Intrusion Detection System Based on Neural Networks Using Bipolar Input with Bipolar Sigmoid Activation Function

Adel S. Issa
College of Eduction
University of Duhok

Adnan M. Abdulazeez
College of Eduction
University of Duhok

Received on: 24/10/2010 **Accepted on: 16/03/2011**

ABSTRACT

Vulnerabilities in common security components such as firewalls are inevitable. Intrusion Detection Systems (IDS) are used as another wall to protect computer systems and to identify corresponding vulnerabilities. The purpose of this paper is to use Backpropagation algorithm for IDS by applying bipolar input "input is represented as (1, -1)", and bipolar sigmoid activation function.

The KDD Cup 99 dataset is used in this paper. Number of train dataset is 4947 connection records, and number of test dataset is 3117 connection records. The results of the proposed method show that the PSP is 88.32 and CPT equal to 0.286.

Keywords: Firewalls, Intrusion Detection Systems (IDS), Backpropagation algorithm, bipolar sigmoid function, KDD Cup 99 dataset.

نظام كشف التطفل القائم على الشبكات العصبية باستخدام الإدخال Bipolar مع دالة Bipolar Sigmoid

عادل عيسى
كلية التربية، جامعة دهوك

عدنان عبد العزيز
كلية التربية، جامعة دهوك

تاريخ استلام البحث: 2010/10/24 **تاريخ قبول البحث: 2011/03/16**

الملخص

إن وجود الثغرات في اغلب مكونات نظم الحماية مثل نظم الجدران النارية هو أمر محتوم. ولهذا استخدمت نظم كشف التطفل (IDS) كجدران ثانوية لحماية أنظمة الحاسبات وتحديد الثغرات. هذا البحث يهدف إلى استخدام خوارزمية (Backpropagation) لبناء نظام IDS وذلك باستخدام إدخالات الـ Bipolar (الإدخالات تمثل بـ (1 ,1-)، وكذلك استخدام دالة Bipolar Sigmoid. اعتمد البحث على بيانات الـ KDD Cup 99. عدد السجلات التي أُستخدمت في تدريب الشبكة هي 4947 سجل، وعدد السجلات التي أُستخدمت لاختبار الشبكة هي 3117 سجل. أظهرت النتائج أن قيمة الـ PSP تساوي 88.32 وقيمة الـ CPT تساوي 0.286.

الكلمات المفتاحية: نظم الجدران النارية، نظم كشف التطفل (IDS)، خوارزمية (Backpropagation)، نظام IDS، دالة Bipolar Sigmoid، بيانات الـ KDD Cup 99.

1. Introduction

Reliance on Internet and world wide connectivity has increased the potential damage that can be inflicted by attacks launched over Internet against remote systems. Successful attacks inevitably occur despite the best security precautions. Therefore, intrusion detection has become an essential component of computer security to detect these attacks with the aim of preserving systems from widespread damages and identifying vulnerabilities of the intruded system [3].

There are two main approaches to the design of IDSs [1] and [2] namely; misuse and anomaly systems. In a misuse detection based IDS, intrusions are detected by looking for activities that correspond to known signatures of intrusions or vulnerabilities. On the other hand, anomaly detection based IDS detect intrusions by searching for abnormal network traffic. The abnormal traffic pattern can be defined either as the violation of accepted thresholds for frequency of events in a connection or as a user's violation of the legitimate profile developed for his/her normal behavior.

Several soft-computing approaches were proposed in recent years for the development of IDS. Mehdi Moradi and Mohammad Zulkernine 2004 [8], the paper presents a NN approach to intrusion detection. A multi-layer perceptron is used for intrusion detection based on an off-line analysis approach and applying the early stopping validation method on the proposed NN. Yacine Bouzida and F. Cuppens 2006 [12] proposed two different techniques for anomaly intrusion namely (NN) and (DT) in order to detect new attacks that are not present in the training data set. They improve them for anomaly intrusion detection and test them over the KDD Cup 99 data sets and over real network traffic in real time. Yuehui Chen et al., 2007 [13] proposed an IDS model based on a general and enhanced Flexible Neural Tree (FNT). Based on the predefined instruction/operator sets, the framework allows input variables selection. Over layer connections and different activation functions for the various nodes involved. Arman Tajbakhsh, et al., 2008 [3] proposed a new intrusion detection framework based on classification algorithm using fuzzy association rules for building classifiers. The fuzzy association rulesets are exploited as descriptive models of different classes. The method proposed to speed up the rule induction algorithm. Rachid Beghdad 2008 [9] aimed to determine which of the NN classifies well the attacks and leads to a higher detection rate of each attack. The paper focused on two classification types of records: a single class (normal, or attack), and a multiclass, where the category of attack is also detected by the NN. Five different types of NNs were tested: Multi-Layer Perceptron (MLP), generalized feed forward (GFF), radial basis function (RBF), self-organizing feature map (SOFM), and principal component analysis (PCA) NN.

This paper deals with a Backpropagation algorithm to build a Neural Network (NN). For the input of the NN, bipolar inputs "the input will represent as (1, -1)", and bipolar sigmoid as activation function are proposed.

Besides this introduction, the rest of the paper is organized as follows:

Section 2 presents an introduction to neural network and Backpropagation (BP) algorithm. Section 3 describes the motivation and the proposed work. The section also deals with the dataset, evaluation criteria, and the features used for classifying network connection records in this study. Section 4 presents the experimental results. Section 5 gives some conclusions.

2. Introduction to Neural Network and Backpropagation Algorithm

2.1. Neural Network (NN)

An Artificial Neural Network (ANN) is an information processing system that is inspired by the way biological nervous systems, such as the brain, process information. It is composed of a large number of highly interconnected processing elements working with each other to solve specific problems. Each processing element is basically a summing element followed by an activation function. The output of each neuron (after applying the weight parameter associated with the connection) is fed as the input to all neurons in the next layer. The learning process is essentially an optimization process in which the parameters of the best set of connection coefficients (weighs) for solving a problem are found and include the following basic steps [11, 5, 7]:

- Present the NN with a number of inputs (vectors each representing a pattern).

- Check how closely the actual output generated for a specific input matches the desired output.

2.2. Backpropagation Algorithm

In order to train an NN to perform some tasks, the weights must be adjusted for each unit in such a way that the error between the desired output and the actual output is reduced. This process requires that the NN computes the error derivative of the weights. In other words, it must calculate how the error changes as each weight. The BP (Rumelhart and McClelland, 1988) [6] is the most widely used method for error determination.

3. The Proposed Work

ANN will be exploited to solve a multi-class problem of intrusion detection using a classic multi-layer feed-forward NN trained with the BP algorithm to predict intrusions. In this work, the aim is to design NN with three layers (input layer, one hidden layer, and output layer), for the input layer, 83 inputs are proposed; each is represented by 1 or -1. The number of neurons in hidden layer will be determined during the experiments, while the output layer is represented with five neurons which is equal to the number of the total classes corresponding to the five classes considered in the KDD Cup 99 contest (Normal, Probing, DoS, U2R and R2L, respectively). The activation function that used in this work is bipolar sigmoid, which is in the range of (-1, 1) and is defined as:

$$f(x) = \frac{2}{1+\exp(-x)} - 1 \qquad \ldots(1)$$

Functionality of the proposed system is divided into five phases they are:

- Input Data and Partition process.
- Labeling continuous values.
- Convert each value to 1 or -1 (vector of 83 input).
- Training
- Classification.

In the first phase, the continuous attributes will be partitioned to 3 groups (A, B, C) by applying K_mean algorithm on input data. In the second phase, each continuous value will be assigned to one of this three groups (A or B or C), in the third phase, the data is converted into suitable input data by convert each value to (1 or -1) and make input vector of 83 value, the number of input vector is calculated as follows:

For each attribute, appropriate number of neurons is assigned depending on the number of bits that can represent them. For example for protocol_type (tcp, udp, icmp), there are 2 inputs, say i_0, i_1 assigned to this attribute, each unit is initialized to -1. If the protocol_ type of the current connection record is “tcp” then i_0 is set to 1, if it is “udp” then i_1 is set to 1, and so on. For service attribute, there are 66 different values which need 7 digits to represent them, which are assigned to 7 neurons. While for flag attribute will be assigned to 4 neurons because, there are 11 different values. For the Boolean attribute which takes (0 or 1), only one neurons is assigned and can be represent by one digit. For each remaining continuous attributes the number of neurons will be 2 neurons. This because each continuous attributes will be grouped to three clusters (A, B, and C) so, only 2 digits to represent this three groups are needed. The final number of the input layer will be 83 input neurons, so that the input is given to BP algorithm. In the training phase, the system gathers knowledge about the normal and attacks from the preprocessed input data, and stores the acquired knowledge. In classification phase, the system detects normal behavior or specific attack based on the knowledge, which is achieved during the training phase. The main task is to generalize and classify each connection record to one of the five classes considered in the KDD Cup 99 dataset (Normal, Probing, DoS, U2R and R2L). Figure (1) shows the block diagram of the proposed system.

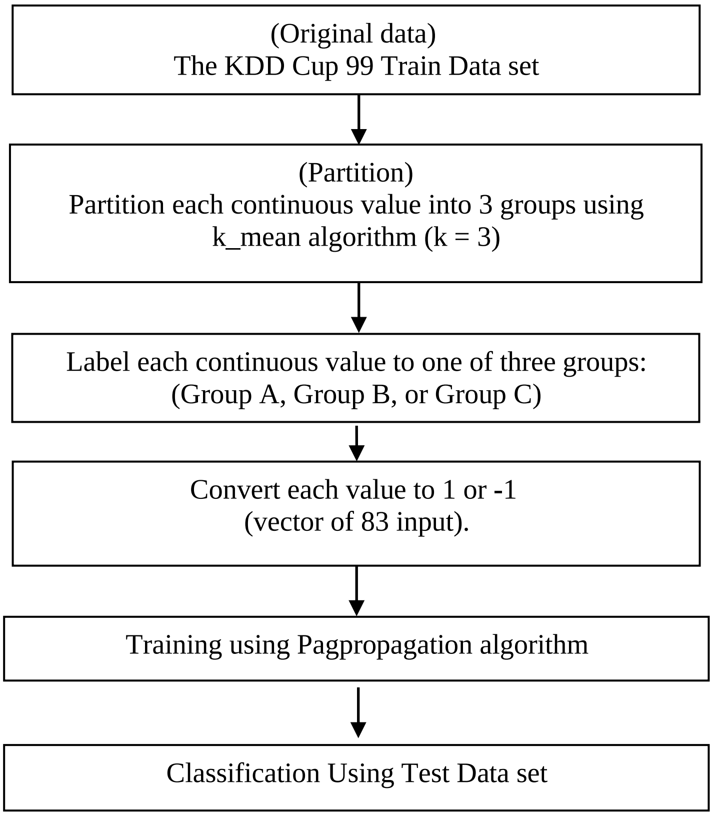
(Original data)
The KDD Cup 99 Train Data set
(Partition)
Partition each continuous value into 3 groups using k_mean algorithm (k = 3)
Label each continuous value to one of three groups: (Group A, Group B, or Group C)
Convert each value to 1 or -1 (vector of 83 input).
Training using Pagpropagation algorithm
Classification Using Test Data set

Figure (1). The block diagram of the proposed system

3.1. Input Data (KDD Cup 99 Data Set)

As mentioned before, KDD Cup 99 dataset [10] is used to evaluate the proposed framework for intrusion detection. This dataset is a common benchmark for evaluation of intrusion detection techniques. The '10%KDD Cup 99' train dataset is about 494020 connection record and test dataset is about 311028 this data is too large to use in this paper. For this reason a subsets of KDD Cup 99 train and test datasets are extracted for this work and shown in Table (1) that describes different attack types and their corresponding occurrence number in the training and test data, respectively. The number of training data is equal to 4947 and test data is equal to 3117, which are selected randomly from KDD Cup 99 dataset. From Table (1) Probing (41; 42) means that the number of records in train dataset of attack Probe is equal to (41 connection records), while the number of records in test dataset of this attack is equal to (42 connection records).

Table (1). Different attack types and their corresponding occurrence number respectively in the training and test dataset.

Normal(973;606)			
Probing (41; 42)	DoS(3915 ; 2299)	U2R(5 ; 10)	R2L(13 ; 160)

ipsweep (12 ; 3), Mscan (0 ; 11), Nmap (2 ;1), Portsweep(11 ; 4), Saint (0;7), Satan (16 ; 16).	apache2 (0 ; 8), back (22 ; 11), land (0 ; 0), mailbomb (0 ; 50), Neptune(1072;580) processtable (0;8), Pod (3 ;1), udpstorm (0 ; 0). Smurf (2808 ;1641), Teardrop (10 ; 0),	buffer_overflow(3 ;1), httptunnel (0 ;3), loadmodule (0; 0), perl (0; 0) rootkit (2;2), xterm (0 ; 2). Ps (0 ; 2), Sqlattack (0 ; 0)	ftp_write (0 ; 0), imap (0 ; 0), guesspasswd(2 ; 44), named (0 ; 0), multihop (0 ; 0), phf (0 ; 0), sendmail (0; 0), snmpgetattack(0;77), snmpguess (0 ;24), spy(0 ; 0), warezclient (10 ; 0), worm (0 ; 0), warezmaster (1; 15), xsnoop (0 ; 0). xlock (0 ; 0),

Each record contains values of 41 independent variables (fields) describing the different features of the connection, and the value of the dependent variable labeled as either normal, or as an attack, with exactly one specific attack type. The sample of four connection records corresponding to the attack types is shown for each type of attack as:

```
0,tcp,http,SF,334,1684,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,9,0.00,0.00,0.00,0.00,1.00,0.00,0.3
3,0,0,0.00,0.00,0.00,0.00,0.00,0.00,normal.

0,tcp,private,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,271,13,1.00,1.00,0.00,0.00,0.05,0.07,0.0
0,255,13,0.05,0.07,0.00,0.00,1.00,1.00,0.00,0.00,neptune.

0,icmp,ecr_i,SF,1032,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,511,511,0.00,0.00,0.00,0.00,1.00,0.
00,0.00,255,255,1.00,0.00,1.00,0.00,0.00,0.00,0.00,0.00,smurf.

0,udp,private,SF,28,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00
,73,1,0.01,0.05,0.01,0.00,0.00,0.00,0.00,0.00,teardrop.
```

3.2. Evaluation Criteria

To rank different results, a cost matrix C is defined [4]. Given the cost matrix illustrated in Table (2) and the confusion matrix obtained subsequent to an empirical testing process, a cost per test (CPT) is calculated by using the following formula:

Table (2). The Cost Matrix

	Normal	Probing	DoS	U2R	R2L
Normal	0	1	2	2	2
Probing	1	0	2	2	2
DoS	2	1	0	2	2
U2R	3	2	2	0	2
R2L	4	2	2	2	0

$$CPT = \frac{1}{N}\sum_{i=1}^{m}\sum_{j=1}^{m} CM(i,j) * C(i,j) \qquad \ldots(2)$$

where CM and C are, respectively confusion matrix and cost matrix, N represents the total number of test instances, and m is the number of the classes in classification. The accuracy is based on the Percentage of Successful Prediction (PSP) on the test data set, which is given by:

$$PSP = \frac{number\ of\ successful\ instance\ classification}{number\ of\ instances\ in\ the\ test\ set} * 100 \qquad \ldots(3)$$

Higher values of PSP and Lower of CPT show better classification for the intrusion detection system. In this paper, the Detection Rate (DR), PSP and CPT measures are used to rank different results. Table (3) illustrates the confusion matrix for the winner on KDD Cup 99 [4].

Table (3). Present the confusion matrix related to the best percentage of successful predication for the winner on KDD Cup 99

Predicted Actual	Normal	Probing	DoS	U2R	R2L	%DR
Normal(60,593)	**60262**	243	78	4	6	99.5
Probing (4,166)	511	**3471**	184	0	0	83.3
DoS (229,853)	5299	1328	**223226**	0	0	97.1
U2R (228)	168	20	0	**30**	10	13.2
R2L (16,189)	14527	294	0	8	**1360**	8.4
***PSP* = 92.71%**			***CPT* = 0.2331**			

4. Experimental Results

The system of Figure (1) is implemented under Visual Studio.NET 2008 environment using Visual C# language. The number of hidden layers considered in the proposed NN architecture is limited to only one hidden layer. The NN is constructed with 83 neurons at the input layer, 50 neurons at the hidden layer and 5 neurons at the output layer. The momentum is fixed to (0.6) after many experiments where this parameter varied over the interval (0.1, 0.9). The learning rate is fixed to (0.3) after varying it over the interval (0.1, 0.5). However, the weights values of the different connections in the whole network are randomly initialized in the interval [-0.5, 0.5]. Since each neuron on the output layer corresponds to a specified class, the neuron with the highest value defines the predicted class. Using this technique, every sample will be assigned to a class among the five classes defined a priori. Table (4) illustrates the confusion matrix relation to the DR, PSP, and CPT obtained after 500 epochs using the proposed method.

Table (4). The DR for each classification type, PSP and CPT using proposed algorithm.

Predicted Actual	Normal	Probing	DoS	U2R	R2L	%DR
Normal(973,606)	**571**	7	22	5	1	94.22
Probing (31,42)	5	**28**	8	1	0	66.66
DoS (2096,2299)	52	101	**2145**	0	1	93.3
U2R (4,10)	3	3	1	**2**	1	20.0
R2L (13,160)	138	3	8	4	**7**	4.3
***PSP* = 88.322%**			***CPT* = 0.286**			

From Table (4), it can be seen that the DR for both Normal and DoS attack indicates good accuracy. For Probing attack it gives about 66.66 but it indicates worst accuracy for both U2R and R2L. That because, the number of connection record for both attacks in train dataset, is very small (as it can be seen that the number of connection record in train dataset for U2R is only 4 connection records) while for R2L, it is about 13 records. Table (4) also shows that the PSP is about 88.32 where CPT is equal to 0.286.

5. Conclusions

An approach for a K_mean algorithm based partition system to split continuous data to three groups has been proposed. Data preprocessing and data converting to represent inputs as (1, -1) has been presented in this paper. The implementation of intrusion detection system based on Neural Network using Backpropagation algorithm to classify the normal, attack patterns and the type of each attack have been presented. The results show that the system indicates good accuracy for both Normal and DoS attack and worst accuracy for R2L and U2R. Also the results show that the PSP and CPT is about 88.32 and 0.286, respectively. As a future work one may check that if more than three groups give better performance. Also another activation function and another method for data representation may be used.

REFERENCES

[1] A. Alonso Betanzos, N. Sanchez Marono, F.M. Carballal Fortes, J. Suarez Romero, and Perez-Sanchez, Classification of Computer Intrusions using Functional Networks. A Comparative Study, European Symposium on Artificial Neural Networks ESANN, ISBN 2-930307-07-2, Bruges Belgium, pp 25-27, April 2007.

[2] Anazida Zainal, Mohd Aizaini Maarof, Siti Mariyam Shamsuddin, and Ajith Abraham, Ensemble of One-class Classifiers for Network Intrusion Detection System Fourth International Conference on Information Assurance and Security FICIAS, Computer Society Washington, DC, USA, IEEE, pp 180-185, 2008.

[3] Arman Tajbakhsh, Mohammad Rahmati, and Abdolreza Mirzaei, Intrusion detection using fuzzy association rules, Applied Soft Computing ASOC-509, Elsevier B.V, 2008.

[4] Charles Elkan, Results of the KDD'99 Classifier Learning, SIGKDD Explorations, ACM SIGKDD, Issue 2, Volume 1, Page 67, January 2000.

[5] JIANG Lukan, On line monitoring of crystallization process using FBRM and artificial neural network, Master of SIDS, Thesis, University of Claude Bernard Lyon 1, 69622 Villeurbanne cedex, France, 2006.

[6] Laurene Fausett, Fundamentals of Neural Networks Architectures, Algorithms, and Applications, Prentice Hall, ISBN-10: 0133341860, ISBN-13: 978-0133341867, pages 289-300. 1994.

[7] Maher Hamdan Khalil Abu-Mutlaq, Dataflow Processor for Back Propagation Neural Networks: Architecture and Performance Evaluation Master of Science in Computer Engineering Thesis, King Fahd University of Petroleum and Minerals, February 1995.

[8] Mehdi Moradi, and Mohammad Zulkernine, A Neural Network Based System for Intrusion Detection and Classification of Attacks, International Conference on Advances in Intelligent Systems, Theory and Applications, Luxembourg, Kirchberg, Luxembourg, IEEE, November 2004.

[9] Rachid Beghdad, Critical Study of Neural Networks in Detection Intursions, Press, Computer and Security, Elsevier, June 2008.

[10] Sandhya Peddabachigari, Ajith Abraham, Crina Grosan, and Johnson Thomas, Modeling intrusion detection system using hybrid intelligent systems, Elsevier, 28 June 2005.

[11] Srinivasa Kumar Devireddy, and Settipalli Appa Rao, Hand Written Character Recognition Using Back Propagation Network, Journal of Theoretical and Applied Information Technology, Volume. 5, No. 3, 2009.

[12] Yacine Bouzida, and Frederic Cuppens, Neural networks vs. decision trees for intrusion detection, IEEE, IST Workshop on Monitoring, Attack Detection and Mitigation MonAM2006 Tuebingen, Germany, September 2006.

[13] Yuehui Chen, Ajith Abraham, and Bo Yang, Hybrid Flexible Neural-Tree-Based Intrusion Detection Systems, International Journal of Intelligent Systems, Volume 22, pp 337-352, 2007.