Application of Network Forensics for Detection of Web Attack using Neural Network

Sudhakar Parate
Dept. of C.S.E.
G.H.R.C.E. Nagpur

S. M. Nirkhi
Dept. of C.S.E.
G.H.R.C.E.Nagpur

R. V. Dharaskar, Ph.D
Director
M.P.G.I. Nanded, MS. India.

ABSTRACT
Neural network help to determine the network attack such as Denial of Service (DoS), User to Root (U2R), Root to Local (R2L) and Probing. These propose technique based on network forensics and forensics work on the basis of post event. In propose application the post event are log files for that kddcup 99 is used as a standard dataset. This dataset is used as a input to the neural network for detecting the network based attack. In this paper Backpropagation algorithm is used for training the feed forward neural network and also help to identify the evidences of data source as intermediate side and end side. This paper provide the information about how to training and testing is perform in the the neural network and error calculation.

Keywords
Network forensics, Attack Detection, Backpropagation algorithm, Neural Network.

1. INTRODUCTION
Neural network is a powerful database modeling tool that is capable to capture and represent complex input/output relationships. The feed forward back propagation neural network architecture is very popular because it can be applied to many different tasks. Neural network architecture help to examine how it is trained and how it processes the pattern behaviour. The first term indicate "feed foreword" describes how this neural network processes the pattern and recalls patterns. Each layer of the neural network contains connections to the next layer[1][12].

Back propagation algorithm describes how this type of neural network is trained. Back propagation is a form of supervised training. When using a supervised training method the network must be provided with sample inputs and anticipated outputs. These anticipated outputs will be compared against the anticipated output from the neural network. Using these anticipated outputs the "back propagation" training algorithm then takes a calculated error and adjusts the weights of the various layers backwards from the output layer all the way back to the input layer[3]. Artificial neural networks have been demonstrated in number of several applications including speech synthesis, signal processing, business and finance, robotic control, diagnostic problems, medicine, computer vision and many other problems that fall under the category of pattern recognition. The propose system help to detect attack by using samples from standard dataset Kddcup99 for training and some of them for testing the system[6][5].

2. RELATED WORK
In 2006 Ren and Jin was propose the first general process model of network forensics, which is comprises with the five steps: capture, copy, transfer, analysis, investigation and presentation. The techniques used for the implementation is the feed forward neural network [6].

McCulloch and Pitts (1943) developed models of neural networks are based on their understanding of neurology. These models are made up of several assumptions about how neurons worked. Their networks were based on simple neurons which are considered to be binary devices with fixed thresholds [7]. The perceptron is the one of the earliest neural networks which are invented at the Cornell Aeronautical Laboratory in 1957 by Frank Rosenblatt. In 1960 Rosenblatt demonstrated the Mark I Perceptron. The Mark I was the first machine which is used to "learn" to identify optical patterns. that were setup manually. Because the multilayered perceptron neuron did not have the ability to learn it was very limited when compared with the infinitely more flexible. In 1969 Minsky and Papert wrote a book in which they described the limitations of single layer Perceptrons. Al-Rashdan [4] has proposed an intelligent model using Hybrid Artificial Neural Networks, supervised and unsupervised learning capabilities to classify and / or detect network intrusions from the KDDCup'99 dataset. The system operation are divided into three categories : Input Data Collection and Preprocessing, Training, and Detection stage. [2][14].

Klopf (A. Henry Klopf) in 1972 developed a basis for learning in artificial neurons based on a biological principle for neuronal learning called heterostasis. Werbos (Paul Werbos 1974) developed and used the back-propagation learning method, however several years passed before this approach was popularised. Backpropagation nets are probably the most well known and widely applied of the neural networks [4][12].

Dorothy Denning (1987) was proposed an intrusion detection model which became a sign in the research in this area. Neural Network offers the potential to resolve a number of the problems encountered by the other current approaches to attack detection. Artificial neural networks (ANN) are alternatives. The first advantage of a neural network in the attack detection would be the flexibility that the network would provide[5].

3. PROPOSED SYSTEM
In the propose system, acquire the data from the data source. In this phase privacy is the major concern . KDDcup99 is a standard dataset used as a evidence. Then pre-process the data before it use as a input to the system. The system should be trained with the backpropogation algorithm. And it will help to detect web attack using neural network.

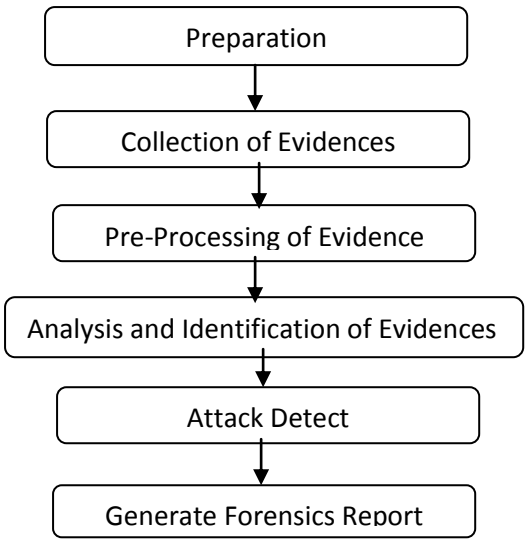

Figure 1: Network forensics investigation process

3.1 A Feed Forward Neural Network

The feed forward neural network architecture used in the propose system. It consists of one input, one hidden and one output layer. The general architecture is shown in the Figure 2.

Feed forward neural network is similar to the types of neural networks that are examined. Just like many other neural network types the feed forward neural network begins with the input layer. This input layer is connected to a hidden layer and this hidden layer can then be connected to another hidden layer or directly to the output layer. There will be any number of hidden layers so long as at least one hidden layer is provided[10][7].

3.2 The Structure of a Feed Forward Neural Network

A feed forward neural network differs from the neural networks previously examined. Figure shows a typical feed forward neural network with a single hidden layer. There are numbers of input with different weight for reducing the error. The input vector is interconnected with hidden layer and then it produces the output.

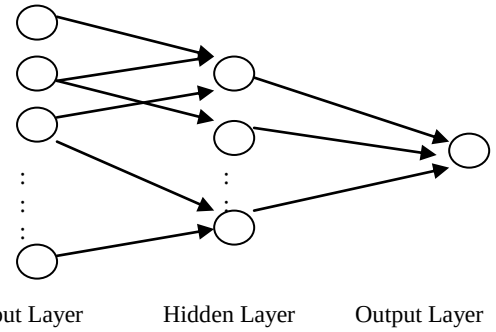

Figure 2: Feed Forward Neural Network

3.3 The Input Layer

In Neural network the input layer is the conduit through which the external environment presents a pattern to the neural network. Once a the pattern is presented to the input later of the neural network and the output layer will produce another pattern. The input layer represents the condition for which we are training the neural network[7][8]. Every input neuron should represent some separate variable that has an influence over the output of the neural network. Neural network also help to process numeric data as well as non numeric data to the neural network. It must develop a process that normalizes this data to a numeric representation[9].

3.4 The Hidden Layers

In Hidden layers there are really two decisions that must be made. The first is how many hidden layers to actually have in the neural network. And secondly, we can determine how many neurons will be in each of these layers. Here we will first examine how to determine the number of hidden layers to used with the neural network [6]. There is currently not having any theoretical reason to use neural networks with any more than two hidden layers. Many practical problems there are no reason to use any more than one hidden layer[7].

3.5 The Number of Neurons in the Hidden Layers

The number of hidden neurons in layers is a very important part of deciding your overall neural network architecture. These layers do not directly interact with the external environment these layers. Both the number of hidden layers and number of neurons in each of these hidden layers must be considered. These few neurons in the hidden layers will result in something called underfitting. Underfitting occurs when there are too few neurons in the hidden layers to the adequately detect the signals in a complicated data set. By using too many neurons in the hidden layers could result in several problems. The total amount of training time can increase enough therefore it is impossible to adequately train the neural network. Some of them are summarized as follows.

- The number of hidden neurons will be in the range between the size of the size of the output layer and input layer.
- The total number of hidden neurons should be 2/3 of the input layer size, with addition of the size of the output layer.
- The number of hidden neurons will be less than twice the input layer size.

The number of hidden neurons is then increased and the process is repeated so long as the overall results of the training and testing improved[12]. The "forward selection method" is summarized.

3.6 The Output Layer

The output layer of the neural network is actually presents a pattern to the external environment. pattern is presented by the output layer can be directly traced back to the input layer. The number of a output neurons should directly related to the type of work that the neural network is perform. The number of neurons to use in your output layer that must consider the intended use of the neural network. If the neural network is to be used to classify items into groups, then it is often preferable to have the one output neurons for each groups that the item is to be assigned into neural network[7][4].

4. TRAINING NEURAL NETWORKS

The individual inputs that is neuron make up a neural network are interconnected through the synapses. These connections permit the neurons to signal each other as information is

processed. all connections are not equal. Each connection is assigned a connection weight. These weights are what determine the output of the neural network. Training is the process for which these connection weights are assigned. Most of the training algorithms begin by assigning random numbers to the weight matrix. Then the validity of the neural network is examined. Next the weights are adjusted based on how valid the neural network performed. And this process is repeated until the validation error is within an acceptable limit. There are many ways to train neural networks [12]. Neural network training methods generally classified into the categories of supervised, unsupervised and various hybrid approaches [9].

Supervised training is accomplished by giving the neural network with a set of sample data it may be KDDCUP 99 along with the anticipated outputs from each of these samples. Supervised training is the most common form of neural network training [1]. As supervised training proceeds the neural network is taken through several iterations, or epochs, until the actual output of the neural network matches the anticipated output, with a reasonably small error[4].

Each iteration is one pass through the training samples. Unsupervised training is similar to supervised training except that are no anticipated outputs are provided. Unsupervised training usually occurs when the neural network is to classify the inputs into several groups. The training progresses through many epochs, it just like as in supervised training. As training progresses the classification groups are "discovered" by the neural network [6][14]. It is very important to understand how to properly train a neural network. There are several methods of neural network training, including backpropagation, simulated annealing, and genetic algorithms. Once the neural network is trained, it must be validated.

5. TESTING NEURAL NETWORKS

When the neural network has been trained it must be evaluated to see that is it ready for actual use. This final step is important so that it can be determined if additional training is required. In order to correctly validate a neural network validation data must be set aside that is completely separate from the training data[8].Consider an example, a classification network that must group elements into three different classification groups. If we are provided with 10,000 sample elements. For this sample data the group that each element should be classified into is known. For such a system you would divide the sample data into two groups of 5,000 elements. The first group would form the training set. When the network has been properly trained then the second group of 5,000 elements would be used to test the neural network. It is very important that a separate group always be maintained for testing. First training a neural network with a given sample set and also using this same set to predict the anticipated error of the neural network a new arbitrary set will surely lead to bad results. The error achieved using the training set will almost always be substantially lower than the error on the new set of sample data[10][12].

6. ERROR CALCULATION

To calculate error is an important aspect of any neural network. When the neural network is supervised or unsupervised, an error rate must be calculated. The goal of virtually all training algorithms is to minimize the error. It will examine how the error is calculated for a supervised neural network. This section we will help to examine an error calculation method that can be employed by supervised training [11]. For supervised training there are two components to the error that must be considered. First, we must calculate the error for each of the training sets as they are processed. Secondly we must take the average across each sample for the training set[14][15].

6.1 Root Mean Square (RMS) Error

The Root Mean Square error which allows the neural network to know when enough training has taken place. The RMS error can be calculated at any time after the "calcErrors" method has been called. This process is necessary because the RMS is an average to take the average error across all training set elements, for that you must know the size of the training set. The RMS error is then calculated by dividing the global error by which the product of the training set length and the number of output neurons. The square root of this ratio produces the RMS. At last after the RMS error has been calculated the global Error is set back to zero. This process is required so that it can help to begin accumulating for a new error[13].

7. CONCLUSION

The proposed system used for identification of network attack using backpropagation neural network and it works to improve the detection rate and high accuracy. The neural network has been trained by using backpropagation algorithm. Finally the proposed systems generate the forensic report which having information about attack scenario, attack log details and analysis of evidences that will help for adding an investigation.

8. REFERENCES

[1] Reyadh Shaker Naoum, Namh Abdula Abid and Zainab Namh Al-Sultani, "An Enhanced Resilient Backpropagation Artificial Neural Network for Intrusion Detection System", IJCSNS International Journal of Computer Science and Network Security, VOL.12 No.3, March 2012.

[2] Yang Xiang,Ke Li, Wanlei Zhou, "Low-Rate Ddos Attacks Detection And Traceback By Using New Information Metrics", IEEE transactions on information forensics and security, vol. 6, no. 2, pp 426-437, june 2011.

[3] Igino Corona, Davide Ariu And Giorgio Giacinto,"HMM-Web: A Framework For The Detect ion Of Attacks Against Web Applications", IEEE International conference on communication, pp1-6, 2009.

[4] Ying Xuan, Incheol Shin, My T. Thai, "Detecting Application Denial-Of-Service Attacks: A Group-Testing-Based Approach",IEEE transactions on p arallel and distributed systems, vol. 21, no. 8, pp 2103-1216, august 2010.

[5] IftikharAhmad, Azween B Abdullah, Abdullah S Alghamdi, "Remote to Local Attack Detection Using Supervised Neural Network",5th International Conference for Internet Technology and Secured Transactions, Nov 8-11, 2010,

[6]Atul Kant Kaushik , R. C. Joshi, "Network Forensic System for ICMP Attacks", International Journal of Computer Applications (0975 – 8887) Volume 2 – No.3, May 2010.

[7]Aida O. Ali, Ahmed I. Saleh, Tamer R. Badawy, "Intelligent Adaptive Intrusion Detection Systems Using Neural Networks", International Journal of Video & Image Processing and Network Security IJVIPNS-IJENS Vol: 10 No: 01.

[8] Sindhu. K. K , Dr. B. B. Meshram, "A Digital Forensic Tool For Cyber Crime Data Mining", IRACST Engineering Science And Technology: An International Journal (ESTIJ), ISSN: 2250-3498, Vol.2, No.1, 2012.

[9]Muna Mhammad T. Jawhar and Monica Mehrotra, "Design Network Intrusion Detection System using hybrid Fuzzy-Neural Network", International Journal of Computer Science and Security, Volume (4): Issue (3) -2009.

[10] Ondrej Linda, Todd Vollmer, Milos Manic,"Neural Network Based Intrusion Detection System for Critical Infrastructures", International Joint Conference on Neural Networks-2009.

[11] Liang Xie, Sencun Zhu,"Message Drop p ing Attacks In Overlay Networks:Attack Detection And Attacker Identification", IEEE 2006.

[12] JeffHeaton," Programming Neural Networks in Java", November 16, 2005.

[13] Andreas Makridakis, Elias Athanasopoulos, Spiros Antonatos, Demetres Antoniades, Sotiris Ioannidis, And Evangelos P. Markatos, " Understanding The Behavior Of Malicious Applications In Social Networks", IEEE 2010.

[14] Tich Phuoc Tran, Longbing Cao , Dat Tran , Cuong Duc Nguyen, "Novel Intrusion Detection using Probabilistic Neural Network and Adaptive Boosting", (IJCSIS) International Journal of Computer Science and Information Security,Vol. 6, No. 1, 2009.

[15] Nidal Qwasmi, Fay y az Ahmed, Ramiro Liscano, "simulation of ddos attacks on p2p networks" , IEEE International conference on HPCC, pp 610-614, 2011.

[16] Ram Prasad Viswanathan, Youssif Al-Nashif, Salim Hariri," Application Attack Detection System (AADS):An Anomaly Based Behavior Analysis Approach", IEEE 2011.