</td>
<td>296</td>
<td>0.29024</td>
<td>2311</td>
</tr>
<tr>
<td>9.01944</td>
<td>5732</td>
<td>1629.35</td>
<td>9551</td>
<td>123.1521</td>
<td>235</td>
<td>660.7778</td>
<td>565</td>
<td>1.23764</td>
<td>2668</td>
</tr>
<tr>
<td>8.15211</td>
<td>776</td>
<td>1669.16</td>
<td>8831</td>
<td>46.88692</td>
<td>477</td>
<td>185.9397</td>
<td>726</td>
<td>0.32143</td>
<td>1061</td>
</tr>
<tr>
<td>9.78973</td>
<td>4979</td>
<td>1564.33</td>
<td>3333</td>
<td>108.4193</td>
<td>887</td>
<td>688.6777</td>
<td>162</td>
<td>1.01201</td>
<td>838</td>
</tr>
<tr>
<td>8.82137</td>
<td>0882</td>
<td>1635.53</td>
<td>8462</td>
<td>70.41303</td>
<td>081</td>
<td>401.7980</td>
<td>469</td>
<td>0.52689</td>
<td>9445</td>
</tr>
<tr>
<td>9.22601</td>
<td>1605.61</td>
<td>82.65173</td>
<td>478.1532</td>
<td>0.81245</td>
<td>12.9993</td>
<td>44724.5</td>
<td>48669.6</td>
<td>7407484</td>
<td>582763</td>
</tr>
</tbody>
</table>

©IJAET
ISSN: 2231-1963

Vol. 5, Issue 1, pp. 387-398
<table>
<thead>
<tr>
<th>2337</th>
<th>8421</th>
<th>496</th>
<th>06</th>
<th>8579</th>
<th>3727</th>
<th>7763</th>
<th>.211</th>
<th>1903</th>
</tr>
</thead>
<tbody>
<tr>
<td>13.1381</td>
<td>8528</td>
<td>1567.82</td>
<td>60.97717</td>
<td>436</td>
<td>335.7265</td>
<td>847</td>
<td>0.80176</td>
<td>4537</td>
</tr>
<tr>
<td>10.2376</td>
<td>1708</td>
<td>1621.77</td>
<td>107.5001</td>
<td>39</td>
<td>639.6973</td>
<td>908</td>
<td>0.61802</td>
<td>9752</td>
</tr>
<tr>
<td>14.1174</td>
<td>0933</td>
<td>1537.98</td>
<td>6938</td>
<td>55.69721</td>
<td>551</td>
<td>329.7004</td>
<td>828</td>
<td>1.10637</td>
</tr>
<tr>
<td>11.3247</td>
<td>4835</td>
<td>1523.32</td>
<td>46.23485</td>
<td>621</td>
<td>296.8309</td>
<td>553</td>
<td>0.23660</td>
<td>2086</td>
</tr>
<tr>
<td>8.32844</td>
<td>5781</td>
<td>1651.97</td>
<td>48.98892</td>
<td>139</td>
<td>329.3808</td>
<td>661</td>
<td>0.64198</td>
<td>8393</td>
</tr>
<tr>
<td>11.6934</td>
<td>3107</td>
<td>1548.83</td>
<td>46.67733</td>
<td>261</td>
<td>268.6304</td>
<td>243</td>
<td>0.67348</td>
<td>6201</td>
</tr>
<tr>
<td>13.8784</td>
<td>9525</td>
<td>1432.84</td>
<td>40.06764</td>
<td>295</td>
<td>200.9667</td>
<td>528</td>
<td>0.45081</td>
<td>3931</td>
</tr>
<tr>
<td>9.62421</td>
<td>2947</td>
<td>1599.36</td>
<td>52.68971</td>
<td>743</td>
<td>289.4970</td>
<td>927</td>
<td>0.56781</td>
<td>3405</td>
</tr>
<tr>
<td>10.1362</td>
<td>5358</td>
<td>1567.27</td>
<td>80.36205</td>
<td>126</td>
<td>476.6320</td>
<td>696</td>
<td>0.89210</td>
<td>8133</td>
</tr>
<tr>
<td>14.3701</td>
<td>0283</td>
<td>1552.65</td>
<td>32.39166</td>
<td>472</td>
<td>251.1562</td>
<td>525</td>
<td>0.61583</td>
<td>7055</td>
</tr>
<tr>
<td>8.60691</td>
<td>6888</td>
<td>1628.66</td>
<td>51.56334</td>
<td>321</td>
<td>258.0409</td>
<td>433</td>
<td>0.31237</td>
<td>3168</td>
</tr>
<tr>
<td>9.20217</td>
<td>6526</td>
<td>1585.38</td>
<td>140.3664</td>
<td>739</td>
<td>874.0920</td>
<td>392</td>
<td>1.14834</td>
<td>4959</td>
</tr>
<tr>
<td>8.17029</td>
<td>6367</td>
<td>1637.28</td>
<td>131.4596</td>
<td>708</td>
<td>607.5526</td>
<td>303</td>
<td>0.72011</td>
<td>2328</td>
</tr>
<tr>
<td>10.3369</td>
<td>9814</td>
<td>1550.55</td>
<td>84.19619</td>
<td>754</td>
<td>483.3572</td>
<td>935</td>
<td>0.56207</td>
<td>5459</td>
</tr>
<tr>
<td>8.68097</td>
<td>7551</td>
<td>1642.34</td>
<td>120.2821</td>
<td>865</td>
<td>367.8754</td>
<td>246</td>
<td>0.28778</td>
<td>2024</td>
</tr>
<tr>
<td>15.5496</td>
<td>2495</td>
<td>1376.74</td>
<td>66.20661</td>
<td>941</td>
<td>480.5522</td>
<td>772</td>
<td>0.50815</td>
<td>5306</td>
</tr>
<tr>
<td>9.92405</td>
<td>1137</td>
<td>1585.74</td>
<td>67.85154</td>
<td>811</td>
<td>336.0418</td>
<td>492</td>
<td>0.35963</td>
<td>2859</td>
</tr>
<tr>
<td>14.1485</td>
<td>8942</td>
<td>1450.71</td>
<td>69.47134</td>
<td>899</td>
<td>348.4456</td>
<td>721</td>
<td>0.83024</td>
<td>7837</td>
</tr>
<tr>
<td>12.1838</td>
<td>5114</td>
<td>1465.22</td>
<td>127.4100</td>
<td>196</td>
<td>485.9748</td>
<td>84</td>
<td>1.03672</td>
<td>1365</td>
</tr>
<tr>
<td>14.0909</td>
<td>7012</td>
<td>1440.05</td>
<td>54.06844</td>
<td>756</td>
<td>234.6842</td>
<td>2136</td>
<td>0.36904</td>
<td>3016</td>
</tr>
<tr>
<td>9.29341</td>
<td>2873</td>
<td>1583.12</td>
<td>108.4084</td>
<td>986</td>
<td>817.3167</td>
<td>886</td>
<td>1.13076</td>
<td>3968</td>
</tr>
<tr>
<td>8.27161</td>
<td>7418</td>
<td>1650.73</td>
<td>9394</td>
<td>235.4550</td>
<td>432</td>
<td>575.5292</td>
<td>127</td>
<td>0.40006</td>
</tr>
</tbody>
</table>

**Table 2:** Testing dataset with some selective attributes

<table>
<thead>
<tr>
<th>Committe d byte in use</th>
<th>Availab le MBytes</th>
<th>Cache faults/sec</th>
<th>Page faults/sec</th>
<th>Page writes/sec</th>
<th>Page op/sec</th>
<th>Pool Nonpage Allocs</th>
<th>Pool Paged Allocs</th>
<th>System driver total byte</th>
<th>Write copies/sec</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.601389</td>
<td>1700.02</td>
<td>2388</td>
<td>26.0404</td>
<td>3396</td>
<td>132.996</td>
<td>7699</td>
<td>0.08761</td>
<td>4078</td>
<td>1.40176</td>
</tr>
<tr>
<td>4.94660</td>
<td>1690.17</td>
<td>85.3540</td>
<td>371.750</td>
<td>2828</td>
<td>4.52559</td>
<td>32166.5</td>
<td>43517.9</td>
<td>570358</td>
<td>7.65501</td>
</tr>
</tbody>
</table>
4.1. False alarm rate

False alarm rate and detection rate can be calculated using confusion matrix. Confusion matrix is shown below.

<table>
<thead>
<tr>
<th>Actual Class</th>
<th>Predicted Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>C</td>
<td>TN</td>
</tr>
<tr>
<td></td>
<td>FP</td>
</tr>
<tr>
<td>NC</td>
<td>FN</td>
</tr>
<tr>
<td></td>
<td>TP</td>
</tr>
</tbody>
</table>

where,

C – Anomaly class,    NC – Normal class
TN – True Negative, FN – False Negative
TP – True Positive, FP – False Positive

Recall (R) = TP / (TP+ FN), Precision (P) = TP / (TP+FP)

F-measure = \( \frac{RP(1+ \beta^2)}{R \beta^2 + P} \), where R and P denote Recall and Precision, respectively; \( \beta \) is the relative importance of precision vs. recall and it is usually set to 1. Therefore, we can have F-measure as 2RP/(R+P).

4.2. Comparative results

<table>
<thead>
<tr>
<th>Name of Used Techniques</th>
<th>Detection rate</th>
<th>False alarm rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Principal Component Analysis</td>
<td>97.5%</td>
<td>2.5%</td>
</tr>
<tr>
<td>Chi-square distribution</td>
<td>90%</td>
<td>10%</td>
</tr>
<tr>
<td>Cluster with Gaussian Mixture Distribution</td>
<td>97.5%</td>
<td>2.5%</td>
</tr>
</tbody>
</table>
Here, we have used three different techniques for our experiments. We have got the results for PCA technique and Gaussian Mixture Distribution each, which is 97.5% detection rate. We have also got good result in Chi-distribution method. Using these above techniques we can easily detect hardware based intrusion. All these three techniques are shown by implementing for HIDS on the basis of performance log and two (i.e. Principal Component Analysis and Cluster with Gaussian Mixture Distribution) among these three give good results. The PCA and Gaussian mixture distribution give the detection rate maximum 97.5% each and the Chi-square distribution gives 90%. Generally, anomaly detection systems suffer from false alarm rate. Another problem in anomaly detection systems is that they are not able to classify the boundary data i.e. sometimes detection systems show the normal data as intrusion data and vice versa. That’s why we have used confusion matrix that gives accurate detection rate and false alarm rate. We can apply these techniques for any type of outlier detection and huge dataset also.

V. FUTURE WORK

In future work, we will try to produce better results in terms of better detection rate and lesser false alarm rate. We will also explore other statistical techniques that can be used in our work for improving the results.

VI. CONCLUSIONS

In this paper, we have analyzed an intrusion detection system using Chi-square distribution and Gaussian mixture distribution. We have shown the comparative results of Principal Component Analysis, Chi-square distribution and Gaussian mixture distribution. Our experimental results show that the PCA and Gaussian mixture distribution each give detection rate maximum 97.5% and the Chi-square distribution 90%. Generally, anomaly detection system suffers from false alarm rate. We can apply these methods for any type of outlier detection and we can apply these techniques for huge dataset also.

ACKNOWLEDGEMENT

The authors express their sincere thanks to Prof. S. Chand for his invaluable comments and suggestions.

REFERENCES


AUTHORS

Hari Om is presently working as Assistant Professor in the department of Computer Science & Engineering at Indian School of Mines, Dhanbad, India. He earned his Ph.D. in Computer Science from Jawaharlal Nehru University, New Delhi, India. He has published more than 50 research papers in International and National Journals including various Trans. of IEEE, Springer, Elsevier etc., International and National conferences of high repute. His research areas are Video-on-Demand, Cryptography, Network security, Data Mining, and Image Processing.

Tanmoy Hazra has completed his M.Tech in Computer Applications, department of Computer Science and Engineering from Indian School of Mines, Dhanbad in 2012. He is presently working as Assistant Professor in the department of Computer Engineering at ISB & M School of Technology, Pune. His research interest is mainly on Intrusion Detection Systems.