ISSN: 2319-5967
ISO 9001:2008 Certified
International Journal of Engineering Science and Innovative Technology (IJESIT)
Volume 2, Issue 6, November 2013

Intelligent Monitoring System –A network based IDS

SONALI M. TIDKE,
Dept. of Computer Science and Engineering,
Shreeyash College of Engineering and Technology, Aurangabad (MS), India

Abstract – Network security is a crucial part for every network in every organization. With introduction of emerging technologies; new attacks and new intrusions are also emerging and network is becoming more vulnerable to attacks. Unfortunately, in this digital world it is difficult to hide yourself from the attacks and intrusions. For this purpose, Intrusion Detection (ID) is becoming more important to safeguard networks. This paper focuses on importance of ID system (IDS) for detecting intrusions. Implemented system is trained using offline data and online captured data packets. Intelligent Monitoring System (IMS) implements pre defined algorithm of Artificial Neural Network (ANN) for identifying attacks. Multi layer Perceptron (MLP) is used for detecting intrusions in off-line and on-line mode of the system. Most of the previous IDS implementations are in off-line mode and mainly concentrates on identifying records as normal or abnormal. But here we are classifying records in various classes by identifying type of attack also. IDS can be converted to Intrusion Prevention System (IPS) by programming router.

Keywords **- Intrusion Detection System, Intrusion Prevention System, Artificial Neural Network, Multi Layer Perceptron, SYN_FLOOD, PING_FLOOD, JPCap**

I. INTRODUCTION

Today's network security infrastructure promisingly depends upon Network Intrusion Detection System (NIDS). NIDS provides safety from known intrusion attacks. It is not possible to stop intrusion attacks, so organizations need to be ready to handle them. Intrusion Detection System (IDS) is a defensive mechanism whose primary purpose is to keep work going on considering all possible attacks on a system.

Intrusion detection (ID) is a process used to detect suspicious activity both at network and host level. Two main ID techniques available are anomaly detection and misuse detection. In anomaly based detection system, audit data is used to differentiate abnormal data from normal one. On the other hand, misuse detection system, also called as signature based IDS, uses patterns of well known attacks to match with audit data and identify them as intrusions. Functioning of misuse detection models is in a sense very much similar to that of antivirus applications. Misuse IDS can analyze network or system and compare its activities against signatures of known intrusions and network behaviors. For recognizing traffic as attack, IDS must be taught to recognize normal activity. Various ways are available to accomplish this like use of artificial intelligence techniques. Audit data used for testing and creating rules or define patterns can be collected from various sources like network traffic data, system logs from hosts and system calls from various processes. IDS require sensor. Sensor is the system on which an IDS is installed and running. Network sensor monitors network packets like TCP/IP headers, duration of connection, and number of bytes transferred etc. while host sensor monitors system logs, memory usage on host etc.

Fig. 1 demonstrates the traditional IDS model. Here sensor machine generates security events, management console monitors those events and controls sensor. The intrusion detector engine records events logged bye the sensor into database and generates alerts based on rules from security events [1].
Section I provide the basic introduction about the IDS and need/purpose of IDS. In Section II, basic Artificial Neural Network (ANN) concepts have been given. Section III concentrates on dataset use for implementation of the system and classification technique used for identifying intrusions. Section IV provides general implementation details of the project. Section V concludes the paper with future scope and benefit of system.

ISSN: 2319-5967
ISO 9001:2008 Certified
International Journal of Engineering Science and Innovative Technology (IJESIT)
Volume 2, Issue 6, November 2013

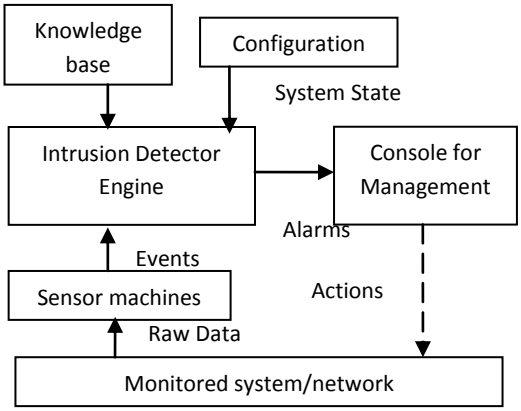


Fig 1. Traditional IDS model [1]

A. Purpose of the system

The purpose of the system is to detect certain well known intrusion attacks on the host system and display warnings to the user and also store information regarding the IP addresses and allow the traffic based on that information [2].

B. Scope of the System

The designed system works on off-line data and on-line data captured thru the host machine. As it uses supervised learning, once the network is trained thru back propagation algorithm, it identifies attacks 100% and no false negatives are generated for on-line data while off-line is also showing good results.

II. CONCEPTS OF ARTIFICIAL NEURAL NETWORK (ANN) FOR IDS

An artificial neuron is a computational model inspired from the natural neurons. Artificial neurons basically consists of inputs (like synapses), which are multiplied by weights (strength of the receptive signals) & then computed by a mathematical function which determines the activation of the neuron. Another function (which may be the identity) computes the output of the artificial neuron (sometimes in dependence of a certain threshold). ANNs combine artificial neurons in order to process information [3]. Soft computing techniques deals with partially true and uncertain data which makes them attractive to applied for designing of IDS. For example, genetic algorithms have been used along with decision trees to automatically generate rules for classifying network connections [4].

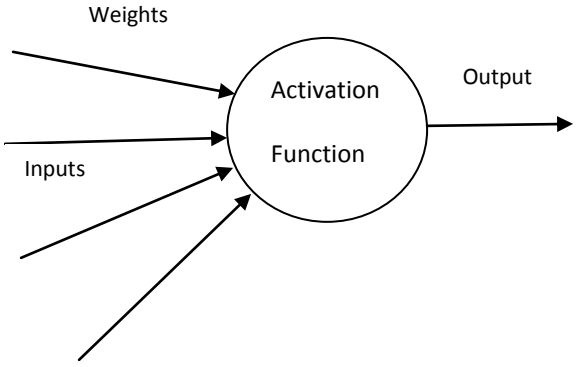


Fig 2. An Artificial Neuron [3]

However ANNs are the most commonly used soft computing technique in IDSs [5],[6],[7],[8],[9],[10]. Learning process in neural network is essentially an optimization process in which the parameters of the best set of connection coefficients (weights) for solving a problem are found and includes the following basic steps [6],[11]:

1. Present the neural network with a number of inputs vectors (each representing a pattern).
2. Check how closely the actual output generated for a specific input matches the desired output
3. Change the neural network parameters (weights) to better approximate the outputs.

The most basic use of neural network in IDS is for training the network. Once the network is trained using required learning method with an ANN algorithm, it is available for capturing data.

ISSN: 2319-5967
ISO 9001:2008 Certified
International Journal of Engineering Science and Innovative Technology (IJESIT)
Volume 2, Issue 6, November 2013

III. DATASET USED IN THE SYSTEM

The training and learning of the system uses off-line and on-line captured data both. While after implementation, the system uses online packets to detect intrusions. As the system works both in on-line and off-line mode, it considers common attacks like TCP/IP flood, ICMP(Ping) Flood, UDP Flood, SYN Flood attack. For off-line mode, DARPA dataset is considered. From the downloaded dataset, only required 11 features and few records are copied in a sample file to train the network and then the testing is done with remaining data in the file. Similarly for on-line mode, same 11 features are considered and packets are captured on-line. These features are broadly categorized into two sets: Set I contains features related to connection details of the captured packets like protocol type, basic flags, length of packet, hop limit etc. while set II concentrates on instructions used for the connection establishment. Since only 11 most important features are required for identifying attack in four classes, the 11 dimension vector is considered. For selected features, a numerical value is attributed. Numerical conversion of feature vector is necessary as the input vector for neural network must be numerical. Since the ranges of the features were different and this made them in comparable, the features were normalized by mapping all different values for each feature to [0,1] range[6].

IV. IMPLEMENTATION OF THE SYSTEM

The system is implemented using Java programming language. JPCap is used to capture packets online while java.io package is used for reading data from the DARPA set. The DARPA dataset is divided into a small file for testing purpose, with 11 features extracted from the original file and about 100 records from each type of class. Similarly, same 11 features are extracted from the online packets and used for training the network. In both on-line and off-line mode, same network is trained with different input vector. The neural network developed is 2 Multi Layer Perceptron (MLP) with one hidden layer. While developing system, first two hidden layers were chosen. With 3MLP network, rate of correct classification in off-line mode was 92 % and in on-line mode, it was 100 % while with 2MLP, it is 88 % for off-line mode and 100 % for on-line mode and no false negatives were generated.

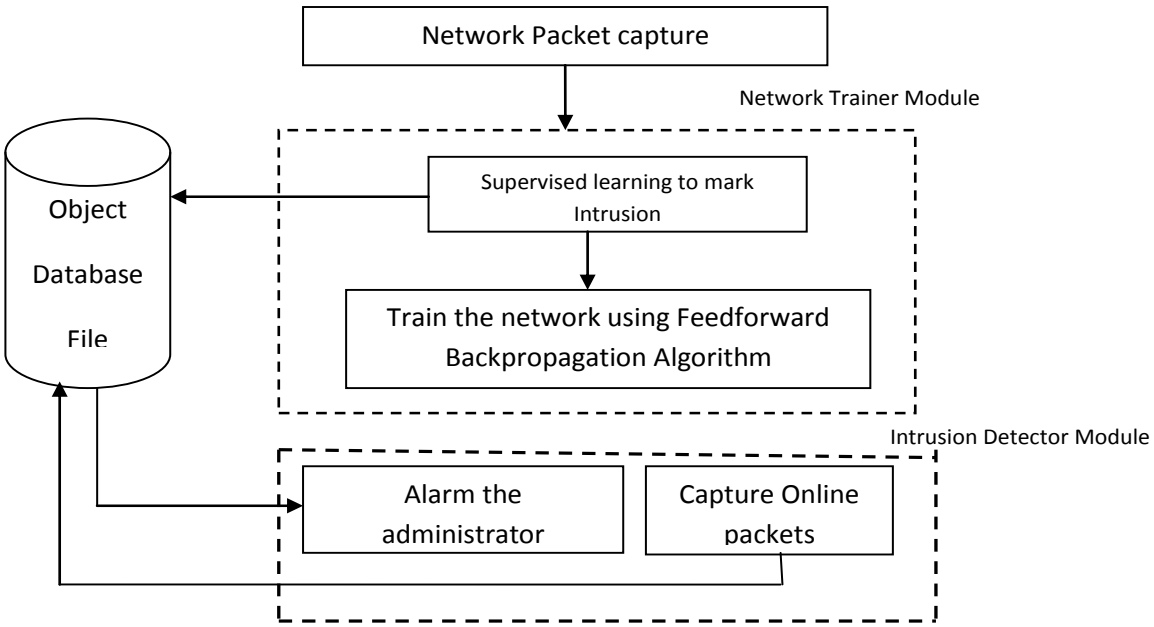
A. Learning Method and Algorithm Used

Supervised learning method with Feed forward back propagation algorithm is used for implementing system. In Feed forward neural network, neurons are only connected in forward direction. Each neuron in every layer is connected with the neurons in the next layer but no connection is back direction. One more neural network can be considered where neurons are fully connected in forward and backward direction which is called as Hopfield neural network. The term back propagation determines the training method of neural network. Back propagation is a type of supervised learning method. In this training method, the network must be fed with sample input and its expected output. This output is compared with actual output for given input vector. With this expected output, back propagation training algorithm calculates the error and adjusts weights of various layers backwards from the output layer to the input layer. The back propagation and Feed forward algorithms are often used together.

B. System Details

The System is divided into basically three parts: Implementation of Algorithm, Training of network and Artificial traffic generator to test network

Fig 3. Block Diagram of IMS



ISSN: 2319-5967
ISO 9001:2008 Certified
International Journal of Engineering Science and Innovative Technology (IJESIT)
Volume 2, Issue 6, November 2013

C. Implementation of Algorithm

In this section the neural network training algorithm *i.e.* Feed forward Back propagation Algorithm is developed. For this, three different user defined classes are used.

1. Single Neuron class : This class is used to calculate weight of a single neuron by assigning some random weight at the beginning to all the dendrites connected to the neuron. A random function is used to assign random weight to every dendrite and all these weights are used to calculate initial weight for every incoming neuron.
2. Single Layer class: This class is the class used to calculate weight for each neuron in a layer. An array containing weights for each neuron in a single layer is created in this layer.
3. Neural Network Class: Neural Network class is the class which is used to train the neural network using Feed forward method. In this, learning Rate, total number of layers in the neural network and neurons in each layer is provided. Above declared class, Single Layer along with Single Neuron class is used to find number of neurons in each layer along with initial weight of input layer. Here, number of neurons in next layer is one more than previous layer and only output layer are having predefined number of neurons which is equal to the number of outputs classes based on network requirement. Following are the various functions declared in the Neural Network class:

- set Inputs(): this function is used to assign initial weight to the input layer. The weights for the input layer are accepted as an argument of type array with data type double.
- Limiter (): 1.0 / (1 + Math.exp(-x)) formula is used to calculate limiter value of the function where x is the input argument provided to the function.
- run Network(): This function is used to update all the old values to new set of values. A temporary output array is created which will store the outputs. Initially each neuron in every layer other than input layer, value 0 is assigned as default value. Now the new values for each neuron in every layer other than output layer will be calculated by multiplying weights and value of each neuron in previous layer and then adding them with value of previous layer. After calculating new value for each neuron in every layer, bias is added and limiter function is applied to every neuron. These new values are set as output value of every layer.
- SigmaWeightDelta():Back propagation algorithm needs sum of weights multiplied by delta for each neuron in every layer. This function is used to calculate it.
- Train (): This is one of the most important function in the network. This function is used to actually implement back propagation algorithm. It calls set Inputs() function to initialize values of input layer and runNetwork() function to calculate and update all the initial/default or old values.

For Back propagation, we need to start from last layer as first to back propagate after getting output value for each layer.

D. Training Network

For training network, supervised learning is used. As we are using feed forward method with back propagation algorithm, supervised learning is the best method to train the network. While training network, the captured packets will be monitored by the administrator and then admin will mark the packets either as ok or intrusion. All the packets marked as intrusion by the admin will be stored in an Object Output Stream class file and a object file will be created.

- updateDB(): Method updateDB() is used to create a database file to store all the packets which are marked as intrusions. The method writeObject from ObjectOutputStream class of Java in-built class is used to write those intrusions in the database.
- readDB() : This is the method used to read intrusions from the database file, convert them in packets and then display in the form of packets in table form on the java frame.

E. Artificial traffic generator to test network

To test the network, an artificial traffic generator program is created. This program is used to generate all the four type of intrusions *i.e.* FLOOD_SYN, PING_SYN, UDP_SYN and TCP_SYN attacks. The intrusions generated will be captured by the network and will be displayed as intrusions.

V. CONCLUSION

Different kinds of techniques for intrusion detection are studied before the actual implementation of the proposed model. The motivation behind the adopted approach for Intrusion Detection presented in the design is the strength and capability of Back propagation method used primarily for classification. The design is of IDS is so flexible that it can be customized easily for new types of intrusion. On identification of the signature of the new attack the used algorithm in the implemented system can be trained to counter the future attacks of that type.

ISSN: 2319-5967
ISO 9001:2008 Certified
International Journal of Engineering Science and Innovative Technology (IJESIT)
Volume 2, Issue 6, November 2013

An approach for a neural network based intrusion detection system, intended to classify the normal and attack patterns and the type of the attack, has been presented. It should be mentioned that the long training time of the neural network was mostly due to the huge number of training vectors of computation facilities. However, when the neural network parameters were determined by training, classification of a single record was done in a negligible time. Therefore, the neural network based IDS can operate as an *online* classifier for the attack types that it has been trained for. A two layer neural network is used for the classification of on-line and off-line records. Although the classification results were better in the three layer network, application of a less complicated neural network is more efficient memory wise.

From the practical point of view, the experimental result simply that lot of innovations can be done in the field of artificial neural network based intrusion detection systems. The implemented system solved a four class problem. However, its further development to several classes is straight forward. As a possible future development to the present study, one can include more attack scenarios in the dataset. Practical IDSs should include several attack types. In order to avoid unreasonable complexity in the neural network, an initial classification of the connection records to normal and general categories of attacks can be the first step. The records in each category of intrusions can then be further classified to the attack types.

The system does not completely shield network from Intruders, but IDS helps the Network Administrator to track down anomalies on the Internet whose very purpose is to bring your network to a breach point and make it vulnerable to attacks. The present system is trained only on the known attacks. In future the system can be trained on various network flow features like Flow Count, Average Flow Packet Count, and Average Packet Size *etc.* for clear and better classification of traffic with low false positive and false negative rate. This can be extended by incorporating Intelligence into it in order to gain knowledge by itself by analyzing the growing traffic and learning new Intrusion patterns. The present system runs on an individual host machine. This can be extended to make it a network application where different modules of the same system running on different machines may interact with each other providing distributed detection and protection.

REFERENCES

[1] S. Selvakani and R.S. Rajesh, “Genetic Algorithm for Framing Rules for Intrusion Detection” IJCSNS International Journal of Computer Science and Network Security, Vol. 7 No. 11, November 2007.

[2] Allam Appa Rao, P.Srinivas, B, Chakravarthy, K. Marx & P. Kiran, “A Java Based Network IDS”.

[3] Callos Gershenson, “Artificial Neural Network for Beginners”, c.gershenson@sussea.ac.uk.

[4] C. Sinclairm L. Pierce and S. Matzner, “An application of machine learning to network intrusion detection”, proceedings of 15th Annual Computer Security applications Conference (ACSAC ’99), Phoenix, AZ, pp 371-377, 1999.

[5] James Cannady, “Artificial neural networks for misuse detection”, Proceedings of the 1998 National Information Systems Security Conference (NISSC'98), Arlington, A, 1998.

[6] Mehdi MORADI and Mohammad ZULKERNINE, “A Neural Network Based System for Intrusion Detection and Classification of Attacks”

[7] K. Fox, R. Henning, J. Reed, and R. Simonian, "A neural network approach towards intrusion detection", Proceedings of 13th National Computer Security Conference, Baltimore, MD, pp. 125-134, 1990.

[8] H. Debar, M. Becker, and D. Siboni, “A neural network Component for an intrusion detection system”, Proceedings of 1992 IEEE Computer Society Symposium on Research in Security and Privacy, Oakland, California, pp. 240 – 250, 1992.

[9] Srinivas Mukkamala, “Intrusion detection using neural networks and support vector machine,” Proceedings of the 2002 IEEE International Honolulu, HI, 2002.

[10] R. Cunningham and R. Lippmann, “Improving intrusion detection performance using keyword selection and neural networks,” Proceedings of the International Symposium on Recent Advances in Intrusion Detection, Purdue, IN, 1999.

[11] Sergios Theodorios and Konstantinos Koutroumbas, Pattern Recognition, Cambridge: Academic Press, 1999.

AUTHOR BIOGRAPHY

Mrs. Sonali M. Tidke is working as HOD and Asst. Professor in Maharashtra. She has completed her Post Graduation from Government College of Engineering, Aurangabad, and Maharashtra, India. She has also completed PGDIT for SCDL, Pune, and Maharashtra, India. She has published 2 papers in International Journal in 2012. Her area of research is network security. She is RHCE certified. She has trained students for certification exams like OCJP, OCP etc. She is IEEE member and life time member of ISTE.