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.](image-url)
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 backpropagation” algorithm. The structure is Perceptron Multilayer [11].

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).

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

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

<table>
<thead>
<tr>
<th>Vector Numbers</th>
<th>Number of Nodes</th>
<th>Training Time (Sec)</th>
<th>Convergence Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>200</td>
<td>2</td>
<td>140</td>
<td>40.2</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>194</td>
<td>79.5</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>588</td>
<td>45.7</td>
</tr>
<tr>
<td>5200</td>
<td>2</td>
<td>560</td>
<td>69.3</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>6235</td>
<td>92.8</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>11653</td>
<td>74.8</td>
</tr>
</tbody>
</table>
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).

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. 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.

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.
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


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.