ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 3, Issue 3, March 2015

Detecting Network Intrusion Using BPA & RBF Neural Network Algorithms

Vinit Kumar Gunjan[1], Vijayalata Y[2]

Calorx Teacher's University, Ahmedabad, India [1]

Professor, Department of IT, GRIET, Hyderabad, India [2]

ABSTRACT: Intrusion detection is one of the core technologies of computer security which is capable of providing security to network of computer systems. Due to the expansion of high-speed internet access, the need of secure and reliable networks become more critical. The computer networks are facing intelligent attacks has increased drastically in recent era. This paper focuses on two type's classification types: a single class (normal, or attack), and a multi class (normal, DOS, PRB, R2L, U2R), where the category of attack is detected by the combination of Back Propagation Neural Network (BPA) and Radial Basis Function Neural Network. Most of existing intrusion detection systems use all features in the network packet to look for known intrusive patterns. A well-defined feature extraction algorithms makes the classification process more effective and efficient. The feature extraction step aims at representing patterns in a feature space where the attack patterns are attained. In this paper, a combination of Back Propagation Algorithm Neural Network along with Radial Basis Function Neural Networks are used for detecting intrusion activity. The testing has been performed on KDD-99 dataset.

KEYWORDS: Intrusion Detection System, KDD-99, IDPS, IDS, IDS Neural Network, RBF IDS, BPA IDS

I. INTRODUCTION

With the tremendous growth of network based service and sensitive information on networks, network security is getting more and more importance than ever. Intrusion [Alireza Osareh and BitaShaadgar, 2008] are in many forms: attackers accessing a system through the internet or insider attackers; authorized users attempting to gain and misuse non-authorized privileges. Intrusion consists of the set of activities which are threatto the integrity, availability and confidentiality of a network resource. Intrusion detection [Tich Phuoc Tran, et al, 2009] is the process of monitoring the events occurring in a computer system or network [Aida O. Ali, et al, 2010] and analysing them for signs of intrusion.

Classification of Attack Detection

Attack/invasion Detection: It is used to detect the unauthorized access by outsiders or intruders.

Misuse Detection: Tries to detect misuse by insiders like users trying to access services in the internet, bye-passing security directives. By using prior knowledge, Misuse detection tries to detect attacks based on the specific pattern of known attacks.

Anomaly Detection:

Tries to detect abnormal states within a network. Anomaly intrusion detection uses normal usage behaviour patterns to identify the intrusion. It also constructs the standard usage patterns are constructed from the statistical measures of the system features. If there is any deviation from the constructed standard behaviour, the behaviour of the user is observed and the same is detected as intrusion. There are many types of IDSs architecture and the same is divided into two groups based on the type of events they monitor, and the way in which they are deployed. They are: Host Based Intrusion Detection System (HIDS) & Network Based Intrusion Detection System (NIDS).

II. RELATED WORKS

Zhang, et al, 2005, proposed a hierarchical IDS framework using RBF to detect both anomaly and misuse detection. A serial hierarchical IDS identifies misuse detection accurately and identifies anomaly detection adaptively. The purpose of parallel hierarchical IDS is to improve the performance of serial hierarchical IDS. Both the systems train themselves for new types of attacks automatically and detect the intrusion activity in real time.

Meera Gandhi et al, 2009, propose a Polynomial Discriminant Radial Basis Function (PRBF) for intrusion detection to achieve robustness and flexibility. Based on several models with different measures, PRBF makes the final decision of whether current behaviour is abnormal or not. Experimental results with some real KDD data show that the proposed fusion produces a viable intrusion detection system.

Ray-I Changet al.,2007 proposed a learning methodology towards developing a novel intrusion detection system(IDS) by BPN with sample-query and attribute-query. The proposed method is tested by a bench mark intrusion datasettoverify its feasibility and effectiveness. Results showed that choosing attributes and samples will not only have impact on the performance, but also on the overall execution efficiency.

Ahmed Fares, et al., 2011, proposed two systems to identify intrusion, the first system was the back propagation neural network intrusion detection system (BPNNIDS) and the second system was the RBF neural network intrusion detection system which was capable of classifying the attacks into two classification types: a single class (normal, DOS, PRB, R2L, U2R). The model was tested against traditional and other machine learning algorithms using a common dataset KDD99 & DARPA 98.

III. MATERIALS & METHODOLOGIES

A. KDD CUP 99 DATASET DESCRIPTION

The KDD Cup 1999 dataset has been used for the evaluation of anomaly detection methods. The KDD Cup 1999 contains 41 features and is labelled as either normal or an attack, with exactly one specific attack type.

DataCollection:KDD Cup 1999 dataset has the different types of attacks: back, buffer_overflow, ftp_write, guess_passwd, imap, ipsweep, land, loadmodule, multihop, neptune, nmap, normal, perl, phf, pod, portsweep, rootkit, satan, smurf, spy, teardrop, warezclient, warezmaster. These attacks can be divided into 4 groups.

The table 1 shows the category wise list of attacks.

DoS	**R2L**	**U2R**	**Probe**
Back Land Neptune Pod Smurf Teardrop	ftp_write guess_passwd imap multihop phf spy warezclient warexmaster	Buffer_overflow Loadmodule Perl Rootkit	Ipsweep Nmap Portsweep Satan

Table. 1 List of Attacks

S.No.	KDD Patterns
1	0,udp,private,SF,105,146,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,254,1.00,0.01,0.00,0.00,0.00,0.00,0.00,0.00,normal
2	0,udp,private,SF,105,146,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,254,1.00,0.01,0.00,0.00,0.00,0.00,0.00,0.00,normal
3	0,udp,private,SF,105,146,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,254,1.00,0.01,0.00,0.00,0.00,0.00,0.00,0.00,normal
4	0,udp,private,SF,105,146,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,254,1.00,0.01,0.00,0.00,0.00,0.00,0.00,0.00,snmpgetattack
5	0,udp,private,SF,105,146,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,1.00,0.00,0.00,255,254,1.00,0.01,0.01,0.00,0.00,0.00,0.00,0.00,

Table. 2 Sample KDD Data

B. Neural Network IDS

Two ANN networks are used: BPA and the RBFnetwork.

Back Propagation neural network (BPNN)

The BPNN [Reyadh Shaker Naoum, et al, 2012, Meera Gandhi, et al, 2008] searches for weight values that minimize the total error of the network over a set of training examples. It consists of the repeated presentation of two passes: a forward pass andabackward pass. In the forward pass, the network is activated and the error of each neuron of the output layer is computed. In the backward pass, the network error is used for updating the weights. This process is more complex, because hidden nodes are not directly linked to the error but are linked through the nodes of the next layer. Therefore, starting at the output layer, the error is propagated backwards through the network, layer by layer. This is achieved by recursively computing the local gradient of each neuron.

The training algorithm BPA are as follows:

1. Initialize the weights of the network randomly.
2. Present a training sample to the network where, in our case, each pattern consists of 41 features.
3. Compare the network's output to the desired output. Calculate the error for each output neuron.
4. For each neuron, calculate what theoutput should have been, and a scaling factor i.e. how much lower or higher the output must be adjusted to match thedesired output. This is the local error.
5. Adjust the weights of each neuron to lower the local error.

$$W_N=W_N+\Delta W_N$$

With W_N computed using generalized delta rule

1. Repeat the process from step 3 on the neurons at the previous level.

Training Phase

A connection in the KDD-99 dataset is represented by 41 features. The features in columns2, 3, and 4 in the KDD99 dataset are the protocol type, the service type, and the flag, respectively. Thevalue of the protocol type may be tcp, udp, or icmp;the service type could be one of the different network services such as http and smtp; and the flag has 11 possible values such as SF or S2.

Weight Updating Methods

The neural network maps the input domainsonto output domains. The inputs are packet parameters and the outputs are classification of attacks information. The combination of input and output constitutes a pattern. During training of ANN, the network learns the training patterns by a weight updating algorithm. The training of ANN is stopped when a desired performance index of the network is reached. The weights obtained at this stage are considered as final weights. During implementation of ANN for intrusion detection, the data coming from the network are transformed with the full weights obtained during the training of ANN. Every output ofthe network is checked. If the outputs are within the desired values detection is enabled.

Radial Basis Function network (RBFN)

A Radial basis function (RBF) network is a special type of neural network that uses a radial basisfunction as its activation function. A Radial BasisFunction (RBF) neural network has an input layer, a hidden layer and an output layer. The neurons in the hidden layer contain radial basis transfer functions whose outputs are inversely proportional to the distance from the centre of the neuron. In RBF networks, the outputs of the input layer are determined by calculating the distance between the network inputs and hidden layer centres. The second layer is the linear hidden layer and outputs of this layer are weighted forms of the input layer outputs. Each neuron of the hidden layer has a parameter vector called centre.The RBF is applied to the distance to compute the weight for each neuron. Centres are chosen randomly from the training set.
The following parameters are determined by the training process:

1. The number of neurons in the hidden layer.
2. The center of each hidden layer RBF function.
3. The radius of each RBF function in each dimension.
4. The weights applied to the RBF function outputs as they are passed to the summation layer.

The BPRABF method have been used to train the network.

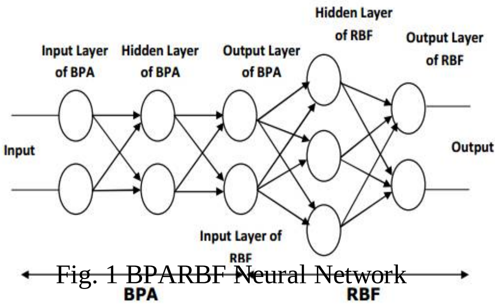

Fig. 1 BPARBF Neural Network

The input layer provides elements of the input vector to all the hidden nodes. The nodes in the hidden layer holds the RBFs centers, computes the basis function to the Euclidean distance between the input vector and its centers. The nodes of hidden layer generates a scalar value, depends upon the centers it holds. The outputs of the hidden layer nodes are passed to the output layer via weighted connections. Each connection between the hidden output layers is weighted with the relevant coefficient. The node in the output layer sums its inputs to produce the network output.

Training RBF

Step 1: Initialize number of Inputs
Step 2: Create centers=Number of training patterns
Step 3: Calculate RBF as exp (-X) where X= (patterns-centers).
Step 4: Calculate Matrix as G=RBF and A=GT*G.
Step 5: Calculate B=A-1 and E=B * G*T.
Step 6: Calculate the final weight as F= (E*D) and store the final weights in a File.
Testing RBF
Step 1: Read output of BPA
Step 2: Calculate RBF as exp (-X) where X=(pattern- centers)
Step 3: Calculate Matrix as G=RBF
Step 4: Calculate Final value=Final weight * G.
Step 5: Classify the intrusion as an attack or normal.

IV. Results & Discussion

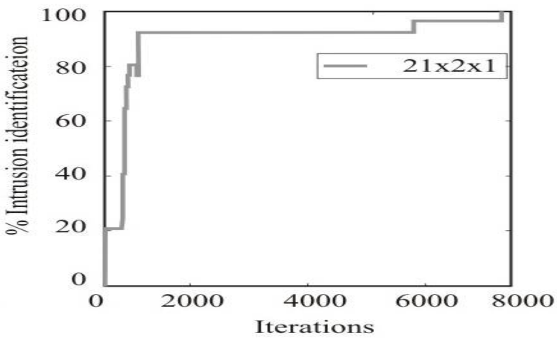

Fig. 2 Back Propagation Network for Intrusion Detection

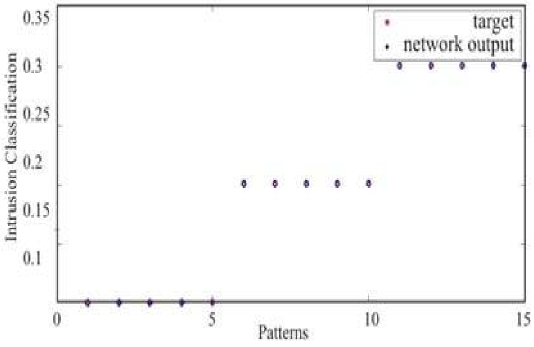

Fig. 3 Radial Basis Function for Intrusion Detection

V. CONCLUSION

Current intrusion detection systems (IDS) examine data features to detect intrusion or misuse patterns. The purpose of this paper is to present combination of BPA with RBF for intrusion.

REFERENCES

[1]. Ahmed H. Fares, Mohamed I. Sharawy, 2011, Intrusion Detection: Supervised Machine Learning, Journal of Computing Science and Engineering, Vol.5, No.4, pp.305-313.

[2]. Aida O. Ali, Ahmed saleh, Tamer Ramdan, 2010, Multilayer perceptrons networks for an Intelligent
Adaptive intrusion detection system, IJCSNS International Journal of Computer Science and Network Security, Vol.10, No.2, pp.275-279.

[3]. Alireza Osareh, Bita Shadgar, 2008, Intrusion Detection in Computer Networks based on Machine Learning Algorithms, International Journal of Computer Science and Network Security, Vol.8, No.11, pp.15-23.

[4]. Asmaa Shaker Ashoor and Sharad Gore, 2011, Importance of Intrusion Detection System, International Journal of Scientific and Engineering Research, Vol.2, Issue 1, pp.1-4.

[5]. Bahrololum M., Salahi E., Khaleghi M., 2009, Anomaly Intrusion Detection Design Using Hybrid Of Unsupervised And Supervised Neural Network,International Journal of Computer Networks and Communications (IJCNC), Vol.1, No.2, pp.26-33.

[6]. Meera Gandhi, Srivatsava S.K., 2008, Application of Back propagation Algorithm in Intrusion Detection in Computer Networks, International Journal of Soft computing, Vol.3, No.4, pp.277-281.

[7]. Meera Gandhi, Srivatsa S.K, 2009, Polynomial Discriminant Radial Basis Function for intrusion detection, International Journal of Cryptography and Security, Vol.2, No.1, pp.25-32.

[8]. Ray-I Chang, Liang-Bin Lai, Wen-De Su, Jen-Chieh Wang, Jen-Shiang Kouh, 2007, Intrusion Detection by Back propagation Neural Networks with Sample-Query and Attribute-Query, International Journal of Computational Intelligence Research. Vol.3, No.1, pp.6-10.

[9]. Reyadh Shaker Naoum, Namh Abdula Abid, Zainab Namh Al-Sultani, 2012, An Enhanced Resilient Backpropagation Artificial Neural Network for Intrusion Detection System, International Journal of Computer Science and Network Security, Vol.12, No.3, pp.11-16.

[10]. Samaneh Rastegari, Iqbal Saripan M., Mohd Fadlee A., Rasid, 2009, Detection of Denial of Service Attacks against Domain Name System Using Neural Networks, IJCSI International Journal of Computer Science Issues, Vol.6, No.1, pp.23-27.

[11]. Tich Phuoc Tran, Longbing Cao, Dat Tran, Cuong Duc Nguyen, 2009, Novel Intrusion Detection using Probabilistic Neural Network and Adaptive Boosting, International Journal of Computer Science and Information Security, Vol.6, No.1, pp.83-92.

[12]. Zhang Chunlin, Ju Jiang, Mohamed Kamel, 2005, Intrusion detection using hierarchical neural networks, Pattern Recognition Letters 26, Vol.9, No.45, pp.779– 791.