ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal
Regular Issue, Vol. 7 N. 1 (2018), 49-65
eISSN: 2255-2863
DOI: http://dx.doi.org/10.14201/ADCAIJ2018714965

Improving Intrusion Detection Systems Using Artificial Neural Networks

Yaser A. Jasim

Accounting Dept. Cihan University-Erbil\Iraq
Yaser.a.Jasim@Cihanuniversity.edu.iq

KEYWORD

Artificial Neural Networks; BP; Intrusion Detection System; MATLAB

ABSTRACT

In this paper, some of the methods used in the intrusion detection system were described using the neural network as a tool in intrusion detection system, which became very necessary in computer systems because it provides protection against attacks by hackers that are becoming increasingly destructive to computer systems.

*The Backpropagation Neural Network was chosen from among the neural networks due to its ability, speed and intelligence to recognize packet patterns captured from the network, providing the ability to detect intrusion of the system. The speed of the network in giving the diagnosis is one of the most important reasons for choosing the neural network. Therefore, many Attacks features have been analyzed of the standard packets that allow traffic through the network as well as the unusual packets, especially on these protocols (*TCP, UDP*).*

The results of these analyzes have been used to learn the neural network on the structure and pattern of standard and unusual packets. There are many algorithms for learning the neural network, but the researcher used the Standard Backpropagation Algorithm. Therefore, for increasing the intelligence and speed of the network and its ability to classify, the researcher used the Resilient Backpropagation Algorithm, provided by MATLAB *programming language which is smarter and more accurate than the first algorithm.*

The output of this system can detect the standards packets from the unusual packets and classify them into five types with the efficiency up to 100% of the defined packets. However, the detection of the unknown attacks is not known, and rating score is zero.

This paper contains a lot of tables and figures that illustrate the results and analysis of the results. It should be noted that any intrusion detection system must be up-to-date, as there is no effective and successful intrusion detection system without updating its database.

1. Introduction

Nowadays, Intrusion Detection System (IDS) on computers or the Internet has brought the attention of many researchers as a result of large network inflation and the increase of complexity in computer technologies to prove and develop intrusion detection systems that have become very necessary. Generally, any attempt to detect intrusion or attack on the computer or Networks has become known as intrusion detection systems (IDS) [4,1].

The designers of any computer network who are interested in increasing the size of the network and its types are always thinking of the two main factors; firstly, security and protection of the network; and secondly services and facilities provided by the network [1].

In general, these two factors negatively impact each other, the increase in one reduces the efficiency of the other, so technologies such as intrusion detection systems are working on the compatibility of these factors. The researcher hopes that the proposed software in this paper will clarify the problems of security and disruption of the system [1,4].

2. Security and Intrusion Detection System

Computer security is defined as the techniques and administrative procedures applied to the computer systems to ensure their availability, effectiveness and confidentiality in the transfer of information in computer system and ensure access. Computer security can be classified into three areas which are prevention, detection, and reaction. The second area is covered by the intrusion detection system (IDS). Which (IDS) is defined as the identification and response of any malicious behavior targeting the computer and network resources; it can be classified into two main types: signature base and misuse base [12,1] table (1) shows the comparison of Misuse Base and Signature Base of the intrusion detection system.

The signature base depends on the signatures of the attackers, while the misuse base acts to search for any abnormal behavior of the user. The intrusion detection system is a requirement for the identification and effectiveness of the network (IDS), which is classified into two fields; Network Based (NIDS) or Host Based (HIDS) networks, both are used for monitoring [12,1,2].

Table 1: Comparison of Misuse Base and Signature Base of the Intrusion Detection System.

Misuse Base	Signature Base
High percentage of false alarms	Lowest percentage of false alarms.
Difficult for the security officer to determine the threshold value, which is considered an abnormal behavior and classified as an attack.	Requires an expert security officer to identify the intrusion.
Effective against unknown attacks	Not effective against unknown attacks.
Requires a lot of training to create natural patterns.	Requires strength and continuity to update signature attacks.

2.1. Intrusion Detection System

Intrusion detection system (IDS) is defined as the work that uses certain techniques and special mechanisms to detect intrusion. Intrusions are defined as an attempt to contain security, safety, effectiveness, or overlap security mechanisms in systems or networks. IDS monitor and analyzes events within the computer or network system to detect signs of intrusion, where an intrusion detection system is a programmable or physical component that automatically works on monitoring as mentioned above to identify security problems. Attackers who intrude on the system by entering as authorized users of the network to obtain additional privileges than the normal user who is not authorized and authorized users who misuse the privileges granted to them. However, the intrusion detection system (IDS) detects the attack or abnormal behavior occurring within the network and immediately generates an alarm that the person responsible for security in the network is aware of when an attack occurs and asks him to take the necessary action to complete his or her work [4,12,5,1], figure (1) illustrates the basic architecture of the intrusion detection system.

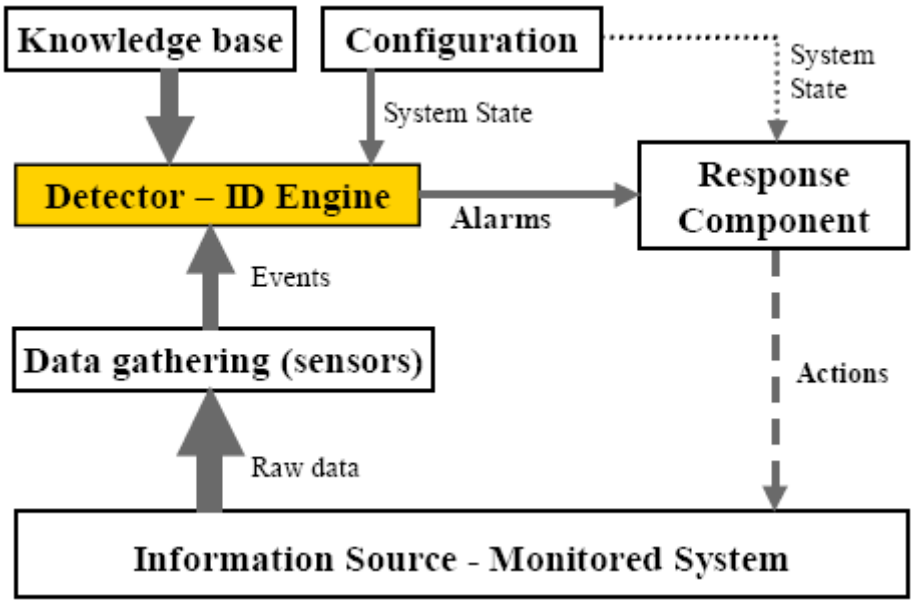

Figure 1: Basic Architecture of The Intrusion Detection System.

2.2. The Purpose of Using IDS Systems

The intrusion detection system allows the organization to protect its systems from threats arising from increased network connections and from the reliability of sharing information between systems, which means protecting their data and the integrity of the systems from attackers and intruders. The goal of security professionals is to use the most efficient intrusion detection to detect intrusion [4,5].

The reasons for using the intrusion detection system are as follows:

1. To detect the intrusion that the security system has not been able to detect or prevent.
2. To detect and deal with attacks.
3. To document threats against the institution.
4. Acts as a quality control of security and management systems especially for large and complex companies.
5. To create useful information about the attack that gives us a good diagnosis and re-correction of causes.

In general, intrusion detection system provides us with an additional layer of security in the system. However, we must remember that the intrusion detection system must detect, prevent and respond as a real-time system because only a few seconds and minutes is enough for the intruder to do it. Also, keep in mind that a good system against hackers and attackers must always be up to date [12].

2.3. Efficiency of Intrusion Detection System

There are three determinants of intrusion detection system efficiency [4,1]:

1. Accuracy; This measure deals with the correct identification of the attack and the absence of the false alarm, and the error occurs when the intrusion detection system warns a legitimate activity as an abnormal or intrusive act.
2. Performance; The performance of an intrusion detection system is measured by the rate of audits performed by the system. The system is considered weak if it does not function as a real-time system.

3. Completeness; The possibility of intrusion detection system to detect all attacks. The decrease occurs when the system fails to detect the attack. This measure is very difficult to evaluate from the rest because it is impossible to obtain general information about attacks or misuse of privileges.

3. Artificial Neural Network

The artificial neural network consists of simple processing units, each unit having a small amount of local memory. These units are linked by some types of connectivity, which usually carry digital data encrypted in a variety of ways. Although the neural network can calculate any arithmetic function, in practice it is used especially in the classification and problem planning, where the network is easy to learn but sometimes it may be difficult especially if the learning laws are complex and fast, Artificial Neural Network (ANN) is compatible with incomplete data and has the ability to train data to show correct outcomes this is the cause of using ANN [1,7].

The artificial neural network can be distinguished by the use of many names such as connections, neural intelligence systems, and distributed parallel processing.

3.1. Machine Learning Methods

There are three types of artificial neural network learning methods:

1. Supervised Learning: In which the Target output vector is inserted in addition to the input vector to calculate the actual output, as a figure (2) depicts the schematic form of supervised learning [12].

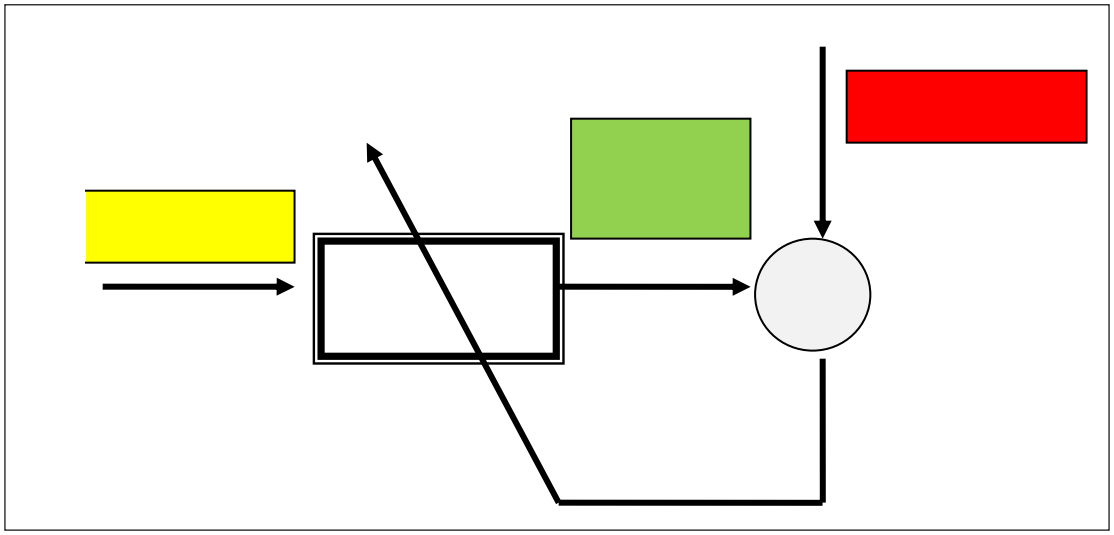

Figure 2: Supervised Learning Diagram.

2. Unsupervised Learning: This method does not enter the Target output, but only enter the input to the network and the output is calculated according to the rules, statistical standards and a detailed explanation of each of the two methods. As figure (3) Shows schematic form of unsupervised learning.

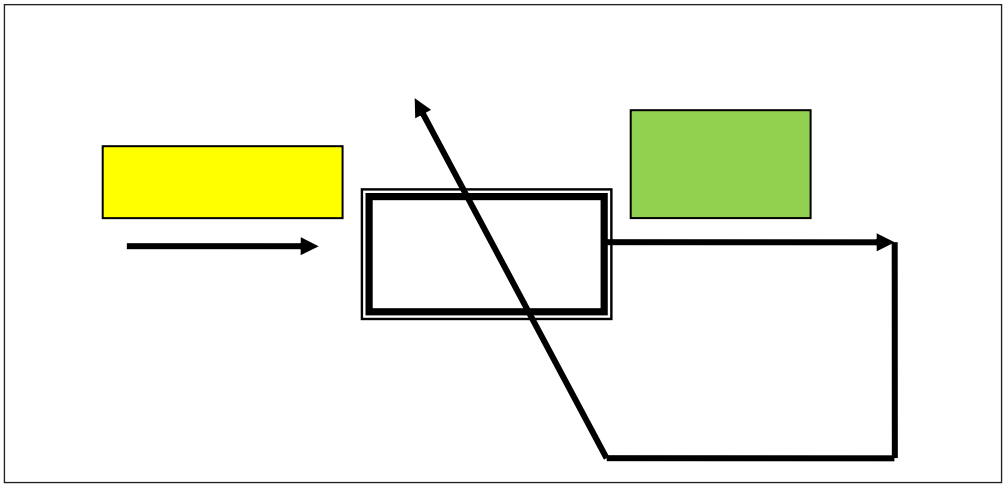

Figure 3: Unsupervised Learning Diagram.

3. Reinforcement learning: a learning method that considers learning to control a system so as to maximize a numerical performance measure that goes for a long-term objective [10].

3.2. Backpropagation Neural Network

The Error Backpropagation Network(EBP), which is a multi-layered and a Feedforward neural network. The Learning method of Multiple Layer network is known as Backpropagation (BP).

The EBP network is a widely distributed algorithm because of its ease and ability to store information implicitly in links, which represent the weights that connect a node to another. The Feedforward of single-layer, multi-layer artificial neural networks are used in a variety of applications (image processing, image recognition, pattern recognition, etc.). It contains a change in the state of effectiveness (either inhibition or stimulation) of all the nodes in all layers in the network and also of the weights that connects layer nodes to other layer nodes based on certain equations. The calculation of the error is applied by subtracting the actual output from the desired output and network training stops when the desired output is obtained and access to the Global Minimum. On the other hand, recurrent is only one step, which applies for the entries with the final weights resulting from the training phase and in one step we get the desired output [3,12].

3.3. The Architecture of Backpropagation Neural Network

The Backpropagation network (BP) is composed of at least three layers of nodes: the input layer, the middle layer called the hidden layer and the output layer. Each layer of the network is associated with the layer that follows it, any node in the input layer is sent out to all the nodes in the middle layer, and the middle layer nodes are sent out to each node in the Output layer.

The number of nodes in the middle layer depends on the complexity of the issue and the size of input information. If the number of nodes in the hidden layer is too large for the number of input nodes, it is not resolved. On the other hand, if the number of nodes in the hidden layer is very small, it will take a large number of steps to train the network. Figure (4) shows the three-layer of the recurrent neural network as the input layer, the middle layer, and the output layer. Each node or processing unit in any layer is connected to all nodes or processing units of the other layer by the weight of the links. The input for the network is represented by an X vector (x1, x2, x3, ... xn), output by a vector Y = (y1, y2, y3, ..., ym) and (n, m) are the exclusion of input vectors [4].

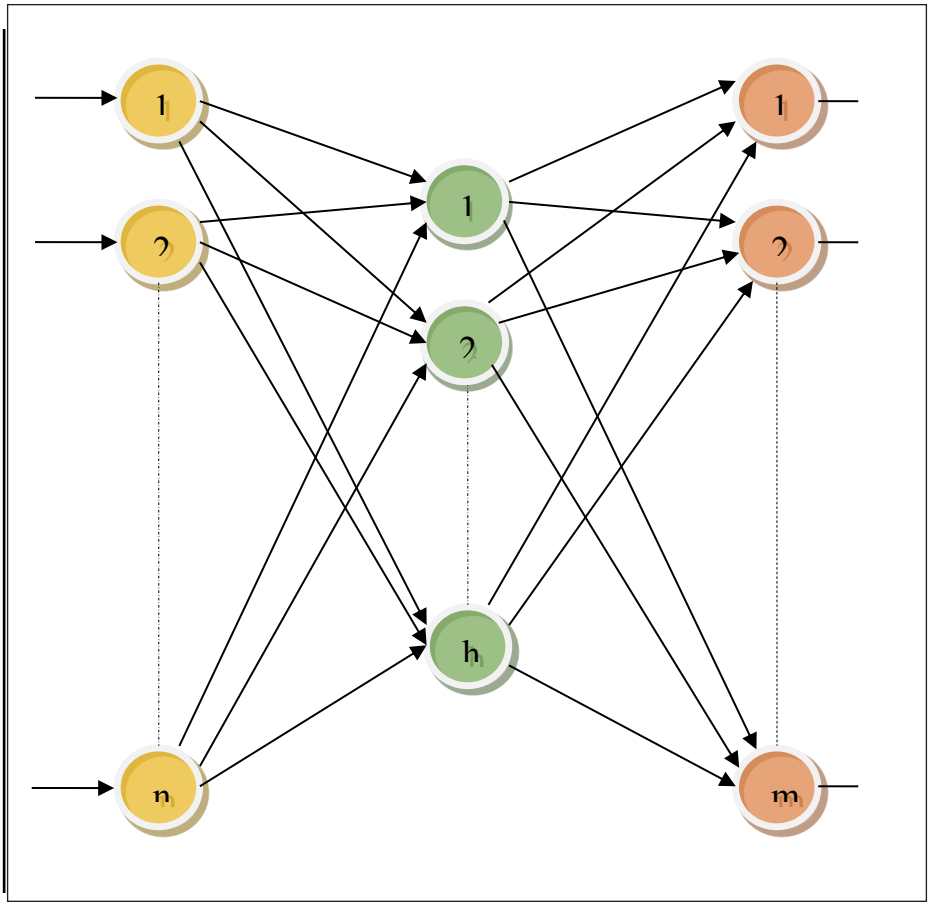

Figure 4: Three Layers Backpropagation Network

Sigmoid Function

Sometimes the Sigmoid Function is called the Logistic Function or the Squashing Function because it limits the output range as well as the actual output out between zero and one [4,7], where the sigmoid function is used in the implementation of the BP algorithm.

$$Out = f(NET)$$
$$= 1/\ 1+e^{-net}$$
$$F(NETj) = out\ (1\text{-}out)$$

3.4. An Overview of Training in BP Network

The purpose of training is to change the weights of the network to obtain the desired output sets for the given inputs. Input and output sets are referred to as vectors. The training pair is the Input Vector with the Target Vector, usually the network trains on a number of training pairs. These pairs of training are called the training set, pre-starting the training process, all weights must start in small random numbers this ensures that the network is not saturated with high-value weights. Training pair is used for training and is available to the network many times during the training phase.

Each model trains and calculates the output. This output compares with the target output and determines the error value, where these errors are propagated from the output layer towards the input layer and this occurs only in the learning phase [6,8].

3.5. Adding a Neuron Bias

In the BP network, it is preferable to add a bias node to speed up the network's work and approach the solution, which is similar to the Threshold Value in the network. The weight of this network is changed like the rest of the cells of the network except that the input of the bias node is always (+1) instead of the input as in the rest of the network nodes calculated from the output of the previous layer [8].

3.6. Network Paralysis

According to the wrong choice of weights, especially if the weights are large numbers, the change in these weights will result in very large numbers. Which means that all or most of the nodes in the network will produce large values for the actual output, while the derivative of the activation function is very small values for this will cause the paralysis of the network so choose small initial weights. To solve this problem, reduce the learning rate but will increase the training time and training periods, if the proportion of learning is very small it will lead to approach the right solution but very slow, either if the proportion of learning is too large it will lead to paralysis [8,7].

3.7. Local Minimum

The reason for the occurrence of the network in local minimum is the wrong selection of the number of hidden layer nodes as well as the wrong selection of the primary weights. In addition to this problem, the number of nodes in the middle layer is increased or decreased until the solution or the nearest solution is reached with the lowest error rate by placing appropriate initial weights on the net [8].

4. Intrusion Detection System Based on Neural Network

In this section, the researcher will discuss attacks targeting the network, specifically on the Transport Layer and especially on (TCP, UDP) communication, but before that, we should know how to extract the features of the attacks on which this system was designed by analyzing the network packet.

4.1. Network Packet Analysis

A network is a set of devices connected to each other through a particular communication medium, and the transmission of data using the (TCP-IP) protocol is through the passage of data in four layers and in each layer, is added a header to the data in preparation for sending from one computer to another in the network, for networks like wireless networks, wired networks, uses the different kinds of communication patterns such as UDP, TCP, IP, figure (5) depicts the headers added to the data in each layer [11,9,2,13].

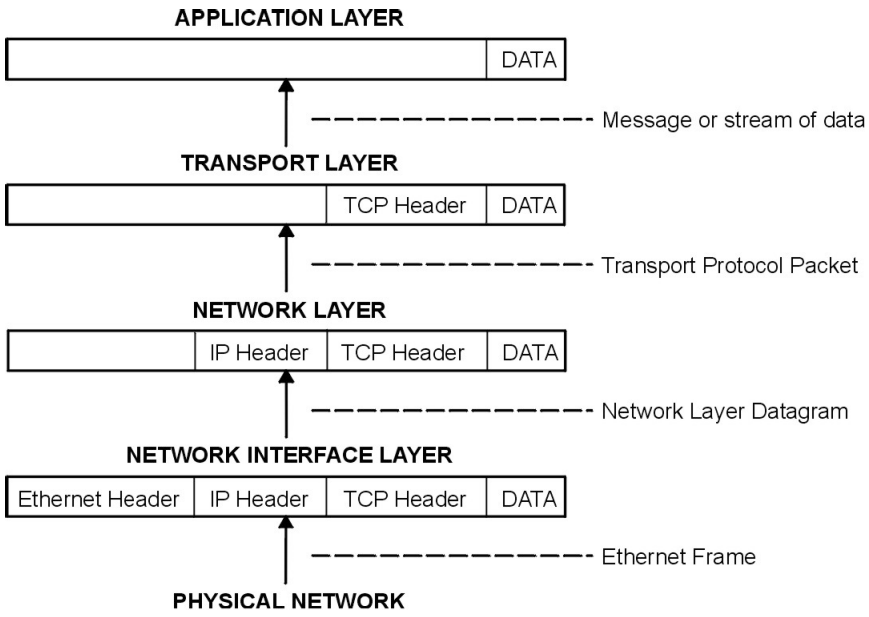

Figure 5: Levels of passing data in TCP-IP Protocol.

Where in each layer the header is added to the data and ranges [20-60] Bytes according to the options that are added to the data in each layer where the general form of the package as shown in figure (6) [9,2].

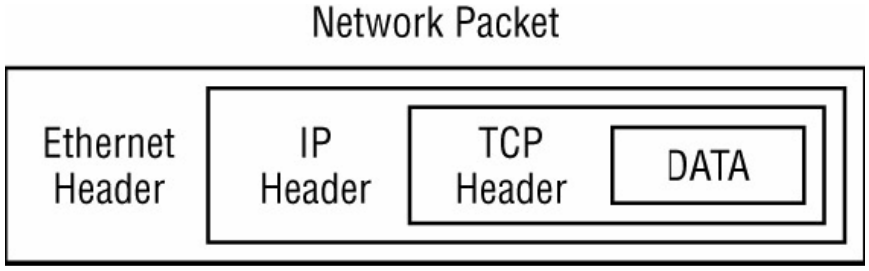

Figure 6: Diagram of encapsulated data in a packet.

4.2. TCP Encapsulation

The TCP connection is characterized by all the data flow control or Buffer size. It also supports the Delivered-on Sequence, which is evident from the installation of its header address as shown in figure (7) [9,2].

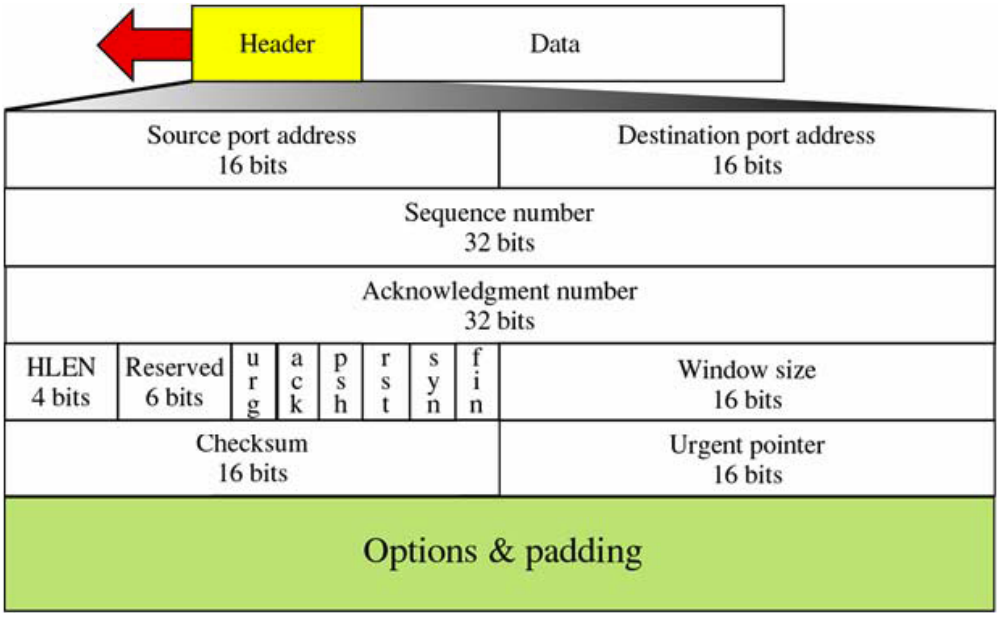

Figure 7: Header of TCP Protocol.

4.3. UDP Encapsulation

UDP connection, where there are no data flow controls or buffer size, nor does it support Delivered on Sequence, nor is it possible to divide data, there is no guarantee of rearrangement in the original form and this is evident from the installation of its header as shown in figure (8) [9,2].

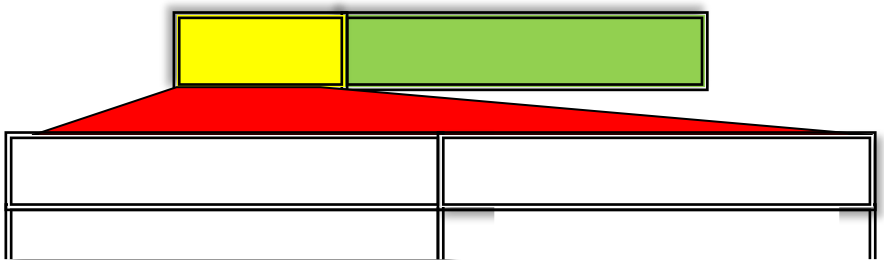

Figure 8: Header of UDP Protocol.

4.4. Work Description

The intrusion detection system in the abuse base relies on a data set with the input layer, then how to train the data, and finally how the network is selected. On this basis, the work begins with the preparation and processing of the input data, then selecting the algorithm used in the training and testing, and store weight values.

Here the work is divided into two phases: the first is the process of configuring the features of attacks, and the second depends on the extract features obtained as the definition of the neural network structure.

4.4.1. *Input Data Preprocessing*

The data inserted into the system as shown in figure (9) is not the output of any sort or classification from a software to capture the packet from the network but is obtained by studying and analyzing the physical connection to the network segments as shown in Table (2). The signatures that we need to capture to detect attacks will explain how the system works.

Port-S	Port-D	SYN	ACK	FIN	RES	UM	Prtc	F1	F2	F3	F4	F5	F6
1024	20	1	0	0	0	0	1	1	0	0	0	0	0
1027	20	1	0	0	0	0	1	1	0	0	0	0	0
1025	20	1	0	0	0	0	1	1	0	0	0	0	0
80	84	1	0	0	0	0	1	1	0	0	0	0	0

Figure 9: Input Data to The System.

4.4.2 *Attack's Features*

In general, each attack has its own signature and event, and to be an excerpt from these attacks it is necessary to study these attacks and analyze their own signatures and events. Table (2) shows the name of the attack and the target header and then the signature of the attack [4,5,9,1].

Table 2: Attacks and their Signatures.

Attacks	Header	Signature
NULL TCP PACKET	**TCP**	Intruder makes flags (SYN, FIN, ACK, RES) inactive, which at the same time makes their values "0".
XMAS TREE ATTACK	**TCP**	Intruder makes flags (SYN, FIN, ACK, RES) active, which at the same time makes their values "1".
SIN/FIN ATTACK	**TCP**	Intruder makes flags (SYN, FIN) active, which at the same time makes their values "1".
CHARGEN DOS ATTACK	**UDP**	The intruder sends from port 7,19 as an exporter or port 135 as a receiver.
SNORK ATTACK	**UDP**	The intruder sends from port 7 as an exporter or port 19 as a receiver.
IOS UDP Bomb	**UDP**	The intruder sends from port 514 as an exporter with activating (SYN) flag at the same time.

4.4.3. *Attack's Features Description*

Here are the properties used by the system through which the system can identify attacks that target the network:

1. Source Port: is the number of port allowed for communications and must be greater than (1023) because the ports whose serialization is less than that are dedicated to the operating system. The largest port number is (1031) because the system is designed for a network of eight computers.
2. Destination Port: The port is the port (port 20) because the inspection process is on the FTP protocol.
3. Synchronize Sequence Number-SYN: Bit number (111) in the TCP header is used to initiate the connection and its value is (1,0).
4. Acknowledgment is Valid-ACK: Bit is the number (108) in the TCP header and is used to accept the connection from the receiver and is (1) if the receiver is "On-Line" and (0) if the receiver is "Off-Line".

5. Terminate the Connection-FIN: Bit is the number (112) in the TCP header and is used to terminate the connection and has a value of (1) if the connection is requested to terminate.
6. Reset Connection-RES: Bit is the number (101) in the TCP header and is used in the event of a Router failure, and when it reaches the value of (1), it terminates the connection for an emergency.
7. Urgent Pointer is Valid-UM: Bit is the number (107) in the TCP header and it is used in case if there are important or emergency data where the value of (1) is set if the receiver knows that it is an emergency event.
8. PRTC: Its value is (1) if the header address is for TCP, and its value is (2) if the header address is for UDP.

5. The Neural Network Structure

As mentioned above, the network architecture must be compatible with the body of the problem to be solved. The network needs (8) nodes together, which represent the input vector, and represents the resulting features of the network analysis, and six nodes representing the output vector of the network which are sufficient to represent the cases of the examined package as shown in Table (2) and there are four nodes in the hidden layer as shown in figure (10). The input is configured to suit the used network, known as the Feedback Error Propagation.

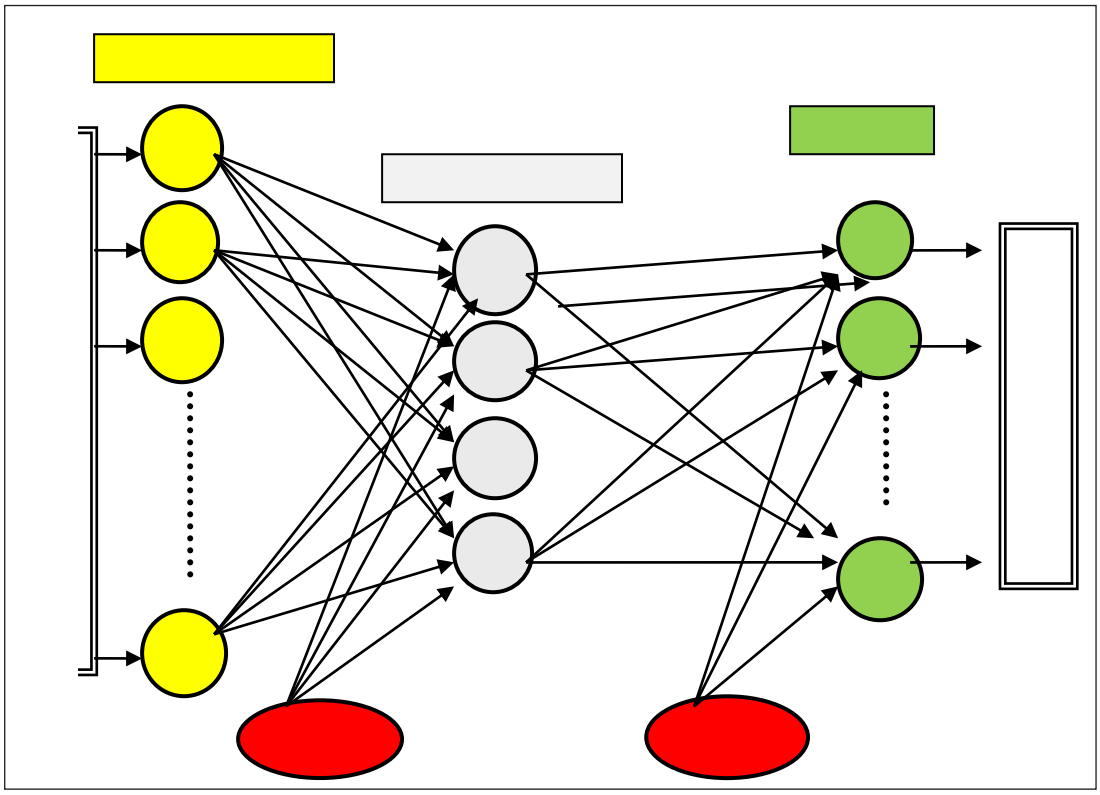

Figure 10: Structure of used Neural Network.

6. Implementation System

The network was trained using Resilient Backpropagation Algorithm to speed up the network and overcome the standard learning algorithm defects in this network. The (newff) function in the MATLAB R2016a was used, and because the learning algorithm has similar properties except in Optimal Step Size. This is why the researcher will explain the BP standard learning steps as shown in figure (11).

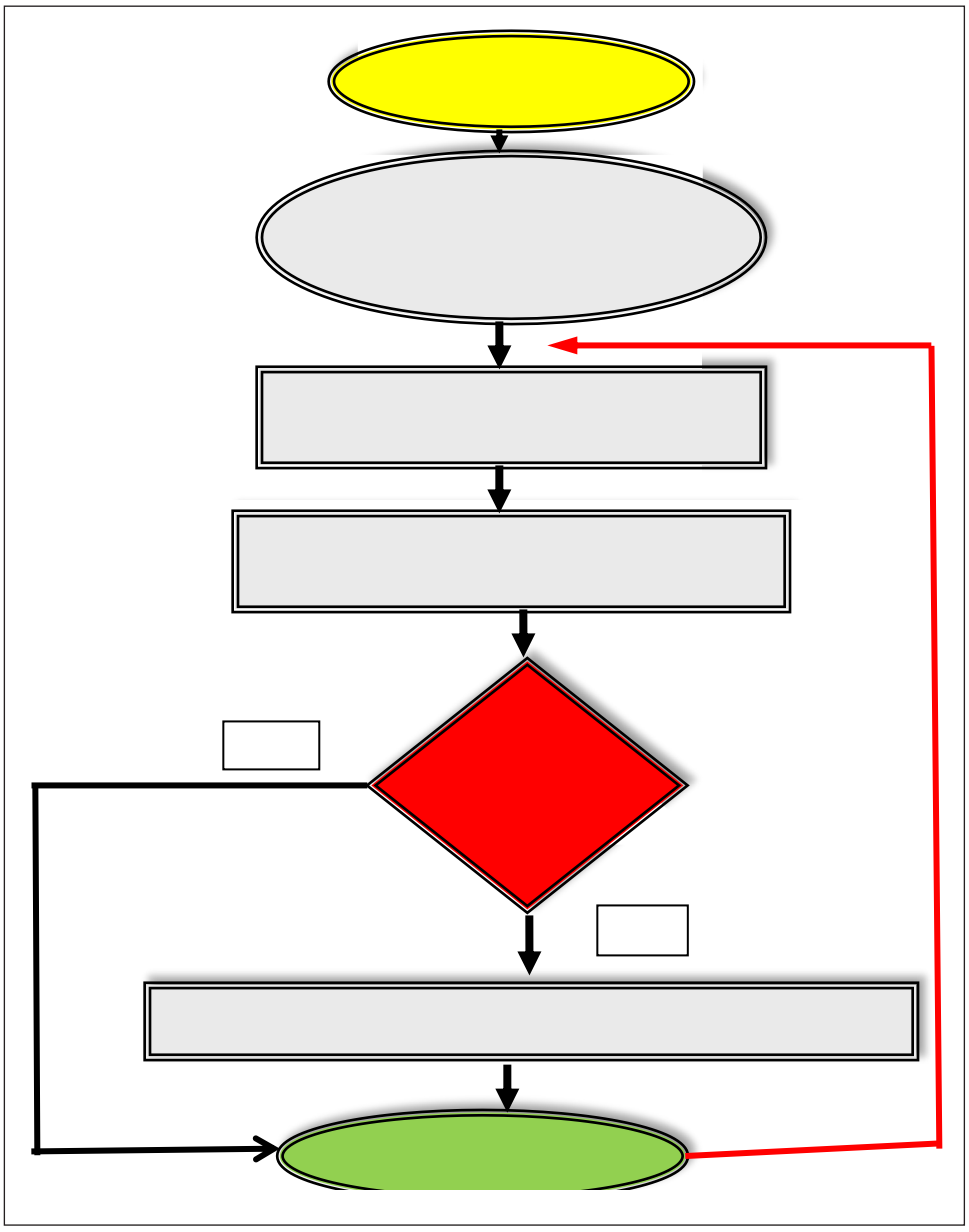

Figure 11: BP Training Algorithm.

Once we have knowledge about the learning algorithm, the steps to implement the program will be three stages:

1. Input stage.
2. Training phase.
3. Results presentation stage.

7. Implementation New Smart Network for Intrusion Detection

➢ The program was written by (MATLAB R2016a).

Since the standard backpropagation (SBP) network has multiple negatives, which leads to slow access to learning, the network is slow in intelligence. The reason is that it adopts an SD vector that leads to the Local Minimum or the Zigzag. To get rid of this problem and to obtain a fast-absorbing smart network, a new network called (Rprop) has been used. The network adopts the basic network (SBP) and the following development: (LR--> δ)) where BP typically uses a constant value (0.5) while in this network the learning rate (δ) is used based on the following conditions:

$$\delta = \begin{cases} \delta + & \text{if } dg > 0 \\ \delta - & \text{if } dg < 0 \end{cases}$$

where (0<δ-<1<δ+); (+δ=0.5) and (-δ=1.2).

This means that the value of the learning rate is limited between the highest value and the least value because the lack of limited values (δ) will reduce the response and the quick intelligence of the network. In figure (12) we will show the Local Minimum in SBP.

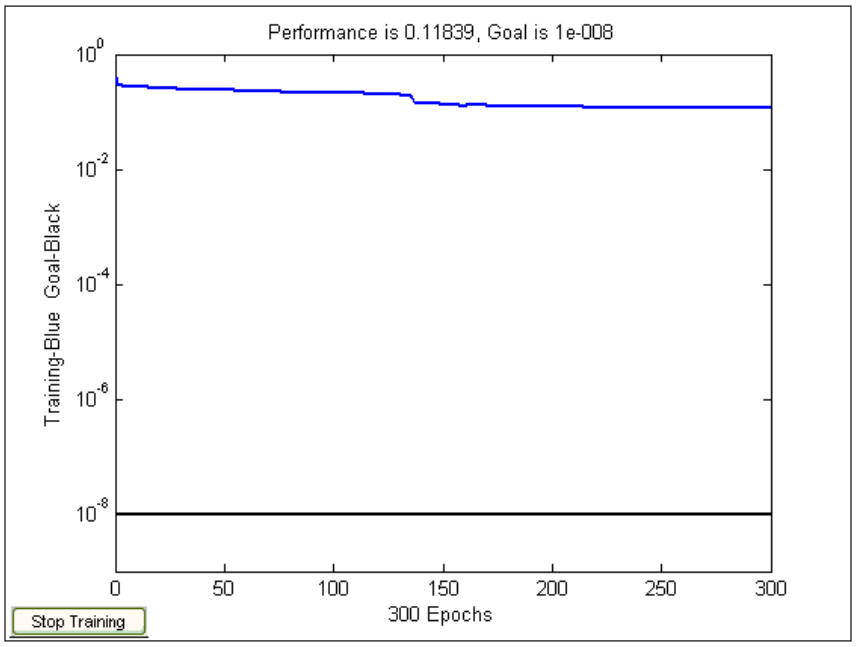

Figure 12: Local Minimum in SBP.

Phase 1: The data entry in this program is from a database file (Indat1.xlsx). All data will be read from this file and converted to a matrix after transporting the rows to columns until it's ready to enter the network. The weights in this program are the type of (Batch). The following is a section of the program that explains the reading process.

```
[Data, headertext] = xlsread('indat1.xlsx');
data1=data (1:107, 1:14);
data1=data1';
```

Phase 2: The training phase, which is the training of the network to obtain the optimal weights and the lowest line ratio as in figure (13).

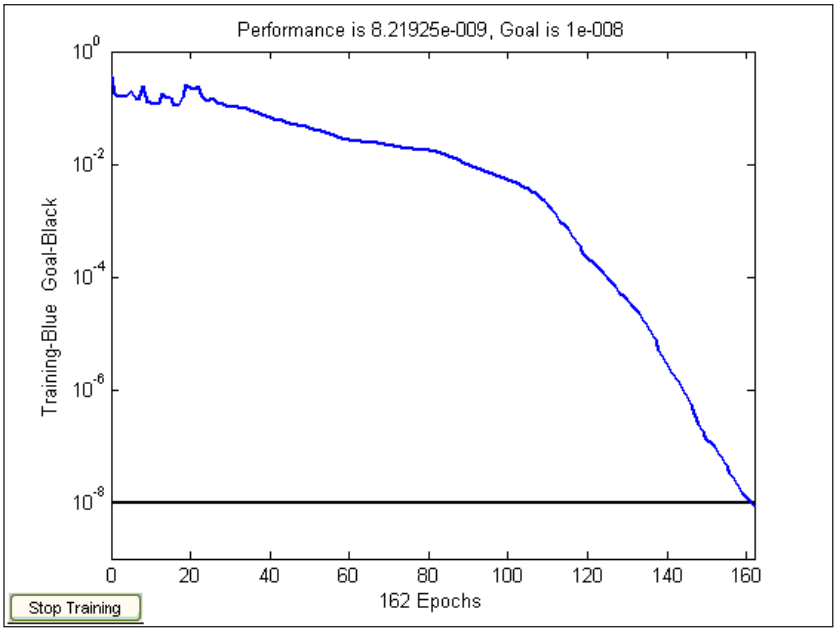

Figure 13: Training Process after Reaching the Target.

After the training process, we will store the optimal weights in the following files: First layer weights in (tw12A1.xlsx), and Second layer weights in (tw12A2.xlsx). In this process, the training process will be completed.

Phase 3: Results display. At this phase, the network has reached the optimal weights and the error rate is (1e - 008) as specified in the program. The system execution algorithm will show the implementation of the program as shown in figure (14).

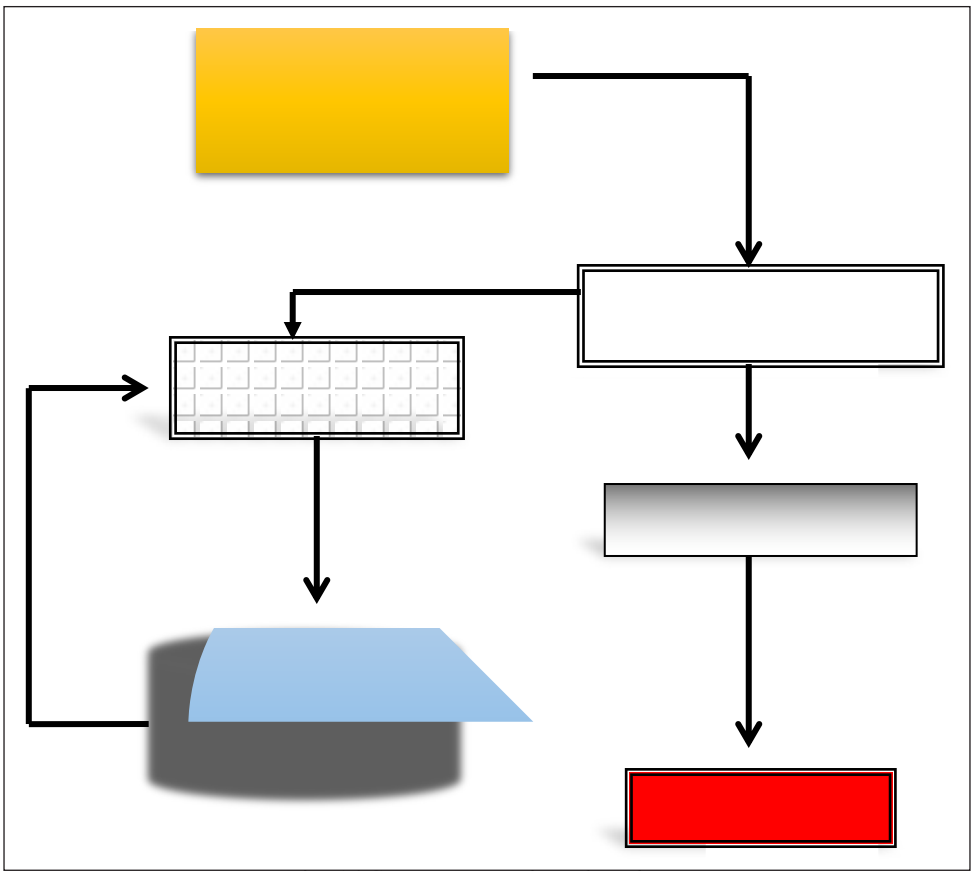

Figure 14: System Execution Flowchart.

8. Results

After the network training process has been completed and the network stabilized, optimal weights were fixed. The optimal weights were as follows:

First Layer Weights							
0.086215	-0.57436	3.533441	-2.93876	-0.44767	2.751792	-1.59311	-1.01671
-1.68782	4.650491	2.960291	-2.36005	3.035467	-3.48811	-2.58521	1.93409
-0.867	-0.23238	-1.07187	-3.56577	-0.4036	2.14511	-3.62901	4.161768
0.65528	0.579847	5.10014	1.337328	-1.02599	-1.55992	2.377706	1.255857
1.039172	0.609302	-3.46601	-1.79237	2.732336	2.617784	-1.5329	0.028184
0.395555	1.019462	-1.69634	-3.52505	0.288602	0.033322	0.430488	4.446128
0.364437	0.607263	2.547216	-3.9192	-2.4094	1.703695	-2.80917	2.680096
-0.55634	-0.54893	-4.4613	2.22663	1.561395	-0.64726	1.777918	1.269109

Output Layer Weights							
4.873026	1.552632	0.551137	0.237712	4.416713	-2.41254	0.949172	0.982889
1.449573	-2.42792	-0.63885	3.706028	2.046153	1.166462	-1.57987	2.453318
-1.51668	3.000589	2.000011	-3.72829	-4.11052	-2.59101	-0.21106	0.30809
1.133963	-2.19818	1.114627	4.406384	-4.49996	-0.2915	-1.52938	1.720377
-4.10399	0.628306	2.642804	-1.75251	3.862018	-3.55477	-0.2617	-2.59305
-1.61042	3.236769	3.734545	-2.5718	-3.07409	3.396116	-2.84271	-1.34713

First Layer Bias
-6.41264
0.425854
-6.13718
-6.79056
-2.44007
-6.56816
-2.08327
-3.09667

Second Layer Bias
-9.16432
-7.88765
2.823688
-0.31931
1.019339
1.908785

- Iteration =162
- Error rate = 1e-008
- The time spent by the (Rporp) network in learning is 37 seconds
- The time spent by the (SBP) network in learning is 3.26 minutes
- Number of input vector for training = 2616

We then tested the system on the following data: Number of input vector to test the network = 6

Test Data													
1028	20	1	0	0	0	0	1	1	0	0	0	0	0
1031	20	0	0	1	0	0	1	1	0	0	0	0	0
1025	20	1	1	1	1	0	1	0	0	1	0	0	0
1030	20	1	1	1	1	1	1	0	0	1	0	0	0
1031	514	1	0	0	0	0	1	0	0	0	0	0	0
1027	514	1	0	0	0	1	1	0	0	0	0	0	0

The first four rows are relational data and the remaining non-relational, the results of the test as follows:

Out =					
1	0	0	0	0	0
1	0	0	0	0	0
0	0	1	0	0	0
0	0	1	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0

Target =					
1	0	0	0	0	0
1	0	0	0	0	0
0	0	1	0	0	0
0	0	1	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0

1 Detection of the relational package 100%.
2 Detection of non-relational package 0%.
3 Detection of the parasitism trained on the network 100%
4 Detection of intrusive non-trained network 0%

9. Conclusion

Several conclusions have been made, so there will be conclusions about the intrusion detection system, conclusions about the neural network, and a combination of the two, as mentioned below:

1. intrusion detection system is a very important tool and must be used in the computer system.
2. All types of intrusion detection systems need to be updated, so there is no complete intrusion detection system for long periods of time.
3. Neural network methods and types allow the researcher to select the appropriate and suitable network for the problem.
4. The more the type and quantity of data used in training, the better the accuracy and efficiency of the network.
5. The quality of data used for training should be appropriate to the real problem and if not appropriate, the training will make the neural network away from solving the problem.

10. References

1. Abdulla, Saman Mirza. (2006), "*Misuse Intrusion Detection System Using Neural Network"*, College of Engineering, Koya University, Iraq.
2. Abdulqadir, Fadi M, (2006), "*Dot Net Networks & TCP/IP Programming"*, Tariq, Amman, Jordan.
3. Anastasiadis, Aristoklis D., (2005), *"An Efficient Improvement of the Rprop Algorithm"*, Dept. of Information Systems & Computing, Brunel University, United Kingdom.
4. Cannady, J., (1998), *"Artificial Neural Network for Misuses Detection"*, School of Computer and Information Science, Nova Southeastern University.
5. David Wagner, (2002), *"Mimicry Attacks on Host Based Intrusion Detection Systems"*, University of California, USA.
6. Jasim, Yaser A, Hasoon, Safwan O, (2013), *"Diagnosis Windows Problems Based on Hybrid Intelligence Systems"*, Journal of Engineering Science & Technology(JESTEC), Vol. 8, Issue 5, Pages 566-578, School of Engineering, Taylor's University, Malaysia.
7. Jasim, Yaser A, Thabit, Thabit H, (2015) *"A Design of Windows 7 Troubleshooting Software Using Hybrid Intelligence Systems"*, International Journal of Engineering Research & Management Technology, Vol.2, Issue 2, India.
8. Negnevitsky, Michael, (2011), *"Artificial Intelligence: A Guide to Intelligent Systems"*, 3rd edition, Addison Wesley.
9. Northcuttt, Stephan, Novak, Judy, (2001), *"Network Intrusion Detection _An analyst 's Handbook"*. www.newwriders.com
10. Szepesvari, Csaba, (2009), *"Algorithms for Reinforcement Learning"*, Alberta University, Canada.
11. Tan, K.M.C., Collie, B.S., (1997), *"Detection and Classification of TCP/IP Network Services"*, In proceeding of the Computer Security Applications Conference, pp. 99-107.
12. Thabit, Thabit H., Jasim, Yaser A., (2017), *"Applying IT in Accounting, Environment and Computer Science Studies"*, 1st Edition, Scholars' Press.
13. Yasen, Khaled N, Jasim, Yaser A, Thabit, Thabit H, (2015), *"Benefits of Relay Station to Enhance Network Signal"*, International Journal of Education and Science Research Review, Vol. 2, Issue 3, India.