I. J. Computer Network and Information Security, 2012, 11, 32-38
Published Online October 2012 in MECS (http://www.mecs-press.org/)
DOI: 10.5815/ijcnis.2012.11.04

Using Adaptive Neuro-Fuzzy Inference System in Alert Management of Intrusion Detection Systems

Zahra Atashbar Orang
Islamic Azad University, Tabriz Branch, Tabriz, Iran
atashbarorang_z@yahoo.com

Ezzat Moradpour
Islamic Azad University, Shabestar Branch, Shabestar, Iran
e.moradpoormail@yahoo.com

Ahmad Habibizad Navin
Islamic Azad University, Science and Research, Tabriz, Iran
ah_habibi@iaut.ac.ir

Amir Azimi Alasti Ahrabi
Islamic Azad University, Shabestar Branch, Shabestar, Iran
amir.azimi.alasti@gmail.com

Mir Kamal Mirnia
Islamic Azad University, Science and Research Branch, Tabriz, Iran
mirnia-kam@tabrizu.ac.ir

Abstract **—** By ever increase in using computer network and internet, using Intrusion Detection Systems (IDS) has been more important. Main problems of IDS are the number of generated alerts, alert failure as well as identifying the attack type of alerts. In this paper a system is proposed that uses Adaptive Neuro-Fuzzy Inference System to classify IDS alerts reducing false positive alerts and also identifying attack types of true positive ones. By the experimental results on DARPA KDD cup 98, the system can classify alerts, leading a reduction of false positive alerts considerably and identifying attack types of alerts in low slice of time.

Index Terms **—** Intrusion detection system, alert classification, ANFIS, false positive alert reduction

I. Introduction

An Intrusion Detection System (IDS) is a software program or hardware device which monitors computer system and/or network activities for malicious activities and produces alerts to security experts. In IDS there are three major problems namely generating many alerts, huge rate of false positive alerts and unknown attack types per generated alerts. Alert management methods are used to manage with these problems. One of the methods of alert management is alert reduction and alert classification [1].

This paper proposes a new method to manage the alerts using Adaptive Neuro-Fuzzy Inference System (ANFIS) [2]. Presented system can classify alerts and detect false positive alerts with a more accuracy than previous methods. This system can be used in active IDSs because it determines the attack type with a low slice of classification time. In the proposed alert management system results from ANFIS, a preprocessing and alert filtering process, is applied to the alerts during train and test phases.

The rest of the this paper is organized as follows: In section 2 related works are discussed, the suggested system for classifying the alerts is proposed in section 3, the experimental results are shown in section 4 and finally section 5 is a conclusion and future works.

II. Related works

One of the methods in IDS alert management techniques is clustering of alerts. The clustering method based on forming a generalized view of false alerts has been introduced by K. Julisch [3]. This method is based on discovering the roots leading false positive alerts. Julisch noticed that a small number of main implies 90% of alerts. By removing those root causes, the total number of alerts will come down to 82%.

Another clustering technique is used in Mirador project with expert systems by Cuppens. In this method the expert system algorithm decides whether alerts be merged into a cluster [4, 5]. Genetic algorithm used to clustering IDS alerts by Jianxin Wang, et al. [6]. Also two clustering

algorithms, based on GA and IGA are compared together [7]. Wang applied GA and IGA instead of Julisch's heuristic algorithm for "root cause" clustering.

Maheyzah Md Siraj compared EM, SOM, K-means and FCM clustering algorithms on Darpa 2000 data set [16]. They showed that Algorithm EM is the best for clustering, since the received alerts by algorithms are not filtered.

Azimi et. al. introduced another alert management system based on Self-Organizing Maps (SOM) [8]. The proposed system (SOM) [8] uses several operations such as alert filtering, alert preprocessing and cluster merging and could cluster and classify true positive and false positive alerts more accurate than other techniques. These operations improve the accuracy of the results. Our proposed alert management system is designed based on alert management system presented by Azimi et. al.

Seven genetic clustering algorithms named GA, GKA, IGA, FGKA, GFCMA, GPCMA and GFPCMA are used to cluster and classify true positive and false positive alerts, and then prioritized generated clusters with Fuzzy Inference System [9]. The proposed system presented in [9] is very similar to the system in [8] only by the difference in clustering and classification mechanisms.

In another work Learning Vector Quantization (LVQ) algorithm is used as a classifier in proposed system by Azimi and Bahbegi [17]. LVQ is a special type of Kohonen network [18] can classify test data set after training. It has some disadvantages; one of them is low accuracy rate in results and another is LVQ could not be able to identify attack type of alerts.

Here we use an alert management system similar to the system proposed by Azimi et. al. which uses ANFIS to classify generated alerts instead of SOM. The main advantages of the proposed system are obtaining the results with higher accuracy, identifying the attack type of alerts accurately and also reducing the number of false positive alerts considerably.

III. Alerts Classification System

The structure of proposed system is shown in Fig. 1. DARPA 98 dataset [10] and Snort tool [11] are used to generate alerts. Snort is an open source signature based IDS which gets DARPA 98 online traffic and then generates alert log files [8]. The security alerts log files generated by Snort tool are imported to the proposed system as its inputs. The units of the above system are introduced in the next sections.

A. Labeling Unit

Labeling unit [8] accepts generated alert from IDS and tcpdump.list files of DARPA 98 data set to label alert with a specific attack type. tcpdump.list files contain information about all traffics in DARPA 98 intrusion detection dataset. Label of alerts are used in training phases to train ANFIS and to evaluate the correctness results in the test phase. Fig. 2 shows labeling algorithm used in [8] and [9].

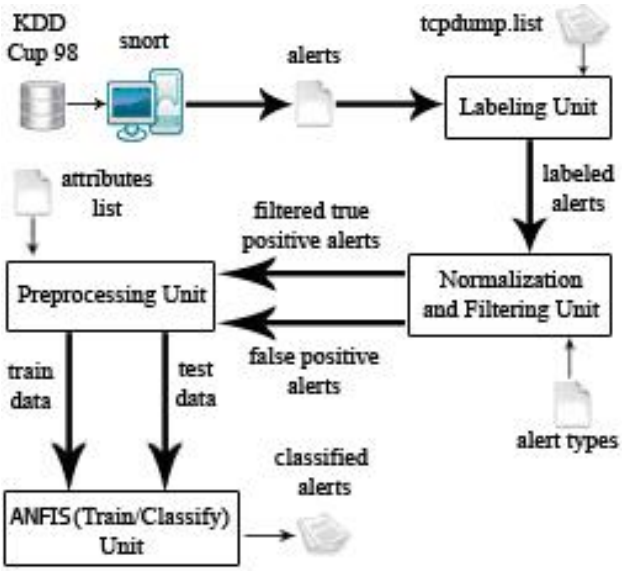

Figure 1. Alert management system

1. Input TCPDUMP list files.
2. Input alert log files.
3. Create an empty Attack List set.
4. Create an empty Alert List set.
5. For each row in TCPDUMP list files:
   5.1. If the row is a labeled attack then add the row to the Attack List set.
6. For each row in alert log files:
   6.1.Create key with the five attributes: source ip, destination ip, source port, destination port, ICMP code/type.
   6.2. If the key exists in the Attack List set then label the selected row with the type of found attack from Attack List set.
   Else
   Label the selected row with the False Positive attack type.
   6.3. Add the selected row to the Alert List set.
7. Return the AlertList set.

Figure 2. The algorithm of alerts labeling [8] and [9].

B. Normalization and Filtering Unit

As mentioned in [12], snort cannot detect some of attacks such as Portsweep and Smurf. It means that among the available attack type in Darpa 98 dataset, it can detect only eight cases with high accuracy [8, 9]. So this unit takes accepted attack names as input to filter labeled alert for generating accepted alert file. The output of this unit is filtered true positive and filtered false positive alerts.

Eight attributes are chosen [8, 9] in the normalization process. The chosen attributes are: Signature ID, Signature Rev, Source IP, Destination IP, Source Port, Destination Port, Datagram length and Protocol [13]. Azimi and Bahrbegi [8, 9] showed that filtering the similar alerts wouldn't remove two alerts with two different types of attack.

C. Preprocessing Unit

Values of attributes of alerts with string type in this unit are converted to the numerical values. Also the attributes that composed of several parts such as IP addresses are converted to the numerical values too. Range reduction is applied for the values of all attributes of alert [8, 9]. This unit accepts filtered true positive alerts, false positive alerts and attributes list as inputs and generates train data file and test data file. By using (1) and (2)

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

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

The string values are converted into the numerical values and by using the Improved Unit Range (IUR) formula (3) the attribute value ranges are reduced in to [0.1, 0.9].

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

D. Training And Classification Unit

In this unit we use Adaptive Neuro-Fuzzy Inference System (ANFIS). ANFIS is trained by train data and then classifies the test data. In the next section we describe the important concepts and definitions of ANFIS.

1) Adaptive Neuro-Fuzzy Inference System

The ANFIS architecture proposed by Jang [2] made up of adaptive networks. ANFIS has an edge over other hybrid architecture due to its mathematical framework devised to decompose the parameter set (of the adaptive network nodes). Such decomposition helps to implement a hybrid learning algorithm composed of the Least Squares Estimator and the Gradient Descent Method. ANFIS implements the Sugeno Fuzzy models. The advantage of ANFIS is that learning can be interpreted from neural and fuzzy systemsnpoint of view. Also such a system enables to view the solution of the problem in term of a linguistic function.

2) Training the ANFIS

As you can see in Fig. 1, train data set and test data set are used as inputs of this unit. The designed ANFIS has eight numerical input variables. Neural network section of ANFS in training phase accepts all of data vectors in train data set and generates IF-THEN rules of Fuzzy Inference System. The training of ANFIS is very time consuming. After training of ANFIS, it is used to classify vectors of test data set.

3) Test the ANFIS

Test data set is given to ANFIS in this unit. It is expected that all given data vectors from alerts with attack labels are resulted the corresponding output. Each data vector entering in this section is converted to fuzzy values, and then appropriate rule is selected from learned rules. After selecting of specific rule for an input data vector, ANFIS generates output value. Since there are some errors in ANFIS results, the output value is a floating point number which should be rounded and then mapped in to an attack label.

IV. Experimental Results

Matlab software is used to implement the system and Fuzzy toolbox is used to simulate ANFIS [14, 15].

ANFIS is a Sugeno type Fuzzy Inference System which is composed of 555 neurons, 2304 nonlinear parameters and 48 linear parameters. The extracted fuzzy rules are 256 rules. The number of input data vector attributes is 8, number of each of which has 2 membership functions. The initial step size of ANFIS training is 0.2. In this paper hybrid learning algorithm is used to train the system. Train data contains 10166 data vectors or 70% of total filtered alert data vectors. The false positive count in the training dataset is 4113. Test data set includes 30% of the data vectors of labeled alerts is; it means 2591 data vectors of true positive, and 1764 data vectors of false positive alerts. The reason of adding the false positive alerts to the test data set is that IDSs always produce this type of alerts in addition to the true positive alerts. The resulted ANFIS is shown in Fig. 3.

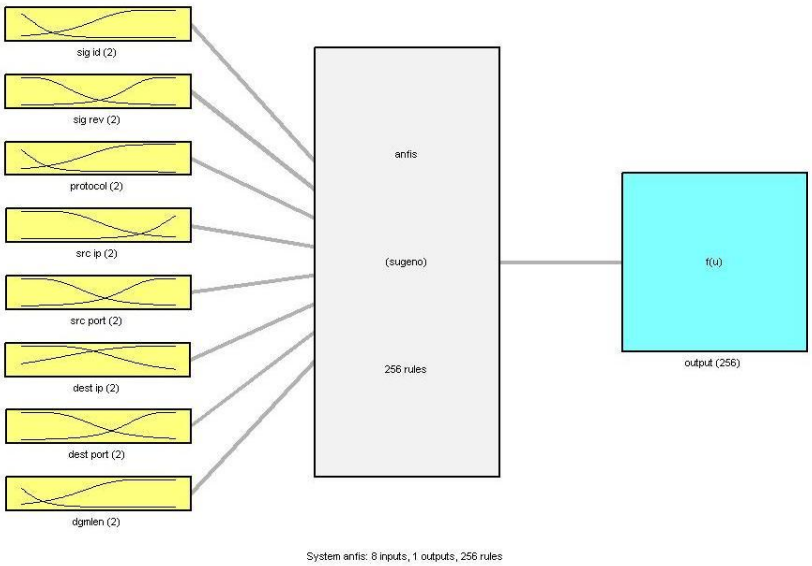

Figure 3. Generated ANFIS diagram

Membership functions for each input attributes are shown in Fig. 4. These functions are conducted and tuned by ANFIS automatically. Some of extracted fuzzy rules of ANFIS are shown in Fig. 5. These rules are used to make decision when an input vector is entered. One of these rules is selected when an input vector is accepted. Defuzzification operation of ANFIS converts the generated output values to the system of real values.

To evaluate the performance of algorithms, five measurements are introduced, which are Train phase Error Rate (TrainErR), Test phase Error Rate (TestErR), Classfication Error (ClaEr), Classfication Accuracy percent (ClaAR) (4), and Average Alert Classification Time (AACT) (5).

$$ClaAR=100-((ClaEr \div Total\ Number\ of\ Alerts\ Observed\ From\ Train\ Data)\times 100) \quad (4)$$

$$AACT=Total\ of\ Execution\ Time\ of\ Classifing\ Test\ Dataset \div Total\ Number\ of\ Alerts\ Observed \quad (5)$$

As you can see in table I, the value of TrainErR metric is 0.0631 that is very low. It shows the proposed system is trained and tuned accurately. Leading an improvement in the accuracy of test phase result. The TestErR is 0.0576 that can be ignored. If training phase is accurate then the results of test phase are accurate and acceptable too. The values of ClaE and ClaAR are 18 and 99.59 respectively that depend on classification error rate directly(Table I). It means that is the low rate error in train and test phases are resulted to produce more accurate in classification of alerts leading an increase in correctness of attack types of alerts. The value of AACT measurement is 0.00003 showing that the proposed system can be used in active IDS alert management systems which analyze alerts in addition to the alert produces by IDS concurrently. Table II shows the results of accuracy of proposed system in identifying attack type of each alert vector in test phase. As it can be seen in table II, the proposed system can identify all of attack types of alerts with high rate of accuracy except Phf attack type. An important point is accurate percent of false positive identification. That is the proposed system can reduce false positive alerts with 99.60 percent. Which shows to be a solution to an important problem of IDSs. Proposed alert management system reaches 100 percent for Back, Land, Dict and Nmap attack types. For attack types Pod, Imap and Rootkit accuracy percent values are 97.96, 33.34 and 28.57 respectively. By the little number of alerts with Phf attack type in training phase, system could not identify any alert of this attack type. That is ANFIS could not be trained for this type of attack.

Bahrbegi et. al. in [9] proposed a framework that uses genetic algorithm families to clustering and classification propose. As two works are similar we have to compare our results with their work. These results are shown in table III. For all metrics the proposed system has high value in contrast of all GA based techniques. As shown in table III, these algorithms could not be able to work actively because of the execution times are high. Although the proposed method earns high accuracy results per alert attack type.

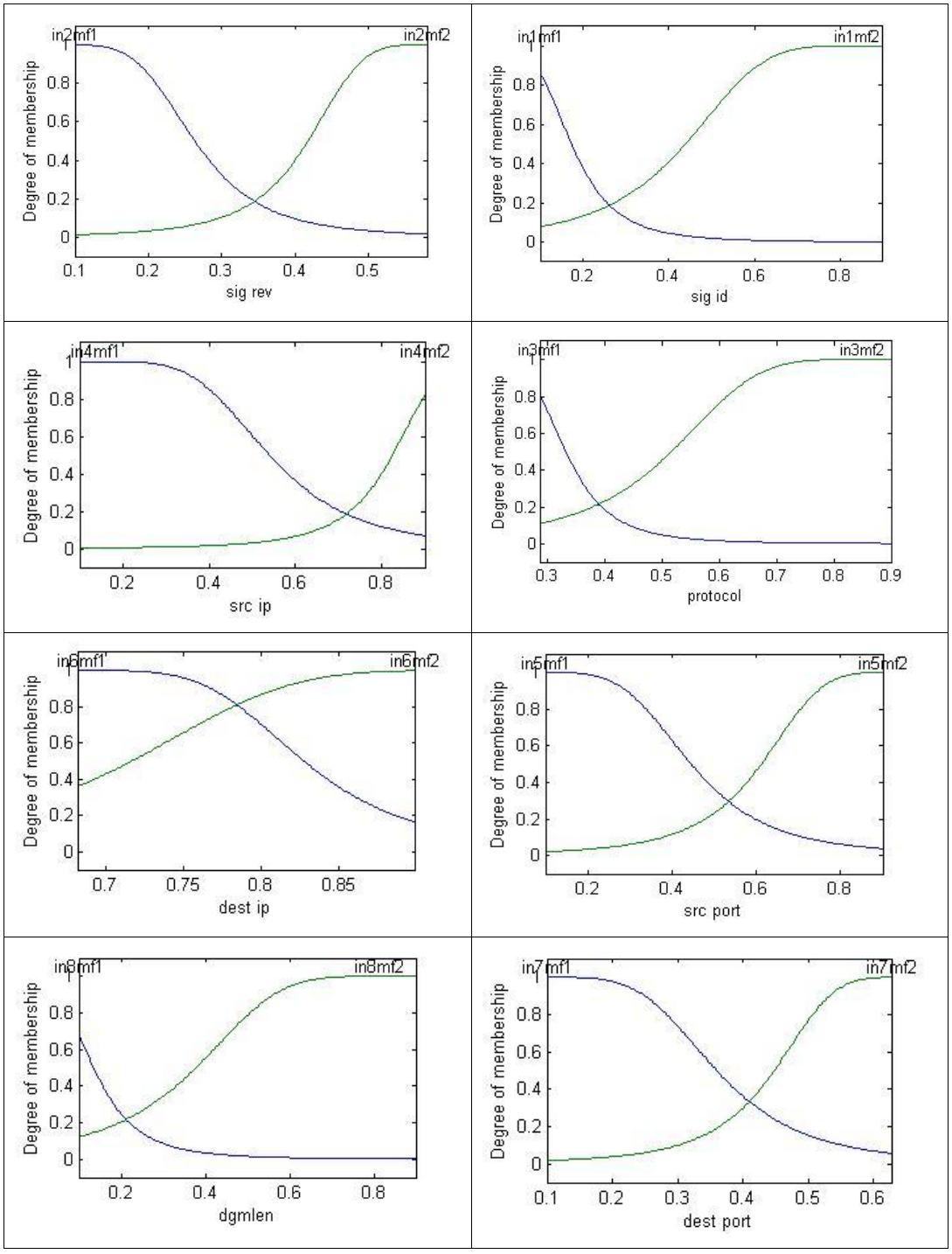

Figure 4. Conducted membership function for each input attributes in ANFIS

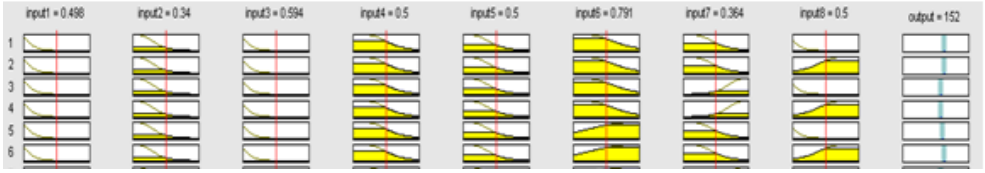

Figure 5. Extracted fuzzy rules

TABLE I. PROPOSED SYSTEM PERFORMANCE METRICS

TrainErR	TestErR	ClaE	ClaAR	AACT
0.0631	0.0576	18	99.59	0.00003

TABLE II. PROPOSED SYSTEM ACCURACY PERCENT FOR EACH ATTACK TYPE OF ALERTS

Back	Land	Pod	Phf	Rootkit	Imap	Dict	Nmap	False Positive
100	100	97.96	0	28.57	33.34	100	100	99.60

TABLE III. RESULTS OF PERFORMANCE METRICS FOR GA BASED ALGORITHMS

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

V. CONCLUSIONS

In this paper an ANFIS based system is presented which can classify the alerts with high accuracy and reduce number of false positive alerts considerably. Also the system is able to identify the attack types of the alerts more accurate.

The one of advantages of ANFIS is its online training capability. That is ANFIS can be trained when it is in testing or classification mode. In other words it is able to be trained when it accepts alerts for classifying. It gets feedback from output values and tunes IF-THEN rules. To implement this new idea a unit should be designed to calculate a metric which can explain the accuracy of ANFIS output values. So this metric can be used as input for the ANFIS to train it. Using this unit in the proposed system lets alert management system always can classify alerts with unknown attack types and separates false positive alerts from true positive alerts.

Alert correlation is one of the main techniques that used in alert management systems. ANFIS can be used to correlate generated alerts.

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] Jang J., “ANFIS: Adaptive-Network-Based Fuzzy Inference System”, IEEE Transactions on Systems, Man, and Cybernetics, Vol.: 23, Issue: 3, pp: 665-685, 1993.

[3] 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.

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

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

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

[8] 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.

[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] S Terry Brugger and Jedidiah 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] Fuzzy Toolbox, "Fuzzy Toolbox for Matlab", www.mathworks.com/products/fuzzy-logic/index.html, 2011.

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

[16] 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.

[17] Amir Azimi Alasti Ahrabi, Hadi Bahrbegi, Elnaz Safarzadeh, Mehdi Bahrbegi, “Using Learning Vector Quantization in Alert Management of

Intrusion Detection System", International Journal of Computer Science and Security (IJCSS), Vol.: 6, Issue: 2, unpublished, 2012.
[18] Kohonen, T, "Self-Organized Maps", Springer series in information. Science Berlin Heidelberg: 1997.

Zahra Atashbar Orang was born in 1984. She received B.S. degree in software engineering and M.S. degree in hardware engineering from University of Islamic Azad University, Tabriz Branch in 2006 and 2012 respectively.

Ezzat Moradpour was born in 1982. He received the B.S. degree in software engineering from Payamenour University in 2006 and M.S. degree in software engineering from Islamic Azad University in 2012. His research interest includes intrusion detection systems and artificial intelligence.

Amir Azimi Alasti Ahrabi. He was born in 1983. He received the B.S. and M.S. degrees in software engineering from University of Payamenour and Islamic Azad University, in 2007 and 2010 respectively. His research areas are information security, and artificial intelligence.

Ahmad Habibizad Navin. He was born in 1950. He received the B.S. degree in applied mathematics from Tabriz University, Tabriz, Iran, in 1999. He received the M.S. degree in computer architecture from University of Science and Research Branch, Islamic Azad University, Tehran, Iran, in 2003 and the Ph.D. in computer architecture from University of Science and Research Branch, Islamic Azad University, Tehran, Iran, in 2007. His research interest includes data-oriented approach, robotic, soft computing and probability and statistic.

Mir Kamal Mirnia. He was born in 1946. He received the B.S. degree in joint mathematics and physics from University of Mashhad, Iran, in 1967. He received the M.S. degree in pure mathematics from University of Tehran, Iran, in 1969. He received his second M.S. degree in numerical analysis and computing from University of Manchester, UK, in 1975, and the Ph.D. in applied mathematics from University of Andrews, UK, in 1979. He is currently a Professor in the Department of Computer Engineering. His research interest includes numerical analysis, soft computing, nonlinear programming and unconstrained optimization. He is member of Associate Fellow of the Institute of Mathematics and its Applications (AFIMA).

