Managing Intrusion Detection Alerts Using Support Vector Machines

Majid Ghonji Feshki *ghonji.majid@yahoo.com*
Department of computer science
Islamic Azad University, Qazvin Branch
Qazvin, Qazvin, Iran

Omid Sojoodi Shijani *o_sojoodi@qiau.ac.ir*
Department of computer science
Islamic Azad University, Qazvin Branch
Qazvin, Qazvin, Iran

Minoo Deljavan Anvary *Minoo.deljavan@yahoo.com*
IT Department School of e-Learning
Shiraz University
Shiraz, Fars, Iran

Abstract

In the computer network world Intrusion detection systems (IDS) are used to identify attacks against computer systems. They produce security alerts when an attack is done by an intruder. Since IDSs generate high amount of security alerts, analyzing them are time consuming and error prone. To solve this problem IDS alert management techniques are introduced. They manage generated alerts and handle true positive and false positive alerts. In this paper a new alert management system is presented. It uses support vector machine (SVM) as a core component of the system that classify generated alerts. The proposed algorithm achieves high accurate result in false positives reduction and identifying type of true positives. Because of low classification time per each alert, the system also could be used in active alert management systems.

Keywords: Intrusion Detection System, Alert Management, Support Vector Machine, Security Alert Classification, Reduction of False Positive Alerts, Classifying True Positive Alert Based on Their Attack types.

1. INTRODUCTION

An intrusion detection system (IDS) monitors network traffic and suspicious activity and then alerts the system or network administrator. In some cases the IDS may also responds to anomalous or malicious traffic by taking action such as blocking the user or source IP address from accessing the network [1]. Common problem with IDSs is producing many alerts each day that many of them are false positive alerts. Since amount of fake alerts is high so alerts of real attacks are hided among them. IDSs divided in to two categories, passive and active. 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 as active IDSs, work in online mode. Some of problems of IDS are: huge amount of generated alerts and high rate of false positive alerts among generated alerts. Also most alert management system has low speed, and has low accuracy in classification results. In this paper a new alert management system proposed that uses Support Vector Machine (SVM) as a classification engine [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 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

Clustering and classification of alerts is a technique of alert management systems. A method of clustering based on root causes is proposed by K. Julisch [3] 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 technique. In [4, 5] two genetic clustering algorithm based, named Genetic Algorithm (GA) and Immune based Genetic Algorithm used to manage IDS alerts. Their proposed methods are depended on underlying network information same as method presented by Julisch.

Three algorithms with dimension reduction techniques are used to cluster generated IDS alerts from DARPA 2000 dataset [6]in [4] and then compared results. The problems of that system are: row alert without preprocessing are entered to the algorithms and system is not tuned. Cuppens proposed another method as a part of MIRADOR project that uses expert system to make decision [7, 8].

Debar et al. [9] designed a system by placing them in situations aggregates alerts together. Situations are set of special alerts. Some attributes of alert are used to construct a situation.

In [10] Azimi et.al. is proposed a new system that manage alerts generated from DARPA 98 dataset with snort. Some algorithm such as alert filtering, alert preprocessing and cluster merging are used in that system. The main unit of the system is cluster/classify unit that uses Self-Organizing Maps (SOM) [11] to cluster and classify IDS alerts. Results of [10] show that SOM was able to cluster and classify true positive and false positive alerts more accurate than other techniques.

In another work, authors of previous article develop an alert management system [12] similar to [10]. In that work usage of seven genetic clustering algorithms named Genetic Algorithm (GA) [13], Genetic K-means Algorithm (GKA) [14], Improved Genetic Algorithm (IGA) [15], Fast Genetic K-means Algorithm (FGKA) [16], Genetic Fuzzy C-means Algorithm (GFCMA) [17], Genetic Possibilistic C-Means Algorithm (GPCMA) [12] and Genetic Fuzzy Possibilistic C-Means Algorithm (GFPCMA) [12] 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 [12].

Also they were develop another approach based on same alert management system in [10, 12] by replacing classification engine of previous work by Learning Vector Quantization (LVQ) [18, 19]. In that work LVQ technique is used in the classification unit. The experimental results shows that LVQ has can be a solution to alert management systems that used in active mode because of high classification speed. The accuracy of the suggested approach is acceptable [18].

In [20] a fast and efficient vulnerability based approach that addresses the above issues is proposed. It combines several known techniques in a comprehensive alert management framework in order to offer a novel solution. Their approach is effective and yields superior results in terms of improving the quality of alerts.

Alert Correlation is one of alert management techniques is used to find any relation between alerts and tries to find attack scenarios. In a new alert correlation technique named ONTIDS is

proposed. It is a context-aware and ontology-based alert correlation framework that uses ontologies to represent and store the alerts information, alerts context, vulnerability information, and the attack scenarios. ONTIDS employs simple ontology logic rules written in Semantic Query-enhance Web Rule Language (SQWRL) to correlate and filter out non-relevant alerts. they illustrate the potential usefulness and the flexibility of ONTIDS by employing its reference implementation on two separate case studies, inspired from the DARPA 2000 [21] evaluation datasets.

In this paper an alert management system based on system proposed by Azimi et.al. [10] is proposed that uses SVM as a tool to classify input alert vectors. It covers the problems of previous work such as low accuracy of results, inability to identify the types of generated alerts accurately. The system will be able to improve accuracy of results, to detect type of true positive alerts and also to reduce the number of false positive alerts.

3. USING SVM IN ALERT MANAMENT SYSTEM

In this section we investigate alert management framework introduced by [10, 12]. The suggested framework consists of several units. In this investigation we use Snort [11] IDS to generate alerts from DARPA 98 dataset [22]. Figure 1 shows the proposed framework. Snort is an open source, network based and signature based IDS. Snort gets tcpdump binary files of DARPA 98 dataset as input and produces security alerts. After producing alerts with snort; they are entered to labeling unit. The labeling unit and others units are investigated in next sections.

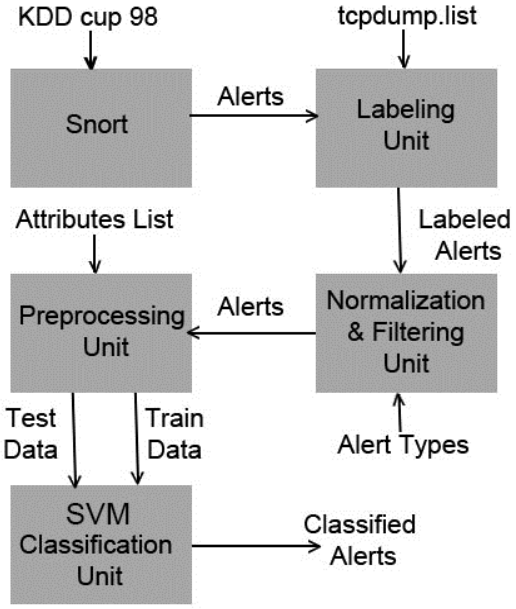

FIGURE 1: Proposed alert management system.

3.1 Labeling Unit

Labeling unit according to tcpdum.list files label generated security alerts. It means that this unit appends attack type of each alert as an attribute of processed alert to proper alert. This operation used in next unit to train and to specify correctness of generated results of the system. Figure 2 shows the labeling algorithm.

1. Input TCPDUMP list files.
2. Input alert log files.
3. Create an empty *AttackList* set.
4. Create an empty *AlertList* set.
5. For each row in TCPDUMP list files:
   5.1. If the row is a labeled attack then add the row to the *AttackList* 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 *AttackList* set then label the selected row with the type of found attack from *AttackList* set.
   Else
   Label the selected row with the False Positive attack type.
   6.3. Add the selected row to the *AlertList* set.
7. Return the *AlertList* set.

FIGURE 2: Alert labeling algorithm used in Labeling Unit [10, 12].

3.2 Normalization and Filtering Unit

According to [23] snort is unable to detect all attacks in DARPA 98 dataset then we use "alert types" file to specify accepted attack types in this investigation [10, 12]. In this unit alerts according to its labels are filtered. All of accepted attack types are entered to this unit and then this unit removes all of alerts that their labels are not in "alert types" file. Also this unit removes redundant alerts from set and keeps one of them. In this paper accepted alerts attack types are: Back, Land, Pod, Phf, Rootkit, Nmap, Imap, and Dict.

3.3 Preprocessing Unit

This unit transforms character form values of attributes of alert to numerical data to construct data vectors from alerts (1) and (2). The range of attribute values is reduced by this unit (3).

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

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

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

3.4 SVM Training and Classification Unit

In this unit SVM algorithm is used as a classifier. SVM is a technique to solve classification problems [2]. It maps pattern vectors to a high dimensional feature space to generate best separating hyperplane. SVM uses linear model to implement nonlinear models. When a linear model constructed in new space can represent nonlinear decision boundaries in original space. One of main characteristics of SVM is that it simultaneously minimizes the empirical classification error and maximizes the geometric margin [24]. There are many types of kernels that may be used in an SVM [25]. Some nonlinear kernel type is Polynomial and Gaussian (radial based Kernel). Equations Polynomial and Gaussian are respectively.

$$\left(coeff + x^T y\right)^d \tag{4}$$

$$\exp\left(-\left|x-y\right|^2 / \delta^2\right) \tag{5}$$

In this paper we use Polynomial and Gaussian kernel functions to evaluate SVM classifier.

4. EXPERIMENTAL RESULTS

To simulate the proposed system C#.net programming language, MATLAB and OSU SVM toolbox are used[26, 27]. The parameters of simulation are shown below.

In the investigation three type of kernel used for SVM named Linear, Polynomial and Radial Basis Function (RBF) kernels. Each of these kernels has own parameters for example Linear SVM depends on C and Epsilon parameters where Epsilon is used to termination threshold. The C parameter tells the SVM optimization how much you want to avoid misclassifying each training example. For large values of C, the optimization will choose a smaller-margin hyperplane if that hyperplane does a better job of getting all the training points classified correctly. Conversely, a very small value of C will cause the optimizer to look for a larger-margin separating hyperplane, even if that hyperplane misclassifies more points. For very tiny values of C, you should get misclassified examples, often even if your training data is linearly separable. Polynomial kernel depends of four parameters named Degree, Gamma, Coefficient and C. Also RBF kernel depends on two factor named Gama and C.

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.

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

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

According to tables 1 to 3 the classification error rate reduces by increasing the value of C. Results show that RBF kernel is best to classify IDS alerts. According to table 3 when Gama is 10 and C is 16 best results is achieved.

The best values of ClaAR are 99.82%. The value of AACT measurement is 0.0.00032 that shows the proposed system can be used in active IDS alert management systems that evaluate alerts beside alert production by IDS concurrently.

The tables show the results of accuracy of proposed system based on Linear, Non-linear and RBF kernels in identifying attack type of each alert vector in test phase. As it can be seen in table 3, the proposed system can identify all of attack types of alerts with high rate of accuracy when RBF kernel is used.

An important point is accuracy percent of false positive identification. That is the proposed system can reduce false positive alerts with 99.94 percent. Which shows to be a solution of an important problem of IDSs. Proposed alert management system reaches 100 percent for Back, Land, Dict and Nmap attack types. For attack types Phf, Imap and Rootkit accuracy percent values are 66.67, 66.67 and 42.86 respectively. The number of generated SVM is 211.

Bahrbegi et. al. in [12] 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 3. For all metrics the proposed system has high value in contrast of all GA based techniques. As shown in table 4, 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.

C	CAR	Time	SVM No	Back	Land	Pod	Phf	Rootkit	Imap	Dict	NMap	FPRR
1	99.01	0.000065	1029	99.22	0	97.96	0	0	0	100	100	99.15

TABLE 1: Extracted performance metric values from simulation with Linear Kernel.

C	CAR	Time	degree	SVM No	Back	Land	Pod	Phf	Rootkit	Imap	Dict	NMap	FPRR
1	99.01	0.000222	1	1029	99.22	0	97.96	0	0	0	100	100	99.16
1	99.61	0.000082	5	337	99.92	0	97.96	33.33	0	66.67	100	100	99.94
1	99.77	0.000054	10	238	100	100	97.96	66.67	14.29	66.67	100	100	99.94
0.5	99.74	0.00054	10	209	100	100	97.96	66.67	28.57	33.33	100	100	99.89
1	99.77	0.000047	15	209	100	100	97.96	66.67	28.57	66.67	100	100	99.89
1	99.74	0.000039	20	140	100	100	100	33.34	28.57	66.67	100	100	99.83

TABLE 2: Extracted performance metric values from simulation with Non-Linear Kernel.

C	CAR	Time	Gama	SVM No	Back	Land	Pod	Phf	Rootkit	Imap	Dict	NMap	FPRR
1	99.01	0.00017	0.5	972	99.22	0	97.96	0	0	0	100	100	99.15
1	99.33	0.000079	2	511	99.45	0	97.96	0	0	0	100	100	99.77
1	99.31	0.000122	3	443	99.45	0	97.96	0	0	0	100	100	99.72
1	99.63	0.000054	10	326	99.84	100	97.96	0	0	66.67	100	100	99.89
2	99.72	0.000061	10	269	99.92	100	97.96	33.34	42.86	66.67	100	100	99.83
4	99.79	0.000043	10	246	100	100	97.96	33.34	42.86	66.67	100	100	99.94
8	99.79	0.000032	10	224	100	100	97.96	33.34	42.86	66.67	100	100	99.94
16	99.82	0.000032	10	211	100	100	97.96	66.67	42.86	66.67	100	100	99.94
1	99.60	0.000061	15	369	99.84	100	97.96	0	0	66.67	100	100	99.83

TABLE 3: Extracted performance metric values from simulation with Non-Linear Kernel.

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 4: Results of performance metrics for GA-Based Algorithms [12].

5. CONCLUSION AND FUTURE WORKS

In this paper a SVM based system is presented which can classify the IDS 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 in little time slice.

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

6. REFERENCES

[1] Debar, H., M. Dacier, and A. Wespi, *Towards a taxonomy of intrusion-detection systems.* Computer Networks, 1999. **31**(8): p. 805-822.

[2] Cortes, C. and V. Vapnik, Support-vector networks. Machine learning, 1995. 20(3): p. 273-297.

[3] Julisch, K., Clustering intrusion detection alarms to support root cause analysis. ACM Transactions on Information and System Security (TISSEC), 2003. 6(4): p. 443-471.

[4] Maheyzah, S.Z., Intelligent alert clustering model for network intrusion analysis. Journal in Advances Soft Computing and Its Applications (IJSCA), 2009. 1(1): p. 33-48.

[5] Wang, J., H. Wang, and G. Zhao. A GA-based Solution to an NP-hard Problem of Clustering Security Events. 2006. IEEE.

[6] DARPA 2000 Intrusion Detection Evaluation Datasets, M.L. Lab., Editor. 2000.

[7] Cuppens, F. Managing alerts in a multi-intrusion detection environment. 2001.

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

[9] Debar, H. and A. Wespi. Aggregation and Correlation of Intrusion-Detection Alerts. in Proceedings of the 4th International Symposium on Recent Advances in Intrusion Detection. 2001.

[10] Ahrabi, A.A.A., et al., A New System for Clustering and Classification of Intrusion Detection System Alerts Using Self-Organizing Maps. International Journal of Computer Science and Security (IJCSS), 2011. 4(6): p. 589.

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

[12] Bahrbegi, H., et al. A new system to evaluate GA-based clustering algorithms in Intrusion Detection alert management system. 2010. IEEE.

[13] Krovi, R. Genetic algorithms for clustering: a preliminary investigation. 1992. IEEE.

[14] Krishna, K. and M. Narasimha Murty, Genetic K-means algorithm. Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, 1999. 29(3): p. 433-439.

[15] Fuyan, L., C. Chouyong, and L. Shaoyi. An improved genetic approach. 2005. IEEE.

[16] Lu, Y., et al. FGKA: a fast genetic K-means clustering algorithm. 2004. ACM.

[17] Di Nuovo, A.G., V. Catania, and M. Palesi. The hybrid genetic fuzzy C-means: a reasoned implementation. in International Conference on Fuzzy Systems. 2006. World Scientific and Engineering Academy and Society (WSEAS).

[18] Ahrabi, A.A.A., et al., Using Learning Vector Quantization in IDS Alert Management System. International Journal of Computer Science and Security (IJCSS), 2012. 6(2): p. 1-7.

[19] Kohonen, T., Learning vector quantization, in M.A. Arbib (ed.), The Handbook of Brain Theory and Beural Networks. 1995: MIT Press.

[20] Njogu, H.W., et al., A comprehensive vulnerability based alert management approach for large networks. Future Generation Computer Systems, 2013. 29(1): p. 27-45.

[21] DARPA 1998 Intrusion Detection Evaluation Datasets, M.L. Lab., Editor. 2000.

[22] DARPA 1998 Intrusion Detection Evaluation Datasets, M.L. Lab., Editor. 1998.

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

[24] Vapnik, V.N., The nature of statistical learning theory. 2000: Springer-Verlag New York Inc.

[25] Webb, A.R., Statistical pattern recognition. Second Edition ed. 2002, Malvern UK: Wiley.

[26] Matlab, www.mathworks.com/products/matlab/, Editor. 2009, Mathworks.

[27] Ma, J., Y. Zhao, and S. Ahalt, OSU SVM classifier matlab toolbox (ver 3.00). Pulsed Neural Networks, 2002.