Using Learning Vector Quantization in Alert Management of Intrusion Detection System

Amir Azimi Alasti Ahrabi *amir.azimi.alasti@gmail.com*
Department of Computer
Islamic Azad University, Shabestar Branch
Shabestar, East Azerbaijan, Iran

Kaveh Feyzi *kavehfeizi@gmail.com*
Department of Computer
Ataturk University
Erzurum, Turkey

Zahra Atashbar Orang *atashbarorang_z@yahoo.com*
Department of Computer
Islamic Azad University, Tabriz Branch
Tabriz, East Azerbaijan, Iran

Hadi Bahrbegi *hadi.bahrbegi@gmail.com*
Department of Computer
Islamic Azad University, Shabestar Branch
Shabestar, East Azerbaijan, Iran

Elnaz Safarzadeh *elnaz_safarzadeh@yahoo.com*
Department of Computer
Islamic Azad University, Shabestar Branch
Shabestar, East Azerbaijan, Iran

Abstract

Intrusion detection system (IDS) is used to produce security alerts to discover attacks against protected network and/or computer systems. IDSs generate high amount of security alerts and analyzing these alert by a security expert are time consuming and error pron. IDS alert management system are used to manage generated alerts and classify true positive and false positives alert. This paper represents an IDS alert management system that uses learning vector quantization technique to classify generated alerts. Because of low classification time per each alert, the system also could be used in active alert management systems.

Keywords: IDS, Alert Management, Learning Vector Quantization, Alert Classification, True Positive and False Positive Classification.

1. INTRODUCTION

An intrusion detection system (IDS) inspects all inbound and outbound network activity or computer system events and identifies suspicious patterns that may indicate a network or system attack from someone attempting to break into or compromise a system. [1]. IDSs are producing many alerts each day that many of them are false positive alerts. Big amount of the false positive alerts crowd and cover true positive alerts from security experts. Also identifying true positive from false positives are time consuming and error prone therefore IDS alert management system are introduced to manage generated IDS alerts. IDSs can be used as active or passive. In passive usage of IDS, it analyzes traffics or events in offline mode but active IDSs work in online mode. To manage alerts concurrently with alerts generation, active alert management systems are used. Active alert management systems same work in online mode as active IDSs. These

types of alert management systems should have little amount of alert analyze time to be used in online mode. Some of problems of IDS are: huge amount of generated alerts and high rate of false positive alert among generated alerts. Also most alert management system has low speed.

In this paper authors change their previous work and proposed a new alert management system by using Learning Vector Quantization (LVQ) [2]. It classifies the generated alerts based on attack type of alerts, detects false positive alerts, high speed classification to use with alert generation in IDSs. The proposed system uses some techniques of previous work techniques [3] such as alert filtering, alert preprocessing, and alert filtering to improve accuracy of the results.

In Section 1 the alert management system is introduced. Section 2 reviews related works, section 3 explains the suggested alert management system and describes all component of the proposed system, the experimental results are shown in section 4 and finally section 5 is a conclusion and future works.

2. RELATED WORKS

Alert management systems use various method and techniques. Clustering and classification of alerts is one of these techniques. A method of clustering based on root causes is proposed by K. Julisch [4] which clusters IDS alerts by discovering main cause of their occurrences. He proves that a small number of root causes imply 90% of alerts. By removing alerts related with these root causes total number of alerts come down to 82%. The system uses information about underlying network so it is not portable and this problem is a disadvantage of the algorithm.

Three artificial intelligence techniques with some dimension reduction techniques are used to cluster generated IDS alerts from DARPA 2000 dataset in [5] then produced results are compared. The problems of that system are: row alert without preprocessing are entered to the algorithms and system is not tuned. Cuppens proposed another method that uses expert system to make decision [8, 16]. In [6, 7] two genetic clustering algorithm based, named Genetic Algorithm (GA) and Immune based Genetic Algorithm (IGA) used to manage IDS alerts. Their proposed methods depend on underlying network information same as method proposed by Julisch.

Wespi et al. [17] design a system that aggregates alerts together by placing them in situations. Situations are set of special alerts. To construct a situation, source, destination and attack class attributes of alert are used.

Authors of this paper propose a system that manages alert generated from DARPA 98 dataset [3]. Some algorithms such as alert filtering, alert preprocessing and cluster merging are used in the system. The main unit of the system is cluster/classify unit that uses Self-Organizing Maps (SOM) [2] to cluster and classify IDS alerts. Results of [3] show that SOM was able to cluster and classify true positive and false positive alerts more accurate than other techniques.

In another work, authors have developed an alert management system [9] similar to [3]. In that work usage of seven genetic clustering algorithms named Genetic Algorithm (GA) [18], Genetic K-means Algorithm (GKA) [19], Improved Genetic Algorithm (IGA) [20], Fast Genetic K-means Algorithm (FGKA) [21], Genetic Fuzzy C-means Algorithm (GFCMA) [22], Genetic Possibilistic C-Means Algorithm (GPCMA) [9] and Genetic Fuzzy Possibilistic C-Means Algorithm (GFPCMA) [9] to cluster and classify true positive and false positive alerts, are explained. The system after clustering alerts then prioritized produced clusters with Fuzzy Inference System [9].

In this paper an alert management system based on system proposed by authors in [3] is proposed that uses LVQ as a tool to classify input alert vectors. Propose of this paper evaluating another type of Kohonen networks named LVQ [2] in alert management system field. The system will be able to improve accuracy of results and also to reduce the number of false positive alerts.

3. USING LVQ IN ALERT MANAGEMENT SYSTEM

The proposed system is shown in Figure 1. In this paper we use binary traffics files of a network named DARPA 98 dataset [10] instead of real network traffics. Snort tool [11] is used to produce alerts of DARPA 98 dataset network traffics. Snort is an open source signature based IDS which gets DARPA 98 online traffic and then generates alert log files [3]. After generating alert log files with Snort tool, these files are entered to the proposed system as its input.

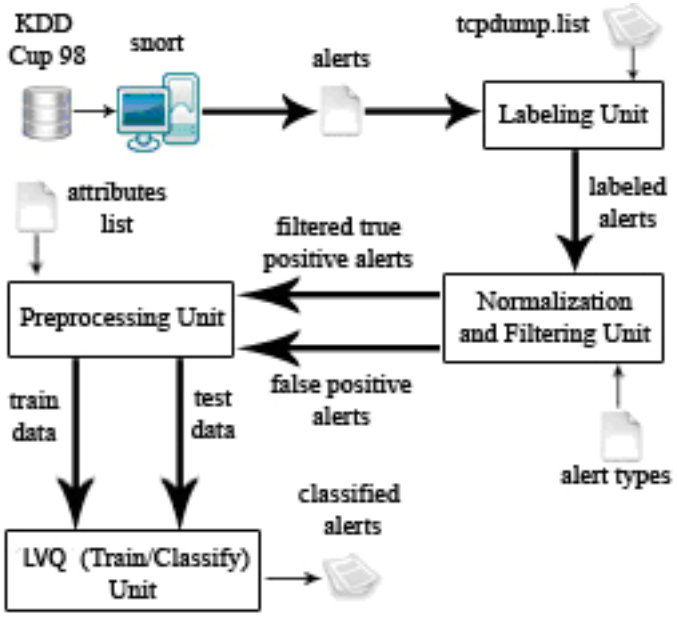

FIGURE 1: Proposed alert management system.

3.1 Labeling Unit

Labeling unit gets generated alert from Snort and `tcpdump.list` files of DARPA 98 dataset and then generate labeled alert which each alert has own attack type. `tcpdump.list` files contain information about all packets in DARPA 98 dataset. These labels are used to train LVQ and evaluate results of LVQ [3, 9].

3.2 Normalization and Filtering Unit

In this phase accepted attack types are entered to the unit and only alerts that are in class of these attack types are selected [3, 9,12]. This unit uses eight attributes of alert to filter alert, this attributes are: Signature ID, Signature Rev, Source IP, Destination IP, Source Port, Destination Port, Datagram length and Protocol [13]

3.3 Preprocessing Unit

Preprocessing unit converts string values of attributes of alert to numerical data. It also reduces the range of attribute values and converts alerts to data vectors (1), (2) and (3).

$$IP = X_1.X_2.X_3.X_4,$$
$$IP\_VAL = (((X_1 \times 255) + X_2) \times 255 + X_3) \times 255 + X_4 \quad (1)$$

$$protocol\_val = \begin{cases} 0, protocol = None \\ 4, protocol = ICMP \\ 10, protocol = TCP \\ 17, protocol = UDP \end{cases} \quad (2)$$

$$IUR = 0.8 \times \frac{x - x_{\min}}{x_{\max} - x_{\min}} + 0.1 \quad (3)$$

3.4 LVQ Training and Classification Unit

In this unit we use LVQ as a classifier. LVQ should be trained with train dataset and then gets test dataset to classify them.

- **Learning Vector Quantization**

LVQ is a special artificial neural network; it applies a winner-take-all Hebbian learning-based approach. LVQ was invented by Teuvo Kohonen. It is a forerunner to SOM and related to Neural gas, and to the k-Nearest Neighbor algorithm (k-NN) [2].

An LVQ system is represented by prototypes W=(w(i),...,w(n)) which are defined in the feature space of data vectors. In winner-take-all training algorithms, the prototype which is closest to the input vector according to a given distance measure for each vectors of input data are determined. The position of this so-called winner prototype is then adapted, i.e. the winner is moved closer if it correctly classifies the data point or moved away if it classifies the data point incorrectly.

4. EXPERIMENTAL RESULTS

To simulate the proposed system C#.net programming language, MATLAB software and SOM toolbox is used [14, 15]. The parameters of simulation are shown below.

Suggested LVQ has 80 neurons in hidden layers. The LVQ gets a data vector of train data that each data vector consists of 8 attributes as input to the system. Training phase consists of 50 epochs. Learning function is learnlv2. Because of Input data vectors consist of 9 alert attack types, each attack type have typical class percentage 0.1 except false positive. False positive typical class percentage is 0.2. The attack types used in this simulation are: Back, Pod, Nmap, Imanp, Dict, Rootkit, Land and Phf. Train data contains 70% of total filtered alert data vectors or 10166 data vectors. The false positive count in the training dataset is 4113. Test dataset includes 30% of the data vectors of labeled alerts; it means 2591 data vectors of true positive, and 1764 data vectors of false positive alerts.

Figure 2 shows Mean Square Error (MSE) for each epoch. As you can see in this figure the error value is reduced when we moved forward on epoch axis; and minimum value of the error achieved in last step.

To evaluate the performance of algorithms four measurements are introduced, they are:

1- Classfication Error (ClaE),
2- Classfication Accuracy percent (ClaAR),
3- Average Alert Classification Time (AACT),
4- False Positive Reduction Rate (FPRR).

In table 1 value of these metrics are shown. The values of ClaE and ClaAR are 490 and 88.75% respectively (Table 1). The value of AACT measurement is 0.000018 that shows the proposed system can be used in active IDS alert management systems that evaluate alerts while IDS produces them simultaneously. False positive alert type identification known FPRR is an important point of extracted values. Because of production of false positive alerts beside true positive ones then this metric value is very important in modern IDS alert management systems. The value of this metric is 88.27% percent.

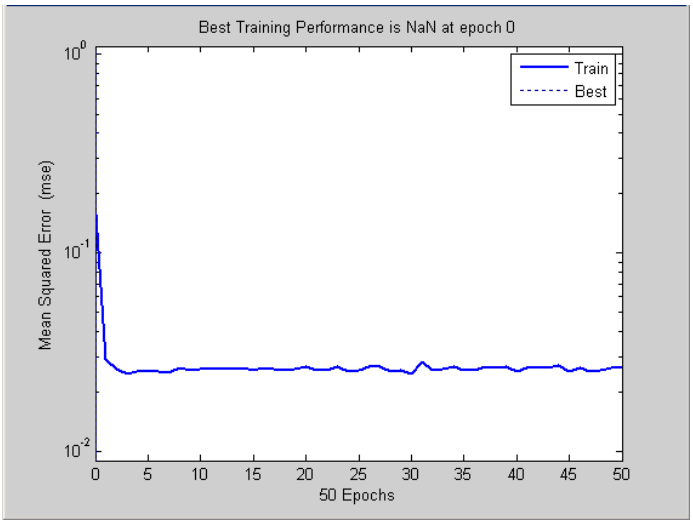

FIGURE 2: Errors of ANN output values per targets.

ClaE	ClaAR	FPRR	AACT
490	88.75	88.27	0.000018

TABLE 1: Extracted performance metric values from simulation.

In [9] GA based algorithms are used to cluster and classify alerts. These results are shown in table 2. For ClaE, ClaAR and FPRR metrics the proposed system has high value in contrast of GA and GKA. But other methods such as IGA, FGKA, GFCMA, GPCMA and GFPCMA have better performance in contrast proposed system. In AACT performance metric, LVQ based alert management system has better result than all of GA based techniques. It means that LVQ could be used in active alert management system.

Algorithm	ClaE	ClaAR	FPRR	AACT
GA	1218	72.03	52.15	Offline
GKA	1011	75.2	62.11	Offline
IGA	306	92.97	95.24	Offline
FGKA	314	92.79	97.51	Offline
GFCMA	148	96.60	97.51	Offline
GPCMA	91	97.91	96.03	Offline
GFPCMA	148	96.60	97.51	Offline

TABLE 2: Results of performance metrics for GA based algorithms.

5. CONCLUSION AND FUTURE WORKS

In this paper a LVQ based system is presented that is able to classify IDS alerts. The system solved some problems of IDSs such as generating high amount of alerts and false positive alert. The system could classify true positive alert and could identify false positive ones. The system identifies and drastically reduces the number of false positive alerts. The results of the proposed system are compared to GA based techniques. The comparison shows that in contrast of GA based systems LVQ algorithm can be used in active alert management systems.

It seems to be useful using LVQ to correlate alerts to discover attack sequences so this idea is another future work of this paper.

6. REFERENCES

[1] H. Debar, M. Dacier, and A. Wespi. "Towards a taxonomy of intrusion-detection systems", COMPUT. NETWORKS, Vol. 31, Issue: 8, pp.: 805-822, 1999.

[2] Kohonen, T, "Self-Organized Maps", Springer series in information. Science Berlin Heidelberg, 1997.

[3] Amir Azimi Alasti Ahrabi, Ahmad Habibizad Navin, Hadi Bahrbegi, Mir Kamal Mirnia, Mehdi Bahrbegi, Elnaz Safarzadeh, Ali Ebrahimi, "A New System for Clustering and Classification of Intrusion Detection System Alerts Using Self-Organizing Maps", International Journal of Computer Science and Security (IJCSS), Vol. 4, Issue 6, pp. 589 – 597, 2010.

[4] K. Julisch, "Clustering intrusion detection alarms to support root cause analysis", ACM Trans. on Information and System Security, Vol. 6, Issue 4, pp. 443 – 471, 2003.

[5] Maheyzah, M. S., Mohd Aizaini, M., and Siti Zaiton, M. H. (2009), "Intelligent Alert Clustering Model for Network Intrusion Analysis", Int. Jurnal in Advances Soft Computing and Its Applications (IJASCA), Vol. 1, Issue 1, pp. 33 – 48, 2009.

[6] Wang, J., Wang, H., Zhao, G., "A GA-based Solution to an NP-hard Problem of Clustering Security Events", IEEE, pp. 2093- 2097, 2006.

[7] Wang J., Baojiang Cui, "Clustering IDS Alarms with an IGA-based Approach", ICCCAS, pp. 586-591, 2009.

[8] Cuppens F., "Managing alerts in a multi-intrusion detection environment", Proceedings of the 17th Annual Computer Security Applications Conference on, pp. 22-31, 2001.

[9] Bahrbegi H., Navin A.H., Ahrabi A.A.A., Mirnia M. K., Mollanejad A., "A new system to evaluate GA-based clustering algorithms in Intrusion Detection alert management system", Nature and Biologically Inspired Computing (NaBIC), Second World Congress on, pp. 115 – 120, 2010.

[10] MIT Lincoln Lab., DARPA 1998 Intrusion Detection Evaluation Datasets. Available: http://www.ll.mit.edu/mission/communications/ist/corpora/ideval/data/1998data.html, 1998.

[11] Snort: The open source network intrusion detection system. Available: http://www.snort.org/.

[12] Brugger S. T., J. Chow, "An Assessment of the DARPA IDS Evaluation Dataset Using Snort", UC Davis Technical Report CSE-2007-1, Davis, CA, 2007.

[13] Snort Manual, www.snort.org/assets/82/snort_manual.pdf.

[14] Neural Network Toolbox, "ANN Toolbox for MATLAB", www.mathworks.com/products/neural-network, 2011.

[15] Matlab Software, http://www.mathworks.com.

[16] E. MIRADOR, "Mirador: a cooperative approach of IDS", European Symposium on Research in Computer Security (ESORICS). Toulouse, France, 2000.

[17] Debar H., Wespi A., "Aggregation and Correlation of Intrusion-Detection Alerts", Proceeding RAID '00 Proceedings of the 4th International Symposium on Recent Advances in Intrusion Detection, pp.:87-105, 2001.

[18] Krovi R., "Genetic Algorithm for Clustering: A preliminary investigation", Proceeding on 25th Hawaii International Conference on Systems Sciences (HICSS), pp. 540–544, 1992.

[19] Krishna K., Murty M., "Genetic K-means algorithm", IEEE Transactions on Systems, Man and Cybernetics - Part B: Cybernetics, pp. 433-439, 1999.

[20] Fuyan L., Chouyong C., Shaoyi L., "An Improved Genetic Approach", International Conference on Neural Networks and Brain, pp. 641-644, 2005.

[21] Lu Y., Lu S., Fotouhi F., Deng Y., Brown J. S., "FGKA: a Fast Genetic K-means Clustering Algorithm", Proceeding of the ACM Symposium on Applied computing (SAC), Nicosia, Cyprus, pp. 622-623, 2004.

[22] Nuovo A. D. G., Catania V., Palesi M., "The Hybrid Genetic Fuzzy C-means: a Reasoned Implementation", Proceedings of the 7th WSEAS International Conference on Fuzzy Systems, ACM, pp. 33-38, 2006.