International Journal of Scientific Research in Computer Science, Engineering and Information Technology
© 2018 IJSRCSEIT | Volume 3 | Issue 7 | ISSN : 2456-3307

To Achieve an Unified Intrusion Detection System Based on Artificial Neural Network

Dr. Abid Hussain[1], Mr. Praveen Kumar Sharma[2]
[1]Assistant Professor, School of Computer Applications, Career Point University, Kota, Rajasthan, India
Email Id : abid.hussain@cpur.edu.in
[2]Vardhman Mahaveer Open University, Kota, Rajasthan, India
Email Id : praveenvmou@gmail.com

ABSTRACT
An intrusion detection system is an important component of the computer and information security framework. Its main goal is to differentiate between normal activities of the system and behaviours that can be classified as suspicious or intrusive. The research aims at the design, implementation and evaluation of an intelligent Intrusion Detection System based on artificial neural network that can promptly detect attacks, no matter they are known or not. In this system, neural network is used to learn about the normal users' behaviour to form the network traffic that only contains information about normal users. When the learning is over, the system is tested with the network traffic that contains both attacks and normal data. A simulated computer network is used to test the system performance. In experiments, system performance has been compared with other research works and the results in experiments are very promising.

Keywords : Anomaly-based system; Network security, Intrusion Detection System, Artificial Neural Network.

I. INTRODUCTION

The detection methods of intruders in the computer networks have drawn attentions of many researchers in recent years. An intrusion detection system is the attempt at detecting intruders in a computer system or network. It is very important to design security mechanism to prevent all these malicious activities to the system resources and data. However, it is not possible to have a complete prevention. At present, an intrusion detection system becomes necessary for detecting the attacks so that the appropriate actions can be taken to repair the damages. Artificial neural network is a kind of information processing paradigm that ii inspired by the biological nervous systems, such as the brain, to process information.

It tries to represent the physical brain and thinking process by means of an electronic circuit or software. Artificial neural network is the network of individual neurons. Each neuron in a neural network acts as an independent processing element. Like human or other brains, neural networks also learn by examples or training, and they cannot be programmed to perform a specific task. It can be configured for any specific application with a learning process. Neural networks perform very successfully for recognizing and matching complicated, vague, or incomplete patterns. The most successful applications of neural network are classification and pattern recognition.

This paper describes an intelligent intrusion detection system based on artificial neural network for network anomaly detection. The system takes network traffic data to analyse, classify the

behaviours of the authorized users, and recognize the likely attacks. System performance has been tested and satisfactory results have been obtained.

II. RELATED WORK

More and more computers are connected to the Internet every day and keeping pace with the development of computer technology, security is becoming a major concern. Much effort has been given to prevent the intruders from the computer systems as well as from the networks. Some preventive methods are working quite well but intruders remain a big threat to the computer security. Thus, research trends to pay attention not only to prevention but also to detection. Intrusion Detection System, hence, becoming a favourite research topic.

A. Artificial Intelligence in IDS

Issues relevant to intrusion detection include data collection, data reduction, behaviour classification, reporting and response. There have huge amount of collected data like audit data or network traffic data. Focusing on data reduction and classification, it is found that Artificial Intelligence techniques have been used in much intrusion detections system for performing these tasks. So many AI techniques have been used for solving intrusion detection tasks. Some mentionable AI techniques that have been used so far are: Expert systems, Rule based induction, Classifier systems like Neural Networks and Decision tree, Feature selection, Clustering etc. Expert system solves problems by using the computer model of expert human reasoning and it requires continuous maintenance and upgrading for performing well. Behaviour classification of intrusion detection systems can be done using expert systems techniques by encoding the security policy and known attacks as well as system vulnerabilities as a fixed set of rules. User behaviour that matches those rules indicates that an attack in under way. Unlike expert systems, Rule based induction derives rules that explain the set of instances describing the problems and steps of solutions.

In an IDS, rule based systems create and manage rules corresponding to anomalous behaviour. Generally, Classifier systems classify different types of patterns from a set of patterns. A classifier like Neural Network uses a model of biological system to perform classification. Another type of classifier called Decision Tree tries to separate the data into two or more groups. Then it tries to separate these groups into further groups and *so* on until small groups of examples are left.

Feature selection is accomplished by searching subsets of features, of information sources, and testing the ability of those features to perform the intended task. It normally reduces the amount of information required for a particular task. For instance, some data may hinder the classification process, which can be eliminated by feature selection. Moreover, some features may be redundant as the information belongs to those features is already held by other features.

In data clustering, data are grouped together according to some common characteristics or criteria. It is used to find the hidden pattern in data that might be missed. Data clustering techniques are also used for reducing the amount of data by dealing with the characteristics of the clusters instead of the actual data.

Artificial intelligence technique like artificial neural network can be used for classification purpose in an intrusion detection system. Neural networks learn by training. This type of learning methods are used to train a neural network supervised, where the net is trained by both the input and output pattern; and unsupervised, where the net is trained only by the input pattern. A neural network has one input layer,

one output layer, and zero or more number of hidden layers. All these layers contain a number of neurons, the basic element of neural networks. All neurons in different layers are connected each other in two ways: feedback and feed forward. Neural networks are classified from their internal structure as well as learning methods used for training.

B. Back propagation Neural Networks

Artificial neural network is a network of many simple processing units, each possibly having a small amount of local memory. These units are connected by some sorts of connections, which usually carry numeric data, encoded by any of various means. In principle, neural network can compute any computable function, in practice, they are especially useful for classification and function approximation problems which are tolerant of some imprecision, which have lots of training data available, but to which hard and fast rules cannot easily be applied. There are many types of neural networks available. The types of neural networks can be distinguished by their physical structure, learning method and connection structures etc. Back propagation neural network is one of the most powerful neural networks. It has the same structure as multilayer perception and mainly used in complex logical operations, pattern classification and speech analysis. Like in multilayer perception, back propagation neural network has three layers: input, output and hidden layers. Before the training of the net, i.e. any data has been run through the network; weights are simply random numerical values. When presented with an input pattern, each input node takes the value of the corresponding attribute in the input pattern. Then, each node in the hidden layer multiplies each attribute value by a weight and adds them together.

The same process is repeated in the output layer with the value from the hidden layer, and if the threshold value is acquired for this layer, it represents the classified pattern for the corresponding input pattern, which is compared with the actual classification pattern and error value is calculated. This error value is then back propagated through the network, and the weights of output and hidden layers are adjusted with these error values. This process is repeatedly carried out until it satisfies a predefined termination condition. The net is then assumed 'trained' and the weights are stored. However, the weight change is performed in such a way that the current point on the error surface will descend into a valley of the error surface, in a direction that corresponds to the steepest gradient or slope at the current point on the error surface. The advantages of back propagation neural nets are that they are great at prediction and classification. On the other hand, there is always a lack of explanation of what the net has learned.

III. DESIGN AND IMPLEMENATATIONS

The objective of this work is to develop an Anomaly Based Network Intrusion Detection System. In anomaly-based system, the main task is to recognize and identify intruders from that knowledge. Behaviour of a particular user can be formed from his habit of using computer. For instance, a particular user may use only some specific types of protocols, or he may only use some particular machines i.e. particular IP addresses, or he may only work within a specific time range e.g. during office hours, and so on. This information can be obtained from the network traffic. Thus, it is possible to store normal users' behaviour and intruders can be recognized from the distortion of normal behaviour.

A. Overview

System objective is to design a behaviour-based system to detect intruders in a computer network. Operation of the system is divided into three phases: Input Data Collection and Pre-processing, Training and Detection. In pre-processing phase, network traffic is collected and processed for use as input to

the system. In the training phase, this system gathers knowledge about the normal behaviour of the network users from the pre-processed input data, and store the acquired knowledge. In the detection phase, the system detects attacks based on the knowledge, which is achieved during the training phase, and notify the system administrator. The system overview is given in Figure 1.

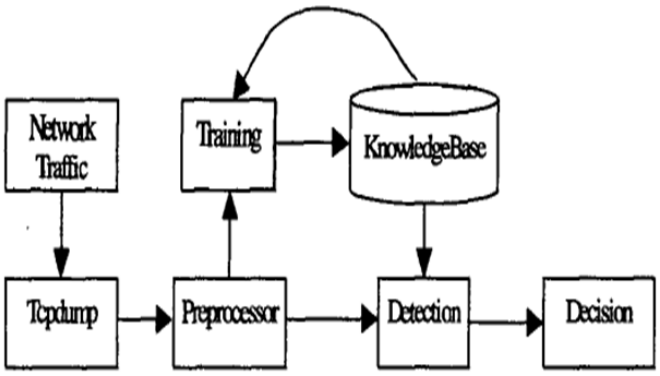

Figure 1. System Overview

The main task is to generalize and classify user behavior and detect intruders from this classification. This task may be carried out using various approaches like expertise systems, rule based induction, artificial neural network approaches etc. We have selected neural networks to accomplish our task. This is because they have some preferable properties that make them suitable for the task of classification.

The preferable properties of artificial neural networks are: It is their inherent property that they can learn through training. Their complex internal structures enable them to learn and accommodate large number of patterns. They can generalize the knowledge acquired through training for similar patterns. They have efficient storage capability of a large set of patterns. Because of these attractive characteristics of neural network, we have chosen it for classifying user behaviour in our system. Since, in the training phase of our classification task, both the input and output Patterns **are** available, we can use a supervised learning method i.e. we can take the help of a teacher to train the neural net. Hence, we have decided to use back propagation neural network in this regard.

B. Input Data Collection and Reprocessing

Network traffic can be collected either from a real computer network or from a simulation of network. Due to the unavailability of a suitable real computer network, we propose to use network traffic data from a simulated network. Each session from the network traffic is used as an input pattern of our system. These traffic sessions are represented in text form in our collected data. To suit the input format of the back propagation neural network, we convert these traffic patterns into binary form.

The data format includes Start time, Duration, Source Port, Destination port, Source IP Address, Destination IP address, and Attack code. These seven types of information of each session can be divided into two groups. First six pieces of information can be concatenated and represented as a single line of binary bits to form the first group, which will be considered as the input pattern to the system. And the single bit of attack state can be considered **as** the second group which will be used as corresponding output pattern. We have, in this stage, kept both of the groups together in a Single line of binary bits keeping in mind that we will feed them separately to the system for carrying on the next phase of operation. All these preprocessed sessions can be stored in one or more files.

C. Training

In training phase, user behaviour is classified from the network traffic sessions. Back propagation neural network is used for this purpose. The input to the neural network is the first group of pre-processed traffic session and the output of the neural network is the second group i.e. either attack or normal. In this

phase, first of all we retrieve sessions from the file and separate them into two groups: one is for input and the other is for target output. And secondly, we train the network with the input data, check if the generated output pattern satisfies the target output pattern and calculate the error from the distortion of target output. We retrain or stop training the network depending on this error value. Once the training is over, we store the knowledge it acquired.

The input pattern made from each session consists of total M binary bits. A is a constant value which is used as a single input to the network. There have one hidden layer and it consists of N nodes. Figure 2 is this network architecture.

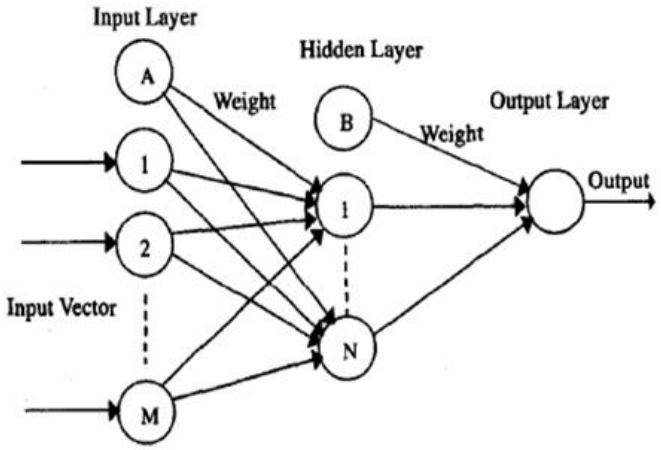

Figure 2. Neural Network Architecture

Our pre-processed data is stored in the files and it has to be retrieved and fed to the system. One file contains data for all sessions of one single day. We retrieve all the data of one external file and store it in a temporary file. We take each single session at a time from this temporary file and feed the network. Each session is nothing more than a vector consisting both the input and output patterns. All the sessions of current file can be fed to the network using an inner loop. The whole process can be repeated for all the external files using an outer loop.

In the beginning of the training, the weight values of the back propagation neural network have to be initialized. For the first time of training, they are initialized with random values. This is because, if all weights in the network are initially set to the same value and the solution to the problem requires unequal weights, the network will never learn since error changes are proportional to the weight values. For consecutive training periods, they are just initialized with the previous stored values.

There are many patterns available for training the network. We train the network with one pattern at a time by calculating the error value and updating the weights. Training the network with all available input patterns defines an epoch. We continue the training for a specific number of epochs depending on the termination condition. The termination condition might be a reasonably predefined number of epochs or until an inferior error value is achieved. When the training is terminated, the updated weight values are stored in external files. These weight values can be used for further training or for classification.

D. DETECTION

In this phase, the system is already trained with the normal behaviour of user and from that knowledge it will detect the intruders or abnormal behaviour of authorized users. In this phase, the input pattern is same as training but there is no output pattern. This step has the same network architecture like training. We retrieve input data from file in the same way as in training step except that we discard the output pattern i.e. the last bit of session vector, as there is no use of output pattern in this case. Like we have trained the network, in the beginning of detection, the network has to be initialized with the weights. In this step, we initialize the network with the weight values that we have stored after training. This. is the knowledge of the network that has to be used to find the abnormal behaviour. After that, we feed the

input vector to the network and get the output produced by the network. This output will show us whether the given input pattern represents normal behaviour or an attack. Unlike training phase, we do not calculate any error value in the detection phase. The same process is continued for all the sessions of all the test files and only for one single epoch.

IV. EXPERIMENTS AND EVALUATION

The system has been implemented, trained and experimented using different data sets in the Linux environment (Red Hat Linux version 9.0) on PC (Intel Pentium4 1.7GHz Processor). The system has been trained in several different ways and tested with different sets of test data. For all approaches, the detection rate for normal data (both known and unknown) is 100%. It means we have 0% false positive rate for each of the approaches. However, among those approaches, the last approach performs very well and returns the best throughput. In this approach, the system has a detection rate of 100% for known attacks and 96% for unknown attacks for small amount of test data. It means, overall performance for small amount of test data is 98%. For big amount of test data, the system raises some false alarms but this problem is solved with retrain technique.

After retraining, false positive rate becomes 0%. However, detection rate for known and unknown attacks are 100% and 90% respectively. Therefore, overall performance for detecting attacks is 95%. The system performance has been compared with that of other related works and better performance has been reported to some extent.

V. CONCLUSION

As showed in this research, neural networks can be used successfully as a method for training and learning an Intrusion Detection System. In this research, the neural network has been used to classify normal traffic correctly and detect attacks on intrusion detection, so the usefulness and drawbacks of these approaches are investigated. The architecture of neural network was designed in such a way that any number of input and hidden units can be used for the purpose of modification of the neural net for different training approaches.

The results of evaluation show that the system performs better than some of other existing intrusion detection systems. In this work, only back propagation neural network is used, but there are many other neural networks available. Further research might be carried out to build a system with other types of neural networks. Moreover, a combination of more than one type of neural networks can also be used to get better results.

VI. REFERENCES

[1]. D Joo, T. Hong and I. Han, "The Neural Network Models for IDS based on the asymmetric costs of false negative errors and false positive errors", Expert Systems with Applications, Vol. 25, pp. 69-75,2003

[2]. S Mukkamala, G. Janoski and A. Sung, "Intrusion Detection Using Neural Networks and Support Vector Machines", Proc. of the 2002 International Joint Conference on Neural Networks, Vol. 2, pp.

[3]. R P. Lippmann and R. K. Cunningham, "Improving Intrusion Detection Performance using Keyword Selection and Neural Networks", Computer Networks,

[4]. J Cannady, "Next Generation Intrusion Detection: Autonomous Reinforcement Learning of Network Attacks", hoc. 23rd National Information Systems Security

Conference, pp. 1-12, Baltimore, 16-19 October 2000

[5]. J Ryan, M. J. Lin and R. Miikkulainen, "Intrusion Detection with Neural Networks", Advances in Neural Information Processing Systems, Vol. 10, pp. 943-949, Cambridge, MA: MIT Press, 1998

[6]. G Giacinto, F. Roli and L. Didaci, "Fusion of Multiple Classifiers for Intrusion Detection in Computer Networks", Pattern Recognition Letters,

[7]. T Verwoerd an dR. Hunt, "Intrusion Detection Techniques and Approaches", Computer Communications, Vol. 25, No. 15, pp. 1356-1365,2002

[8]. D Dasgupta and F. Gonzalez, "An Immunity-Based Technique to Characterize Intrusion in Computer Networks", IEEE Trans. On Evolutionary Computation, Vol. 6, No 3