IJDACR
ISSN: 2319-4863

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary research
Website: www.ijdacr.com (Volume 1, Issue 1, August 2012)

A Genetically Optimized Network Intrusion Detection System using K-means Clustering via Principal Component Analysis

Shweta Gupta
Shwetagupta_285@yahoo.com

Prashant dutta
prashantdutta786@gmail.com

***Abstract*— Intrusion detection is to detect attacks against a computer system. It is an important technology in business sector as well as an active area of research. In Information Security, intrusion detection is the act of detecting actions that attempt to compromise the confidentiality, integrity or availability of a resource. It plays a very important role in attack detection, security check and network inspect.**
This paper presents the performance of k-mean algorithm for various values of number of clusters, based on experiments. The optimization of output is done using genetic algorithm by selecting initial through GA. Preliminary experiments with KDD cup'99 Data set show that the k-mean clustering can effectively detect intrusive attacks and achieves a low false positive rate. Here PCA is used to reduce the dimensionality of feature vectors extracted from data for analysis and visualization.
***Keywords*— Intrusion detection, Principle Component Analysis, K-means Clustering, Genetic Algorithm, Data mining.**

I. INTRODUCTION

Intrusion detection systems have evolved from monolithic batch-oriented systems to distributed real-time networks of components [2] as shown in Figure 1.

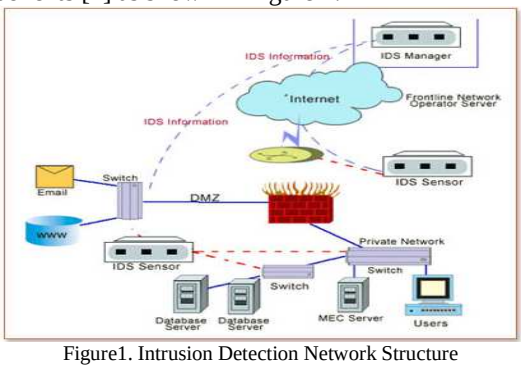

Figure1. Intrusion Detection Network Structure

In current systems, a number of common building functional blocks can be distinguished:

- Sensor, Probe: These modules form the most primitive data-gathering components of IDS. Implemented in a highly system-specific fashion, they track network traffic, log files or system behaviour - translating raw data into events useable by the IDS monitors [1].
- Monitor: Monitor components, the main processing segment of IDS, receive events from sensors. These events are then correlated against the IDS behaviour models, potentially producing model updates and alerts. Alerts, events in themselves, indicate occurrences significant to the security of a system, and may be forwarded to higher-level monitors, or to resolver units [6].
- Resolver: Resolver components receive suspicion reports from monitors (in the form of one or more alerts), and determine the appropriate response - logging, changing the behaviour of lower level components, reconfiguring other security mechanisms (e.g. adding firewall rules), and notifying operators.
- Controller: Facilitating component configuration and coordination, controller components are most significant in distributed ID systems, where manually upgrading, configuring and starting a network-wide series of components would be infeasible. In addition, controller units offer a single point of administration and interrogation for IDS, and may act in a supervisory capacity, restarting failed components.

Categories of intrusion detection

IDSs can also be categorized according to the detection approaches they use. Basically, there are two detection methods: misuse detection and anomaly detection [7]. The major deference between the two methods is that misuse detection identifies intrusions based on features of known attacks while anomaly detection analyzes the properties of normal behaviour [5]. IDSs that employ both detection methods are called hybrid detection-based IDSs [3].

A. Anomaly Detection Systems

IJDACR
ISSN: 2319-4863

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary research
Website: www.ijdacr.com (Volume 1, Issue 1, August 2012)

Anomaly-based intrusion detection techniques depict an intrusion as a deviation from nor-mal behaviour. The so called 'normal behaviour needs to be defined, or learned [8][9].

B. Misuse detection systems

Misuse detection systems define what an intrusion is in the observed system and uses a knowledge-base of system vulnerabilities and patterns of known security violations - so- called signatures - as a model of the intrusive process [4].

Dos attacks in our data set

S.No	Attack	Category
1	Smurf	DOS
2	Neptune	DOS
3	Back	DOS
4	Teardrop	DOS
5	Pod	DOS

II. Methodology

In this project, PCA is used to reduce the dimensionality of feature vectors extracted from data for analysis, and separation of components are done by dividing it into clusters by applying K-means clustering, and Genetic algorithm is used to determine optimum number of clusters in analyzed data. These methods are described below:

Principal Component Analysis (PCA)

The Principal Component Analysis (PCA) is one of the most successful techniques that have been used in feature extraction. PCA is a statistical method under the broad title of factor analysis. Before getting to a description of PCA, this portion first introduces mathematical concepts that will be used in PCA. It covers standard deviation, covariance, eigenvectors and Eigen-values. This background knowledge is meant to make the understanding of PCA very straightforward.

Eigen Vector, Eigen Value:

Transformations of space such as translation (or shifting the origin), rotation, reflection, stretching, compression, or any combination of these; other transformations could also be listed may be visualized by the effect they produce on vectors. Vectors can be visualized as arrows pointing from one point to another.

a) *Eigenvectors* of transformations are vectors which are either left unaffected or simply multiplied by a scale factor after the transformation. An *eigenvector* of a transformation is a non-null vector whose direction is unchanged by that transformation.
b) An eigenvector's *eigenvalue* is the scale factor that it has been multiplied.
c) The *geometric multiplicity* of an eigenvalue is the dimension of the associated eigenspace.
d) The *spectrum* of a transformation on finite dimensional vector spaces is the set of all its eigenvalues.

K–means Clustering Algorithm

A clustering method that doesn't require computation of all possible distances is K-means clustering. It is one of the simplest unsupervised learning algorithms that solve the well known clustering problem. The procedure follows a simple and easy way to classify a given data set through a certain number of clusters (assume k clusters) fixed a priori. The main idea is to define k centroids, one for each cluster. These centroids should be placed in a cunning way because of different location causes different result. So, the better choice is to place them as much as possible far away from each other. The next step is to take each point belonging to a given data set and associate it to the nearest centroid. When no point is pending, the first step is completed and an early groupage is done. At this point we need to re-calculate k new centroids as barycenters of the clusters resulting from the previous step. After we have these k new centroids, a new binding has to be done between the same data set points and the nearest new centroid. A loop has been generated. As a result of this loop we may notice that the k centroids change their location step by step until no more changes are done. In other words centroids do not move any more.

$$J = \sum_{j=1}^{k} \sum_{i=1}^{x} ||x_i^{(j)} - c_j||^2$$

Genetic Algorithm

In a genetic algorithm, a population of strings (called chromosomes or the genotype of the genome), which encode candidate solutions (called individuals, creatures, or phenotypes) to an optimization problem, is evolved toward better solutions. Traditionally, solutions are represented in binary as strings of 0s and 1s, but other encodings are also possible. The evolution usually starts from a population of randomly generated individuals and happens in generations. In each generation, the fitness of every individual in the population is evaluated, multiple individuals are stochastically selected from the current population (based on their fitness), and modified (recombined and possibly randomly mutated) to form a new population. The new population is then used in the next iteration of the algorithm.

A typical genetic algorithm requires:

- A genetic representation of the solution domain,

IJDACR
ISSN: 2319-4863

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary research
Website: www.ijdacr.com (Volume 1, Issue 1, August 2012)

- A fitness function to evaluate the solution domain.

Here, fitness function is our principle component analysis based K-means clustering algorithm, for the optimization of our solution domain.

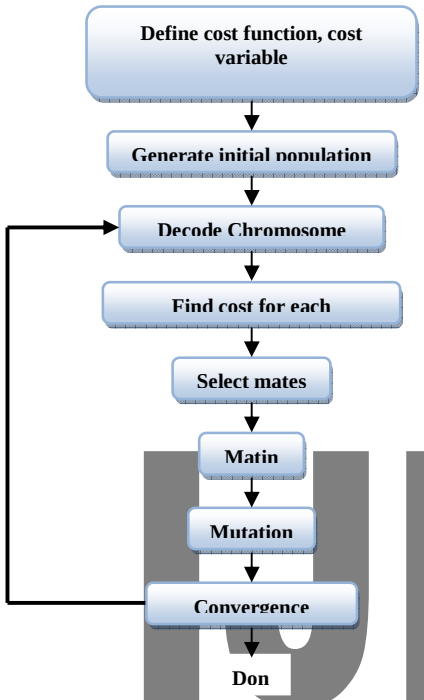

Figure.2 A path through the components of the GA

KDD cup'99 Dataset

With the enormous growth of computer networks usage and the huge increase in the number of applications running on top of it, network security is becoming increasingly more important. As it is shown in [10], all the computer systems suffer from security vulnerabilities which are both technically difficult and economically costly to be solved by the manufacturers. Therefore, the role of Intrusion Detection Systems (IDSs), as special-purpose devices to detect anomalies and attacks in the network, is becoming more important.

The pseudo code for the adapted k-mean algorithm used after PCA is presented as below:

/**********start of pseudo code

1. Choose random k data points as initial Clusters Mean (cluster centre)

2. Repeat

3. For each data point x from D

4. Computer the distance x and each cluster mean (centroid)

5. Assign x to the nearest cluster.

6. End for

7. Re-compute the mean for current cluster collections.

8. until reaching stable cluster

9. Use these centroid for normal and anomaly traffic.

10. Calculate distance of centroid from normal and anomaly centroid points.

11. If distance(X, Dj) > = 5

12. Then anomaly found; exit

13. Else then

14. X is normal;

*/ end of pseudo code.

Principal component analysis (PCA) is a widely used statistical technique for unsupervised dimension reduction. K-means clustering is a commonly used data clustering for unsupervised learning tasks. Here we prove that principal components are the continuous solutions to the discrete cluster membership indicators for K-means clustering. Equivalently, we show that the subspace spanned by the cluster centroids are given by spectral expansion of the data covariance matrix truncated at K−1 terms. These results indicate that unsupervised dimension reduction is closely related to unsupervised learning. On dimension reduction, the result provides new insights to the observed effectiveness of PCA-based data reductions, beyond the conventional noise-reduction explanation. Mapping data points into a higher dimensional space via kernels, we show that solution for Kernel K-means is given by Kernel PCA. On learning, our results suggest effective techniques for K-means clustering. Here we explore the connection between these two widely used methods. We prove that principal components are actually the continuous solution of the cluster membership indicators in the K-means clustering method, i.e., the PCA dimension reduction automatically performs data clustering according to the K-means objective function. This provides an important justification of PCA-based data reduction. K-means method uses K prototypes, the centroids of clusters, to characterize the data. They are determined by minimizing the sum of squared errors.

IJDACR
ISSN: 2319-4863

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary research
Website: www.ijdacr.com (Volume 1, Issue 1, August 2012)

Evaluation and Analysis

To evaluate the effectiveness of the data clustering K-mean algorithm over the DARPA test data, it describes the results using Detection Rate (DR), False Positive Rate (FPR), and Accuracy (ACC). Each metric is defined below.

- *Detection Rate (DR):*

Detection rate is the ratio of correctly classified intrusive examples to the total number of intrusive examples. The detection rate of our detection approach over the DARPA 1998 testing data is computed,

$$DR = \frac{\sum_{service=1}^{n} no.of\ detected\ attacks}{\sum_{service=1}^{n} no.of\ total\ attacks}$$

- *False Positive Rate (FPR):*

False positive rate is the ratio of incorrectly classified normal examples (false alarms) to the total number of normal examples. The false positive rate of the detection approach is calculated,

$$FRP = \frac{\sum_{service=1}^{n} no.\ of\ false\ alarms}{\sum_{service=1}^{n} no.\ of\ total\ normal\ process}$$

- *Accuracy (ACC):*

Accuracy is the ratio of correctly classified examples to the total number of classified examples. The accuracy of our experiments is computed.

$$ACC = \frac{\sum_{service=1}^{n} no.of\ correct\ classification}{\sum_{service=1}^{n} no.of\ total\ classification}$$

Basically, the detection rate and the false positive rate are tradeoffs in machine learning algorithms. When the detection rate is increased; the algorithm generally generates more false alarms. Hence, introduce the third metric, accuracy, to evaluate the overall prediction accuracy of the algorithm.

III. Result & Discussions

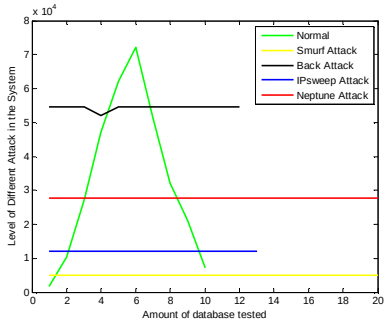

Figure.3 Level of different attacks in amount of database tested

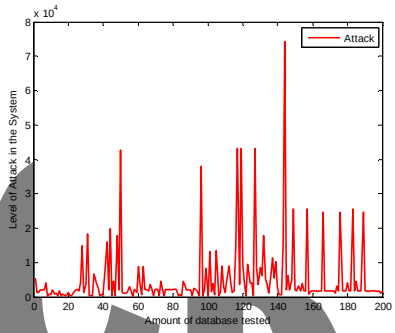

Figure.4 Level of attacks in amount of database tested

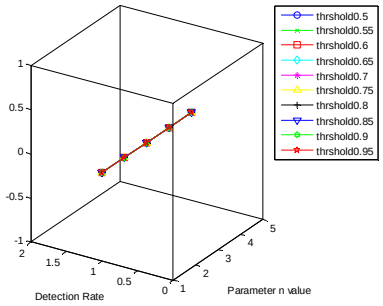

Figure.5 Detection rate in corrected data

Table1: Level of Attack

Attack	Samples	Category
Smurf	6000+	dos
neptune	50000+	dos
back	55000+	dos
IPsweep	12000+	dos
normal	71000+	normal

IJDACR
ISSN: 2319-4863

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary research
Website: www.ijdacr.com (Volume 1, Issue 1, August 2012)

```
Top 8 components are retained; cumulative eigenvalue contribution is 1.00

temp_c =

   -0.9900   -0.1407         0    0.1407    1.0000

Iteration: 0,  Best: 0.623699,  F: 0.500000,  CR: 0.800000,  NP: 5
Iteration: 1,  Best: 0.623699,  F: 0.500000,  CR: 0.800000,  NP: 5
Iteration: 2,  Best: 0.592311,  F: 0.500000,  CR: 0.800000,  NP: 5
Iteration: 3,  Best: 0.582581,  F: 0.500000,  CR: 0.800000,  NP: 5
Iteration: 4,  Best: 0.571200,  F: 0.500000,  CR: 0.800000,  NP: 5
```

Figure.6 Genetic Iterations in Command window

Discussions

Though the other clustering algorithm is popular for its simplicity, it has some drawback in choosing optimal number of clusters. However, the K-means clustering methods had shown tremendous achievements in areas of image processing and pattern recognition, and intrusion detection. The K-means is a good choice for circular and spherical clusters, but if the orientation of natural clusters is not spherical, then the algorithm leads to among almost wrong clusters. Another drawback of previous algorithm is that it imposes equal size clusters on the data set which is again a deviation from the natural clusters. The performance of any K-means clustering method is the best when the number of clusters is known appropriately. But most of the time, it is not the case and so researchers have devised a number of methods known as cluster validation indices to evaluate the clusters formed. Though out neural network has following disadvantages:

1. They are black box - that is the knowledge of its internal working is never known
2. To fully implement a standard neural network architecture would require lots of computational resources - for example you might need like 100,000 Processors connected in parallel to fully implement a neural network that would "somewhat" mimic the neural network of a cat's brain or may say it's a greater computational burden
3. Remember the No Free Lunch Theorem - a method good for solving 1 problem might not be as good for solving some other problem - Neural Networks though they behave and mimic the human brain they are still limited to specific problems when applied.
4. Since applying neural network for human-related problems requires Time to be taken into consideration but it's been noted that doing so is hard in neural networks.

Traditional data reduction perspective derives PCA as the best set of bilinear approximations (SVD of Y). The new results show that principal components are continuous (relaxed) solution of the cluster membership indicators in K-means clustering. These two views (derivations) of PCA are in fact consistent since data clustering also is a form of data reduction. Standard data reduction (SVD) happens in Euclidean space, while clustering is a data reduction to classification space (data points in same cluster are considered belonging to same class while points in different clusters are considered belonging to different classes). This is best explained by the vector quantization widely used in signal processing where the high dimensional space of signal feature vectors are divided into Voronoi cells via the K-means algorithm. Signal feature vectors are approximated by the cluster centroids, the code-vectors. That PCA plays crucial roles in both types of data reduction provides a unifying theme in this direction.

The use of genetic algorithm allows us to get an optimal solution of our proposed technique, by selecting the initial variables through the genetic algorithm.

IV. CONCLUSIONS

Intrusion detection systems (IDSs) play an important role in computer security. Many of the today's anomaly detection methods generate high false positives and negatives.

A Data clustering using K-mean algorithm is a good technique to address these problems, on dimension reduction; the result provides new insights to the observed effectiveness of PCA-based data reductions, beyond the conventional noise-reduction explanation. The results also show that a low false positive rate can be achieved. With the frequency-weighting method where each entry is equal to the number of occurrences of a system call during the TCP communication, finally genetic algorithm optimized or moves our results to a best one.

REFERENCES

1. Shi Zhong, Taghi Khoshgoftaar, And Naeem Seliya, "Clustering-Based Network Intrusion Detection", IJRQSE, Vol. 14, No. 02, pp. 169-187, 2007
2. Multiagent systems for network intrusion detection, a review. Alvaro herrero, EmilioCorchado, 2009
3. Intrusion Detection Techniques and Approaches Theuns Verwoerd and Ray Hunt, University of Canterbury, New Zealand.

IJDACR
International Journal Of Digital Application & Contemporary Research

International Journal of Digital Application & Contemporary research
Website: www.ijdacr.com (Volume 1, Issue 1, August 2012)

4. CERIAS Autonomous Agents for Intrusion Detection Group, "COAST Autonomous Agents for Intrusion Detection", Project homepage,7 September 1999.
5. James P. Anderson, "Computer Security Threat Monitoring and Surveillance", Technical Report, James P. Anderson Co., Fort Washington, PA, April 1980.
6. Rebecca Gurley Bace, "Intrusion Detection", 2001, ISBN 1-57870-185-6
7. Andrew Plato, Network ICE "BlackICE Defender User's Guide version1.0", 1999.
8. Vern Paxson "Bro: A System for Detecting Network Intruders in Real-Time" USENIX Security Symposium, January 1998, ftp://ftp.ee.lbl.gov/papers/bro-usenix98-revised.ps.Z
9. Marcus J. Ranum "Intrusion Detection: Challenges and Myths",1998.
10. C. E. Landwehr, A. R. Bull, J. P. McDermott, and W. S. Choi, "A taxonomy of computer program security flaws," ACM Comput. Surv., vol. 26, no. 3, pp. 211–254, 1994.
11. P.S.Prabhu, "Network Intrusion Detection Using Enhanced Adaboost Algorithm", International Journal of Communications and Engineering Volume 03– No.3, Issue: 02 March 2012.
12. Dalila BOUGHACI, Mohamed Lamine HERKAT, Mohamed Amine LAZZAZI, "A Specific Fuzzy Genetic Algorithm for Intrusion Detection", ICCIT, 2012.
13. R. Shanmugavadivu, Dr.N.Nagarajan, "Network Intrusion Detection System Using Fuzzy Logic" IJCSE Vol. 2 No. 1, 2011.
14. Nasser S. Abouzakhar And Abu Bakar , "A Chi-Square Testing-Based Intrusion Detection Model",CFET, 2010.
15. Debdutta Barman Roy, Rituparna Chaki, Nabendu Chaki, "A New Cluster-Based Wormhole Intrusion Detection Algorithm for Mobile Ad-Hoc Networks", IJNSA, Vol 1, No 1, April 2009.