International Journal Of Advanced Engineering & Innovative Technology (IJAEIT)
ISSN: 2348 7208

ISRA IMPACT FACTOR: 0.42

Malicious Packet Detection Using Artificial NeuralNetwork

Mukesh A. Patil, Nitin A. Dethe, Sagar V. Gudadari and Prof. Samarsinh Jadhav

Abstract- In this research, malicious packet detection using neural network is introduced. This research aims to experiment with user behavior as parameters in malicious packet detection using back propagation neural network. Here we wanted to see if a neural network is able to classify normal traffic correctly, and detect known and unknown attacks without using a huge amount of training data. For the training and testing of the neural network, we create our own data sets. In our final experiment, we have got a classification rate of 99% on known and unknown attacks. Simulation result shows that the proposed approach detects the intrusions accurately and is well suitable for real time applications.

Keywords:Network Security, Intrusion Detection System, Artificial Neural Networks, Back propagation Neural Network, Malicious packet detection, Datasets, Detection Rate.

1. INTRODUCTION Confidentiality, integrity and availability of the system resources are the major concerns in the development and exploitation of network based computer systems. Enlargements of computer infrastructure have raised the vulnerability of these systems to security threats, attacks and intrusions. Some specific examples of intrusions that concern system administrators are Attempted break-in, Masquerading or successful break-in, Penetration by legitimate user, Leakage by legitimate user, Inference by legitimate user, Trojan Horse, Virus and Denial-of-Service. Generally these intrusions would cause loss/damage to our system resources in terms of unauthorized modifications of system files, user files or information and any other system information in network components. Hence a system is needed that detects any unauthorized modification forced by an attacker and able to run continually with minimal human supervision.An intrusion detection system (IDS) is one that inspects all inbound and outbound network activity and identifies suspicious patterns that may indicate a network or system attack from someone attempting to break into or compromise a system. According to the detection principles there are two types of intrusion detection system: Misuse and Anomaly detection. In Misuse detection, attack patterns or the behavior of the intruder is modeled (attack signature is modeled). Here the system will signal the intrusion once a match is detected. In Anomaly detection system, the normal behavior of the system is modeled and the system will raise an alarm once the behavior of the network does not match with its normal behavior. According to the source of data, there are two types of intrusion detection: Network-based IDS (NIDS) and Host-based IDS (HIDS). A network based IDS captures all network traffic and analyzes the content of individual packets for malicious traffic where as a host-based IDS identifies intrusions by analyzing system calls, application logs, file system modifications (binaries, password files, capability/acl databases) and other host activities and state.In the literature Statistical Techniques like Hidden Markov Model [1], Multivariate Adaptive Regression Splines [2], Bayesian Network and Classification and Regression Trees (CART) [3] have been applied to Intrusion detection. These statistical approaches usually results in an inflexible detection system that is unable to detect an attack if the sequence of events slightly different from the predefined profile. Rule-based systems have been proposed by Denning [4] for intrusion detection. Expert systems are the most common form of rule-based approaches. They permit the incorporation of an extensive amount of human expertise into a computer application that then utilizes that knowledge to identify activities that match the defined characteristics of misuse and attack. 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. In [5] fuzzy logic approach has been combined with data mining techniques for discovering association rules which can be applied for detecting intrusions.Recently, Artificial Neural Networks have

Mukesh A. Patil, Nitin A. Dethe, Sagar V. Gudadari, Research students of Padmabhooshan Vasantdada Patil Institute Of Technology,Bavdhan , Pune 411 021

been successfully applied for developing the IDS. ANN has the advantage of easier representation of nonlinear relationship between input and output and is inherent by fast. Even if the data were incomplete or distorted, a neural network would be capable of analyzing the data from a network. A Multilayer Perceptron (MLP) was used in [6] for misuse detection with a single hidden layer. A Similar approach was applied in [7] but generic keywords were selected to detect the attack preparations and actions after the break-in. The weakness of neural network based approaches is that if the dimension of the input data is very large then it is difficult for it to interpret the relationship between inputs and outputs. Clustering can be performed to find hidden patterns in data and significant features for use in detection. Self-Organizing Map was applied to perform the clustering of network traffic and to detect attacks in [8]. A hybrid model of the SOM and the MLP was proposed in [9] to detect the dispersing and possibly collaborative attacks. In [10], the self-organizing map was combined with the Resilient Propagation Neural Network (RPROP) for visualizing and classifying intrusion and normal patterns. If the system is complex and input features are numerous, clustering the events can be a very time consuming task. Feature extraction methods like Principal Component Analysis (PCA) or Singular Value Decomposition (SVD) [11] can be an alternative solution but extraction of features will lead to a less accurate detection model.Recently Feature selection is found to be more relevant to Intrusion detection System since the selected features retain their physical interpretation. In [12], a trial and error approach is employed for feature selection by deleting one feature at a time. Each reduced feature set was then tested on Support Vector Machines and Neural Networks to rank the importance of input features. The reduced feature set that yielded the best detection rate in the experiments was considered to be the set of important features. Bayesian networks used in [3] not only classify the data, but also select features based on the Markov blanket of the target variables. The CART algorithm proposed in [3] classifies data by constructing a decision tree and identifies the important features based on predictor ranking (variable importance). In general if a model which captures the relationship between different features or between different attacks and features the intrusion detection process would be simple and straightforward. In this paper we reported a Mutual Information [13] based Technique for selecting the important features and it is used as the input for a simple feed forward neural network trained by back propagation algorithm for detecting intrusions. Since the mutual information measures the arbitrary dependencies between random variables and is independent of the coordinates chosen they seem to be an appropriate approach for feature selection in ANN based Intrusion Detection.The rest of this paper is organized as follows. In section 2, we give a brief description about the proposed model for intrusion detection. In section 3, we give a brief introduction about artificial neural network. In section 4, we present the proposed system. Finally we present our conclusion in section 5

2. PROPOSED MODEL FOR INTRUSION DETECTION

The proposed methodology for Intrusion Detection in Computer Networks is based on using Artificial Neural Network (ANN) for detecting the Normal and Abnormal conditions of the given parameters, which leads to various attacks. The neural network approach for this purpose has two phases; training and testing. During the training phase, neural network is trained to capture the underlying relationship between the chosen inputs and outputs. After training, the networks are tested with a test data set, which was not used for training. Once the networks are trained and tested, they are ready for detecting the intrusions at different operating conditions. The following issues are to be addressed while developing an ANN for Intrusion Detection [14]:

I. Data Collection
II. Data preprocessing, representation and Normaliztion
III. Dimensionality Reduction
IV. Selection of Network Structure
V. Network Training and Testing

2.1 DATA COLLECTION

There are two ways to build IDS, one is to create our own simulation network, and collect relevant data and the other one is by using previously collected datasets. Issues like privacy, security, and completeness greatly restrict people from generating data. The advantage of using previously collected datasets is that the results can be compared with others in the literature. Some of the popularly used IDS datasets [15] are DARPA 1998 data set, DARPA 1999 data set and KDD Cup 1999 data set which are available in the MIT Lincoln Labs. In this work, we are creating our own data set for developing the IDS.

2.2 DATA PREPROCESSING, REPRESENTATION AND NORMALIZATION

Before training the neural network, the dataset should be preprocessed to remove the redundancy present in the data and the non-numerical attributes should be represented in numerical form suitably. During training of the neural network, higher valued input variables may tend to suppress the influence of smaller ones. Also, if the raw data is directly applied to the network, there is a risk of the simulated neurons reaching the saturated conditions. If the neurons get saturated, then the changes in the input value will produce a very small change or no change in the output value. This affects the network training to a great extent. To minimize the effects of magnitudes among inputs as well as to prevent saturation of the neuron activation function, the input data is normalized before being presented to the neural network. One way to normalize the data x is by using the expression:

$$x_n = \frac{(x - x_{\min})\,range}{(x_{\max} - x_{\min})} + \text{starting value} \qquad (1)$$

where $x_n$is the normalized value , and $x_{\min}$ *and* $x_{\max}$are the minimum and maximum values of the data.

2.3 DIMENSIONALITY REDUCTION

The ability to scale neural network applications to realistic dimension of intrusion detection problem is a major issue. The amount of audit data that an IDS needs to examine is very large and contains more number of variables even for a small network. If all the measured variables are used as inputs to neural network, it results in large size of the network and hence larger training time. To make the neural network approach applicable to large scale intrusion detection problems, some dimensionality reduction is mandatory. Also, networks involving too many input variables suffer from curse of dimensionality. A network with fewer inputs has fewer adaptive parameters to be determined, and these are more likely to be properly constrained by a data set of limited size, leading to a network with better generalization properties. There are two approaches to achieve dimensionality reduction: Feature Extraction and Feature Selection. In this work, feature selection is used for dimensionality reduction

2.4 SELECTION OF NETWORK STRUCTURE, NETWORK TRAINING AND TESTING

To make a Neural Network to perform some specific task, one must choose number of input neurons, output neurons, hidden neurons and how the neurons are connected to one another. For the best network performance, an optimal number of hidden-units must be properly determined using the trial and error procedure. The hidden layer neurons have tangent hyperbolic function as the activation function and the output have linear activation function. Once the appropriate structures of the network are selected, the ANN model is trained to capture the underlying relationship between the input and output using the training data. In this work, Back propagation algorithm is used to train the network, which propagates the error from the output layer to the hidden layer to update the weight matrix. After training, the networks are tested with the test data set to assess the generalization capability of the developed network.

3. REVIEW OF ARTIFICIAL NEURAL NETWORK

Artificial Neural Networks [16] can be viewed as parallel and distributed processing systems which consists of a huge number of simple and massively connected processors. The MLP architecture is the most popular paradigm of artificial neural networks in use today. Fig.1 shows a standard multilayer feed forward network with three layers. The neural network architecture in this class shares a common feature that all neurons in a layer are connected to all neurons in adjacent layers through unidirectional branches. That is, the branches and links can only broadcast information in one direction, that is, the "forward direction". The branches have associated weights that can be adjusted according to a defined learning rule.

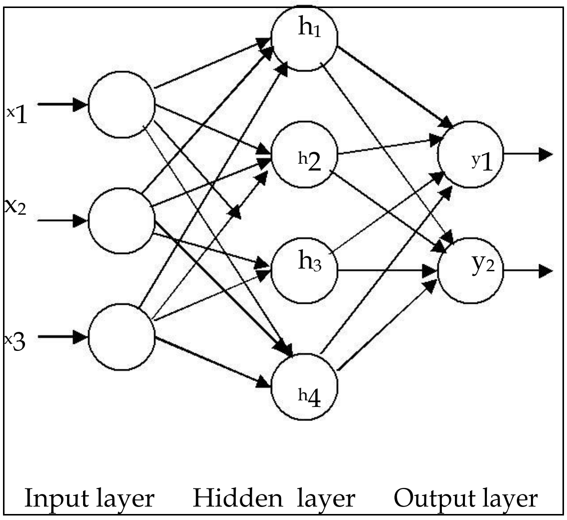


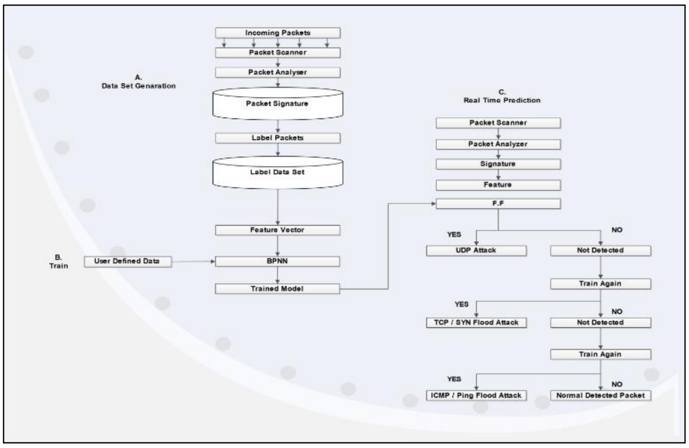
Fig.1. Architecture of feed forward neural network

Feed forward neural network training is usually carried out using the called back propagation algorithm. Training the network with back propagation algorithm results in a non-linear mapping between the input and output variables. Thus, given the input/output pairs, the network can have its weights adjusted by the back propagation algorithm to capture the non-linear relationship. After training, the networks with fixed weights can provide the output for the given input.

The standard back propagation algorithm for training the network is based on the minimization of an energy function representing the instantaneous error. In other words, we desire to minimize a function defined as

$$E(m) =-{}^{1}/_{2}\sum_{q=1}^{q}[d_{q}-y_{q}]^{2} \qquad (2)$$

where d_q represents the desired network output for the q^{th} input pattern and y_q is the actual output of the neural network. Each weight is changed according to the rule:



$$w =-k \frac{dE}{dw_{ij}} \text{re} \qquad (3)$$

where, k is a constant of proportionality, E is the error function and w_{ij} represents the weights of the connection between neuron *j* and neuron*i*. The weight adjustment process is repeated untilthe difference between the node output and actual output are within some acceptable tolerance.

4. THE PROPOSED SYSTEM

The prime goal of our project is to protect a server side resource that is to make the clients a valid request and if the any malicious activity is found then it should be handled at the IDS side and not at the server side. In the sense we can also call our system as "The Packet Inspection system". The architecture of our system is: Figure 3 illustrates the system architecture of our approach. The architecture of the system is:

The architecture of the our system is: In the above to say that the IDS is situated between the client and the server, there can have multiple number of clients as well as the servers. So that each of the packet going from the client to the server is inspected at the IDS.

There is total three models such as:

A. Data Set Generation

B. Train Module

C. Real Time Prediction

A Data Set Generation:-

Data set generation is work as like a First module of our system, where the incoming packets are capture here then that packet will be scan. After that packet analyzer will analyze the packets that which packets are coming in network that has affected or malicious packet then it will create the signature for each packet and stored the signature of such packets in data set. Label packet will label every packet with unique signature then it will label the data set then it will extract the features of that signature.

B. Train Module:-

Train module has user defined data set it will uses the BPNN Algorithm i.e., Back Propagation Neural Network it will help to trained module for easy access of packets and there signatures.

C. Real Time Prediction:-

In real time prediction here our actual work of IDS will started here is Some steps are involve such as Packet scanner, Packet analyzer, Signature, Features Extraction, if it has detect the affected packet then it will analyzed that it has Intrusion, if the any specified signature is not trained in trained module then it will also show that it is intrusion but it will match with any similar intrusion and then it will train such unknown type of intrusion for next time intrusion.

5. CONCLUSION

As showed in this research, neural networks can successfully be used as a method for training and learning an Intrusion Detection System. The main problem with today's Intrusion Detection System is that they produce many false alarms, and this takes up much of a system administrator's time and resources.

Here the neural network had a classification rate of 100 %, which gives a false positive rate of 0 %. This means that none of the normal sessions were classified as an attack. If normal traffic was classified as an attack a false alarm would be raised.

In this research, we have tested the ability of a backpropagation neural network to classify normal traffic correctly and to detect attacks without a huge amount of training data. The results of our study show that a neural network do not need huge amount of training data to be able to classify traffic correctly.

Thus, the proposed approach detects the intrusions accurately and is well suitable for real time applications.

6. LIMITATIONS AND FURTHER WORK

In this work there is need to regular update of the signature database, need to consider known andunknown attack. Only detect, cannot prevent the intrusions and it's an offline system. In the future

work this system can be extended to an online system by little effort. There has been a lot of research on intrusion detection, and also on the use of neural networks in intrusion detection. As showed in this thesis, backpropagation neural networks can be used successfully to detect attacks on a network. The same experiments should also be conducted with other types of neural networks to see if these types can improve the detection rate we got from the experiments with a backpropagation neural network.

7 REFERENCES

[1]Zhong and C.F. Jia. 2004, "Study on the applications of hidden Markov models to computer intrusion detection," in Proceedings of the Fifth World Congress on Intelligent Control and Automation WCICA, Vol. 5, pp. 4352-4356.

[2]M.Analoui, A.Mizaei, and P.Kabiri. 2005, "Intrusion detection using multivariate analysis of variance algorithms," in Third International Conference on Systems,Signals & Devices SSD05, Vol. 3.

[3]Chebrolu, S., A. Abraham and J.P. Thomas. 2005. Feature deduction and ensemble design of intrusion detection system. Computers & Security. Vol.24, No.4: pp.295-307.

[4]Denning, D. E., 1987. An Intrusion-Detection Model. IEEETransactions on Software Engineering. Vol.13, No.2:pp.222-232.

[5]Luo J., S. M. Bridges. 2000. Mining Fuzzy Association Rules and Fuzzy Frequency Episodes for Intrusion Detection. International Journal of Intelligent Systems. Vol.15, No.8: pp.687-704.

[6]J. Cannady, 1998, "Artificial Neural Networks for Misuse Detection," Proceedings, National Information SystemsSecurity Conference (NISSC'98), pp.443-456.

[7]Lipmann, R.P., and R.K Cunningaham1999. Improving Intrusion Detection Performance using keyword selection and neural networks. Computer Networks. Vol.34, No.4: pp.597- 603.

[8]M.Ramadas, S.Ostermann, and B.Tjaden, 2003, "Detecting anomalous network traffic with self organizing maps," in Recent Advances in Intrusion Detection, 6th International Symposium,(RAID-2003), pp. 36-54.

[9]J.M. Bonoficio, 1998. "Neural Networks Applied in Intrusion Detection Systems," IEEE World Congress onComputational Intelligence, Vol.1, pp. 205-210.

[10]C.Jirapummin, N.Wattanapongsakorn and P.Kanthamanon, 2002, "Hybrid Neural Networks for Intrusion Detection System," Proceedings of the International TechnicalConference on Circuits / Systems, Computers and Communications (ITC CSCC 2002), pp.928-931.

[11]Kabiri, P., and A. A. Ghorbani. 2005. Research on Intrusion Detection and Response: A Survey. International Journal ofNetwork Security. Vol.1, No.2: pp.84102.

[12]A. H. Sung, S. Mukkamala, 2003, "Identifying important features for intrusion detectio using support vector machines and neural networks," in Proceedings ofInternational Symposium on Applications and the Internet (SAINT 2003), pp. 209-17.

[13]R. Battiti. 1994. Using Mutual Information for Selecting Features in Supervised Neural Net Learning. IEEETransaction on Neural Networks. Vol.5, No.4:pp.537-550.

[14]P.GaneshKumar, D.Devaraj, V.Vasudevan, 2006, "Artificial Neural Network for Misuse Detection in Computer Network," Proceedings of the International Conference onResource Utilisation and Intellige Systems (INCRUIS-2006), pp.889-893.

[15]DARPA Intrusion Detection Evaluation – MIT Lincoln Laboratory – (http://www.ll.mit.edu/IST/ideval)