International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:4, No:10, 2010

Identify Features and Parameters to Devise an Accurate Intrusion Detection System Using Artificial Neural Network

Saman M. Abdulla, Najla B. Al-Dabagh, Omar Zakaria

***Abstract*—**The aim of this article is to explain how features of attacks could be extracted from the packets. It also explains how vectors could be built and then applied to the input of any analysis stage.

For analyzing, the work deploys the Feedforward-Back propagation neural network to act as misuse intrusion detection system. It uses ten types if attacks as example for training and testing the neural network. It explains how the packets are analyzed to extract features.

The work shows how selecting the right features, building correct vectors and how correct identification of the training methods with nodes' number in hidden layer of any neural network affecting the accuracy of system. In addition, the work shows how to get values of optimal weights and use them to initialize the Artificial Neural Network.

***Keywords*—**Artificial Neural Network, Attack Features, Misuse Intrusion Detection System, Training Parameters.

I. INTRODUCTION

THERE is a wide usage of the artificial neural Network to build many models of Intrusion detection System [17]. The abilities like learning and predicting cases making Artificial Neural Network (ANN) to be a good tool for building whatever models proposed for the Intrusion Detection System (IDS). (IDS)s are classified into two categories; Misuse and Anomaly based systems. The approach of misused IDS depended on the rule based methods, while the anomaly detection systems depended on some behaviors of attacks [3].

We can deploy an Artificial Neural Network to work an approach of Intrusion Detection System through the vectors that used to learn the neural network model [6]. The vectors will contain elements that representing attacks features. The elements will be applied to some rule based methods to detect misuse situation or they will explain some behaviors of an attack that used for anomaly detection. The (ANN) will receive these vectors at its input layer and used them in training phase [8]. The training or learning phase will teach the (ANN) to work as a model to achieve required functions or activities that needed to solve problems [10].

This work has built a misuse (IDS) using (ANN). The next section explains, in brief, the Misuse (IDS). Section three explains the structure and the functions of the (ANN). Section four explains, how to extract the features that represents your problem domain. Section five will simulate the (ANN) as misuse (IDS) and will show the effect of each (ANN).s parameters on the predication process. The last two sections (Six and seven) will discus and conclude this work.

II. INTRUSION DETECTION SYSTEM

The Intrusion is defined as attempts to compromise the confidentiality, integrity, availability, or to bypass the security mechanisms of computer or network systems [1], while the Intrusion Detection System is the process of monitoring the events occurring in a computer system or over a network and analyzing them for signs intrusions [2]. They could be software or hardware. The structure of any intrusion detection system, as shown in the fig. 1 [5], has some parts, such as; information system, detection engine, Countermeasures, and configuration.

The Intrusion detection systems could be classified based on their locations; Host Based or Network Based, or based on their Techniques or methods used by; Behavior based or Misuse based [4].

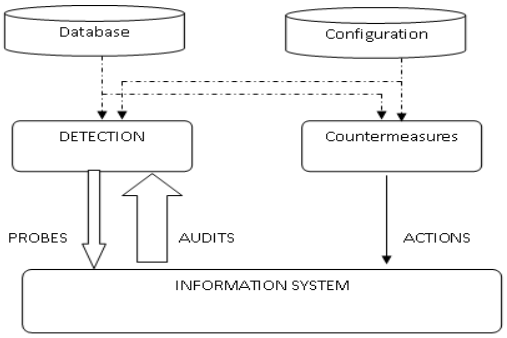

Fig. 1 Typical Structure of Intrusion Detection System.

S.M.Abdulla is with the University of Malaya, 50603, Kuala Lumpur, Malaysia: (e-mail: mamyahias@yahoo.com).

N. B. Al-Dabagh is with Mosul University, College of Computer Science and Mathematics, Mosul - Iraq: (e-mail: najlabadiedabagh@yahoo.com).

O. Zakaria is with Department of Computer Science, Faculty of Science & Defence Technology, National Defence University of Malaya, Sungai Besi Camp, 57000 Kuala Lumpur, (e-mail: manafzack@gmail.com).

The efficiency of any Intrusion Detection Systems will measure by their Accuracy, Performance, and Completeness parameters. In this work, we are going to explain some parameters that affecting the Intrusion Detection System accuracy. The work used Artificial Neural Network for simulation [7].

III. ARTIFICIAL NEURAL NETWORK (ANN)

Artificial Neural Network is a network of many simple processing units; each possibly has a small amount of local memory [6]. These units are connected by some sorts of connections which usually carry numeric data, encoded by any of various means. It somewhat resembles the way human brain works. Its idea comes from attempts of researchers to invent a system that can learn like people brain [7].

Any (ANN) could be classified based on Activate function (Transfer Function) used by neurons (units); Feed forward or feedback, or based on the training algorithms; Supervised or Unsupervised [14] [16]. Fig. 2 shown a simple structure of (ANN) Neuron (units) with transfer function (activate).

The activate function that used in this work is "logsig" and the training algorithm is supervised "Feed forward back-propagation" algorithm. The structure is Perceptron Multilayer [11].

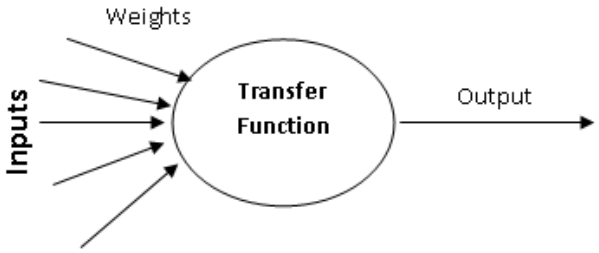

Fig. 2 The Structure of a Neuron

Any (ANN) consist of three of layers; Input layer, Hidden layer, and output layer. The number and attributes of neurons at input and output layers depends on the vector that applied to the input layer of the (ANN) and what output needs at output layer [6]. Figure (3) shows the typical connection between layers in any (ANN).

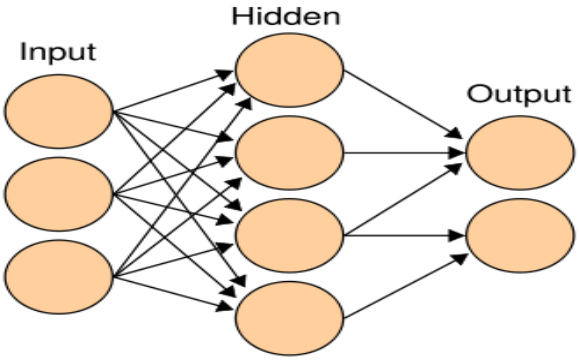

Fig. 3 Layers connection in any (ANN)

IV. (ANN) STRUCTURE

We will study a simple case to explain how the number and attributes of neurons at input and output layer of an (ANN) could be obtained. We need to test the accuracy of the (ANN) model within the changes of some parameters such as; training function, number of learning iteration, number of hidden layer nodes, and number of input vectors applied for trailing.

A. Input Vector

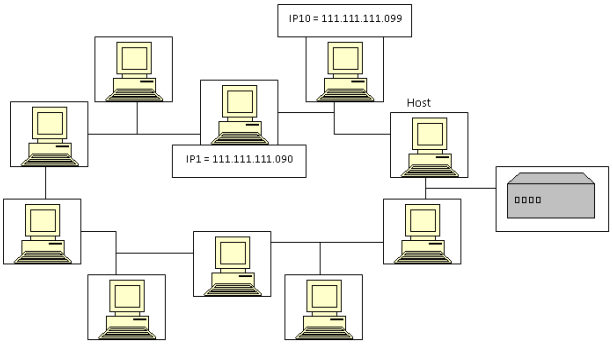

Fig. 4 The Structure of a Neuron

To identify the input structure, it is necessary to show the simple case study that used to build an intrusion detection system for it. Suppose we have a LAN network as mentioned in figure (4), and suppose we have (11) attacks that most possibly found on that LAN [13].

Table 1 explains the characteristics of these (11) attacks and the TCP/IP protocols that related to each attack [11] [12].

From the information located in table 1, which extracted

TABLE I
FEATURES OF STUDIED ATTACKS

Attack's name	Signature / Event	Protocol Name
LAND ATTACK	When the IP number of source and destination host is similar.	IP
Null TCP Packer	Occurred when all flags are not set.	TCP Flags.
Xmas	Occurred if all flags are set to 1.	TCP Flags
SYN/FIN	Occurred if both SYN and FIN flags are set to 1	TCP
IOS UDP Bomb	Occurred if SYN log is initiated to port 514.	UDP
Chargen DoS	Occurred if source port is 7 and destination is 19.	UDP
Broadcast Source address	If the IP address of the source is 255.255.255.255.	IP
Snork	If the source port is on of the 135, 19 and 7 and the destination is 135.	UDP
Orphan FIN Packet	If a FIN packet sent to port less than 1024	TCP
WinNuke (OOB Nuke)	If a TCP packet sent to 139 with setting the urgent mode.	IP
Multi-Cast IP Source address	If the IP source address is 224.X.X.X	IP

from communicating packets [11], we can build the structure of the vector that applied to input layer of (ANN) during training and testing phase. From the attack description, we can know which information should be inserted in input vector to represent a certain attack. Through the following steps we can build the structure of the input vectors.

1) IP numbers of Source and Destination are used to represent "LAND ATTACK", "Broadcast Source Address", and "Multi-Cast IP Source Address" attacks.
2) Port Number of Source and Destination are used to represent attacks of "Chargen DoS", "Snork", and "WinNuke (OOB Nuke)".
3) Flags; there are six flags could cover attacks of "Null TCP Packet", "Xmas Tree", " SYN/FIN", and "IOS UDP Bomb".
4) The attack "Orphan FIN Packet" depends on both port number and Fin flag. So that it doesn.t has any affect on changing the vector size or attributes.
5) Finally, we will include the "Protocol ID" in the vector as a protocol name.
6) The complete structure of applied vector shown in the Fig. 5. It consists of (10) elements. So that, the number of nodes at input layer of our (ANN) will be (10).

B. Output nodes

The format of output vector will identify the node number that needs at output layer. For example, if we need to indentify the normality or abnormality of an applied vector, only one node is enough to give one bit output ('0' for normal case and '1' for abnormal case). While, if the name of the attack needs too, in such cases we need an extra (4) bits to represent (16) cases, for example ('0000' for normal, '0001' for Land Attack, '0010' for Null TCP Packet and …etc.). In this work we will consider (5) nodes at output layer; one node (bit) for detection purpose and four nodes (bits) for classification purpose.

Fig. 6 shows the structure of our (ANN), without the hidden layer nodes. It will be identified in next section.

IP Source	IP Destination	Source Port	Destination Port	SYN	ACK	FIN	RES	UM	Protocol ID
111111111090	111111111091	1024	20	1	0	0	0	0	1
111111111090	111111111092	1025	20	1	0	0	0	0	1

Fig. 5 Vector sample

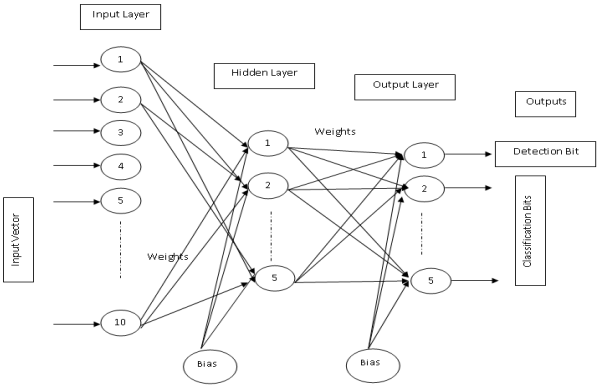

Fig. 6 The Structure of the used Neural Network

V. (ANN) SIMULATION AND EVALUATION

A. Hidden layer Node Number:

The first phase of any Artificial Neural Network simulation is Training phase [15]. In this phase, our work used Supervised Feed forwarded back-propagation training Algorithm and activated the "logsig" as transfer function [17]. We have proposed two sets of training vectors; first, it consists of (5200) vectors. Second, is only (200) vectors. All experiments have been performed on Intel Core (TM) 2 Duo CPU +3.00 GHz, Vista, and (4.00 GB) RAM. During the experiment, we have monitored the time consumed by training phase and the convergence rate, for each data set, while the number of node in hidden layer changed. Table (2) shown how the increasing the nodes will affect the accuracy. However, more increasing will just increasing the training time and will not make more improvement in accuracy of the prediction process, sometimes it affects negatively.

B. Training Functions and Training Phase:

To train our (ANN) model, it is necessary to known what kind of training function is most suitable. We have tried most known training functions [7] and have recorded the performance for each function. Table 3 shows that most suitable training function of our model is "trainlm" function.

TABLE II
CONVERGENCE RATE AND TRAINING TIME RELATION WITH NUMBER OF HIDDEN NODES

Vector Numbers	Number of Nodes	Training Time (Sce)	Convergence Rate
200	2	140	40.2
	5	194	79.5
	10	588	45.7
5200	2	560	69.3
	5	6235	92.8
	10	11653	74.8

TABLE III
COMPRESSION BETWEEN TRAINING FUNCTIONS

Training Function	Performance of Detection	Performance of Classification
trainscg	0.1033	0.0744
traincgb	0.1653	0.0984
traingdm	0.1389	0.2314
traincgf	0.1077	0.144
trainlm	9.8579e-06	8.4967e-06

The next action after fixing the training function is applying the dataset to the model and starting training phase. For each set, we have changed the number of iteration (epoch) to find the most suitable number that gives highest accurate at output. For each dataset, we have changed the number of iteration between (100, 200, and 500) epochs. Then, when the network got the best training, we have saved the optimum value of weights and biases. Fig. 7 showing the structure of the optimum weight values saved after training phase.

The structure of the optimum weights that located between input layer and hidden layer is (5 x 10), while the structure of weights between the hidden layer and output layer is (5 x 5).

-0.11806	-0.09088	-0.0909	-0.0168	4.650021	4.912566	4.99645	4.917442	-0.11526	9.971604
-0.34783	-0.35955	0.049429	0.049586	12.53409	12.54471	12.54476	12.54375	-0.38591	19.3184
-0.02847	0.035283	0.018575	0.018707	1.012593	1.826807	1.208156	1.886831	-0.3977	-3.02252
-0.03055	-0.03008	0.008312	0.008402	-4.41067	1.257494	-4.60158	5.247461	-0.02242	7.843321
0.00697	-0.0081	0.016869	0.0169	10.87328	10.7682	10.8741	10.76657	0.035796	10.2777

The weights value of Input layer

24.70495	-36.7204	-37.6453	1.554167	57.69709
-40.453	4.455295	-45.8734	16.32961	-49.6322
39.93992	-6.71564	-44.8619	-53.4077	51.39306
36.68335	-78.2092	39.58251	19.6116	69.21801
-0.49971	-5.11812	6.430772	52.33284	72.02589

The weights value of output layer

Fig. 7 Saved optimum value of weights and Biases

C. Testing Phase:

For testing the (ANN) model, we have initialized the model with the weight values that obtained during the training phase. The model has been tested with four groups. dataset. Each group consists of (15) vectors and they represent four statuses of the packets that possibly captured on the network.

1) The first group of vectors has chosen from the Normal vectors that already used in training phase.
2) The second group of vector has chosen from the Attack Vectors that also applied to the (ANN) model during the training phase.
3) The third group is for vectors that represented Normal packet; however, they are not applied to the (ANN) model during the training phase.
4) The last group of vectors represented new attacks that not applied to the (ANN) model during the training phase.

From testing phase we have found that the (15) vectors of known normal and other (15) vectors of known attacks are easily recognized by the (ANN) model. These vectors are already used during the training phase; therefore, it was easy for the (ANN) model to recognize them. Table (4.a and 4.b) shows how known vectors have recognized.

TABLE IV A
RATE RECOGNITION FOR (ANN) TRAINED ON (200) VECTORS

		Unknown Normal	Unknown Attack
150 Epochs	Detection	7/15	7/15
	Classification	7/15	0/15
200 Epochs	Detection	8/15	8/15
	Classification	7/15	7/15
500 Epochs	Detection	15/15	8/15
	Classification	15/15	0/15

TABLE IV B
RATE RECOGNITION FOR (ANN) TRAINED ON (5200) VECTORS

		Unknown Normal	Unknown Attack
150 Epochs	Detection	15/15	8/15
	Classification	7/15	0/15
200 Epochs	Detection	15/15	15/15
	Classification	15/15	7/15
500 Epochs	Detection	15/15	15/15
	Classification	15/15	0/15

VI. DISCUSSION

The process of identifying the right parameters and selecting good features to represent input vectors for any (ANN) is an essential step. This work needs to explain the effect of such parameters on training phase, prediction and accuracy of the system, as well as, the performance of the system.

We can improve the accuracy and the performance of an (IDS) through obtaining good training parameters and selecting right features to design any (ANN).

The results in table 2 show that increasing the number of nodes more than required will affect negatively on the convergence rate and will increase the training time required. At the same time it will consume the CPU performance as it take longer duration. Look at fig. 8.

Another factor that should be selected carefully is the vectors that applied during the training phase. These vectors will explain to the (ANN) how to distinguish between the normal and the attack vectors. The type of elements inside the vector and the number of vectors will affect the accuracy of the (ANN) while it makes prediction. Fig. 9 explains the effect of vector.s number and iteration number that used during the training phase, on the process of the recognition.

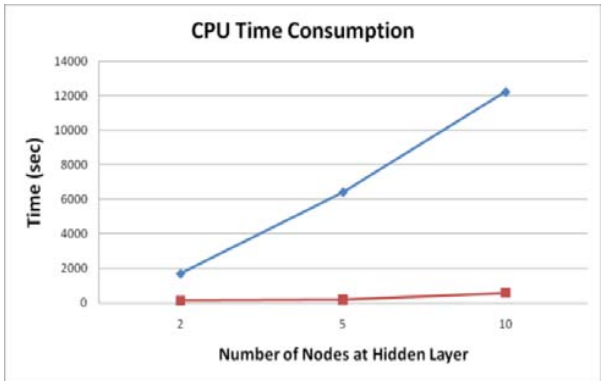

Fig. 8 CPU Time Consumption with increasing Hidden node number

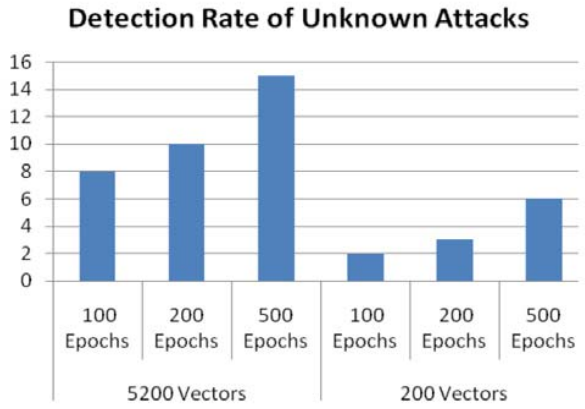

Fig. 9 Effect of Vector number and Epochs Number on Detection Rate or Accuracy

VII. CONCLUSION

There are many scientific fields that propose Artificial Neural Network to simulate systems. To simulate any model correctly, it is necessary to know how to use the parameters of the (ANN) in order to get optimum solution.

This work has proposed (ANN) to implement the Misuse Intrusion Detection System. It explains that intrusion detection system needs to be accurate system and could predict unknown cases. So that, it explains how to analyze attacks and normal packets to build vectors that used for learning an (ANN).

The work found how the type and number of input vectors during training phase will affect the accuracy of the model. It also found that parameters of the (ANN) should be fixed on the correct value in order to get optimum output.

REFERENCES

[1] Karen S. , Peter M., "Guide to Intrusion Detection and Prevention Systems (IDPS)", Recommendations of the National Institute of Standards and Technology, Special Publication 800-94, February 2007.

[2] John Mc., Alan Ch., and Julia A., "Defending Yourself: The Role of Intrusion Detection Systems", IEEE Software, volume 17, No. 5, 0740-7459, September / October 2000.

[3] Rodrigo Rubira Brance, " KIDS-Kernel Intrusion Detection System", Hacker 2 Hacker Conference IV 2007 – Brazil, 11/09/2007.

[4] Bob R., "Hiding Intrusion Dectection System (IDS)", Whitepaper, in www.infosecwriters.com/text_resources/pdf/wp-003.pdf, found on 2010.

[5] Latifur Khan, Mamoun Awad, and Bhavani Thuraisingham, "A new intrusion detection system using support machines and hierarchical clustering" , The BLDB Journal, 1066-8888, Volume 16, No. 4, Octobor-2007, pp (507-521).

[6] Ajith Abraham, "Artificial Neural Network", Handbook of Measuring System Design, 0-470-02143-8, 2005.

[7] Klaus D., Alexander K., and Horst-Michael G., "Transfer Functions in Artificial Neural Network", http//:www.brains-minds-media.org, Accessed on 2010, 2005.

[8] Jake R., Meng-Jang Lin, andRisto Mi., "Intrusion Detection with Neural Networks", Advances in Neural Information Processing Systems 10,Cambridge,MA: MITPress,1998.

[9] Zhimin Yang, Xiumei Wei, Luyan Bi ,Dongping Shi ,Hui Li, "An Intrusion Detection System Based on RBF Neural Network", The 9th International Conference on Computer Supported Cooperative Work in Design Proceedings, 2005.

[10] Wang Jing-xin, Wang Zhi-ying, and Dai Kui, " A Network Intrusion Detection based on the Artificial Neural Network", ACM , 1-58113-955-1, Vol. 85, Proceedings of the 3rd international conference on Information security, 2004.

[11] Allan Liska, " Network Security: Understanding Types of Attacks" http://www.informit.com/articles/article.aspx?p=31964, accessed on 2010.

[12] Simon H. and Ray Hunt, " A taxonomy of network and computer attacks" Computer and Security journal, 0167-4048, 2004.

[13] Kristopher Kendall, "A Database of Computer Attacks for the Evaluation of Intrusion Detection System", A thesis submitted to Department of Electrical Engineering and Computer Science At MASSACHUSETTS INSTITUTE OF TECHNOLOGY, 1999.

[14] Mansor Sh. and Amir Sh., " Fast Neural Intrusion Detection System Based on Hidden Weight optimization Algorithm and Feature Selection", World Applied Sciences Journal 7 (Special Issue of Computer & IT): 45-53, 2009

[15] Jimmy Sh. and Heidar A., "Network Intrusion Detection System Using Neural Networks", Fourth International Conference on Natural Computation, 978-0-7695-3304-9, 2008.

[16] Qinzhen Xu,, Wenjiang Pei, Luxi Yang, and Qiangfu Zhao, "An Intrusion Detection Approach Based on Understandable Neural Network Trees", JCSNS International Journal of Computer Science and Network Security, Vol.6 No.11, November 2006

[17] Vipin Kumar, Jaideep Srivastava and Aleksandar Lazarevic, " Intrusion Detection: Survey" Resource Secured Journal, Vol. 5, 10.1007/b104908v, 2005, pp (19-78).

ACKNOWLEDGMENT

Authors appreciated the role of University of Malaya grant "PS072/2010A" and the fund from "Ahmed Ismail Foundation" due their support my PHD works.