Multilayer perceptrons networks for an Intelligent Adaptive intrusion detection system

Aida O. Ali
Dep. of Computer and system engineering
Faculty of engineering
Mansoura University
Al Mansoura, Egypt

Ahmed saleh
Dep. of Computer and system engineering
Faculty of engineering
Mansoura University
Al Mansoura, Egypt

Tamer Ramdan
Dep. of Computer and system engineering
Misr Engineering and Technology Institute
Al Mansoura, Egypt

Abstrac
Intrusion Detection Systems (IDSs) provide an important layer of security for computer systems and networks, and are becoming more and more necessary as reliance on Internet services increases and systems with sensitive data are more commonly open to Internet access. An IDS's responsibility is to detect suspicious or unacceptable system and network activity and to alert a systems administrator to this activity. We need to use the classification algorithms to discriminate between normal and different types of attacks. The performance of nine artificial neural networks (ANNs) based classifiers was evaluated, based on a selected set of features. The results showed that; the Multilayer perceptrons (MLPS) based classifier provides the best results; about 99.63% true positive attacks are detected using this classifier.

Keywords
Component, Intrusion detection syste, artificial neural network, Multilayer perceptrons

I. INTRODUCTION

Intrusion detection is the process of monitoring the events occurring in a computer system or network and analyzing them for signs of intrusions defined as attempts to compromise the confidentiality, integrity, availability, or to bypass the security mechanisms of a computer or a network. Intrusion are caused by attackers accessing the system for the internet, authorized users of the systems who attempt to gain additional privileges for which they are not authorized, and authorized users who misuse the privileges given to them.

Intrusion detection systems (IDSs) are software or hardware products that automate this monitoring and analysis process.

Intrusion detection allows organizations to protect their systems from threats that come with increasing network connectivity and reliance on information systems.

Given the level and nature of modern network security threats, the question for security professionals should not be whether to use intrusion detection, but which intrusion detection features and capabilities to use.

IDSs have gained acceptance as a necessary addition to every organization's security infrastructure despite the documented contributions intrusion detection technologies make to system security, in many organizations one must still justify the acquisition of IDSs.

We may use IDSs to prevent problem behaviors by increasing the perceived risk of discovery of those who would attack or abuse the system.

The paper is structured as follows. Section 2 is a background section. In Section 3 we discuss related work. Section 4 introduces our proposed framework. Section 5 introduces the suggested algorithm of neural network. The paper is ended with a conclusion.

II. BACKGROUND

The timely and accurate detection of computer and network system intrusions has always been an important goal for system administrators and information security researchers. The individual creativity of attackers, the wide range of computer hardware and operating systems, and the ever-changing nature of the overall threat to target systems have contributed to the difficulty in effectively identifying intrusions. While the complexities of host computers already made intrusion detection a difficult endeavor, the increasing prevalence of distributed network-based systems and insecure networks such as the Internet has greatly increased the need for intrusion detection.

There are two general categories of attacks which intrusion detection technologies attempt to identify - anomaly detection and misuse detection. Anomaly detection identifies activities that vary from established patterns for users, or groups of users. Anomaly detection typically involves the creation of knowledge bases that contain the profiles of the monitored activities.

The second general approach to intrusion detection is misuse detection. This technique involves the comparison of a user's activities with the known behaviors of attackers attempting to penetrate a system. While anomaly detection typically utilizes threshold monitoring to indicate when a certain established metric has been reached, misuse detection techniques frequently utilize a rule-based approach. When applied to misuse detection, the rules

Manuscript received February 5, 2010
Manuscript revised February 20, 2010

become scenarios for network attacks. The intrusion detection mechanism identifies a potential attack if a user's activities are found to be consistent with the established rules. The use of comprehensive rules is critical in the application of expert systems for intrusion detection.

III. RELATED WORK

Most current approaches to the process of detecting intrusions utilize some form of rule-based analysis. Rule-Based analysis relies on sets of predefined rules that are provided by an administrator, automatically created by the system, or both. Expert systems are the most common form of rule-based intrusion detection approaches. The early intrusion detection research efforts realized the inefficiency of any approach that required a manual review of a system audit trail. While the information necessary to identify attacks was believed to be present within the voluminous audit data, an effective review of the material required the use of an automated system. The use of expert system techniques in intrusion detection mechanisms was a significant milestone in the development of effective and practical detection-based information security systems.

An expert system consists of a set of rules that encode the knowledge of a human "expert". These rules are used by the system to make conclusions about the security-related data from the intrusion detection system. Expert systems permit the incorporation of an extensive amount of human experience into a computer application that then utilizes that knowledge to identify activities that match the defined characteristics of misuse and attack.

Unfortunately, expert systems require frequent updates to remain current. While expert systems offer an enhanced ability to review audit data, the required updates may be ignored or performed infrequently by the administrator. At a minimum, this leads to an expert system with reduced capabilities. At worst, this lack of maintenance will degrade the security of the entire system by causing the system's users to be misled into believing that the system is secure, even as one of the key components becomes increasingly ineffective over time.

Rule-based systems suffer from an inability to detect attacks scenarios that may occur over an extended period of time. While the individual instances of suspicious activity may be detected by the system, they may not be reported if they appear to occur in isolation. Intrusion scenarios in which multiple attackers operate in concert are also difficult for these methods to detect because they do not focus on the state transitions in an attack, but instead concentrate on the occurrence of individual elements. Any division of an attack either over time or among several seemingly unrelated attackers is difficult for these methods to detect.

Rule-based systems also lack flexibility in the rule-to-audit record representation. Slight variations in an attack sequence can affect the activity-rule comparison to a degree that the intrusion is not detected by the intrusion detection mechanism. While increasing the level of abstraction of the rule-base does provide a partial solution to this weakness, it also reduces the granularity of the intrusion detection device.

An artificial neural network consists of a collection of processing elements that are highly interconnected and transform a set of inputs to a set of desired outputs. The result of the transformation is determined by the characteristics of the elements and the weights associated with the interconnections among them. By modifying the connections between the nodes the network is able to adapt to the desired outputs.

Unlike expert systems, which can provide the user with a definitive answer if the characteristics which are reviewed exactly match those which have been coded in the rule base, a neural network conducts an analysis of the information and provides a probability estimate that the data matches the characteristics which it has been trained to recognize. While the probability of a match determined by a neural network can be 100%, the accuracy of its decisions relies totally on the experience the system gains in analyzing examples of the stated problem.

The neural network gains the experience initially by training the system to correctly identify pree-selected examples of the problem. The response of the neural network is reviewed and the configuration of the system is refined until the neural network's analysis of the training data reaches a satisfactory level. In addition to the initial training period, the neural network also gains experience over time as it conducts analyses on data related to the problem.

IV. NEURAL NETWORK INTRUSION DETECTION SYSTEMS

A limited amount of research has been conducted on the application of neural networks to detecting computer intrusions. Artificial neural networks offer the potential to resolve a number of the problems encountered by the other current approaches to intrusion detection. Artificial neural networks have been proposed as alternatives to the statistical analysis component of anomaly detection systems. Statistical Analysis involves statistical comparison of current events to a predetermined set of baseline criteria. The technique is most often employed in the detection of deviations from typical behavior and determination of the similarly of events to those which are indicative of an attack. Neural networks were specifically proposed to identify the typical characteristics of system users and identify statistically significant variations from the user's established behavior.

Artificial neural networks have also been proposed for use in the detection of computer viruses.

Neural networks were proposed as statistical analysis approaches in the detection of viruses and malicious

software in computer networks. The neural network architecture may be a self-organizing feature map which uses a single layer of neurons to represent knowledge from a particular domain in the form of a geometrically organized feature map. The proposed network was designed to learn the characteristics of normal system activity and identify statistical variations from the norm that may be an indication of a virus.

While there is an increasing need for a system capable of accurately identifying instances of misuse on a network there is currently no applied alternative to rule-based intrusion detection systems. This method has been demonstrated to be relatively effective if the exact characteristics of the attack are known. However, network intrusions are constantly changing because of individual approaches taken by the attackers and regular changes in the software and hardware of the targeted systems. Because of the infinite variety of attacks and attackers even a dedicated effort to constantly update the rule base of an expert system can never hope to accurately identify the variety of intrusions.

The constantly changing nature of network attacks requires a flexible defensive system that is capable of analyzing the enormous amount of network traffic in a manner which is less structured than rule-based systems. A neural network-based misuse detection system could potentially address many of the problems that are found in rule-based systems.

V. PROPOSED FRAMEWORK

Figure 1, illustrates the proposed framework. The proposed framework is described in terms of four phases; the first phase is the network sensor in this phase we analyze the input packets to obtain the packet parameters and then filtering these parameters to obtain the needed parameters for intrusion detection, the second phase is the event manager which processes the filtered parameters and then compare these parameters with known attacks for determining attacks signatures and also compare these parameters with normal events then we go to the third phase which is the response manager which respond to the attack and normal events in a suitable manner. The fourth phase is the learning model in this phase we use a mixed database of normal and attack events then sending these events to learning model of neural network. After obtaining a learning module, the unlabeled events could be classified as a normal or attack events.

Applying this framework we can obtain a database of normal and attack events then we can use this database for applying our algorithms, we have a database of 145587 of normal and attack event we use this database for our study. We use 70% of input events for training our network using the multilayer perceptrons algorithm and the other 30% of the input events for testing the network.

An institutive goal of classification is to discriminate between normal and attack events, while a more ambitious goal may be to classify different attack types. There is a large number of ANNs based classifiers. The performance of nine of them will be evaluated and assessed.

We use four different measures to evaluate the performance of the artificial neural network based classification techniques: (i) mean-square-error (MSE); (ii) normalized mean-square-error (NMSE); (iii) correlation coefficient (r), and (iv) error percentage (%error). The mean squared error of an individual case (i) is evaluated by the equation:

$$MSE = \sum_{j=1}^{n} \left(P_{ij} - T_j\right)^2 / n \qquad (1)$$

where P(ij) is the value predicted by the individual case i for fitness case j (out of n fitness cases or sample cases); and Tj is the target value for fitness case j.

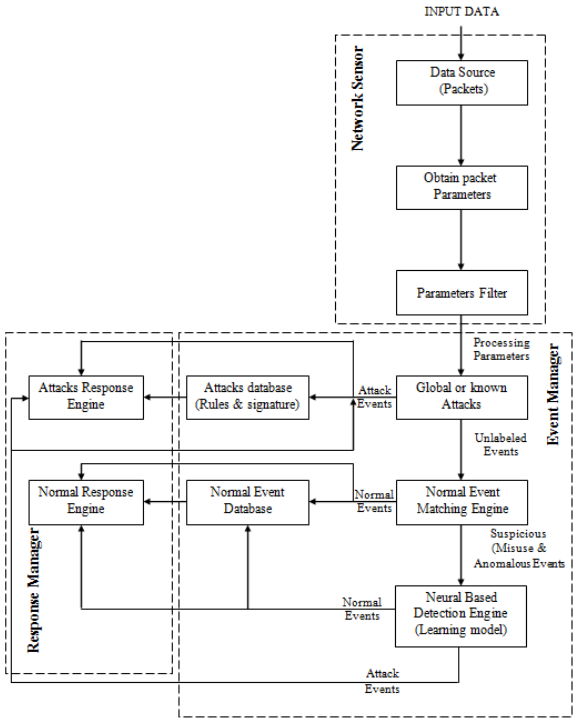

Fig.1. Proposed framework for the learning algorithm

The normalized mean square error is an estimator of the overall deviations between predicted and measured values. It is defined as:

$$MSE = \sum_{j=1}^{n} \left(P_{ij} - T_j\right)^2 / \left(n \times P \times T\right) \qquad (2)$$

where:

$$P = \sum_{j=1}^{n} P_{ij} / n \text{ and } P = \sum_{j=1}^{n} T_j / n$$

The correlation coefficient(r) is a quantity that gives the quality of a least squares fitting to the original image. For two data sets x, y; the auto correlation is given by:

$$r = \mathrm{cov}/(x, y)/(\sigma_x \times \sigma_y) \qquad (3)$$

Where σ_x and σ_y are the standard deviation of image x and y. Finally; the error percentage is calculated as the percentage difference between the measured value and the accepted value.

Multilayer perceptrons (MLPs)

Multilayer perceptrons (MLPs) are layered feedforward networks typically trained with static backpropagation. These networks have found their way into countless applications requiring static pattern classification. Their main advantage is that they are easy to use, and that they can approximate any input/output map. The key disadvantages are that they train slowly, and require lots of training data (typically three times more training samples than network weights).

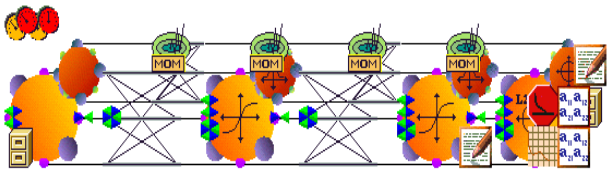

Fig.2. Multilayer perceptrons algorithm

The performance of multilayer perceptrons-based classifiers was evaluated through four performance indices; MSE; NMSE; r, and %Error. Table 1 illustrates the results of the multilayer perceptrons classifier. This classifier is trained with the following parameters: (i) 10 processing elements; (ii) one hidden layer, and (iii) 1000 epochs. We train 70 % of the input data and test the other 30% obtaining the following results.

TABLE 1
RESULTS OF MLP CLASSIFIER

MSE	0.019614889247
NMSE	0.061119179527
r	0.571021701230
% Error	0.711851781093

The classifier based on Multilayer perceptrons provides the results:
MSE=0.019614889247
NMSE=0.061119179527
r=0. 571021701230, and %Error=0. 711851781093.

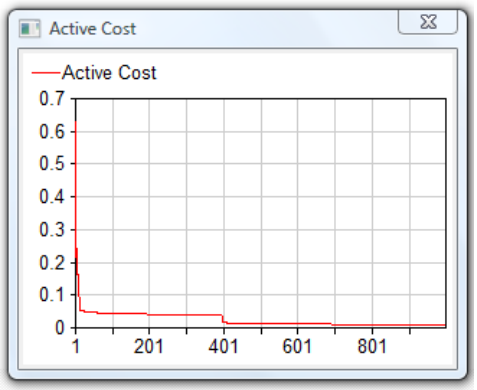

Fig.3. Learning curve of MLPS classifier after 1000 epochs

VII. Conclusion

As a result of the comparative study; the Multilayer perceptrons (MLPS) based classifier provides the best results among nine other classifiers; about 99.63% true positive attacks are detected using this classifier. That indicates that we have only 0.47 % false positive. The main advantage of the MLPs is that they are easy to use, and that they can approximate any input/output map. The key disadvantages are that they train slowly, and require lots of training data.

REFERENCES

[1] Anderson, D., Frivold, T. & Valdes, A (May, 1995). Next-generation Intrusion Detection Expert System (NIDES): A Summary. SRI International Technical Report SRI-CSL-95-07.

[2] Aickelin, Uwe, Twycross, Jamie, Hesketh-Roberts, Thomas. (2007). Rule Generalisation in Intrusion Detection Systems using Snort.

[3] Rebecca C. Leng, (May 4, 2009). Review of web applications security and intrusion detection in air traffffic control systems.

[4] Carpenter, G.A. & Grossberg, S. (1987). A Massively Parallel Architecture for a SelffOrganizing Neural pattern Recognition Machine. Computer Vision, Graphics and Image Processing 37,54-115.

[5] Chung, M., Puketza, N., Olsson, R.A., & Mukherjee, B. (1995) Simulating Concurrent Intrusions for Testing Intrusion Detection Systems:Parallelizing. In NISSC. pp. 173-183.

[6] Cramer, M., et. al. (1995). New Methods ofIntrusion Detection using Control-Loop Measurement. In Proceedings of the Technology in Information Security Conference (TISC) '95. pp. 1-10.

[7] Debar, H., Becke, M., & Siboni, D. (1992). A Neural Network Component for an Intrusion Detection System. In Proceedings of the IEEE Computer Society Symposium on Research in Security and Privacy.

[8] Debar, H. & Dorizzi, B. (1992). An Application ofa Recurrent Network to an Intrusion Detection System. In Proceedings of the International Joint Conference on Neural Networks. pp. (II)478-483.

[9] Denault, M., Gritzalis, D., Karagiannis, D., and Spirakis, P. (1994). Intrusion Detection:Approach and Performance Issues of the SECURENET System. In Computers and Security Vol. 13, No.6, pp. 495-507

[10] Denning, Dorothy. (February, 1987). An Intrusion-Detection Model. IEEE Transactions on Software Engineering, Vol. SE-13, No.2.

[11] Fox, Kevin L., Henning, Rhonda R., and Reed, Jonathan H. (1990). A Neural Network Approach Towards Intrusion Detection. In Proceedings of the 13th National Computer Security Conference.

[12] Frank, Jeremy. (1994). ArtificialIntelligence and Intrusion Detection: Current and Future Directions. In Proceedings of the 17th National Computer Security Coriference.

[13] Fu, L. (1992). A Neural Network Model for Learning Rule-Based Systems. In Proceedings of the International Joint Coriference on Neural Networks. pp. (I) 343-348.

[14] Hammerstrom, Dan. (June, 1993). Neural Networks At Work. IEEE Spectrum. pp.26653.

[15] Helman, P., Liepins, G., and Richards, W. (1992). Foundations ofIntrusion Detection. In Proceedings of the Fifth Computer Security Foundations Workshop pp. 114-120.

[16] Helman, P. and Liepins, G., (1993). Statistical foundations of audit trail analysis for the detection of computer misuse, IEEE Trans. on Software Engineering, 19(9):886-901.

[17] Ilgun, K. (1993). USTAT: A Real-time Intrusion Detection System for UNIX. In Proceedings of the IEEE Symposium on Research in Security and Privacy. pp. 16-28.

[18] Kumar, S. & Spafford, E. (1994) A Pattern Matching Model for Misuse Intrusion Detection. In Proceedings of the 17th National Computer Security Conference, pages 11-21.

[19] Kumar, S. & Spafford, E. (1995) A Software Architecture to Support Misuse Intrusion Detection. Department of Computer Sciences, Purdue University; CSD- TR-95-009

[20] Lunt, T.F. (1989). Real-Time Intrusion Detection. Computer Security Journal Vol. VL Number 1. pp.9-14.

[21] Mukherjee, B., Heberlein, L.T., Levitt, K.N. (May/June, 1994). Network Intrusion Detection. IEEE Network. pp.28-42.

[22] Porras, P. & Neumann, P. (1997). EMERALD: Event Monitoring Enabling Responses to Anomalous Live Disturbances. In Proceedings of the 20th NISSC.

[23] Puketza, N., Chung, M., Olsson, R.A. & Mukherjee, B. (September/October, 1997). A Software Platform for Testing Intrusion Detection Systems. IEEE Software, Vol. 14, No.5

[24] Ryan, J., Lin, M., and Miikkulainen, R. (1997). Intrusion Detection with Neural Networks. AI Approaches to Fraud Detection and Risk Management: Papersfrom the 1997 AAAI Workshop (Providence, Rhode Island), pp. 72-79. Menlo Park, CA: AAAI.

[25] Sebring, M., Shellhouse, E., Hanna, M. & Whitehurst, R. (1988) Expert Systems in Intrusion Detection: A Case Study. In Proceedings of the 11 th National Computer Security Conference.

[26] Tan, K. (1995). The Application of Neural Networks to UNIX Computer Security. In Proceedings of the IEEE International Conference on Neural Networks, Vo!.l pp.476-481.

[27] Tan, K.M.C & Collie, B.S. (1997). Detection and Classification of TCP/IP Network Services. In Proceedings of the Computer Security Applications Conference. pp. 99-107.

[28] White, G.B., Fisch, B.A., and Pooch, U.W. (January/February 1996). Cooperating Security Managers: A Peer-Based Intrusion Detection System. IEEE Network. pp. 20-23.