Journal of Engineering and Applied Sciences 11 (7): 1493-1501, 2016
ISSN: 1816-949X

A Method Based on Data Mining for Detection of Intrusion in Distributed Databases

[1]Amin Mohajer, [2]Abbas Mirzaei Somarin, [2]Mohammadreza Yaghoobzadeh and
[3]Sajad Jahanbakhsh Gudakahriz
[1]Faculty of Information, Communications and Security Technology,
Malek Ashtar University of Technology, Tehran, Iran
[2]Department of Computer Engineering, Islamic Azad University, Ardabil Branch, Ardabil, Iran
[3]Department of Computer Engineering, Islamic Azad University, Germi Branch, Germi, IRan

Abstract: Increasing growth of distributed databases on one hand and attack to distributed networks on the other hand resulted in an important research area in the security of databases. The aim of this study is to introduce systems of intrusion detection based on data mining for detection of intrusion in distributed databases. Here, by proposing a model and another one in its heart, we intend to find the pattern of attacks. To test the efficiency of the model, among test dataset, each one containing 12000 records are selected randomly and the efficiency of the model is evaluated by this dataset. Results of tests illustrate the suitable performance of the proposed system. Criteria such as: CPE criterion and obtained detection rate reveal the suitable performance of the proposed system.

Key words: Data mining, intrusion detection, distributed databases, CPE, Iran

INTRODUCTION

Methods of intrusion detection are classified into two types: methods of abuse detection and abnormal behavior detection. The former acquires the ability of detecting intrusions using patterns recognized from the attack behaviors (Abdoli and Kahani, 2009; Noy and McGuinness, 2000; Tavallaee *et al*., 2009). For detection of the abnormal behavior, administrator of the network defines the normal state of the network traffic and detects abnormal behaviors by observing behaviors which don't follows the normal state (Toosi *et al*., 2006; Song *et al*., 2005). Recently, data miningtechniques are considered as systems for detection of intrusion.

For the sake of improving safety factor, fault tolerance and distribution of the traffic loadof the networkin distributed database, all or a part of the system of intrusion detection such as sensors, loggers and even analyzers are distributed in various regions of the network or many networks. In distributed networks such as (Yeung and Chow, 2002; Sabhnani and Serpen, 2003) or systems based on mobile agents analyzer components are distributed among various parts of the network. Moreover, in other distributed systems such as TaoPeng (Spafford and Zamboni, 2000) and or system introduced by Peng *et al.* (2007), sensing component and logger component are distributed in various parts of the network, respectively. However, in none of the systems, inference component is not distributed. From methods of selection of feature, artificial neural network (Scarfone and Mell, 2012; Andrew and Srinivas, 2003); vector support machine (Zhang *et al*., 2005), Bayesian networks (Xu and Wang 2005), method of finding the best features using cost among others can be noted all of which are used in many of the systems (Tsai *et al*., 2012; Mitrokotsa and Dimitrakakis, 2013; Guo *et al*., 2014; Li *et al*., 2004) intrusion is represented by an undirected graph and then this graph is made directed by means of concepts of centrality measurement (Dash *et al*., 2002).

Other methods are presented in (Memon and Henrik, 2006; Yen, 1971). These techniques can be applied in flexible fields of use. However, Yen (1971) is not appropriate for high volume logs (Yen, 1971) needs a more exact proportionality and is not an automatic method for detection of abnormalities. The aim of the data mining is to extract information relevant to various fields of the logs in distributed databases and is used as a way for analysis of systems and their actual behavior based on logs (Bezerra and Wainer, 2008a).

In the context of intrusion detection in distributed databases based on the process, researches are performed for detection of the abnormal sequence of processes Bezerra and Wainer (2008b) and Yen (1971) using a algorithm in process mining, two methods are presented for detection of abnormality. In these two methods, using a normal log, a reference model is obtained for the

Corresponding Author: Amin Mohajer, Faculty of Information, Communications and Security Technology,
Malek Ashtar University of Technology, Tehran, Iran

process. Then, new events which are recorded in a separate log are compared with the reference model and when an incompatibility is seen, an attack is considered. Owing to unpredictability of the behavior of users, this method has significant errors and is not suitable for flexible applications since there is not a normal log before execution. In four other researches, reference normal model is obtained during the process of abnormality detection and hence, they are suitable for flexible applications. In (Bezerra and Wainer, 2008a), three algorithms for detection of abnormality are presented and compared: sampling, threshold and iterative. Finally, sampling algorithm is presented as the best method. Due to changes and compatibility of the incremental process-mining, these methods have limitations and cannot be used for high volume logs. Jalali and Baraani (2010) and Lopez *et al*. (2013) a four-step method is presented which is based on a formal definition of an abnormal sequence according to proportionality level of the model and compatibility of the model. Using two aforesaid parameters, an appropriate model in accordance with logs is found and sequences which are not proportional to it are considered as attack. However, as stated in the paper, criterion of compatibility of the model is not exact and the method doesn't find the appropriate model automatically and needs manual exploration of the security agent. Bezerra and Wainer (2008b) using a genetic process-mining, a three-step method is presented for detection of abnormalities.

Therefore, the main purpose of this work is to detect intrusion in distributed databases. By means of the proposed model, required rules of determination of the features of various classes of attacks can be set. Obtained model is used as a classifier for the system of intrusion detection in distributed databases.

MATERIALS AND METHODS

Proposed model: In this study, architecture of the system of intrusion detection in process-based distributed databases based on data-mining is presented and the model designed for detection of intrusion is used for distributed databases. Proposed model is a multi-unit which has many units for detection of intrusion in fixed and mobile distributed databases and a central unit for detection of unit which collaborate for intrusion detection in an optimal way. For using and testing the designed model, it was attempted to simulate situations of an actual network for it. Therefore, a multi-unit environment was considered for achieving this goal. Using unit-based environment helps using capabilities and behaviors of the units for communication with each other and integration for better simulation of the network environment to test the proposed system. For design of an environment based on unit, Java programming language was used together with JADE library.

In present study, issues such as flexibility, scalability, independence from platform and reliability are taken into consideration for design of the proposed system. Overall architecture of the system is illustrated in Fig. 1 and in what follows, features of each of the components are explained and function of the system and the way units are integrated is explored.

Static unit (CAD): Role of CAD in the proposed architecture is monitoring or sensing and is located in components installed on all monitored hosts and components available on sensors of the network and in the following, role of this unit will be described.

Mobile Unit (MU): This unit is responsible for switching between CADs, receiving information corresponding to links, removal of additional information, merging of them and delivery of links to central unit.

Management Component (MC): This component includes the model of attacks in the form of a log file and performs inference on this model using central unit and detects the status of the links of mobile units and in case an intrusion is detected from a link, alarm is issued for thealarm unit.

Central Unit (CU): The main unit of proposed system which is equipped with the model of computer attacks designed in this study and in fact plays the ultimate role of detection of intrusion in distributed databases.

Alarm unit (AP): This unit sends alarms received from the management component to a console of the security manager of the network.

Mobile unit dispatchers (MP): It plays the main role of sending mobile units for data collection to hosts having suspicious alarms previously.

Static Units Message board (SMU): Role of this component is to exchange information between static units and central one and is in fact a part of the management component.

Mobile Unit Message board (MMU): Role of this component in the proposed system is to exchange information between all SAs and all monitored hosts as well as available sensors of the network.

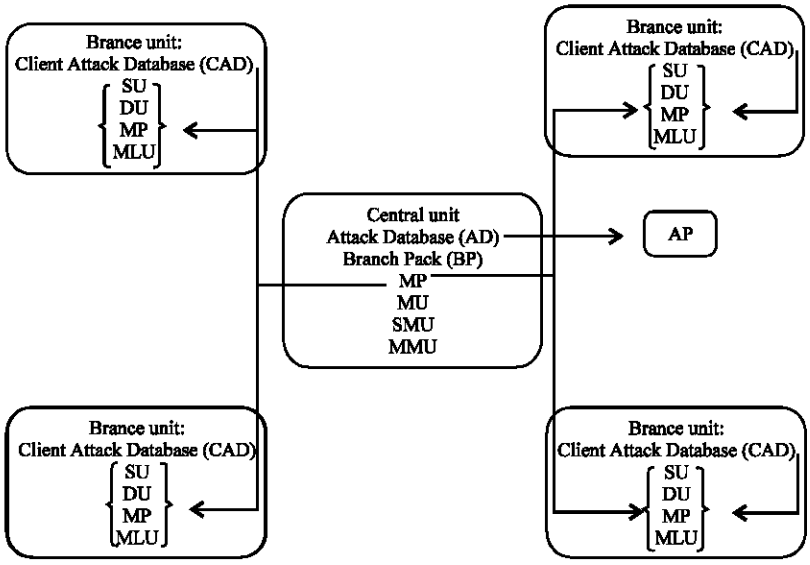

Fig. 1: proposed architecture

Data Unit (DU): Like message board, this unit is a part of the component installed on all monitored hosts and sensors and helps detection of a suspicious link and improves the performance of the system.

Exchanges between units and detection of intrusion in distributed databases: As stated earlier, in proposed architecture, CAD plays a pivotal role as a sensor and a CAD is embedded in all components installed on monitored hosts and components available on sensors of the network. By observing logs of hosts including system, application logs, calls and system status, components installed on monitored hosts and try to extract features based on the host of a link. In each part of the network, a network sensor component is available which includes a CAD which is responsible for extraction of features based on network of a link (traffic and inherent features of a link) based on content of the package passing through the network.

CAD units first extract these features for a link and then, compare them with their data model and if compatible, the link is considered as suspicious and CAD sends an alarm message to MP. This message contains the ID of the link, address of the hos and overall suspicious features of the network and it serves as an informant for the MP to continue the process of detection. To prevent bombardment of the management component by all features of a link, only required and necessary features must be delivered to this component so that it can act more rapidly and efficiently. Sometimes, it may be necessary to have only 2-3 feature of all 41 features of the link and by sending all 41 features to management component, a heavy overhead is imposed to the component. To achieve this goal, central unit first gets messages received from the SMU and looks for semantic similarities between the model and features available in these messages. After finding most of the similarities, set of required features corresponding to the state of the link is given to the MMU together with link ID and host address. MP in turn sends a mobile unit to the corresponding host according to the information available in MMU. This MU extracts all required features based on hosts and features required based on the network from CADs available in monitored hosts and network sensors and after accumulation of the features and making required link with demanded features by management component, link is delivered to the management component. In this stage, management component looks for semantic similarities between this link and model of attacks and after finding most of the similarity, type of the link is determined based on this similarity and if it is an intrusion, appropriate alarm is sent to the alarm unit. AP in turn, sends alarms received from central unit to the console which is monitored by network security manager. In addition, AP is responsible for displaying similar alarms in a certain time interval and stores information of these alarms for probable analyses in the future.

Analysis of intrusion in distributed databases: Distributed databases are comprised on nodes of a network and these nodes are connected to each other. Intrusion is an instance of such connections in which

network nodes send messages to each other. In this type of networks, there is usually a connection pattern between nodes to achieve goals. The aim of the analysis of these networks is to find the most important nodes of the network and the way network nodes are connected to each other. Similarly, in this paper, we intend to find the most important features and the method of connection of nodes. Consequently, for analysis of the relationship between features, methods presented for analysis of the distributed databases can be utilized. For this end, it is necessary to find the model of the connections graph and the way this model is created will be explained below.

For the analysis of the network, a method is used so that by means of the concept of dependence centrality, directed graph will be transformed into a tree. Using tree, important nodes of the network and their followers can easily be differentiated so that the intrusion can be created. The new idea of measuring dependence centrality is very useful for making a tree since this idea shows nodes which are dependent upon certain nodes. By means of measuring the efficiency of the network, effect of removal of each of the nodes of the graph can be shown well. The more the efficiency of the node is reduced by removal of a node, the more the importance of that node will be.

Evaluation of the efficiency of a network is represented in the formula of the centrality of the efficiency. In the formula, the value of dist (i, j) represents the shortest distance between two nodes i and j. centrality of the role index reveals the role of node in network. Role index is calculated using centrality of the network efficiency. If the centrality of IR is sketched on coordinate axis, points above the x-axis represent the important nodes of the network and points below the x-axis represent the less important nodes of the network. Method of calculation of the RI is shown in formula of the RI. In this formula, node (G). Efficiency (G-i) represents the status of network after deactivation of node i. Using centralities introduced in this section, graph of features can be analyzed and the tree model of the importance of features can be obtained.

Proposed algorithm for detection of intrusion: In proposed algorithm, it is tried to use the analysis of the distributed databases to find the set of optimal features. The difficulty in the model of data of intrusion detection systems is that in these systems, no information is exchanged between features and hence, it is necessary to transform data model into a graph model and the method is explained as.

Features graph for importance of nodes: To create the model of features graph, matrix of features is introduced: columns of the features matrix represent the features and rows show the classes of attacks in various methods of classification. This matrix has f columns and n×m rows so that f is the number of features available in dataset, n is the number of classification methods and m is the number of attack types.

Cell (i, j) is the accuracy of the class i if the feature j is not present. In addition to accuracy, another variable referred to as below threshold is located in matrix cells. This variable has either zero or one value. If removal of a feature from the set of features reduces the accuracy of the class to below determined threshold, value of the variable is set as unity. This value states that feature j selected for class i is of great importance for it since its removal reduces the accuracy of the class significantly.

To find the matrix of features, different classification algorithms are implemented on dataset per elimination of each feature. After implementation of each classification method, accuracy of the classification corresponding to each type of attack is recorded in matrix of features using disturbance matrix which shows the accuracy of each class.

The accuracy below the threshold means the importance of the removed feature. Therefore, its below-threshold value will be unity. After calculation of the matrix of features, graph model is created using formula of calculation of weight and direction of the lines of the graph.

Formula of weight and direction of the lines in graph finds the weight and direction of the graph from feature i to feature j using conditional probability. The C stands for all classes whose below-threshold variable in features matrix for a selected feature is one. Unity value of the below-threshold variable reveals the importance of the feature. Similarly, value of class (i) represents the number of classes which is one for feature i and class (i, j) is the number of classes which are unity for features i and j finally, according to the definition of the conditional probability, direction of the line between i and j is obtained using weight and direction of line in the graph. Output of this stage is the graph matrix which has f columns and f rows and the value of cell (i, j) yields the eright of the directed line from feature i to feature j. If only rows corresponding to a certain attack in the matrix of features is used, model of graph corresponding to each type o attack will be obtained as well and the obtained model is used for extraction of the set of optimal features corresponding to each type of attack. In features graph, presence of lines shows the similarity between importance of two features.

Features tree for importance of nodes: After creation and analysis of the graph model, graph model is converted to the tree. Before creation of the tree and to reduce the time required for calculation of the tree, features whose

Table 1: Symbols

Symbols	Description
Node. Center (m, n) = Sum (Dist(m,n)) /All_Dist+Node. essential ()	Dependence centrality
Node. efficiency (G) = Sum (efficiency (I, j)/ N*(N-1))	Efficiency centrality
Node. RI (G) = Node (G). efficiency- Node (G). efficiency (G- i)	Role Index (RI)
Arc ($Feature_i$, $Feature_j$) = Class (I,j)/Class (I)	Line weight and direction in graph
Cost = (Feature. No+1) * Sum (Class)	Time cost
Mem = N*M*F *Mem. size ()	In use memory
Mem = N*M*F *Mem. size ()-Sum (class)	Runtime memory

connection line has unity weight are removed from the network of features and after completing the model, they are added again to the model. The reason of removal of these features in this stage is that they are similar to each other with respect to weight and all of their measured centralities will be the same. For creation of the tree model, following rules are used:

- In graph of features, if there is a line from feature i to feature j, feature i will be selected as the parent of feature j
- No node can have two parents. Therefore, under circumstances in which there is line from two nodes i and k to j, i and k compete with each other. In this case, dependence centrality is used. If centrality of dependence of feature i to I is more than that of k to j, i is selected as the parent of I and the line between k and i is removed

In tree model, features which are in highest level are the most important ones and features in lowest level are the least important and or irrelevant ones (Table 1).

RESULTS AND DISCUSSION

Runtime of the algorithm includes two parts; calculation of the features matrix and calculation of the tree. For creation of the matrix of features, stage of elimination of features is repeated for each of the classification methods. Stage of elimination of features and then, implementation of the classification method are shown in the time cost formula o the classification method. In the formula of time cost, f is the number of features and C is the time cost of the classification method. For calculation of the centralities, time cost is equal to f^2. Among these centralities, only calculation of the centralities of the dependence will yieldk f^5 for finding k-shortest distances between two nodes according to Tao and Christopher. Size of in-use memory of this method for storing the matrix of features is given in the formula of in-use memory. In this relationship, n is the number of classification methods, f is the number of features and mem.size () is the size of the double type number. Size of the memory required for implementation of the algorithm is in accordance with the formula of the algorithm memory. In formula, class is the size of in-use memory and the type of classification.

Centrality of the characteristic values and role index are summarized in Table 2 and 3. According to the concept of centrality of the characteristic value, nodes with highest value are the most important characteristics. In Table 3, nodes with positive values represent the important nodes and nodes with negative values are soldiers and or less important ones. In two tables, characteristics values and role index, features with the same importance have the same values. After calculation of the centralities, using the method explained in previous section, tree model of features is obtained. In obtained tree, three features, 3, 22 and 23 are in the first level. These three features are the most important features. Since feature 2 is similar to 33, 35 and 40 with respect to the importance, six features can be taken as the most important features introduced by the proposed method.

After calculation of the tree and finding the set of optimal features, it is necessary to compare the obtained optimal set with optimal set of the other methods. For this end, set of optimal features of various methods are given using j48 classification method. Then, this method classifies the overall set of data selected from KDD99 in presence of set of optimal features. Finally, accuracy of the classifications is compared. First, set of optimal features corresponding to various methods are presented. Rough-PSO can find the set of optimal features of each attack. However, cost of implementation of each feature is from index type. Set of overall optimal features includes features 2, 4, 24, 27, 34 and 35. Proposed method is able to find the set of optimal features corresponding to any other type of attack. Therefore, it outperforms Rough-PSO method Table 4 and 5.

As can be seen, number of nodes of the set of optimal features introduced by the proposed method is equal to the number of nodes introduced by other methods.

To test the efficiency of the system, from the dataset of the NSL-KDD test, ten test data sample including 12000 records are selected randomly and the efficiency of the system is evaluated by means of them. Results of the classification of the test dataset based on defined criteria and each of the test versions of the proposed system are depicted in Fig. 2. In this table, considerable results for versions of testing using various datasets are presented which reveal the effectiveness of the proposed system and its suitable efficiency.

Table 2: Centrality of the characteristic value

No.	Centrality	No.	Centrality	No.	Centrality	No.	Centrality
1	20	11	0	21	0	31	0
2	23	12	0	22	7	32	3
3	5	13	0	23	2	33	22
4	4	14	0	24	20	34	20
5	0	15	0	25	0	35	22
6	19	16	0	26	0	36	5
7	0	17	0	27	12	37	12
8	3	18	0	28	0	38	25
9	0	19	0	29	25	39	3
10	19	20	0	30	0	40	16

Table 3: Role index

Nodes	Role	Nodes	Role	Nodes	Role	Nodes	Role
1	5	11	-3	21	0	31	-3
2	3	12	-3	22	7	32	-1
3	3	13	2	23	2	33	3
4	3	14	-3	24	20	34	5
5	1	15	-3	25	0	35	3
6	5	16	-3	26	0	36	2
7	-3	17	-3	27	12	37	5
8	1	18	-3	28	0	38	4
9	-3	19	-3	29	25	39	1
10	5	20	-3	30	0	40	0

Table 4: Set of features

Methods	----------Set of features----------
Bayesian	26, 25, 24, 23, 22, 17, 14, 12, 11, 8, 7, 5, 3, 2, 1, 32, 30
CART	35, 33, 32, 31, 28, 25, 24, 23, 12, 6, 5, 3
Normal	40, 37, 36, 35, 33, 32, 31, 12
Checkout attack	40, 36, 34, 23, 3, 2
DoS attack	40, 38, 34, 33, 29, 24, 10, 5
Attack of user to the	22, 17, 14, 6, 4, 3
Remote attack	36, 33, 23, 10, 4, 3
SVDF	33, 24, 23, 5, 4, 2
Linear Genetic Program (LGP)	35, 31, 27, 12, 5, 3

Table 5: Optimal features

Methods	----------Set of features----------
Normal	40, 39, 38, 36, 35, 34, 33, 32, 29, 10, 6, 4, 2, 1
Checkout attack	40, 38, 37, 36, 34, 29, 24, 13, 10, 6, 3, 1
DoS attack	36, 27, 24, 22, 13, 10, 6
Remote attack	37, 36, 34, 27, 13, 5, 4, 3, 1

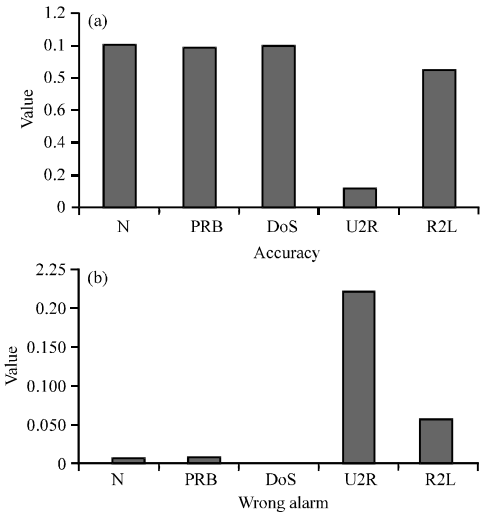

Fig. 2: a, b) Accuracy and error messages of the system of intrusion detection in proposed distributed

In what follows, for better evaluation of the above approach, proposed system is compared to some methods of machine learning which yielded their test results on KDD data. Figure 3 compared the efficiency of the methods. In some attack classes, proposed method has better efficiency compared to other methods and CPE value as much as 0.1027 reveals the ability of the system for detection of intrusion in distributed databases. From the results, it can be inferred that the proposed system has an acceptable performance for detection of intrusion in distributed databases of the network and it provides acceptable detection rate and false alarms as well.

Of course, results of some of rows of the table can be unfair. For instance, Abdoli and Kahani (2009) only the ability to detect DoS attacks is studied and the system is only able to differentiate DoS and non-DoS attack types and gives no information about non-DoS behaviors. This is not to say that since the proposed method presents a four-class classification for attacks, it is a better method but we notice that in this method when a record is

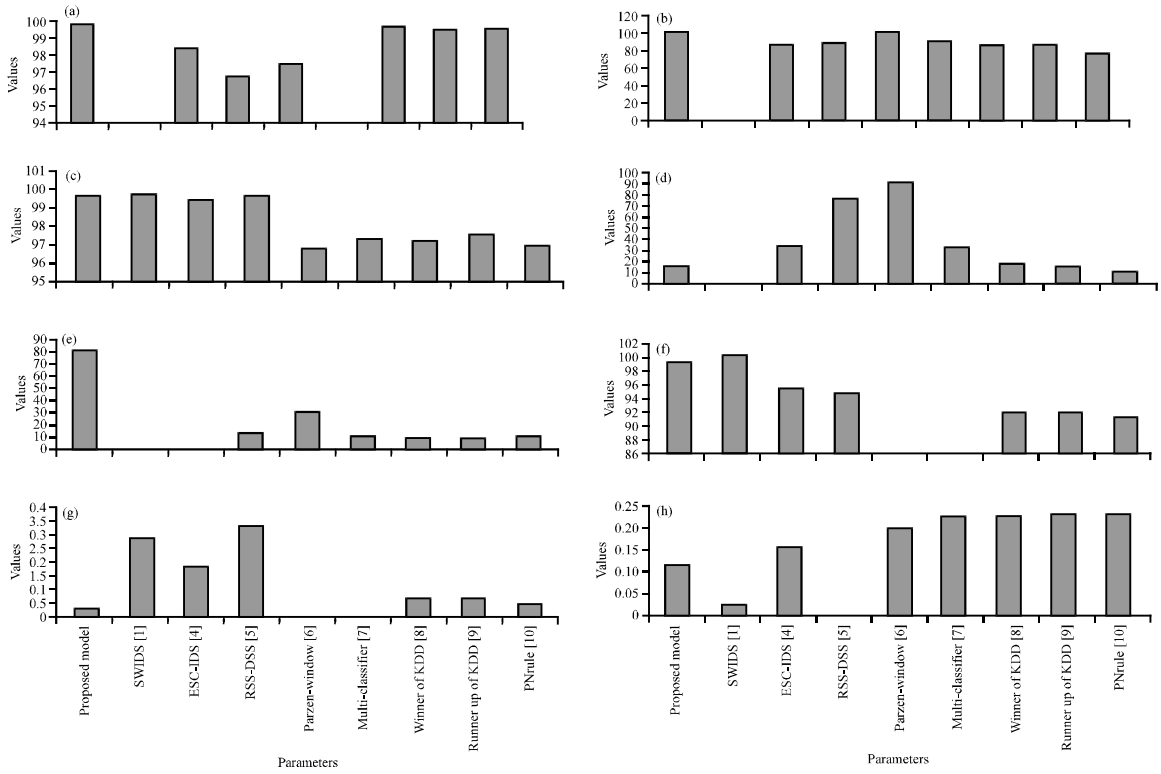

Fig. 3: Comparison of various systems of intrusion detection in distributed databases databases. a) N, b) Prb, c) DoS, d) U2R, e) R2L, f) DTR, g) FA and h) CPE

considered as non-DoS, regardless of its class, it is classified as a pattern classified in that class. However, in our proposed method, despite of detection as non-DoS attack, some of the patterns may be classified wrongly. In other words, in above method, detection of a non-DoS attack means right classification while in methods like one proposed in this study, detection of the attack is not necessarily the right classification and it is possible to classify the attack wrongly. Higher rate of attack detection in our method proves this claim which shows that our method performs well in detection of attacks while it may have errors in classification.

CONCLUSION

In this study, a method of detection of intrusion in distributed databases based on data mining is presented which can be used for detection of the conditions resulted from computer attacks in the system. Each report about the status of the network to the central unit is investigated in the model of computer attacks and result will be sent to the alarm unit. In proposed system, scalability of the system is taken into account seriously and it is necessary that the computational overhead of the central unit is in its minimum level. To achieve this goal, bombardment of the central unit by all links of the network with all 41 features must be avoided and a monitoring method must be applied. That is, instead of sending all links of the network together with all of the features by static unit to the central unit, only suspicious links with features asked by the central unit must be sent to it. As a result, static units are equipped with a data model which plays a vital role in this regard and causes that some of the links are considered as normal using the available data model and static units avoid sending their features to the central unit. This will result in reduction of integration volume and improvement of the efficiency of the central unit. In this paper, NSL-KDD model is used for production of mined data, studying the model of computer attacks and testing the proposed model. Results of tests illustrate the suitable performance of the proposed system. Criteria such as CPE and rate of detection denote the suitable performance of the proposed system. Downfall of the proposed system is in detection of the U2R class which can affect the performance of the system. Of course, it is expected the reason of which is limited number of training samples of the attack class. However, the important issue is the high rate of detection of system which is one of the strengths of the system.

This research is a new step toward data mining concepts and particularly model extracted from data mining for improvement of the systems of intrusion detection in distributed systems. It is possible to extend this work from other aspects. The first step is to prioritize, develop and improve the model of computer attacks so that new and unknown intrusions can be detected easily based on similarities found between values of a feature and using various methods of the semantic similarity.

On the other hand, data used in static units and or sensors are so weak and by means of strengthening of such data model, rate of detection of suspicious cases can be improved. This will result in reduction of the volume of data exchanges between static unit and central one as well as improvement of the scalability of the system. Other ways for continuing research in this context include development of a model so that it can extend the collaboration between units. Moreover, studying the inference over the model so that the integration of the available units in a multi-unit environment can be increased for detection of the distributed attacks can be useful.

REFERENCES

Abdoli, F. and M. Kahani, 2009. Ontology-based distributed intrusion detection system. Proceedings of the 14th International Conference on CSI Computer (CSICC), October 2021, 2009, IEEE, New York, USA., ISBN:978-1-4244-4261-4, pp: 65-70.

Andrew, H.S. and M. Srinivas, 2003. Identifying important features for intrusion detection using support vector machines and neural networks. Proceedings of the 2003 Symposium on Applications and Internet, January 27-31, 2003, IEEE Xplore, London, pp: 209-216.

Bezerra, F. and J. Wainer, 2008a. Anomaly detection algorithms in business process logs. Anomaly detection algorithms in business process logs. June 12-16, 2008, Scite Press, Barcelona, Spain, ISBN: 978-989-8111-37-1, pp: 11-18.

Bezerra, F. and J. Wainer, 2008b. Anomaly detection algorithms in logs of process aware systems. Proceedings of the 2008 ACM Symposium on Applied Computing, March 16-20, 2008, ACM, Fortaleza, Brazil, ISBN:978-1-59593-753-7, pp: 951-952.

Dash, M., K. Choi, P. Scheuermann and H. Liu, 2002. Feature selection for clustering-a filter solution. Proceedings of the 2002 IEEE International Conference on Data Mining (ICDM), December 9-12, 2002, IEEE, New York, USA., ISBN:0-7695-1754-4, pp: 115-122.

Guo, C., Y. Zhou, Y. Ping, Z. Zhang and G. Liu *et al*., 2014. A distance sum-based hybrid method for intrusion detection. Appl. Intell., 40: 178-188.

Jalali, H. and A. Baraani, 2010. Genetic-based anomaly detection in logs of process aware systems. World Acad. Sci. Eng. Technol., 64: 304-309.

Li, J., G.Y. Zhang and G.C. Gu, 2004. The research and implementation of intelligent intrusion detection system based on artificial neural network. Proceedings of the 2004 International Conference on Machine Learning and Cybernetics, August 26-29, 2004, IEEE, New York, USA., ISBN:0-7803-8403-2, pp: 3178-3182.

Lopez, V., A. Fernandez, S. Garcia, V. Palade and F. Herrera, 2013. An insight into classification with imbalanced data: Empirical results and current trends on using data intrinsic characteristics. Inf. Sci., 250: 113-141.

Mitrokotsa, A. and C. Dimitrakakis, 2013. Intrusion detection in MANET using classification algorithms: The effects of cost and model selection. Ad Hoc Networks, 11: 226-237.

Noy, N.F. and D.L. McGuinness, 2000. Ontology development 101: A guide to creating your first ontology. Stanford Knowledge Systems Laboratory. http://www-ksl.stanford.edu/people/dlm/papers/ontology-tutorial-noy-mcguinness-abstract.html.

Peng, T., C. Leckie and K. Ramamohanarao, 2007. Information sharing for distributed intrusion detection systems. J. Network Comput. Appl., 30: 877-899.

Sabhnani, M. and G. Serpen, 2003. Application of machine learning algorithms to KDD intrusion detection dataset within misuse detection context. Proceedings of the International Conference on Machine Learning: Models, Technologies and Applications, June 2003, Las Vegas, NV., pp: 209-215.

Scarfone, K. and P. Mell, 2012. Guide to Intrusion Detection and Prevention Systems (IDPS). Special Publication 800-94, National Institute of Standards and Technology, Technology Administration, U.S. Department Commerce, Gaithersburg, MD., USA., February 2007.

Song, D., M.I. Heywood and Z.A.N. Heywood, 2005. Training genetic programming on half a million patterns: An example from anomaly detection. IEEE. Trans. Evol. Comput., 9: 225-239.

Spafford, E.H. and D. Zamboni, 2000. Intrusion detection using autonomous agents. Comput. Networks, 34: 547-570.

Tavallaee, M., E. Bagheri, W. Lu and A.A. Ghorbani, 2009. A detailed analysis of the KDD CUP 99 data set. Proceedings of the 2nd IEEE Symposium on Computational Intelligence for Security and Defence Applications, July 8-10 2009, Ottawa, Canada.

Toosi, N.A., M. Kahani and R. Monsefi, 2006. Network intrusion detection based on neuro-fuzzy classification. Proceedings of the International Conference on Computing and Informatics, June 6-8, 2006, Kuala Lumpur, pp: 1-5.

Tsai, C.F., J.H. Tsai and J.S. Chou, 2012. Centroid-based nearest neighbor feature representation for e-government intrusion detection. Proceedings of the Conference on World Telecommunications Congress (WTC), March 5-6, 2012, IEEE, Miyazaki, Japan, ISBN:978-1-4577-1459-7, pp: 1-6.

Xu, X. and X. Wang, 2005. An adaptive network intrusion detection method based on PCA and support vector machines. Proceedings of the International Conference on Advanced Data Mining and Applications, July 22-24, 2005, Springer, Berlin, Germany, ISBN:978-3-540-27894-8, pp: 696-703.

Yen, J.Y., 1971. Finding the k shortest loopless paths in a network. Manage. Sci., 17: 712-716.

Yeung, D.Y. and C. Chow, 2002. Parzen-window network intrusion detectors. Proceedings of the 16th International Conference on Pattern Recognition, August 11-15, 2002, IEEE, New York, USA., ISBN:0-7695-1695-X, pp: 385-388.

Zhang, C., J. Jiang and M. Kamel, 2005. Intrusion detection using hierarchical neural networks. Pattern Recognit. Lett., 26: 779-791.