APPLYING ADAPTIVE FUZZY NEURAL ALGORITHM FOR INTRUSION DETECTION

Lec. Mokhtar Mohammed Hasan
Ass. Lec. Noor Adnan Ibraheem
University of Baghdad/ College of Science for Women/Department of Computer Science

الخلاصة

معظم التطبيقات التي يتم الدخول عليها من خلال شبكة الانترنيت لها طرق خاصة لتمييز المستخدمين المخولين بالدخول لها عن المتطفلين, هذه الطرق أغلبها تقليديه تتم عن طريق المقارنه بين المعلومات المدخله من قبل المستخدم وتلك المخزونه بالنظام, النظام المقترح يقوم بعمليه اكتشاف المتطفلين بأستعمال طريقه جديده يطلق عليها طريقه الشبكه الضبابيه التكيفيه والتي لها القدره على أكتشاف المتطفلين بنفس الفتره الزمنيه حتى عند ازدياد عدد مستخدمي النظام. النظام المقترح يتكون من مرحلتين, المرحله الاولى تتضمن مراقبة جميع الحوادث والفعالبات التي تحدث بالنظام وتحليلها,اما المرحله الثانيه فتتضمن تحديد الاختراق والذي يكون بنوعين الاستعمال الطبيعي والاستعمال الخاطىء باستعمال الشبكه الضبابيه التكيفيه وهي الطريقه المقترحه في هذا البحث لتمييز المرور الطبيعي والمرور غير الطبيعي في النظام.

ABSTRACT

Many Network applications used as remote login have some ways for detecting the intruders which are classical ways applied by comparison of operations between login user interface and system stored information. The proposed system tried to detect the intrusions happened by the network intruders using new technique called Adaptive Fuzzy Neural Network which have the ability to detect the intrusions at the same time even if the number of users is large. The proposed system consists of two stages, the first stage is for monitoring all events that happen and analyzing them, and the second stage is to detect intrusions. The detection operation combines anomaly intrusion detection and misuse intrusion detection using the Adaptive Fuzzy Neural Network system, which is a suggested method in our paper used to learn the normal network traffic and detect the abnormal traffic.

INTRODUCTION

Intrusion detection is the process of monitoring the events occurring in a computer system or network and analyzing them for signs of intrusions, defined as attempts to compromise the confidentiality, integrity, availability. Intrusions are caused by attackers accessing the systems from the Internet, authorized users of the systems who attempt to gain additional privileges for which they are not authorized, and authorized users who misuse the privileges given to them. Intrusion Detection Systems (IDS) are software or hardware products that automate this monitoring and analysis process [1].

Intrusion detection is classified into two types: misuse intrusion detection and anomaly intrusion detection. Misuse intrusion detection uses well-defined patterns of the attack that exploit weaknesses in the system and application software to identify the intrusions. These patterns are encoded to match against the user behavior to detect intrusion. Anomaly intrusion detection uses the normal usage behavior patterns to identify the intrusion. The normal usage patterns are constructed from the statistical measures of the system features. The behavior of the user is observed and any deviation from the constructed normal behavior is detected as intrusion [2].

Since the system is concerned with the development of an effective, simple Fuzzy logic system for modeling of ill-define dynamical system used. The system represented as a feed forward neural network. Supervised linear back propagation learning algorithm has been applied to model a system through identifying the Fuzzy parameters. The Fuzzy learning system with the training algorithm is called the Adaptive Fuzzy Neural Network (AFNN). [3].

It is in this context that we present an overview of Intrusion Detection system that uses the content of Fuzzy Neural Network (FNN) for adopting the proposed IDs system. In the next section an overview of Fuzzy Neural Network is explained. After that the proposed system is described. Then the results from different experiments are presented. Finally conclusions are expressed.

FUZZY NEURAL NETWORK: AN OVERVIEW

Unlike classical logical systems, Fuzzy logic aims at modeling the imprecise modes of reasoning that play an essential role in the remarkable human ability to make rational decisions in an environment of uncertainty.

Fuzzy logic allows variables to be partial members of a particular set and uses generalization of the conventional Boolean logical operators to manipulate this information. By allowing partial membership of a set, it is possible to represent the smooth transition from one rule to another as the input is varied smoothly, which is a very desirable property in modeling and control application [4].

Our approach deploy in a different manner that uses the Fuzzy Neural Network. There are several reasons for using such method, which are:

- Most papers used the classical Neural Network such as back propagation for IDS, so we want to adopt a new technique.
- The normal approach will slow down as the number of users increases because we need to compare all the data base in order to reach the specified user. Our approach need no time as an average, so that all the information related to each user is propagated through the Fuzzy Network learning, i.e. we need to supply the input to this net in order to discover the confidentiality of the user.
- The FNN has a significant property that is the ability to confident users even the input data is not complete, i.e. if we use part of the information as the input for Fuzzy Network. This net will still be able to recognize the confident user like supplying the data entering without operations and vice versa.

- Fuzzy Set Basic Operations [5, 6, 7]

A Fuzzy set **F** in a universe of discourse **U** is characterized by a membership function μ_F which takes values in the interval [0,1], namely:

μ_F: U $\rightarrow$ [0,1]

A Fuzzy set may be viewed as a generalization of the concept of an ordinary set whose membership function takes only two values {0,1}.

Let A and B be two Fuzzy sets in U with membership functions μ_A and μ_B respectively. The set theoretic operations of union, intersection and complement for Fuzzy sets are defined via their membership functions as follows :

- The membership function $\mu_{A \cup B}$ of the union **A** $\cup$ **B** is pointwise defined for all u$\in$ U by :

$$\mu_{A\cup B}(u) = Max\{\mu_A(u), \mu_B(u)\}$$

- The membership function $\mu_{A\cap B}$ of intersection **A** $\cap$ **B** is pointwise defined for all u$\in$ U by:

$$\mu_{A\cap B}(u) = Min\{\mu_A(u), \mu_B(u)\}$$

- The membership function μ_A of the complement of a Fuzzy set is pointwise defined for all u $\in$ **U** by :

$$\mu_A(u) = 1 - \mu_A(u)$$

Membership Function (MF) is the function which corresponds to a real number between 0 and 1 to any generic element of the Universe of Discourse. There are different shapes of membership functions like, Triangular, Trapezoidal, Gaussian, and Singleton.

Adaptive Fuzzy Systems Architecture

The fundamental configuration of a Fuzzy logic system consists of four basic blocks; the Fuzzy role base, the Fuzzy interface engine, the fuzzifier, and the defuzzifier, as shown in Figure (1).

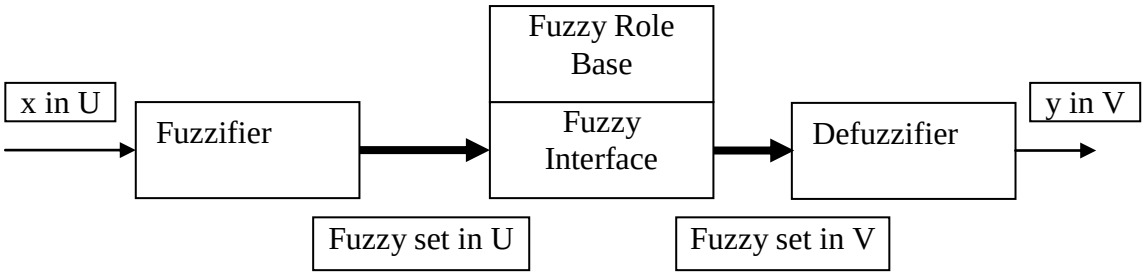

Figure (1) Basic Configuration of Fuzzy Logic Systems

There are different types of fuzzifiers and defuzzifiers. Several combinations of the Fuzzy interface engine, fuzzifier, and defuzzifier may constitute useful Fuzzy logic system. If the Fuzzy logic system can be represented as a feed forward network, then the idea of back propagation training algorithm can be used to train them.

Adaptive Fuzzy System Structure

The most useful class of fuzzifier is the center average [8, 4], of the form:

$$f(\underline{x}) = \frac{\sum_{j=1}^{M} y_j(\mu_{F_j}(y_j))}{\sum_{j=1}^{M}(\mu F_j(y_j))} \quad (1)$$

Where M is the number of Fuzzy rules, y_j is the center of Fuzzy set F_j, which is, a point in the universe of discourse V when μ_{Fj} (y) achieves its maximum value, and μ_{Fj} (y) is given by a product interface engine. Hence using product operator becomes:

$$f(\underline{x}) = \frac{\sum_{j=1}^{M} y_j(\prod_{i=1}^{n} \mu_{F_j}(y_j))}{\sum_{j=1}^{M}(\prod_{i=1}^{n} \mu F_j(x_j))} \quad (2)$$

Where n is the number of input variables.

In order to develop training algorithms for this Fuzzy logic system, the functional form of μ_{Fi} (x_i) must be specified. The bell shaped membership function, based on the normal distribution of the grades of the membership [9, 10], was proposed, i.e. the membership function will be given by:

$$\mu_{F\,i}(x_i) = \exp\left[-\left(\frac{x_i - m_i}{\sigma_i}\right)^2\right] \quad (3)$$

Where m_i and σ_i are, respectively, the center and width of the bell-shaped function of the i^{th} input variable. By combining Equations (2, 3), the overall function of the Fuzzy logic system is:

$$f(\underline{x}) = \frac{\sum_{j=1}^{M} y_j\left[\prod_{i=1}^{n} \exp\left[-\left(\frac{x_i - m_{ij}}{\sigma_{ij}}\right)^2\right]\right]}{\sum_{j=1}^{M}\left[\prod_{i=1}^{n} \exp\left[-\left(\frac{x_i - m_{ij}}{\sigma_{ij}}\right)^2\right]\right]} \quad (4)$$

This equation represents a Fuzzy logic system with center average defuzzifier, product interface rule, non-singleton fuzzifier, and bell-shaped membership function. Equation(4) can be implemented on a Forward Neural Network. This connectionist model combines the approximate reasoning of Fuzzy logic into a five layer neural network structure [11], as in Figure (2).

Associated with each node in a typical neural network is an integration function, which serves to combine information or activation from the other nodes. This function

X_i^L provides the net input of the i^{th} node in layer L. A second action taken by each node is to output an activation value as a function of its net input:

$$O_I^L(k) = g(X_I^L(k)) \quad (5)$$

Where g (.) denotes the activation function.

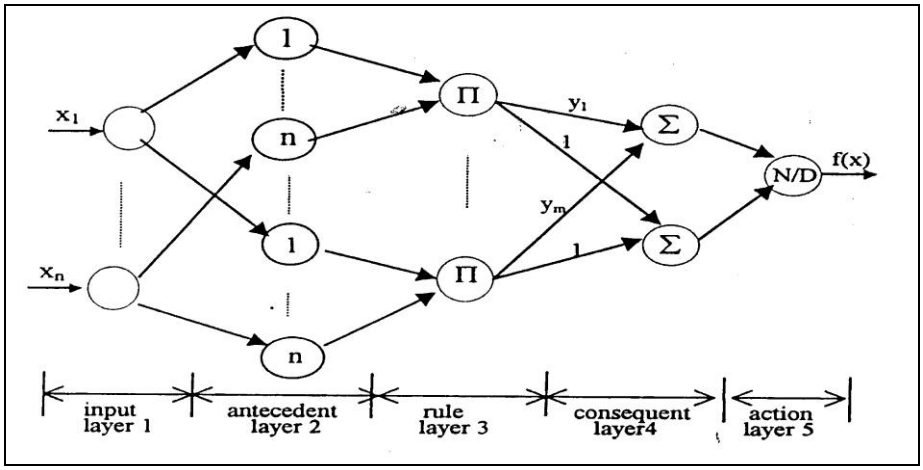

Figure (2) Adaptive Fuzzy System as a FNN

The basic function of the nodes in each layer would be defined as follows:

a) Input layer

The nodes in this layer just transmit their inputs to layer 2:

$$X_1^1 = x_1, X_2^1 = x_2, \ldots, X_n^1 = x_n \quad (6)$$

$$O_i^1 = X_i^1 \quad (7)$$

Where i = 1,2, ……, n and n is the number of the input linguistic variables.

b) Antecedent Layer

The output from this layer is:

$$O_i^2 = \mu_{Fi}(X_i^2) \quad (8)$$

Where X_i^2 is the input to node i in layer 2 and F_i is the linguistic label assigned to Fuzzy set (small, large, etc.). Using Equation(3), Equation(8) can be rewritten to have:

$$O_i^2 = \exp\left[-\left(\frac{x_i^2 - m_{ij}}{\sigma_{ij}}\right)^2\right] \quad (9)$$

Where m_{ij} and σ_{ij} are, respectively, the center and width of the bell-shape function of the i^{th} input of the j^{th} rule.

c) Rule Layer

The output from each node in this layer is dictated by the firing strength of the corresponding rule. With the proposed scheme (i.e. Equation(4)), the rule node perform the Fuzzy product operation; Therefore:

$$z_j = O_j^3 = \prod_{i=1}^{n} X_{ij}^3 \tag{10}$$

Where X_{ij}^3 denotes the i^{th}. Input to node j in layer 3.

d) Consequent Layer

The upper node of these layer sums all outputs from the rule layer with action strengths (y_i) and the lower node those with unity strength:

$$N = O_1^4 = \sum_{j=1}^{M} y_j X_j^4 \tag{11}$$

$$D = O_2^4 = \sum_{j=1}^{M} X_j^4 \tag{12}$$

Where N and D are the numerator and denominator of Equation(4)

e) Action Layer

The network output would be pumped out the single node layer.

$$f(\underline{x}) = O^5 = \frac{X_1^5}{X_2^5} = \frac{N}{D} \tag{13}$$

PROPOSED SYSTEM

Adaptive Fuzzy System Training Algorithm

Fuzzy-neural network will be used here in an intrusion detection system. The data sets from the monitoring system data base are used for the training and the testing of the fuzzy-neural networks.

Based on the error back propagation algorithm, the goal is to determine a Fuzzy logic system $f(\underline{x})$ in the form of Equation(4), which minimizes the error function:

$$E(k) = \frac{1}{2}\sum_{j=1}^{P}\left[f_j(\underline{x}(k)) - d_j(k)\right]^2 \tag{14}$$

Where P is the number of outputs and $d_j(k)$ is the j^{th} desired output at time k. Without loss of generality, Multi-input single-output (MISO) Fuzzy logic system was considered in this paper. A multi-output system can always be decomposed into a group of single-output systems [12,13], therefore for P=1, Equation(14) is reduced to:

$$E(k) = \frac{1}{2}(f(x(k)) - d(k))^2 \quad (15)$$

According to Equation(4), if the number of rules is M, then the problem becomes training the parameters y_j, m_{ij}, and σ_{ij} such that E(k) is minimized.

Based on the back propagation training algorithm the iterative equations for training the parameters y_j, m_{ij}, and σ_{ij} are:

$$y_j(k+1) = y_j(k) - \eta\left(f(\underline{x}(k)) - d(k)\right)\frac{1}{D}z_j \quad (16)$$

$$m_{ij}(k+1) = m_{ij}(k) - 2\eta\frac{z_j}{D}\left(f(\underline{x}(k)) - d(k)\right)\cdot\left(y_j(k) - f(\underline{x}(k))\right) \cdot\left(\frac{x_i^2(k) - m_{ij}}{(\sigma_{ij})^2}\right) \quad (17)$$

$$\sigma_{ij}(k+1) = \sigma_{ij}(k) - 2\eta\frac{z_j}{D}\left(f(\underline{x}(k)) - d(k)\right)\cdot\left(y_j(k) - f(\underline{x}(k))\right) \cdot\left(\frac{(x_i^2(k) - m_{ij})^2}{(\sigma_{ij})^2}\right) \quad (18)$$

Where η is the learning rate, Equation (16), (17), and (18) perform an error back propagation procedure.

Our system combines the two distinct intrusion detection approaches, (anomaly and misuse). Combining these two approaches enables bypassing drawbacks that appear in each approach.

The proposed system has six main components to be maintained along the project implementation:

i. **Subject**: The users and their identification information, for the machine that they are working with, represent the subjects here.
ii. **Object:** This term refers to the resources managed by the system.
iii. **Audit records**: This is the basic element of IDS because it is generated by the server machine (where the IDS is residing) in response to actions performed or attempted by subjects on objects such as user login, command execution, file access, and time of accesses. These records are arranged into database to be referenced along the execution of IDS. Each record field will be stored in the string type to increase the analysis and detection speed.
iv. **Profiles**: are structures that characterize the behavior of subjects. The more important profiles fields are: the IP address, PC name, type of event (delete, create, rename, open), the time of accessing (starting and ending) per day (in hour).
v. **Anomaly records**: are generated when abnormal behavior is detected.
vi. **Activity rules**: actions taken when some condition is satisfied, which update profiles, detect abnormal behavior, relate anomalies to suspected intrusions, and produce reports.

The block diagram for the proposed system is shown in Figure (3). Each part of the proposed system will be discussed in detail in the following subsections.

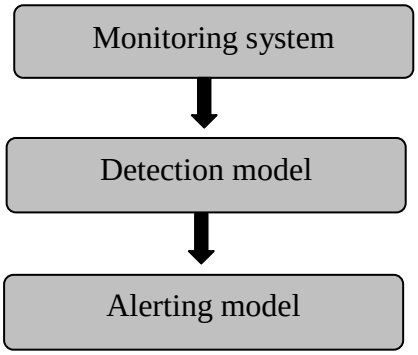

Figure (3) Proposed IDS block diagram

The Proposed IDS Architecture

The architecture of the proposed IDS is shown in Figure (4) and is stated in the following points:

- **i.** An audit record is created. This occurs when an action happens, such as open file.
- **ii.** The detection engine searches for anomaly behavior comparing the current data with the historical data.
- **iii.** Anomaly record is generated when abnormal behavior is observed. The anomaly record is forwarded to a number of different sub-systems for security officer, response, and storage.
- **iv.** The security officer is notified through visual method such as showing the message as alarm.
- **v.** A response is generated. Responses include actions such as shutdown the target, restart the target or notify the security officer.
- **vi.** Store the anomaly record to use it in expecting new type of intrusion.
- **vii.** Reports can be generated and forwarded to the security officer.

The Proposed Monitoring System

Host based monitor activities are normally executed only by an administrator. Operating systems log any event user accounts are added, deleted, and renamed. Host-based IDS can detect an improper change as soon as it is executed. Host-based IDS can also audit policy changes that affect what systems track in their log.

The proposed system does not contain any special features for dealing with complex actions that exploit a known or suspected security flaw in the target system; indeed, it has no knowledge of the target system's security mechanisms or its deficiencies. By detecting the intrusion, however, the security officer may be better able to locate vulnerabilities.

The proposed system was built as an administrator for a LAN network with a two personal computer connected to server. These two computers will operate under windows, the first step is start up the server, no network services without server. Each client registers himself as an employee. The employee has to fill registration form

which asks him about many personal aspects. This personal information is decided previously. After that, employee returns to his office to get himself ready for the next step through which the system learns more about him, as shown in Figure (5). Monitoring Algorithm can be characterized as seen in algorithm (1)

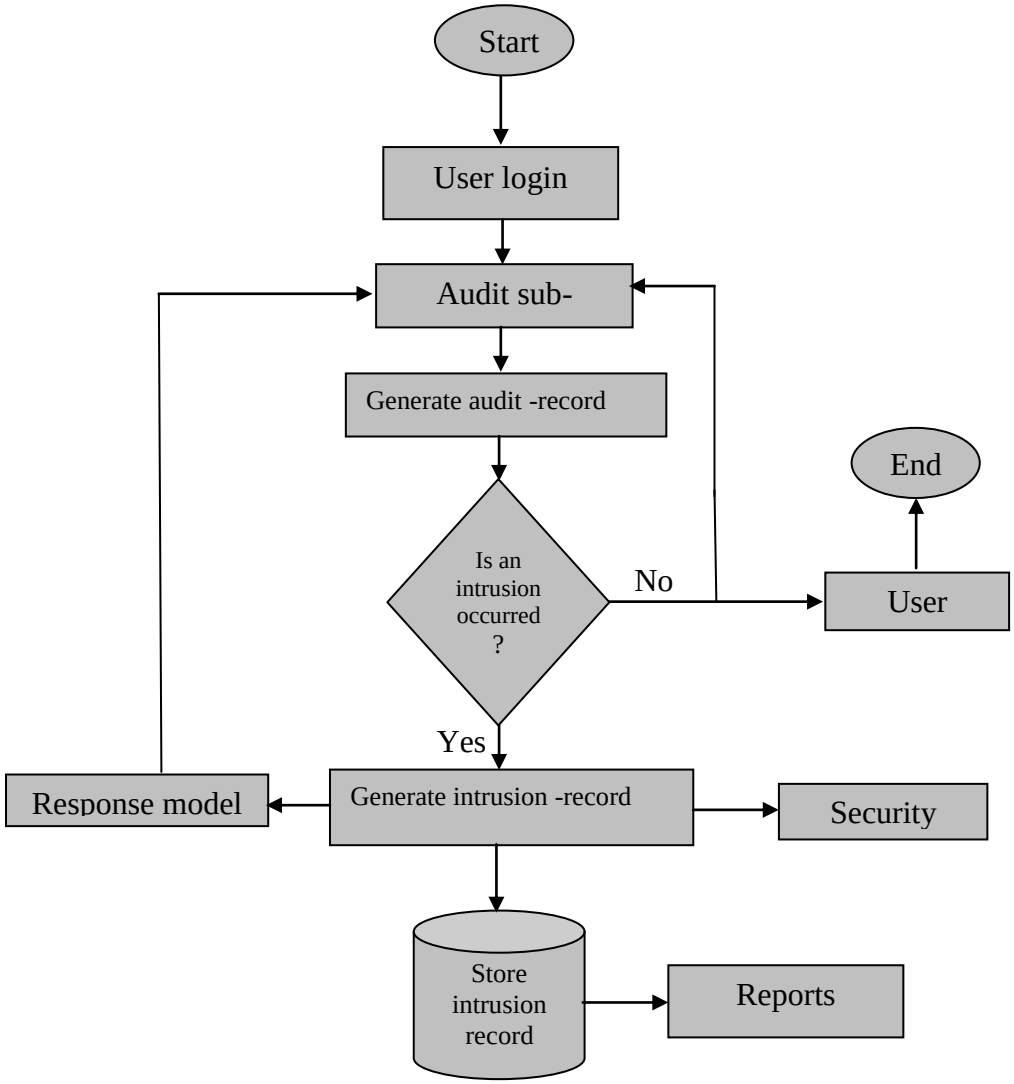

Figure (4) The architecture of the proposed system

Algorithm (1):

- Start when the computer is turned-on.
- Specify the target system that wants to monitor it.
- Monitoring the specified target system.
- Check if any operation of (delete, create, rename, and open) on the file or folder has occurred then add a record in database, go to step 3.
- End.

Detection Model

There are two categories of intrusion detection system. They are called network-based IDS and host-based IDS. Networks have made computer systems easily

accessible from remote location. This allows legitimate users to access computer systems and information on those computer systems and this may lead to a large number of intrusions.

A single intrusion of a computer network can result in the loss or unauthorized utilization or modification of a large amount of data and cause users to question the reliability of all the information on the network. An important shortcoming of IDS is that can detect when an event is unusual, which may or may not indicate an attack.

This model acts the heart of the proposed system, it consists of two phases: Encoding phase and Detection phase.

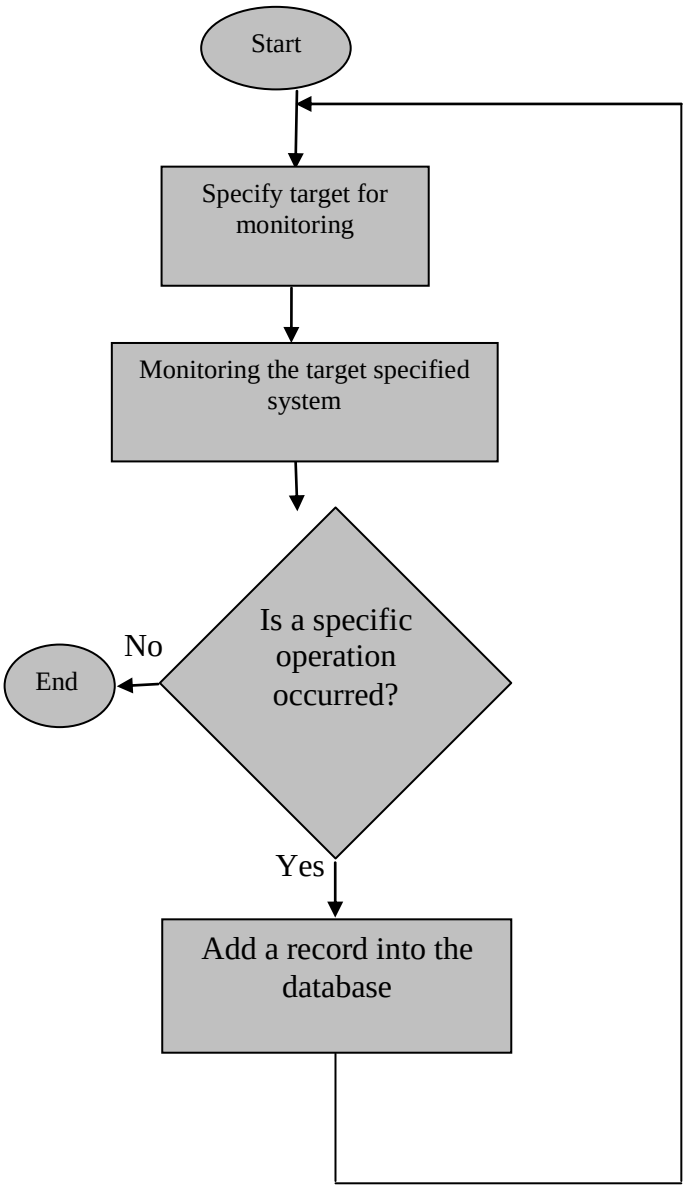

Figure (5) The proposed monitoring system

Encoding Phase

The main function of this phase is reading the records from monitoring system database and encoding it. The input files to the Fuzzy Neural Network must be encoded. These files must be in decimal format, therefore, all log files must be converted to decimal format, as shown in the figure (6). Encoding phase explained in the following algorithm.

Algorithm (2):

- Start while DB contain operation record.
- If rec. no from MSDB larger than zero, then go to step 3.
- Else go to step 11.
- Read record from MSDB.
- Set coding = 0 , x =0.
- x = x +1.
- If x < field no. then go to step 7.
- Else go to step 2.
- 7. Execute Query
- Query = "select target field, code, ser, max (ser). As m ser from code table where target field = record, field (x)"
- Check if rec. no. Of Query larger than zero then go to step 9
    - Else go to step 10
- Coding (x + ser) = Query (code).
- x = x + Query (m ser) and go to step 5
- End.

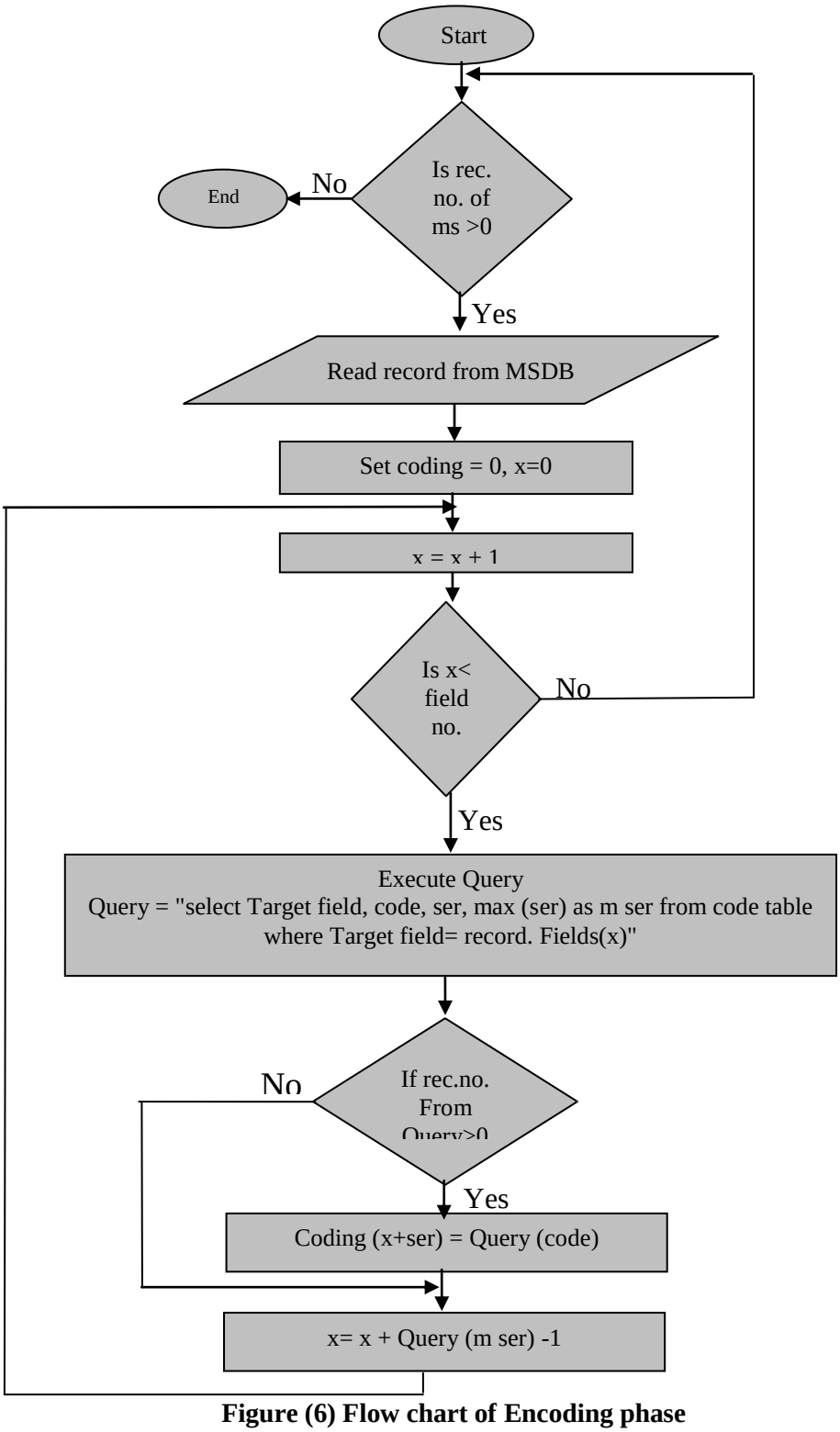

Figure (6) Flow chart of Encoding phase

Detection Phase

The purpose of this phase is to detect intrusions by using Fuzzy Neural Network. In this phase, two kinds of intrusions will be detected, anomaly and misuse intrusion.

a. Anomaly Detection

This is the first detection mechanism in the proposed system. The structure of the anomaly detection is shown in Figure (7).

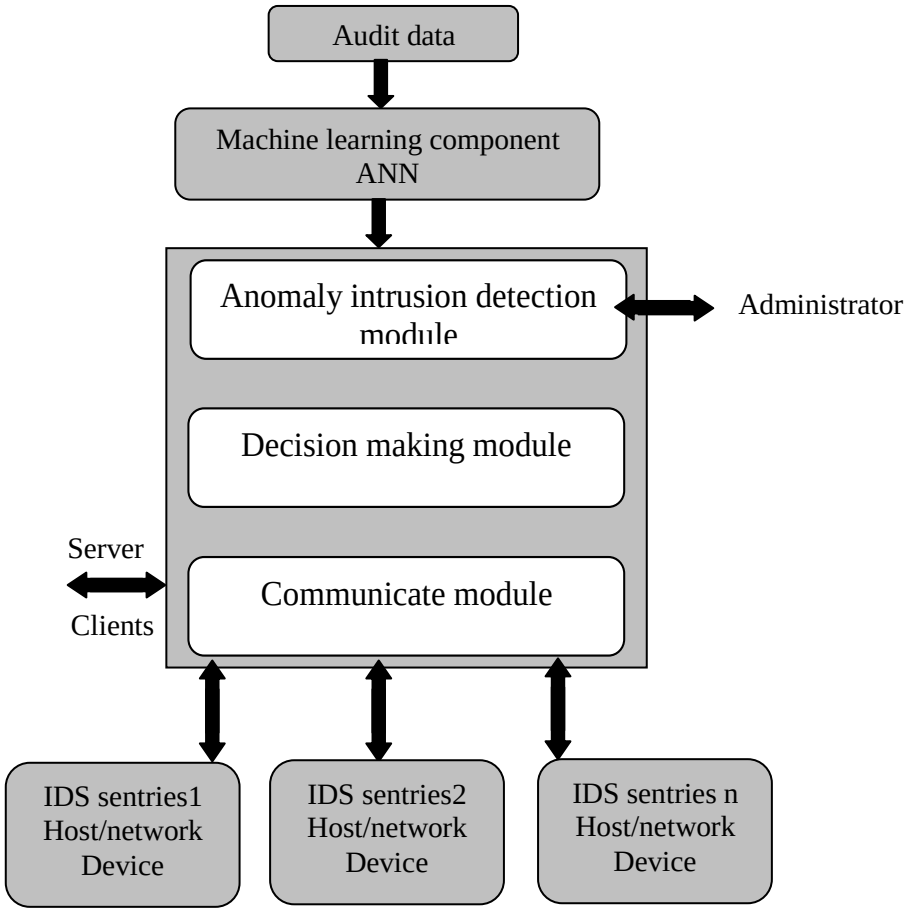

Figure (7) Structure of the anomaly detector

- **Machine Learning Component:** This part uses fuzzy-neural (back propagation) to learn normal patterns of system behavior. This normal behavior is stored in profiles. This allows the system to adapt to new environment.
- **Anomaly Intrusion Detection Module:** This module extracts patterns of an observed audit trail and compares these new patterns with the normal patterns. If the similarity of the sets of patterns is below a specified threshold, the system alarms of an intrusion.
- **Decision- Making Module:** This will decide whether or not to activate anomaly intrusion detection and integrates evaluation results provided by the anomaly intrusion detection.
- **Communication Module:** It is the bridge between the intrusion detection and decision- making module.
- **Intrusion Detection Sentries:** Pre-process audit data and send results to the communicatION module.

Algorithm (3) shows the steps of anomaly detector. Figure (8) shows Neural Network for anomaly detector.

Algorithm (3):

1. Begin
2. Monitoring system turn on.
3. Select specific fields from monitoring system data base and arrange them in vectors
4. While .T. DO
   Begin
   Read vector from monitoring system data base.
   Apply encoding algorithm.
   Input vector to the designed FN
   If the output of FN is (1) then
   Mark the specified Vector as normal
   Else mark the specified vector as abnormal
5. If no more vector found then
   Exit
6. End (while)
7. End.

The input values to the FN are coded as x1, x2… x9.

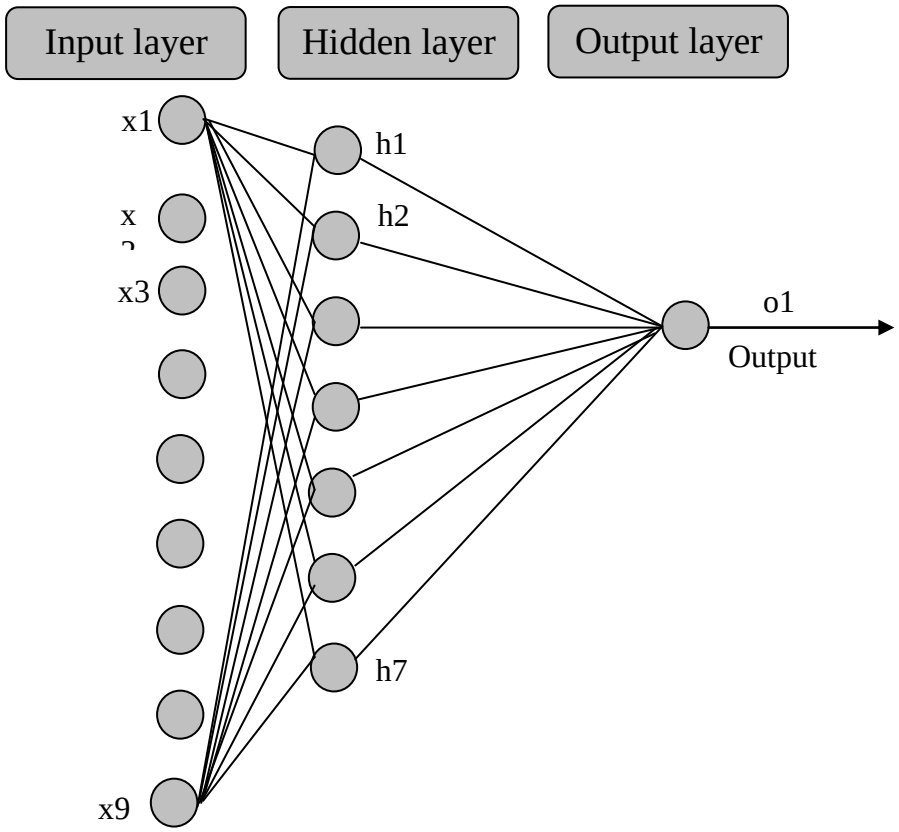

Figure (8) Neural network architecture used for anomaly detector

Where:

x1: the destination name.
x2: create event.
x3: delete event.
x4: rename event.
x6: the attribute of folder (hidden or not).
x7: the file
x8: the folder.
x9: the event in time.

x5: open event.

The main task of this phase is to identify intrusion patterns by considering the threshold that was computed in the fuzzy-neural network with back propagation training algorithm. The detector was designed using Fuzzy Neural Network, which utilizes back propagation architecture that consists of (9) input nodes, the hidden layer consist of (7) nodes and one output node.

The FN is designed to provide an output value (0) to indicate a misuse attack, and if output value is larger than error then anomaly intrusion has occurred. If output value is smaller than error then behavior is normal.

The number of hidden layers and the number of nodes in hidden layer is determined based on the process of trial and error.

The hidden layer consists of seven nodes, because this number of nodes gives good rate of accuracy in learning operation. If the number of nodes in hidden layer is increased then the accuracy will be increased but this number of nodes in hidden layer needs more time because of the computations between the nodes.

The output layer consists of one node, because we need either yes or no. It means intrusion or not- intrusion.

b. Misuse Detector

This type of detection involves a comparison of user's activities with the known behaviors of attackers. This detection mechanism matches the current behavior with a signature of known attacks. A signature is a pattern that we want to look at in event records. The structure of the misuse detector is shown in Figure (9).

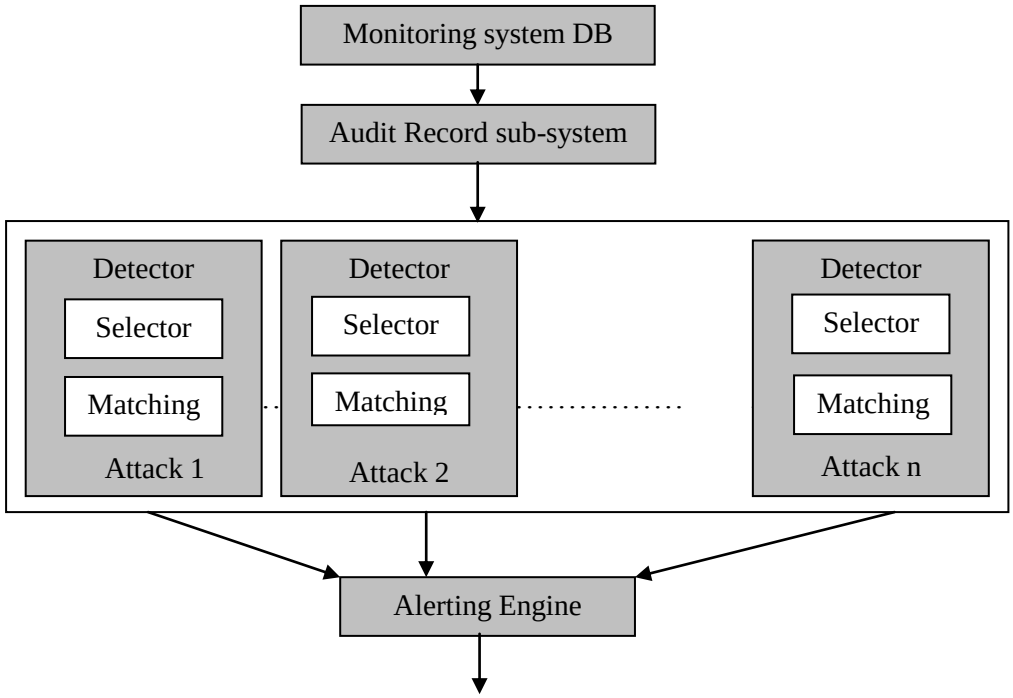

Figure (9) The structure of the misuse detector

The algorithm of the misuse detector is illustrated in algorithm (4).

Algorithm (4):

1. Begin
2. Monitoring system turn on.
3. Select specific fields from monitoring system data base and arrange them in vectors
4. While .T. DO
    a. Begin
    b. Select specific features from records in MSDB depending on the Signature of the attack
    c. Match the selected features with the features of the attack
    d. signature
    e. If the two sets of features match then
        Misuse attack is found
       Else
        Mark the event as normal
    f. If no more attacks found then
        Exit
       Else
       Check another attack type
       End
5. End

Alerting Model

This is the last stage in the proposed system. This stage is concerned with deciding if an event or a set of events is an intrusion or not. It receives the output of anomaly detection or misuse detection part and then gives the result in a report. This report shows if the event is an intrusion and normal. The report also specifies the source that causes the abnormal behavior and the data and time of it. This model is important, since it is used to stop an intruder. Alerting model has many levels of reactions against the intruder, using any one of them depends on the nature of attack. Figure (10) shows this mechanism.

This model may be enough in showing the warning messages or locking some bottoms in the keyboard or freezing the IP that causes that attack.

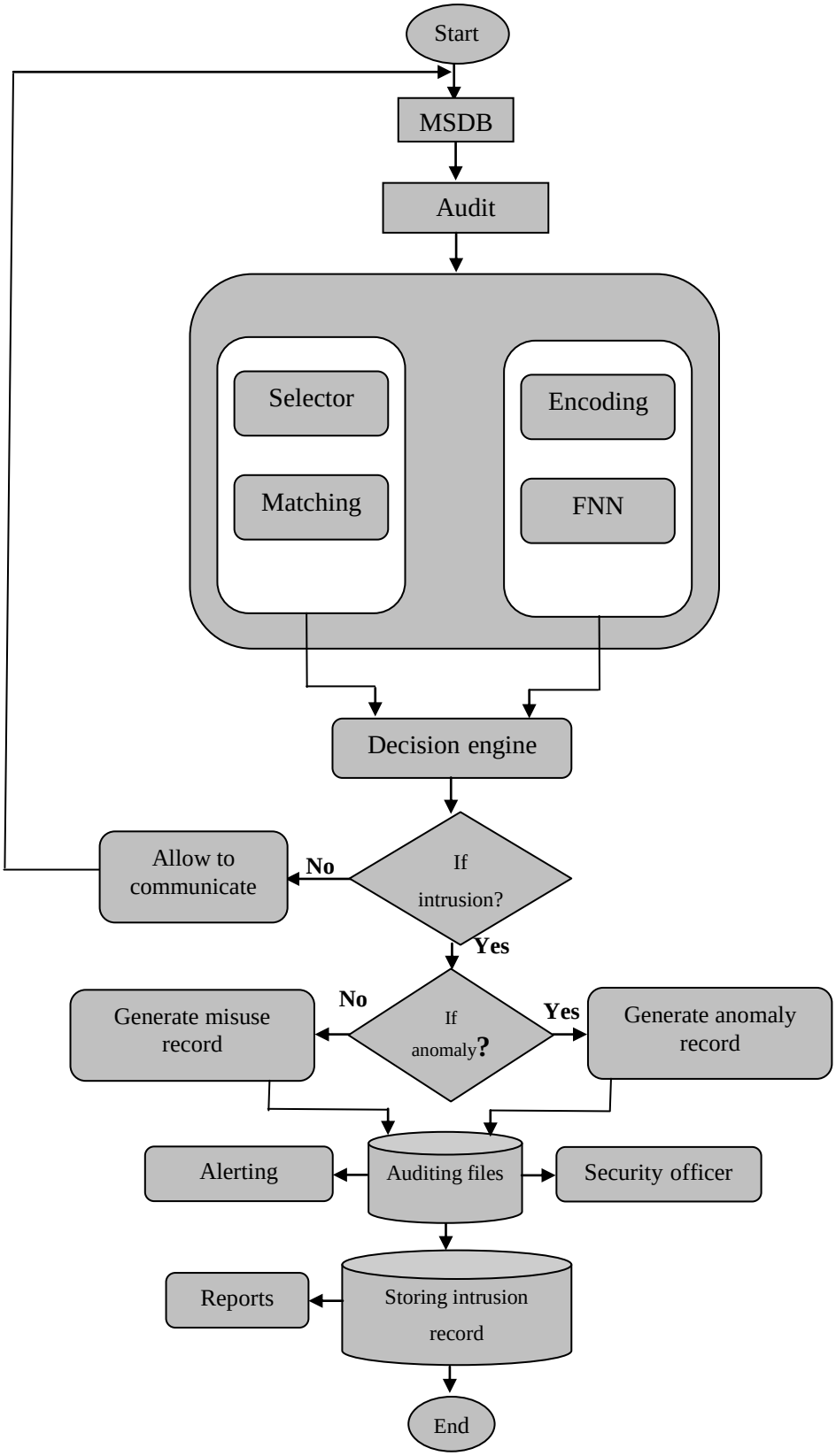

Figure (10) General block diagram of the proposed system

SYSTEM IMPLEMENTATION

The system is divided in two parts: the first one for the user and the second part for Administrator only. The final experiments were conducted to see what percent of the normal patterns and the attacks that are classified correctly.

The first part of the proposed system represents the user behavior by applying the four operation (open, rename, delete, and create) on folder or file and select the file categories. By executing that we obtain the total path and its history in exact time and date. This will record the required information for the user who authorized to work on this IP address by registering correct username and password. The second part of the proposed system that represents the administrator will record the required information to register users like (user name, password, etc).

Since our work is in LAN environment, therefore, all clients in the LAN must be monitored and all events that occurred should be notified and registered in the database. The events that monitoring system monitors are open, rename, create and delete file or folder. Our system monitors the files that have the following extensions (*.sys,*.txt, *.avi, and custom).

The experiments were conducted in two parts, the preliminary experiment and final experiment.

i. **The Preliminary Experiment:** In this test we used both known and unknown attacks in the same file. When we used just three hidden units, no attacks were detected at all. With 5 hidden units, we got a detection rate of 86%. The results from this experiment gave the background for choosing the number of hidden units and iterations used for the training of the fuzzy-neural network in the final experiment. This means that number of hidden units must be over 5; number of iterations must be over 100. There is a huge drop in the error between 3 and 5 hidden units. Table 1 shows Preliminary experiment results, figure (11) shows the error for the preliminary experiment.

Table 1: Preliminary experiment results

Hidden number	Iterations number	Error
2	100	0.3544
3	100	0.3541
4	100	0.0219
5	100	0.0195
6	100	0.0183
7	100	0.0031

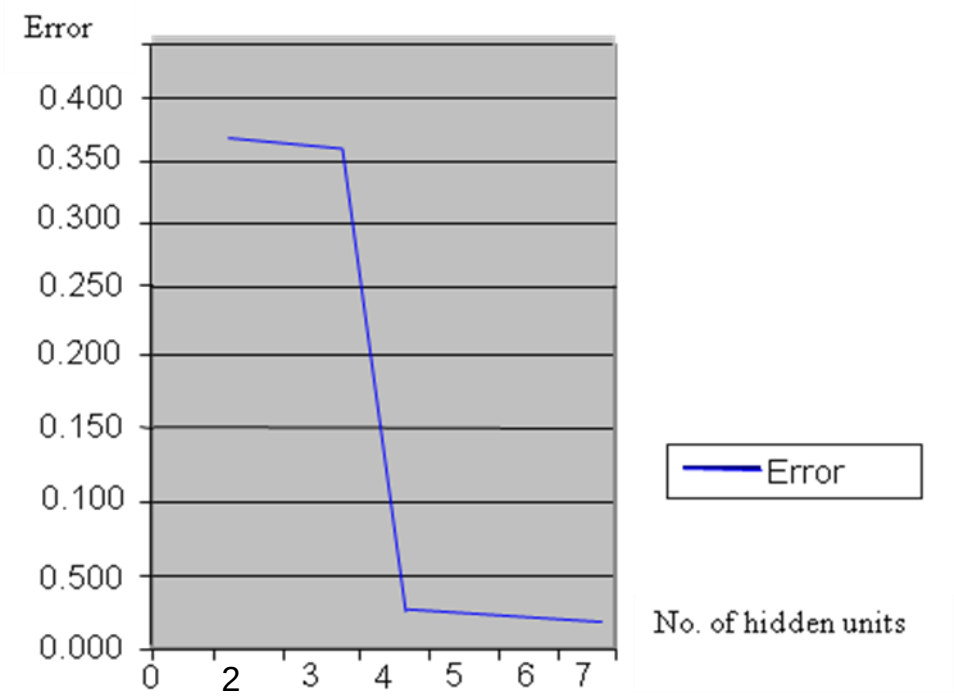

Figure (11): Error for the preliminary experiment

ii. The Final Experiment: The testing was divided into normal traffic, known attacks, and unknown attacks. The classification rate of normal is 82%, but for the known attacks is 82% and the classification rate of unknown attacks is 80%, as shown in table 2:

Table 2: Final experiment results for normal traffic

Hidden number	Iterations number	Error
4	100	0.021995822141
7	100	0.018322100972
12	100	0.018230336374
24	100	0.018132969630
4	1000	0.006137723648
7	1000	0.004811317503
12	1000	0.004608365849
24	1000	0.004337760720
7	5000	0.002035273630
12	5000	0.001919316337
24	5000	0.001778245927
7	20000	0.000985901225
12	20000	0.000917864210
24	20000	0.000841241861

As explained before, the lower error rate is, the better rate normally is, in Figure (12) the differences in the errors when we used 100,1000,5000 and 20000 iterations for the training of the neural network are shown,

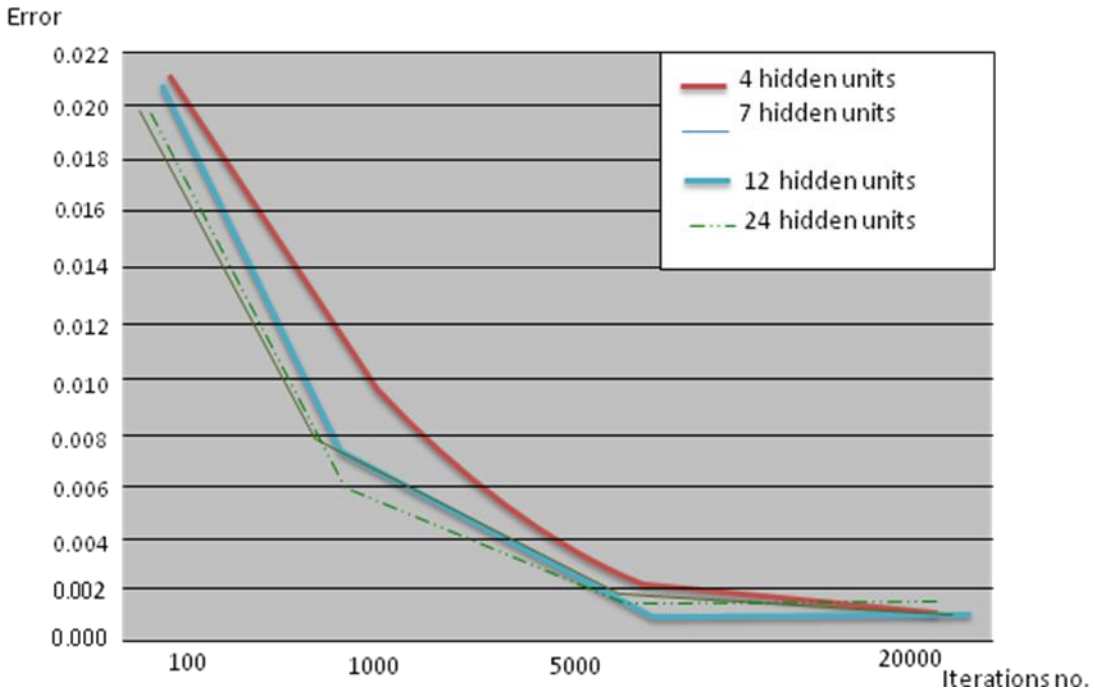

Figure (12): Error comparison

CONCLUSIONS

The proposed system is dedicated to design a network-based intrusion detection system on a network. Using the machine learning component of IDS architecture allows the system to adopt new environments. This makes the proposed system able to detect both of attacks anomaly and misuse attacks. Both anomaly detection and misuse detection are supported by the system. This gives the ability to respond to anomaly and not only to signatures of known attacks. Using Fuzzy Neural Network in anomaly detection shows that it can learn the characteristics of normal behavior and identify instances that are unlike normal behaviors that are anomalies. The training of the neural network requires a very large amount of data to ensure that the results are accurate.

REFERENCES

- Anita K. Jones and Robert S. Sielken, ***"Computer System Intrusion Detection A Survey1"*** Department of Computer Science **,** University of Virginia,2000.
- S.A. Ismaeel Al_Ani & Dr.K.M. Al_Jebory **"*Adaptive Fuzzy System Modeling* "** Control &Computer Eng .Dept. ,University of Technology, Baghdad, 2001.
- H. Debar, M. Dacier, and A. wespi, ***"Towards taxonomy of Intrusion Detection System".*** 1999.
- M. Brown and C. Harris, ***"Neurofuzzy Adaptive Modelling and Control,"*** Prentice–Hall International (UK) Limited, 1994.
- C. C. Lee, ***''Fuzzy Logic in Control System: Fuzzy Logic Controller Part I and II''*** IEEE Transactions on Systems, Man and Cybernetics, Vol. 20, No.pp 404-435, April, 1990.

- J . Barron ***"Putting Fuzzy Login into Focus",*** Byte ,pp. 111-118, April 1993.
- L. Reznik ***"Fuzzy Controllers "***Newnes , 1997.
- S. G. Naoam, "***Design of Fuzzy Logic Controller of Dilatation Column,"*** M.Sc. Thesis, School of Control and Computers, University of Technology, September 1995.
- K.S. Narendra and K. Parthasarathy, ***"Identification and Control of Dynamical Systems Using Neural Networks***", IEEE, Transactions on Neural Networks, vol.1, no.1, pp.4-27, March 1990.
- R. K. Al-Sibakhi, ***"Fuzzy Neural Control of Dynamical Systems,"*** M.Sc. Thesis Electrical Engineering Department, University of Al-Mustonsiriya, Iraq, October 1996.
- Y. M. Chen and K. F. Gill, ***"Application of Fuzzy Neural Network to Control of An Unstable System,"*** IMACS International Symposium on Signal Processing, Robotics and Neural Networks, France, pp.454-457, April 1994.
- L. X. Wang, ***"Adaptive Fuzzy Systems and Control: Design and Stability Analysis,"*** Prentice-Hall, Inc., 1994.
- C. W. Xu, ***"Fuzzy Systems Identification,"*** IEE Proceedings Control Theory and Applications, vol.136, part D, no. 4, pp. 146-150, July 1989.