Intrusion Detection System using Fuzzy Inference System

ANVBS Harikishan[#1], P.Srinivasulu[*2]
[#] *(M.Tech),Department of CSE,QIS College of Engineering and Technology, Ongole, AndhraPradesh,India*
[#] *Professor and HOD,Department of CSE,QIS College of Engineering and Technology, Ongole, AndhraPradesh,India*

***Abstract*—** Network security consists of the provisions and policies adopted by a network administrator to prevent and monitor unauthorized access or denial of a computer network and network accessible resources. Intrusions are the activities that violate the security policy of a system. An Intrusion detection system monitors network or system activities for malicious activities or policy violations and produces reports to a management station. Previously several machine learning algorithms such as neural network ,data mining and many more have been used to detect intrusion behaviour. In the proposed system we have designed fuzzy inference approach for effectively identifying the intrusion activities within a network. The proposed system uses Sugeno Fuzzy Inference system for generation of fuzzy rules and ANFIS editor for experimentation. The experimentation and evaluation of the proposed intrusion detection system are performed with KDDCup99Dataset and we can easily detect whether the records are normal or attack one.

Keywords- IDS, KDD Cup99 Dataset, Fuzzy Inference System, Anfis editor.

I. INTRODUCTION

Network is a group of two or more computer systems linked together. Networked systems are technologies which connect the smart products of embedded systems to form smarter networks. The obvious example of a network system that is open to public access is internet. Due to the commercialization of internet ,computer systems are turning out to be more and more vulnerable to attacks, Though there are a number of ways to provide security such as cryptography, anti-virus, malwares, spywares, etc., it is not possible to provide complete secure systems. So there should be a second line of defence as an Intrusion Detection Systems to detect attacks .An intrusion detection systems watches networked devices and searches for anomalous or malicious behaviours in the patterns of activity in the audit stream. [11]

IDS has emerged as one of the significant field of research in today's world, because there is no chance of a system without vulnerabilities. The main argument here is how to find the attacks in a large quantity of routine communication activities. To detect intrusions of complex and dynamic datasets many data mining and machine learning techniques have been applied previously.[2] Intrusion detection mainly based on two types of techniques.

A) Misuse detection

Catch the intrusions in terms of the characteristics of known attacks or system vulnerabilities[4].

B) Anomaly detection

Detect any action that significantly deviates from the normal behaviour.[4].

II. FUZZY LOGIC AND FUZZY INFERENCE SYSTEM

Fuzzy logic is a form of many-valued logic or probabilistic logic; it deals with reasoning that is approximate rather than fixed and exact. [9].A fuzzy inference system (FIS) is a system that uses fuzzy set theory to map inputs (*features* in the case of fuzzy classification) to outputs (*classes* in the case of fuzzy classification).Fuzzy inference systems (FIS) are one of the most famous applications of fuzzy logic and fuzzy sets theory.[12] They can be helpful to achieve classification tasks, offline process simulation and diagnosis, online decision support tools and process control.There are two types of Fuzzy inference systems.

A) Mamdani fuzzy inference system

Mamdani's fuzzy inference method is the most commonly seen fuzzy methodology. Mamdani's method was among the first control systems built using fuzzy set theory. It was proposed in 1975 by Ebrahim. Mamdani's effort was based on Lotfi Zadeh's 1973 paper on fuzzy algorithms for complex systems and decision processes [1].Mamdani-type inference, as defined for the toolbox, expects the output membership functions to be fuzzy sets .After aggregation, Mamdani method, uses centroid for defuzzification which is a two-dimensional function[16].

B) Sugeno Fuzzy inference system

Sugeno Fuzzy inference system was introduced in 1985. Sugeno output membership function are either linear or constant. Sugeno[14] was one of the first to propose self-learning FIS and to open the way to a second kind of FIS; those designed from data. Even if the fuzzy rules, which are automatically generated from data, are expressed in the same form as expert rules, there is generally a loss of semantic. Since Sugeno's early work, a lot of researchers have been involved in designing fuzzy systems from databases.

III. KDDCUP99DATASET

Since 1999, KDD'99 has been the most wildly used data set for the evaluation of anomaly detection methods. The KDD Cup99 attack dataset is a public repository to promote the research works in the field of intrusion detection.[6]This data set is prepared by Stolfo et al. and is built based on the data captured in DARPA'98 IDS evaluation program. DARPA'98 is about 4 gigabytes of compressed raw (binary) tcp dump data of 7 weeks of network traffic, which can be processed into about 5 million connection records, each with about 100 bytes. The two weeks of test data have around 2 million connection records. KDD training dataset consists of approximately 4,900,000 single connection vectors each of which contains 41 features which includes both of continuous and symbolic type. Among the total 41, 33 are continuous in nature and the rest are symbolic[5]. We use continuous type for our experimentation as majority are continuous in nature. The description of the various features is shown in the Table 1 [3].

Feature name	*Type*	*Description*
Srv_rerror_rate	Continuous	% of connections that have ``REJ" errors
dst_host_same_src_port_rate	Continuous	same_src_port_rate for destination host
Service	Symbolic	network service on the destination, e.g., http, telnet.
Flag	Symbolic	normal or error status of the connection
Src_bytes	Continuous	number of data bytes from source to destination
Dst_bytes	Continuous	number of data bytes from destination to source
Protocol type	Symbolic	type of the protocol, e.g. tcp, udp, etc
Duration	Continuous	length (number of seconds) of the connection
Land	Symbolic	1 if connection is from/to the same host/port; 0 otherwise
Wrong fragment	Continuous	number of ``wrong" fragments
Urgent	Continuous	Number o f Urgent packets
Hot	Continuous	number of ``hot" indicators
Num_failed_login	Continuous	Number of failed login attempts
Num_outbound_cmds	Continuous	number of outbound commands in an ftp session
Is_hot_login	Symbolic	1 if the login belongs to the ``hot" list; 0 otherwise
Is_guest_login	Symbolic	1 if the login is a ``guest" login; 0 otherwise
Num_root	Continuous	number of ``root" accesses
Num_file_creations	Continuous	number of file creation operations
num_shells	Continuous	number of shell prompts
Num_access_files	Continuous	number of operations on access control files
Count	Continuous	number of connections to the same host as the current connection in the past two seconds
Srv_count	Continuous	number of connections to the same service as the current connection in the past two seconds
Logged_in	Symbolic	1 if successfully logged in; 0 otherwise
Num_compromised	Continuous	number of ``compromised" conditions
Root_shell	Continuous	1 if root shell is obtained; 0 otherwise
Su_attempted	Continuous	1 if ``su root" command attempted; 0 otherwise
Same_srv_rate	Continuous	% of connections to the same service
Diff_srv_rate	Continuous	% of connections to different services
Srv_diff_host_rate	Continuous	% of connections to different host
Serror_rate	Continuous	% of connections that have ``SYN" errors
Srv_serror_rate	Continuous	% of connections that have ``SYN" errors
Rerror_rate	Continuous	% of connections that have ``REJ" errors
dst_host_same_srv_rate	Continuous	same_srv_rate for destination host
Dst_host_count	Continuous	count for destination host
Dst_host_srv_count	Continuous	Srv_count for destination host
dst_host_diff_srv_rate	Continuous	diff_srv_rate for destination host.
dst_host_serror_rate	Continuous	serror_rate for destination host
dst_host_srv_serror_rate	Continuous	srv_serror_rate for destination host
dst_host_rerror_rate	Continuous	rerror_rate for destination host
dst_host_srv_rerror_rate	Continuous	srv_serror_rate for destination host
Dst_host_srv_diff_host_rate	Continuous	Diff_host_rate for destination host

TABLE 1: FEATURES OF KDDCUP99DATASET

IV. ATTACKS

In networking an attack is any attempt to destroy or gain unauthorized access to or make unauthorized use

of an asset. Networks are subject to attacks from malicious sources. Our KDDCup99dataset consists of 4,90,000 connection vectors with 41 features each is labelled as either normal or an attack, with exactly one specific attack type[13]. Here Attacks fall into one of four categories described below.

A) Denial of Service (dos):
Excessive consumption of resources that denies legitimate requests from legal users on the system[15].
B) Remote to Local (r2l):
Attacker having no account gains a legal user account on the victim machine by sending packets over the networks[15]
C) User to Root (u2r):
Attacker tries to access restricted privileges of the machine[15].
D) Probe:
Attacks that can automatically scan a network of computers to gather information or find known vulnerabilities[15].

back,land,neptune,pod,smurf,teardrop	Denial of Service attacks
buffer_overflow,loadmodule,perl,rootkit	User to Root Attacks
ftp_write,guess_passwd,imap,multihop,phf,spy,warezclient,warezmaster	Remote to Local Attacks
satan,ipsweep,nmap,portsweep	Probes

TABLE 2 VARIOUS TYPES OF ATTACKS DESCRIBED IN FOUR MAJOR CATEGORIES[15].

V. NETWORK INTRUSION DETECTION SYSTEM USING FUZZY INFERENCE SYSTEM

With the increase in computers getting connected to public access networks (eg:internet), it is impossible for computer systems to get protected from network intrusions. It is better to identify and remove intrusions at the initial moment rather than looking them after they enter the event. Because, there is no ideal solution to avoid intrusions from the event. [7]. One approach to handle suspicious behaviours inside a network is an intrusion detection system (IDS). For intrusion detection, a wide variety of techniques have been applied specifically, data mining techniques, artificial intelligence technique and soft computing techniques. AI techniques such as neural networks and fuzzy logic are applied for detecting suspicious activities in a network, in which fuzzy based system provides significant advantages over other AI techniques. Researchers are focussing on fuzzy rule learning for effective intrusion detection. So based on these research we develop a fuzzy rule base system for identifying the attacks. Here we design anomaly based intrusion detection, which makes use of the generated rules from Sugeno fuzzy inference system [4].The figure 1 describes the design of proposed IDS.

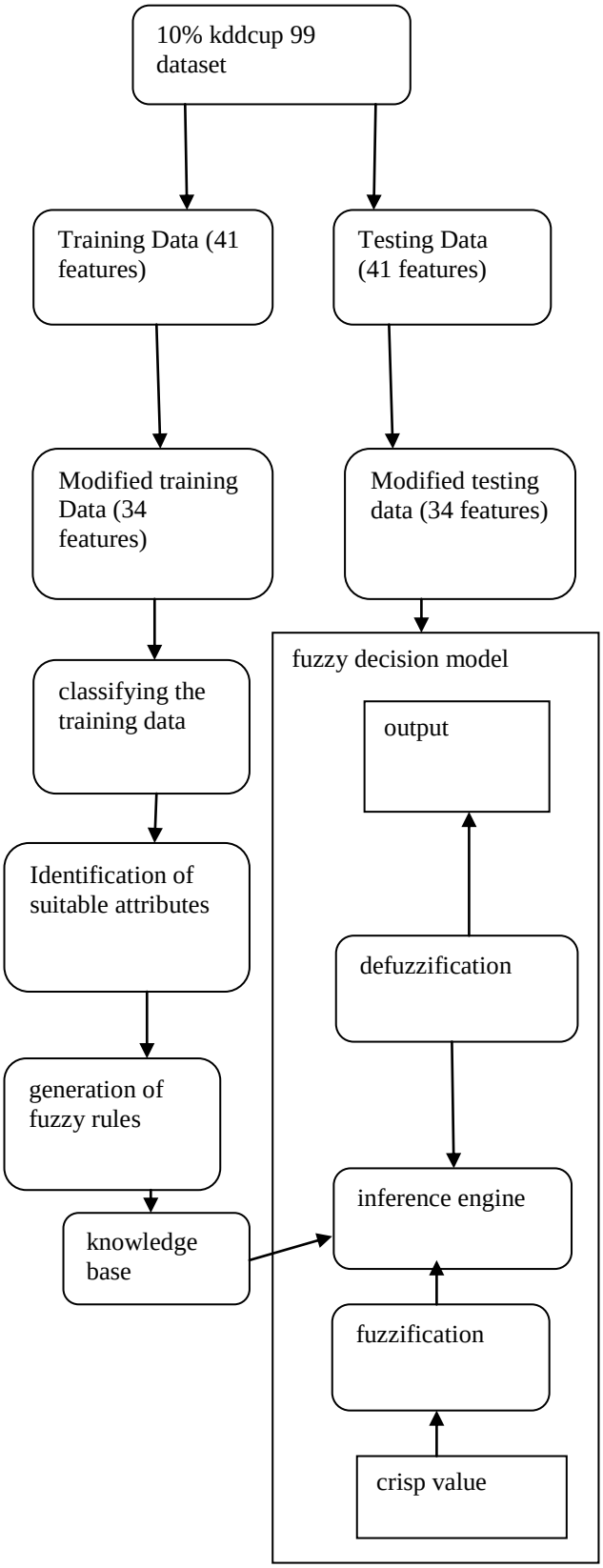

FIG 1:DESIGN OF PROPOSED INTRUSION DETECTION SYSTEM

A)Training data classification: The dataset we have taken for analysing the intrusion detection behaviour is KDD-Cup 1999 data. The KDD-Cup 1999 data contains four types of attacks and normal behaviour

data with 41 attributes that have both continuous and symbolic attributes. The proposed system is designed only for the continuous attributes because the major attributes in KDD-Cup 1999data are continuous in nature. Therefore, we have taken only the continuous attributes for instance, 34 attributes from the input dataset by removing discrete attributes. Then the dataset(D) is divided into five subsets of classes based on the class label prescribed in the dataset. The class label describes several attacks, which comes under four major attacks(Denial of Service, Remote to Local,U2R and probe) along with normal data. The five subsets of data are then used for generating a better set of fuzzy rules automatically so that the fuzzy system can learn the rules effectively.

B)Approach for generation of fuzzy rules: This section describes the designed strategy for automatic generation of fuzzy rules to provide effective learning. In general, the fuzzy rules given to the fuzzy system is done manually or by experts, who are given the rules by analysing intrusion behaviour. But, in our case, it is very difficult to generate fuzzy rules manually due to the fact that the input data is huge and also having more attributes. So we use fis editor for generation of fuzzy rules

1) Identification of suitable attributes for rule generation:

For identifying the suitable attributes we have used CfssubsetEval and best first[8].In this step, we have chosen only the most suitable attributes for identifying the classification whether the record is normal or attack. The reason behind this step is that the input data contain 34 attribute, in which all the attributes are not so effective in detecting the intrusion detection.

C) Fuzzy Decision module

Here we describe the designing of fuzzy inference system for finding the suitable class label of the test dataset. Fuzzy inference system is the process of formulating the mapping from a given input to an output using fuzzy logic. Zadeh in the late 1960s[9] introduced fuzzy logic and is known as the rediscovery of multi valued logic designed by Lukasiewicz. the designed fuzzy system shown in figure 2 contains 34 inputs and one output where inputs are related to the 34 attributes and output is related to the class label(attack data or normal data).Here from thirty four input, we select some attributes using CfssubsetEval and best first ,here we get 10 attributes and so these are used as input and single output of Sugeno Fuzzy inference System with Wtaver area of defuzzification Strategy was used for this purpose. Here ,each input fuzzy set defined in the fuzzy system includes three membership functions and an output fuzzy set contains two membership functions. Each membership function used triangular function for fuzzification strategy. The fuzzy rules obtained from the inference system are fed to the rule base for learning the system.

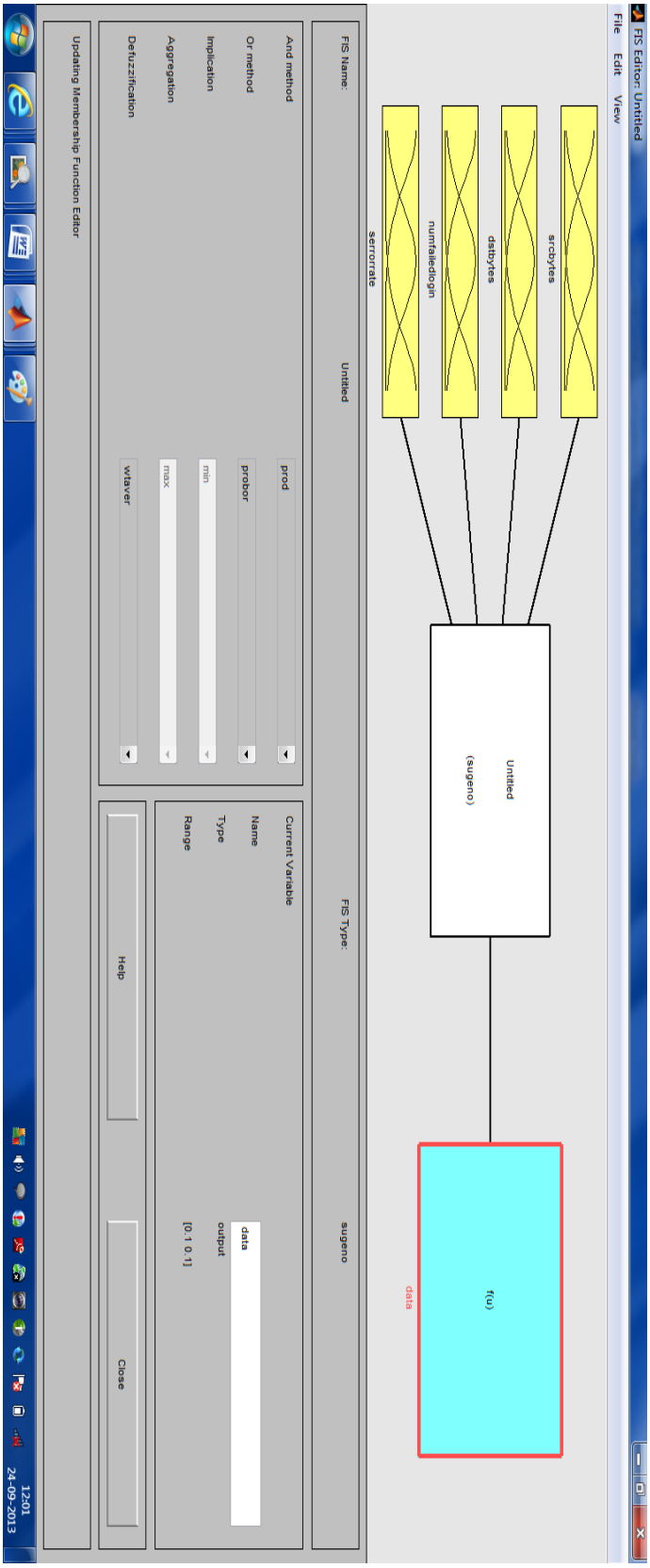

FIGURE2 :SUGENO FUZZY INFERENCE SYSTEM

D) Classification For A Test Input:

For testing phase, a test data from the KDD-cup 99 dataset is given to the designed fuzzy logic system for

finding the fuzzy score. At first, the test input data containing 34 attributes is applied to fuzzifier, which converts 34 attributes (numerical variable) into linguistic variable using the triangular membership function. The output of the fuzzifier is fed to the inference engine which in turn compares that particular input with the rule base. Rule base is a knowledge base which contains a set of rules obtained from the definite rules. The output of inference engine is one of the linguistic values from the following set {Low and High} and then, it is converted by the defuzzifier as crisp values. The crisp value obtained from the fuzzy inference engine is varied in between 0 to 2, where '0' denotes that the data is completely normal and '1'specifies the completely attacked data.

VI. EXPERIMENTATION

This section describes the experimental results and performance evaluation of the proposed system. The proposed system is implemented in MATLAB (7.8) and anfis editor. For experimental evaluation, we have taken KDD cup 99 dataset[10], which is mostly used for evaluating the performance of the intrusion detection system. For evaluating the performance, it is very difficult to execute the proposed system on the KDD cup 99 dataset since it is a large scale. Here, we have used subset of 10% of KDD Cup 99 dataset for training and testing. The dataset which is in .txt extension is converted as a csv(Comma separated value) file. Then using CfssubsetEval and best first methods under select attributes option in the weka tool we select some important attributes ,and store them as a csv file. This csv file is then converted into dat file .The dat file is the only file open in anfis editor. Among the 10% data we take some records for training and some records as testing data before converting them into dat files.

A)Loading, Plotting, and Clearing the Data:

To train a FIS, you must begin by loading a Training data set that contains the desired input/output data of the system to be modelled. Any data set you load must be an array with the data arranged as column vectors, and the output data in the last column. You can also load Testing and Checking data in the GUI. For more information on testing and checking data sets, see Model Validation Using Testing and Checking Data Sets. To load a data set using the Load data portion of the GUI:Specify the data Type. Select the data from a file or the MATLAB workshop .Click Load Data .After you load the data, it displays in the plot. The training, testing and checking data are annotated in blue as circles, diamonds, and pluses respectively. To clear a specific data set from the GUI: In the Load data area, select the data Type. Click Clear Data. This action also removes the corresponding data from the plot. Fig3 describes the loading ,plotting and clearing the data.

B) Generating or Loading the Initial FIS Structure: Before you start the FIS training, you must specify an initial FIS model structure. To specify the model structure, perform one of the following tasks:

Load a previously saved Sugeno-type FIS structure from a file or the MATLAB workspace. Generate the initial FIS model by choosing one of the following partitioning techniques:

1)Grid partition: Generates a single-output Sugeno-type FIS by using grid partitioning on the data.

2)Sub clustering: Generates an initial model for ANFIS training by first applying subtractive clustering on the data.

To view a graphical representation of the initial FIS model structure, click Structure. Later select rules option under the edit option in the anfis editor,the below fig 4 shows the rules generated.

C)Training the FIS

After loading the training data and generating the initial FIS structure, you can start training the FIS. The following steps show you how to train the FIS.

In Optim. Method, choose hybrid or back propagation as the optimization method. The optimization methods train the membership function parameters to emulate the training data. Enter the number of training Epochs and the training Error Tolerance to set the stopping criteria for training. The training process stops whenever the maximum epoch number is reached or the training error goal is achieved. Click Train Now to train the FIS.

This action adjusts the membership function parameters and displays the error plots. Examine the error plots to determine over fitting during the training. If you notice the checking error increasing over iterations, it indicates model over fitting. For examples on model over fitting, see ANFIS Editor GUI

D)Validate the model: validate the model using a Testing or Checking data that differs from the one you used to train the FIS. To validate the trained FIS:

Select the validation data set and click Load Data.Click Test Now. Finally we can detect the data is normal or attack.

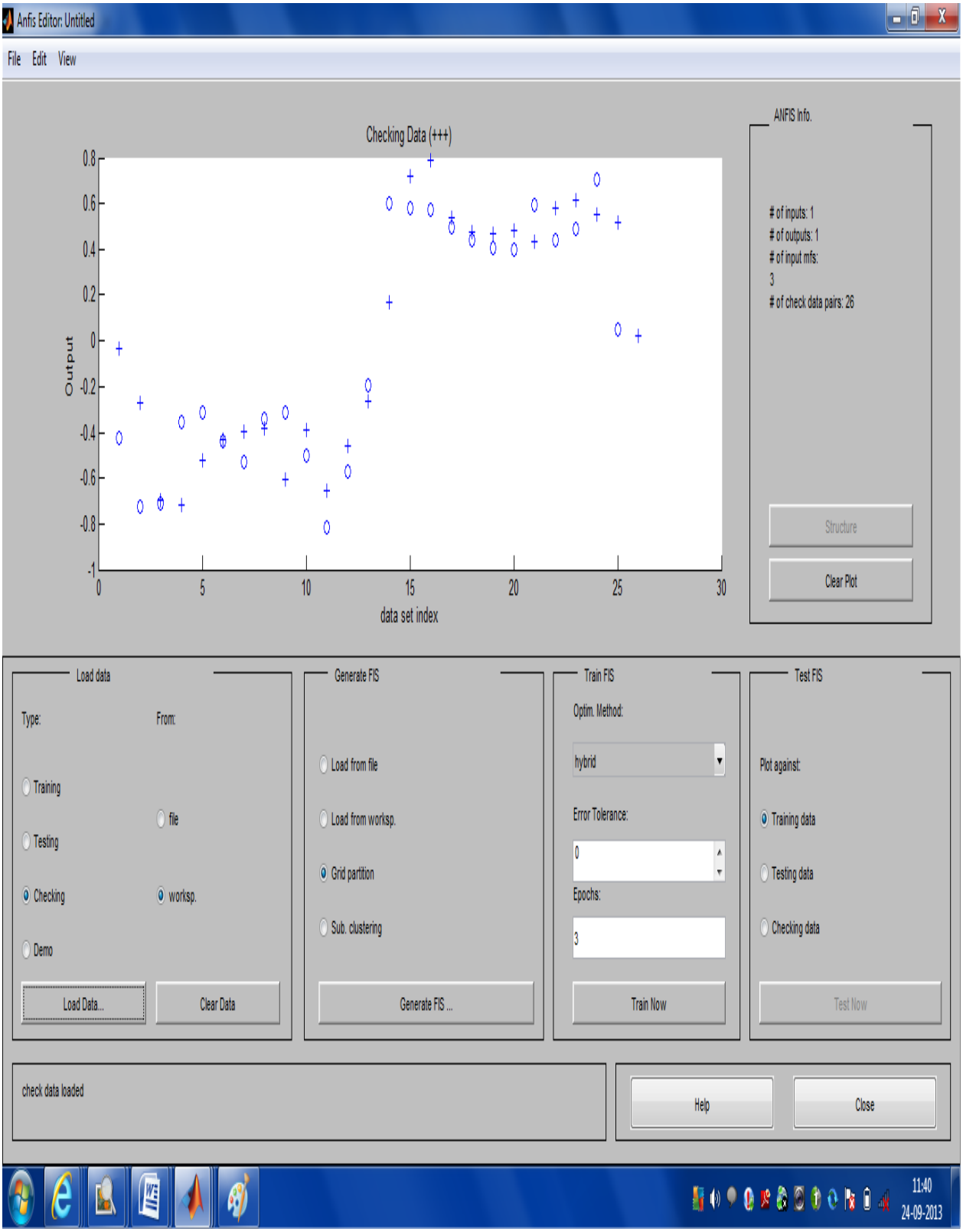

FIG3: LOADING TRAINING,CHECKING,CLEARING DATA

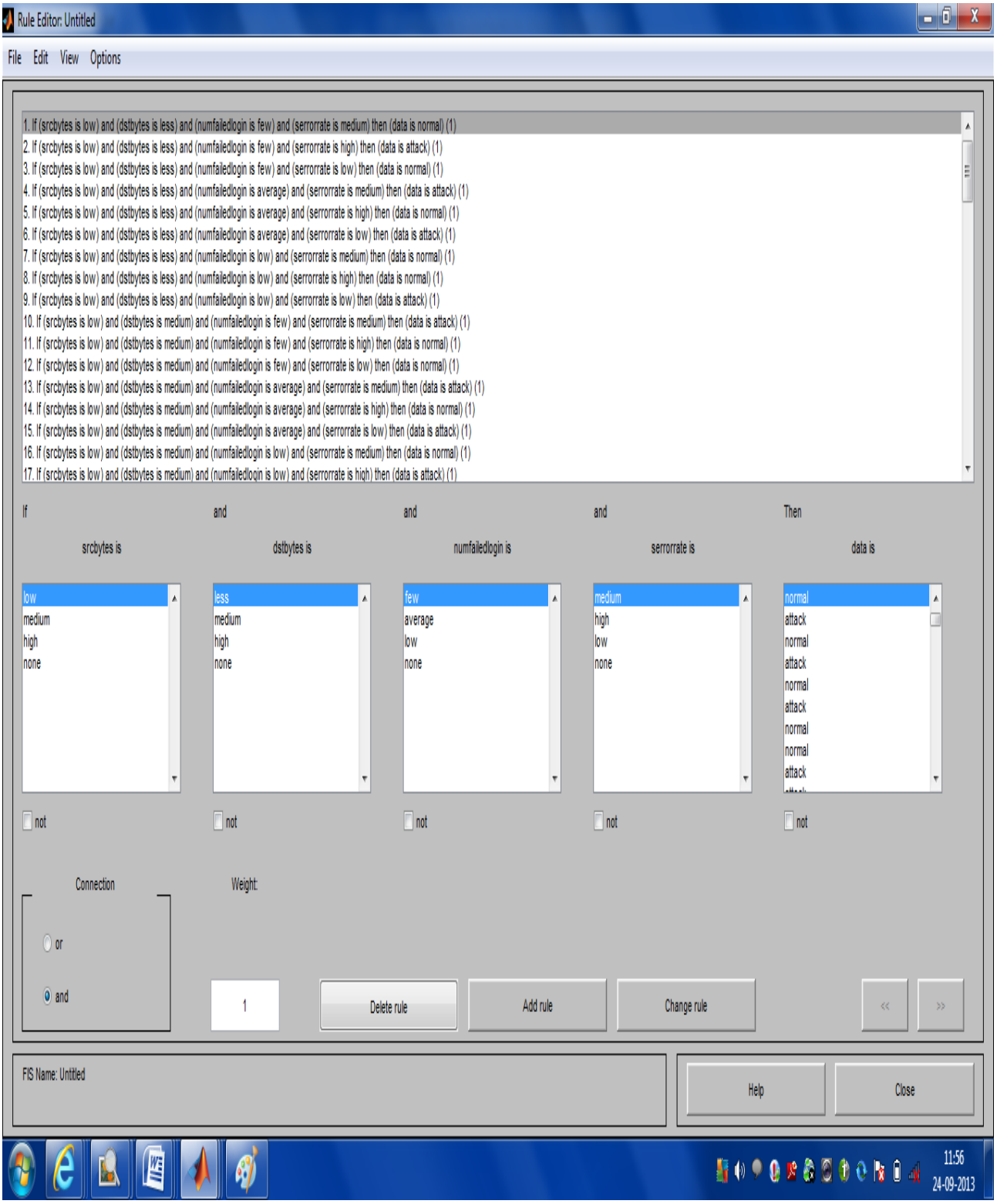

FIG4:FIS

RULES

E) Fis rules:

If(Src_bytes is low) and (dst_bytes is less) and (Num_failed_login is few) and (serrorrate is medium) then (data is normal).

If(Src_bytes is low) and (dst_bytes is medium) and (Num_failed_login is few)and (serrorrate is medium) then (data is attack).

If(Src_bytes is low) and (dst_bytes is less) and (Num_failed_login is average) and (serrorrate is medium) then (data is attack).

If(Src_bytes is low) and (dst_bytes is less) and (Num_failed_login is low) and (serrorrate is medium) then (data is normal).

VII .CONCLUSION

Here we have developed an anomaly based intrusion detection system in detecting the intrusion behaviour within a network. A fuzzy decision-making module was designed to build the system more accurate for attack detection, using the Sugeno fuzzy inference approach. An effective set of fuzzy rules for inference approach were identified automatically by making use of the fuzzy rule learning strategy, which are more effective for detecting intrusion in a computer network System. Then, fuzzy rules were identified by fuzzifying the definite rules and these rules were given to Sugeno fuzzy system, which classify the test data. We have used KDD cup 99 dataset for evaluating the performance of the proposed system and for experimentation we used MATLAB(7.8) and ANFIS editor and results showed that the proposed method is effective in detecting various intrusions in computer networks.

References

[1] Zadeh, L.A., "Outline of a new approach to the analysis of complex systems and decision processes," IEEE Transactions on Systems, Man, and Cybernetics, Vol. 3, No. 1, pp. 28-44, Jan. 1973.

[2] Susan M. Bridges and Rayford B.Vaughn, "Fuzzy Data Mining And Genetic Algorithms Applied To Intrusion Detection", In Proceedings of the National Information Systems Security Conference (NISSC), Baltimore, MD, pp.16-19, October 2000http://www.security.cse.msstate.edu/docs/Publications/sbridges/nissc2000.pdf.

[3] Network intrusion detection system using fuzzy logic Shanmugavadivu Indian Journal of Computer Science and Engineering01/2011 http://www.ijcse.com/docs/IJCSE11-02-01-034.pdf

[4] Intrusion Detection Techniques. Peng Ning, North Carolina State University. Sushil Jajodia, George Mason University. Introduction. Anomaly Detectionhttp://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.89.2492&rep=rep1&type=pdf

[5] J. H. Güneş Kayacýk, A. Nur Zincir-Heywood, Malcolm I. Heywood. Selecting Features for Intrusion Detection: A Feature Relevance Analysis on KDD 99 Intrusion Detection Datasets

[6] Knowledge discovery in databases DARPA archive and Task Description. http://www.kdd.ics.uci.edu/databases/kddcup99/task.html .

[7] Qiang Wang and Vasileios Megalooikonomou, "A clustering algorithm for intrusion detection", in Proceedings of theconference on Data Mining, Intrusion Detection, Information Assurance, and Data Networks Security, vol. 5812, pp.31-38, March 2005.

[8] Combined Feature Selection and classification – A novel approach for the categorization of web pages K. Selvakuberan, M. Indradevi, Dr. R. Rajaram Innovation Lab. http://www.worldacademicunion.com/journal/1746-7659JIC/jicvol3no2paper01.pdf

[9] www.cse.unr.edu/~bebis/CS365/Papers/FuzzyLogic.pdf Available: http://www.aptronix.com/fide/ ... .edu/~fishwick/paper/paper.html.

[10]http://www.sigkdd.org/kddcup/index.php?section=1999&method=data

[11] Discriminant Analysis based Feature Selection in KDD Intrusion Dataset. By Dr.S.Siva Sathya, Dr. R.Geetha Ramani and K.Sivaselvi http://research.ijcaonline.org/volume31/number11/pxc3875527.pdf

[12] Designing Fuzzy Inference Systems from Data:An Interpretability-Oriented Review Serge Guillaume

[13] A Detailed Analysis of the KDD CUP 99 Data Set Mahbod Tavallaee, Ebrahim Bagheri, Wei Lu, and Ali A. Ghorbani

[14] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applicationsto modeling and control," *IEEE Tran. Syst., Man, Cybern.*, vol.SMC 15, pp. 116–132, 1985.

[15] A Database of Computer Attacks for the Evaluation of Intrusion Detection Systems by kristopher kendell http://www.google.co.in/url?sa=t&rct=j&q=&esrc=s&frm=1&source=web&cd=1&ved=0CCwQFjAA&url=http%3A%2F%2Fwww.dtic.mil%2Fcgi-bin%2FGetTRDoc%3FAD%3DADA525629&ei=pHBAUrG_BoLrrQeNp4HQBQ&usg=AFQjCNFeL1Q0zPYhNrb-8-8cZ5G_AVA5Aw&bvm=bv.52434380,d.bmk

[16] E. H. Mamdani and S. Assilian, "An experiment in linguistic synthesiswith a fuzzy logic controller," *Int. J. Man-Mach. Stud.*, vol. 7, pp. 1–13,1975.