An Efficient Intrusion Detection System Based on Fuzzy Genetic approaches

*Azarkasb S.O.

Faculty of Computer Engineering, Qazvin Branch Azad University, Qazvin, Iran.
Azarkasb@ymail.com

Abstract: One of the solutions for the security of the systems and computer networks is the formation of intrusion detection systems. In the design of these systems, the techniques of artificial intelligence such as neural network, data mining techniques, expert systems, genetic algorithms and fuzzy systems are applied. The main aim is the design of the system that besides exact detection is with low error. The main aim of detection in these systems is their analyzer. It seems that if the input of analyzer of the systems is language variables and final detection is done as fuzzy inference, the results of the analysis get better and an exact detection is done. In addition to reduction of error, it can present good information in the form of fuzzy rules to the security expert of the system. Genetic algorithms by flexible and prevalent searching ability can be applied for optimized learning of fuzzy system rule base. The current paper aimed to evaluate, design and implement fuzzy genetic intrusion detection system. The experiments and the results showed the considerable effect of using the proposed architecture on the improvement of detection.

[Azarkasb S.O.. **An Efficient Intrusion Detection System Based on Fuzzy Genetic approaches**. *Life Sci J* 2013;10(8s):6-21] (ISSN:1097-8135). http://www.lifesciencesite.com. 2

Keywords: Intrusion Detection systems, Fuzzy Logic, Genetic Algorithm, Classification, Evolution Methods, Attack Production Tools.

1. Introduction

One of the solutions for the security of the systems and computer networks is the formation of intrusion detection systems. These systems are similar to computer buzzer and alarm suspect events and attacks in the environment. The aim of most of the security solutions as firewall encryption and authentication systems of security and avoiding intrusion is making the systems secure and avoiding intrusion. As system security is costly and is not provided completely, we need a system that instead of avoiding attack and event, detect the reason and report. Intrusion detection systems are applied. The prediction of the attacks and tracing the intruder at complex and higher levels is studied.

There are different kinds of intrusion detection systems, and are different from different aspects. One of the differences of these systems is their analysis method on input data for detection. The method that system analyzer engine applies as the most important component of the system, the system detection power is determined. In misuse detection systems as a special type of these systems from analysis, definite models of attacks are presented to the system for detection. If the implementation is good in these systems, all the defined attacks in the system are detected as without error. But the existing problem is the increasing growth of various attacks and new intrusion techniques. However, we need an intelligent system to detect new intrusions and unknown attacks. In other words, we require an n intrusion detection system to detect the new intrusions not being defined. Anomaly detection systems are raised here and instead of the study of the attacks, it focuses on the description of normal behavior models.

Now, the Anomaly Detection system is mostly research-based and are applied less in commercial systems. This is because of their error in detection of normal behavior as an intrusion or vice versa. To do this, various research projects are performed about Anomaly Detection systems that by the increase of the validity of the detection of the systems, we can benefit more of its benefits. Normally, in the design of these systems, the important techniques are applied in artificial intelligence as neural network, genetic algorithms, data mining, fuzzy systems, expert systems and rule learning algorithms.

The intrusion detection systems with design of analysis method are encountered with new attacks. The more the diversity of the attacks, the more difficult the study of the good model of them for security expert and it is required to extract more criteria to determine the normal and abnormal behavior. In addition, the relevance and composition of various criteria are determined to describe any special behavior as exactly. The fuzzy systems can cover the complexities of such systems and are effective in the reduction of system error.

On the other hand, the raw data of frequency in the existing normal activities and attacks are extracted easily. Using the Machine Leaning techniques and data mining are useful for detection of the good information of the raw data. Thus, the attack or normal behavior models are extracted from raw data automatically and are presented to the analyzer.

The current study aimed to design and evaluate an intrusion detection system with the analysis engine of fuzzy genetic system. Indeed, in the proposed system, genetic algorithms do the design of its analyzer as a fuzzy system.

By using fuzzy logic, the controlled criteria and parameters for detection in the system are expressed by linguistic variables and fuzzy rules are used to describe the intrusion models by the fuzzy criteria. Finally, the intrusion occurrence detection is stated by fuzzy expression.

The genetic algorithms by prevalent and flexible searching capability can be applied for optimized learning of fuzzy system rules base. Thus, the current paper focused on optimized use of the algorithms in the design of mentioned fuzzy systems that is obtained by making each algorithm stages suitable. The system fuzzy analyzer beside the input media giving the suitable criteria and data, form the final intrusion detection system. The data source being used in the proposed system is based on network packets that are extracted by using the attack production tools against the network agent and it is introduced in the following.

This paper is organized as follows: Section 2 provides historical background for this work Section 3 defines the initial fuzzy concepts and the related genetic. Section 4 focuses on the different types of attacks. Section 5 explains about the proposed fuzzy genetic detection system. Section 6 elaborates the implementation system and its performance. In section 7, the extracted rules are tested, evaluated, compared and analyzed. Finally, the paper is concluded with the summary of the attachments and development solutions as future works in sections 8, 9.

Related works

Fuzzy intrusion detection systems

Anomaly Detection method compared to misuse detection method in intrusion detection has the special complexities. In this method, suitable criteria are used to distinguish anomaly and normal behavior. Suitable determination of these parameters is hard. Some of the attacks are recognized by the criteria not suitable for the detection of other attacks. Some of the attacks are complex and are detected only by the investigation of the combination of some various criteria. To remove the complexities in Anomaly Detection, some of the intrusion detection systems applied fuzzy view such as (Boughaci, 2011; Dickerson, 2001; Saniee Abadeh, 2008; Gao, Gomez2, 2002; 2003; Shah, 2003; Sheikhan, 2010).

As various aspects of intrusion detection are fuzzy, they are expressed better by fuzzy logic. For example, an attack is occurred with what possibility, validity and reliability or what is the exact boundary of normal and abnormal event. The fuzzy systems have special capabilities being useful in intrusion detection. In fuzzy system, we can combine different inputs of various resources. This advantage is applied in intrusion detection systems for the combination of various criteria and the analysis. Some examples of research system of fuzzy intrusion detection are introduced.

FIRE

An example of fuzzy detection systems is FIRE system (Dickerson, 2001) that is a distributed intrusion detection system and in analysis stage, rule base and fuzzy inference are applied. The information source of this system is a network and investigates special data of the packets during two weeks. This system is factor based and each factor fazzificate the data separately from the special data collecting and is related to fuzzy analyzer engine. The analyzer by fuzzy rules base detects attack event and alarms. The analyzer in addition to notification of attack occurrence, reports the degree and the attack nature of the event. Security expert defines the rules base of the system. An example of the rules of this base that is used for the detection of the attack of port scanning is as following:

If the COUNT of UNUSAL SDTs is HIGH
And the COUNT of ICMP ERQs is HIGH
Then pingflood ALERT is HIGH

Where SDTs are the combined [Source, Destination, ICMP Type] identifier of ICMP packets. The complete fuzzy system for detecting this type of attack is shown in Figure 1.

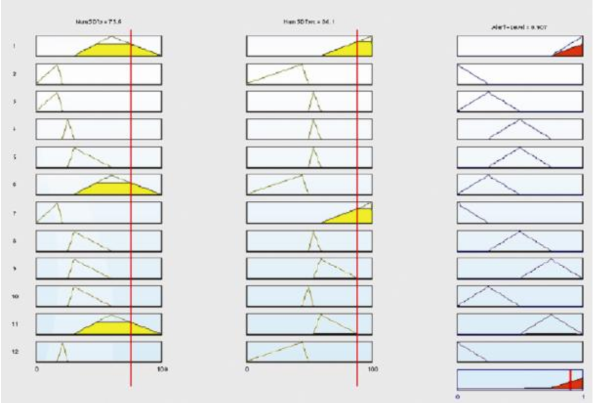
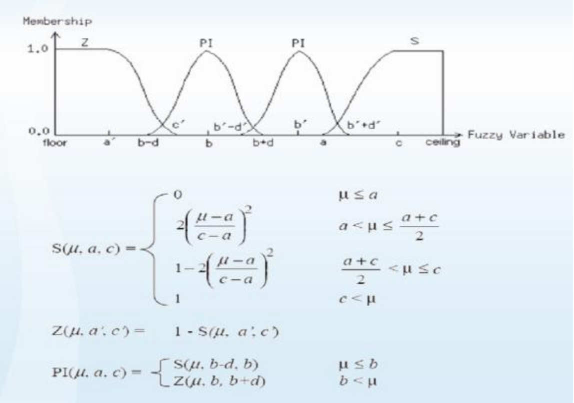


Fig. 1. The fuzzy system applied in FIRE to DoS attack (Dickerson, 2001).

This system detects port scanning, host scanning and service denial easily and acts well in the recognition of the Back Door and Trojan horses.

BRIDGES and Vaughn

An example of other fuzzy intrusion detection system is Vaughn and Bridge. They presented an example of intelligent intrusion detection system in which fuzzy view is applied. This system is consisting of both misuse and anomaly detection section. In misuse detection, expert system method is applied and in anomaly detection, fuzzy data mining technique was applied. This system by data mining method extracts the models and rules for intrusion detection from raw data. This system applies fuzzy variables in extraction of the rules and expressing the values of the criteria and it could extract flexible and abstract models in data mining process. The applied fuzzy sets are PI, Z, S, and the form and relations of membership function are shown in Fig 2. In the design of expert system rules for misuse detection, fuzzy clips are applied and fuzzy and non-fuzzy rules are created. Fuzzy clips is expert system shells fuzzy development CLIPS made by NASA (Bridges, 2000).



$$S(\mu, a, c) = \begin{cases} 0 & \mu \leq a \\ 2\left(\frac{\mu - a}{c - a}\right)^2 & a < \mu \leq \frac{a+c}{2} \\ 1 - 2\left(\frac{\mu - a}{c - a}\right)^2 & \frac{a+c}{2} < \mu \leq c \\ 1 & c < \mu \end{cases}$$

$$Z(\mu, a', c') = 1 - S(\mu, a', c')$$

$$PI(\mu, a, c) = \begin{cases} S(\mu, b\text{-}d, b) & \mu \leq b \\ Z(\mu, b, b+d) & b < \mu \end{cases}$$

Fig. 2. The fuzzy sets applied in the system and formal description (Bridges, 2000).

Evolving Fuzzy Rule for Intrusion Detection (EFRID)

In this system, the aim was the creation of fuzzy classifiers: two fuzzy rules, one determining the normal behavior and another determining the attack. The applied data of the extracted data of the network is during the normal and attack behavior. The fuzzy rule in rule condition is the variables of the investigated features and in consequent section, determines the normal and abnormal class. In the classification, the rule achieving the highest score determines the class correspondent with the object. The fuzzy rule score based on the accuracy degree of the condition determines the rule and the validity of the rule. The designer of the system defined fuzzy variables with five similar triangular set, very low, low, average, much, very much are expressed in the condition section of the rule. The defined space for all the variables is similar. In the above approach for the problem solving of intrusion detection, the determination of suitable rules is the main aim (Gomez2, 2002).

Genetic intrusion detection systems

Several Genetic Algorithms (GAs) and Genetic Programming (GP) has been used for detecting intrusion detection of different kinds in different scenarios. Some uses GA for deriving classification rules (Chittur, 2001; Li, 2004; Lu, 2004; Pillai, 2004; Sadiq Ali Khan, 2011). GAs used to select required features and to determine the optimal and minimal parameters of some core functions in which different artificial intelligence methods were used to derive acquisition of rules (Bridges, 2000; Gomez2, 2002; Middlemiss, 2003). There are several papers such as (Mukkamala, 2005; Peddabachigari 2007; Saniee Abadeh, 2007; Peng 2007) related to IDS which has certain level of impact in network security.

The effort of using GAs for intrusion detection can be referred back to 1995, when Crosbie and Spafford (Crosbie, 1995) applied the multiple agent technology and Genetic Programming to detect network anomalies (Gong, 2005). For both agents they used Genetic Programming to determine anomalous network behaviors and each agent can monitor one parameter of the network audit data.

Li (Li, 2004) described a method using GA to detect anomalous network intrusion (Gong, 2005; Abdullah, 2009). The approach includes both quantitative and categorical features of network data for deriving classification rules. However, the inclusion of quantitative feature can increase detection rate but no experimental results are available. Goyal and Kumar (Goyal, 2008) described a GA based algorithm to classify all types of smurf attack using the training dataset with false positive rate is very low (at 0.2%) and detection rate is almost 100%.

Lu and Traore (Lu, 2004) used historical network dataset using GP to derive a set of classification (Gong, 2005). They used support-confidence framework as the fitness function and accurately classified several network intrusions. But their use of genetic programming made the implementation procedure very difficult and also for training procedure more data and time is required.

Xia et al. (Xia, 2005) used GA to detect anomalous network behaviors based on information theory (Gong, 2005; Abdullah, 2009; Gupta, 2011). Some network features can be identified with network attacks based on mutual information between network features and type of intrusions and then using these features a linear structure rule and also a GA is derived. The approach of using mutual information and resulting linear rule seems very effective because of the reduced complexity and higher detection rate. The only problem is it considered only the discrete features.

Gong et al. (Gong, 2005) presented an implementation of GA based approach to Network Intrusion Detection using GA, and showed software implementation. The approach derived a set of classification rules and utilizes a support-confidence framework to judge fitness function.

The genetic-based rule learning systems

There are three main views for rules learning by genetic algorithms, Michigan, Pittsburg and Iterative Rules Learning (IRL) (Herrera, 1997).

According to Michigan, each chromos shows a rule and the rules set is shown by total population. In this view, genetic operators are applied in the rule level.

According to Pittsburg, each chromosome shows the total predictive rules set. Crossover operator creates a new combination of the rules and mutation operator produces a new rule. In IRL new method, according to Michigan view, each chromosome displays only one rule. However, in this method, only the best chromosome is considered as a part of final solution and the remaining of the chromosomes are ignored. The fitness of each chromosome is determined alone. In this method, genetic algorithm obtains a part of solution. To obtain the total answer of rules set, the algorithm is applied as iterative. In each iteration, the new rule obtained is added to the previous rules set and it is punished. Punishment means that in the next algorithm iterations, this rule is not achieved again. This is done by elimination of the educational examples that the rules current set cover them of the total educational data. The iteration is done until a full answer is obtained for the problem. Michigan view is suitable when the learning is increasing and the system is on line. It means that the educational examples are added gradually and the environment is dynamic. Pittsburg view and more IRLs are applied in the cases that educational data is existing before learning and the environment is stationary. The main problem of Michigan view is that the rules collaborating with each other are competitors in the selection stage and new generation production. This problem is not appeared in Pittsburg view in which each person including total rules set. This method is costly for chromosomes fitness. In IRL method, as the algorithm is only in searching a rule, the search space is limited (Herrera, 1997).

Rule-based fuzzy genetic detection system

In recent years, many studies are performed in which fuzzy system concepts and genetic algorithms are integrated with each other such as (Boughaci, 2012, Bridges, 2000; Gomez1, 2002; Mendes, 2001; Saniee Abadeh, 2007). In most of the cases, genetic algorithms are used for the design of fuzzy systems. The fuzzy genetic system is used for all of the systems. The common example of these systems is rule-based fuzzy genetic systems. Rule-based model of fuzzy systems is one of the most common and successful fuzzy systems model and fuzzy controllers are the example. Figure 3 shows the general view of fuzzy genetic system (Herrera, 1997).

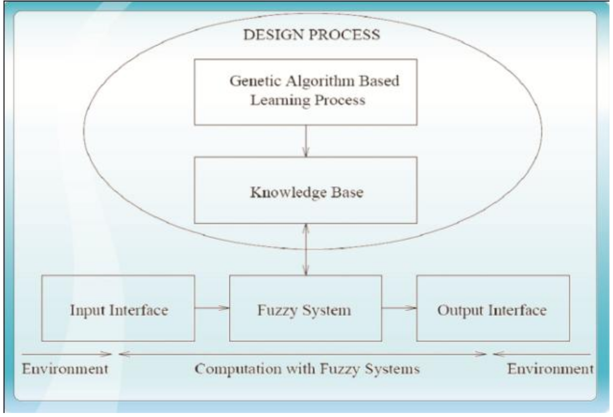


Fig. 3. The fuzzy genetic system architecture (Herrera, 1997).

Knowledge base is composed of data bank and rules base. Membership function and fuzzy sets describing the linguistic concepts are the main components of data bank. Fig 4 shows knowledge base with fuzzification sections and non-fuzzification applied in fuzzy control (Herrera, 1997).

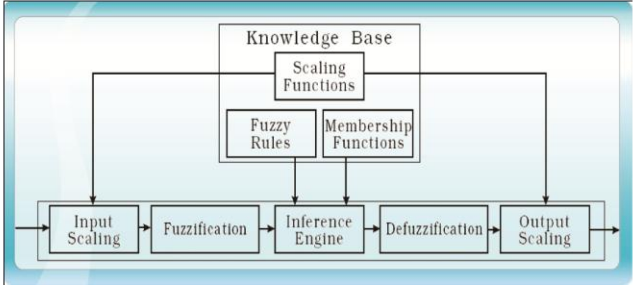


Fig. 4. The structured knowledge base of a fuzzy system (Herrera, 1997).

The genetic algorithm is applied for learning or changing rules base, data bank or both of them. Although genetic algorithms are not learning algorithms but due to good and independent searching of the range is applied in the solution of most of the machine learning issues (Herrera, 1997).

(Boughaci, 2011) proposed a fuzzy genetic algorithm (FGA) for intrusion detection. The FGA system is a fuzzy classifier, whose knowledge base is modeled as a fuzzy rule such as "if-then" and improved by a genetic algorithm. His approach consists of two main steps. The first one is the data normalization The second step is the fuzzy genetic algorithm.

The Fuzzy genetic algorithm (FGA) starts from a population of individuals generated randomly. Each individual is an "*ifthen*" fuzzy rule. In order to optimize the set of fuzzy rules already generated in the first stage, a genetic algorithm process that consists of selection, crossover and mutation operators are applied on the individuals.

Initial concepts

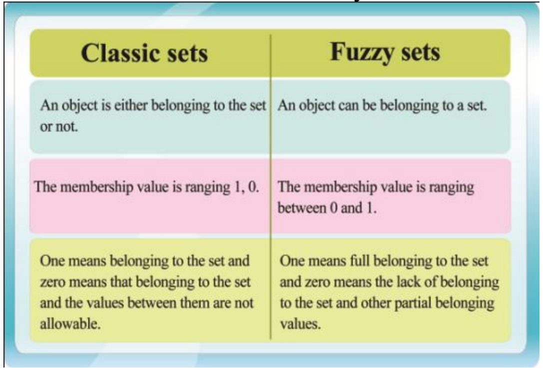
Fuzzy initial concepts

In this section, we review the fuzzy operators, fuzzy aggregation operators, fuzzy classifier and C-mean fuzzy clustering.

Fuzzy sets

In fuzzy logic, the fuzzy sets define the linguistic concepts and membership functions by mathematical description of the sets express the accuracy of the linguistic forms. For example, hot is a linguistic concept that is raised as a degree of belonging to heat. Table 2 shows the difference of fuzzy sets and classic sets easily.

Table. 1. The difference of fuzzy and classic sets.



Classic sets	Fuzzy sets
An object is either belonging to the set or not.	An object can be belonging to a set.
The membership value is ranging 1, 0.	The membership value is ranging between 0 and 1.
One means belonging to the set and zero means that belonging to the set and the values between them are not allowable.	One means full belonging to the set and zero means the lack of belonging to the set and other partial belonging values.

The object membership value in a set in fuzzy logic is expressed by membership function. The function range is the values controlled by an object and the function range is [0,1]. The functions are mostly in triangle and Trapezoid.

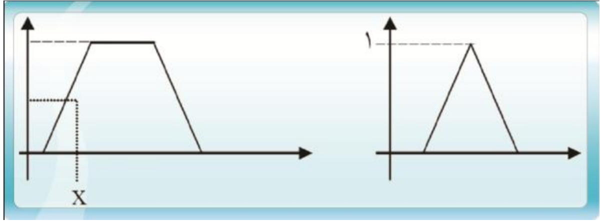


Fig 5. A view of triangle and trapezoid membership functions.

In figure 5, if the set is mean, object X with belonging degree 0.6 is the member of the set. The fuzzy set with a set of ordered pairs, the first member denotes the object and the second member is the membership degree of the object to the set and is shown as following (Zimmermann, 1996).

$A = \{(x, \mu_A(x)) \mid x \in X\}$

Some fuzzy sets form fuzzy space together. This space shows the various sets that an object can belonging to them. For example, Fig. 6 shows the fuzzy space a time variable can have low, medium low, medium high, and high.

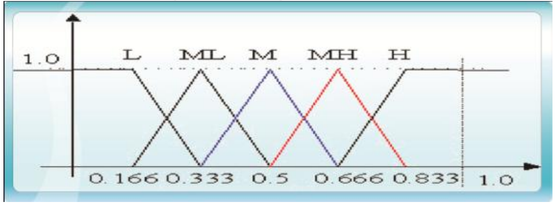


Fig. 6. A view of a fuzzy example consisting of five fuzzy sets.

The advantage of using fuzzy space or the display of quantity space sets with fuzzy sets is such that an object at the same time can belongs to two sets. This advantage is useful when determining the exact component between two sets is difficult.

Fuzzy operators

Two main sets of logical operators for fuzzy sets are S-norm and T-norm operators.

These operators have monotony, displacement and participation. In addition, T-norm operator should satisfy equation 1 and S-norm in equation 2 as following (Zimmermann, 1996).

$1: t(0,0) = 0$

$t(\mu_A(x),1) = t(1, \mu_A(x)) = \mu_A(x), x \in X.$

$2: s(1,1) = 1$

$s(\mu_A(x),0) = s(0, \mu_A(x)) = \mu_A(x), x \in X.$

The operators defined in these two general groups are operator product and share of T-norm and is considered the operator sum of S-norm.

Figure 7 shows the space of S-norm, T-norm operators and averaging operators (Zimmermann, 1996). The weighted or normal geometry and arithmetic means are averaging operators.

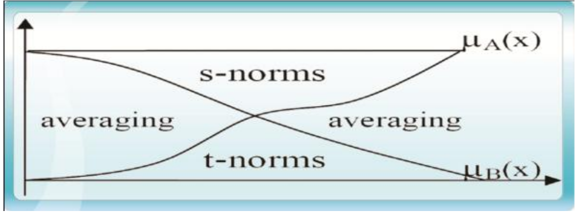


Figure 7: The comparative range of S-norm and T-norm operators and averaging (Zimmermann, 1996).

In Multi criteria Decision Making, various aggregation operators are applied. Weighted arithmetic mean operators, weighted geometry mean and Quasi-arithmetic mean are some examples of the classic aggregation operators (Fodor, 1995).

Two fuzzy sets A and B are considered. The descriptive attributes of the members of these two sets are independent; it means that μ_A values are not affected by μ_B and vice versa. If $\mu_{A\cap B}$ shows aggregation of μ_A and μ_B by share modeling and it is described as following:

$$\mu_{A\cap B} = (W_A \mu_A) \circ (W_B \mu_B)$$

In the above equation, W_A, W_B are corresponding weights and 0 shows the algebra operator. If there is no difference between two sets in the model, the weights are equal (Zimmermann, 1996).

An example of new aggregation operators is Ordered Weighted Averaging (OWA). Yager presented this operator in 1988. In this operator, the aim of the various weights application is based on specific order not specific value of vector. The aim of $M^{(m)}(X_1,...,X_m) \in R$ value to this vector is by M aggregator. m is the non-negative weight $(W_1^{(m)},...,W_m^{(m)})$ to M aggregator operator of OWA type:

$$(1 \le i \le m)\sum W_i = 1$$

And

$$M^{(m)}(X_1,...,X_m) = \sum W_i^{(m)} X_{(i)}, 1 \le i \le m, X_{(1)} < ... < X_{(i)} < ... < X_{(m)}$$

Where, $X_{(i)}$ is the increasing order of X_i's. OWA operator by attributing specific values to the weight are changed into maximum, minimum, arithmetic mean, averaging and other operators (Fodor, 1995).

Fuzzy classifier

Fuzzy rules form the fuzzy classifier. A fuzzy rule is generally is as following:

Fuzzy rule: IF condition THEN consequent [weight]

Condition section and consisting of fuzzy result, fuzzy operators and fuzzy sets. For example, a condition sections is shown as: **Condition: (X1 is [not] A1) and (X2 is [not] A2)… and (Xk is [not] Ak)**

The second section is consequent and in classified discussion shows a special class. Weight is the validity of the rule. In classification, we can apply the above term and for each class of fuzzy rule is obtained as a classifier.

C-Mean fuzzy clustering

C-Mean fuzzy clustering algorithm minimizes the following objective function:

$$J(U,V) = \sum_{k=1}^{n}\sum_{i=1}^{c}(u_{ik})^m \| X_k - V_i \|^2$$

$$u_{ik} : [0,1]$$

$$\forall k, \sum_{i=1}^{c} u_{ik} = 1$$

Matrix u denotes Kth belonging degree (X_k) to ith set being calculated based on X_k distance to the member of ith set (V_i). m is the weight power applied. c the number of sets and n the total number of the data (Zimmermann, 1996).

After applying the algorithm, u shows the fuzzy sets of the corresponding space with the sets and v shows the example membership of each set.

Initial concepts of genetic

The classifying chromosome view

A simple rule is expressed by various methods and chromosome view can be for all of them. These methods are including (Gomez1, 2002):

Conjunctive normal form in this view is composed of conjunction combination of disjunctive normal form. A view of CNF-K is defined as following. In this equation $p = m \times k + 1$, m is a natural number.

$$CNF - k : (A_1 \vee A_2 \vee ... \vee A_K) \wedge (A_{k+1} \vee A_{k+2} \vee ... \vee A_{2k}) \wedge ... \wedge (A_p \vee A_{p+1} \vee ... \vee A_{p+k+!})$$

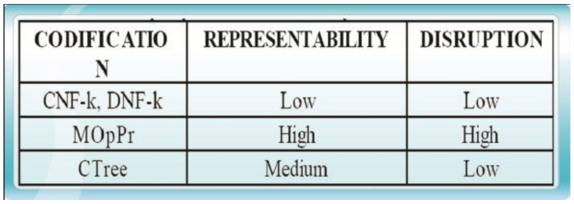
Normal form disjunction in this view, the rule is consisting of disjunctive combination of conjunctive normal form. A view of DNF-K is defined as following:

$$DNF - k : (A_1 \wedge A_2 \wedge ... \wedge A_K) \vee (A_{k+1} \wedge A_{k+2} \wedge ... \wedge A_{2k}) \vee ... \vee (A_p \wedge A_{p+1} \wedge ... \wedge A_{p+k+!})$$

The conditional fixed structure, in this method, a fixed conditional structure is considered as a model for the rule. Linear tree with the priority of the operators, in this method a special priority is determined for the operator. Binary full tree, in this method, the conditional terms are considered as full tree form.

Each of the above methods, have special capabilities. The capability of presenting various rules, suitability for better application of genetic operators and not complicating them, the correct computation of the chromosome fit and not being affected by evolution parameters are the features. A comparison is made between these methods in Table 2 (Gomez1, 2002).

Table 2. The comparison of various methods of rule view (Gomez1, 2002).



CODIFICATIO N	REPRESENTABILITY	DISRUPTION
CNF-k, DNF-k	Low	Low
MOpPr	High	High
CTree	Medium	Low

The important point here is considering rule condition in the methods of review. The necessity of review of the consequent section in chromosome depends upon the application of data mining and required evolution trend. In the mentioned view in (Freitas, 2002), the rule is reviewed by high-level method. It means that the variable, operator and fuzzy set are coded separately. In the review of simple and limited rules, other methods are used.

Genetic operators and classifiers

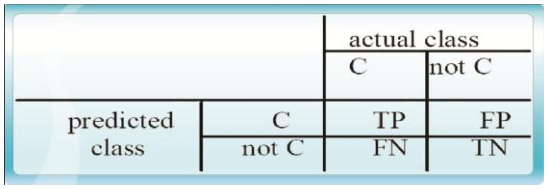
Cross over operator in the evolution of the classifiers should have two features, the local changes in linear review is created and acts well in the chromosomes with varied length. The optimum cross over operator should create a child similar to his parents and it should not create invalid chromosome implicitly. In addition to using general crossover operator, special operators general or specific cross over are defined in data mining and rules evolution. Mutation operator does not have the special considerations and is used normally. In the selected review, the common genetic operators can be applied optimally on the chromosomes and they do not need any additional processing.

The fitness of fuzzy classifiers

Generally, in data mining issue, optimality of the extracted information is dependent upon the application but it can be said that the extracted data is optimum when it is true, perceptible and attractive. Accuracy is the validity of the predicted data as in most of the cases; the determination of the variable future based on the observed data is good. The extracted data should be perceptible. Most of the extracted data is used for decision-making, the user should rely on it, and it should be interpreted. The information is perceived by high-level presentation. As one common method of data mining is using predictive rules. A suitable method for the evaluation of the accuracy of the rule is using confusion matrix (Gupta, 2009). This matrix is shown in Fig 8. Each of the elements means: TP: true positive means the number of the examples satisfy the rule and its set is determined accurately. FT false positive is the number of the elements satisfies in the rule but their set is different from the set determined in the rule. FN false negative is consisting of the number of examples not satisfy in the role but belong to the determined set in the rule and TN true negative is consisting of the number of the elements not satisfy the rule and do not belong the rule set. By this matrix, both Confidence Factor (CF) and Completeness Measure (comp) of the rule are computed as following (Freitas, 2002):

$CF = TP/(TP+FP)$

$Comp = TP/(TP+FN)$



		actual class	
		C	not C
predicted class	C	TP	FP
	not C	FN	TN

Fig. 8. Confusion matrix for the accuracy of the rule calculation (Freitas, 2002).

Fitness function is proposed based on these two criteria:

$Fitness = CF \times Comp$

In completeness criterion, in addition to coving the elements belonging to $Comp$ set, the lack of covering the elements not belonging to the set, $Comp_2$ is defined and two equations are defined as following:

By these two criteria, another accuracy fitness function of weighted sum of completeness (Gomez2, 2002) and the product of these two criteria [9] are presented as following:

$Comp_1 = TP/(TP+FN),$

$Comp_2 = TN/(TN+FP)$

To consider the perception of the rule, besides its accuracy, another criterion should be added to fitness function and it is evaluated by rule simplicity:

$Fitness = W_1 \times (TP/(TP+FN)) + W_2 \times (TN/(TN+FP))$

$Fitness = (TP/(TP+FN)) \times (TN/(TN+FP))$

$Simp$ is simplicity criterion of the rule ranging 0, 1. W_i's are the weights defined by the user. The simplicity criterion is consistent with the number of conditions applied in the rule (Freitas, 2002). The following equation is another example of simplicity criterion estimation of the rule that is defined by chromosome length ($chrom\_length$) (Gomez2, 2002):

$Fitness = W_1 \times Fitness_{accuracy} + W_2 \times Simp$

To calculate the confusion matrix elements for fuzzy classifiers, the following equations are proposed (Gomez2, 2002):

$$TP = \sum_{i=1}^{P} predicted\,(class_1 data_i)$$

$$FP = \sum_{i=1}^{q} prediction\,(class_2 data_i)$$

$$TN = \sum_{i=1}^{q} (1 - prediction\,(class_2 data_i\,))$$

$$FN = \sum_{i=1}^{p} (1 - prediction\,(class_1 data_i))$$

The above equations are for the class 1 classifier in a binary classification. $prediction\,(class_a data_i)$ is the belonging of i th data of classifier n. The number of first class element is p and the number of second-class element is q.

The evolution methods of classifiers

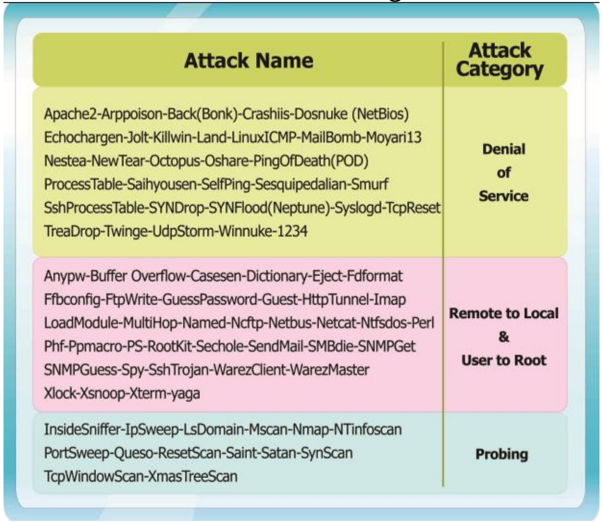
The method of the final process of evolution is related to the review of rules consequent. To review the consequent section of the rule, there are various methods. One method is its review in gene series of population and putting it in evolution process. Another method is attributing all the people of the population to a similar set and after evolution, the corresponding rules are extracted with the set. This trend should be replicated to the number of the existing set. The third method is the absolute determination of the set of the rule after the formation of its condition section. The determined set can be a set achieving the most vote of the existing data examples or the set a person value with that set is maximized in the evaluation of the fitness function. The first and third method has the advantage of the low implementation of evolution algorithm to the second method. The third method is more logical compared to the first method. In most of the applications, namely in cases where the number of the existing sets are low, as intrusion detection, the second view is applied (Gomez2, 2002; Mendes, 2001).

In binary classification of Co-Evolution, the classifiers can be useful and avoids the contradictory classifiers in the conditions the problem space is divided into non-partitioned groups.

Attacks Categories

Based on the (Azarkasb, 2012) attacks broken up into three following categories: Denial of Service (DoS) attacks, User-to-Root (U2R) and Remote-to-Local (R2L) attacks, and Probing. Table 3 shows the 88 different attacks that threat operating systems. The next three sections present detailed descriptions of each class of attacks.

Table. 3. Attacks Categories.



Attack Name	Attack Category
Apache2-Arppoison-Back(Bonk)-Crashiis-Dosnuke (NetBios) Echochargen-Jolt-Killwin-Land-LinuxICMP-MailBomb-Moyari13 Nestea-NewTear-Octopus-Oshare-PingOfDeath(POD) ProcessTable-Saihyousen-SelfPing-Sesquipedalian-Smurf SshProcessTable-SYNDrop-SYNFlood(Neptune)-Syslogd-TcpReset TreaDrop-Twinge-UdpStorm-Winnuke-1234	Denial of Service
Anypw-Buffer Overflow-Casesen-Dictionary-Eject-Fdformat Ffbconfig-FtpWrite-GuessPassword-Guest-HttpTunnel-Imap LoadModule-MultiHop-Named-Ncftp-Netbus-Netcat-Ntfsdos-Perl Phf-Ppmacro-PS-RootKit-Sechole-SendMail-SMBdie-SNMPGet SNMPGuess-Spy-SshTrojan-WarezClient-WarezMaster Xlock-Xsnoop-Xterm-yaga	Remote to Local & User to Root
InsideSniffer-IpSweep-LsDomain-Mscan-Nmap-NTinfoscan PortSweep-Queso-ResetScan-Saint-Satan-SynScan TcpWindowScan-XmasTreeScan	Probing

Denial of Service Attacks

A denial of service attack is an attack in which the attacker makes some computing or memory resource too busy or too full to handle legitimate requests, or denies legitimate users access to a machine. There are many varieties of denial of service (DoS) attacks. Some DoS attacks (like mailbomb, neptune, or smurf attack) abuse a perfectly legitimate feature. Others (Teardrop, Ping of Death) create malformed packets that confuse the TCP/IP stack of the machine that is trying to reconstruct the packet. Still others (apache2, back, syslogd) take advantage of bugs in a particular network daemon.

User-to-Root (U2R) and Remote-to-Local (R2L) Attacks

User to Root exploits are a class of exploit in which the attacker starts out with access to a normal user account on the system (perhaps gained by sniffing passwords, a dictionary attack, or social engineering) and is able to exploit some vulnerability to gain root access to the system. There are several different types of User to Root attacks. The most common is the buffer overflow attack. Buffer overflows occur when a program copies too much data into a static buffer without checking to make sure that the data will fit. Another class of User to Root attack exploits programs that make assumptions about the environment in which they are running. A good example of such an attack is the loadmodule attack. Other User to Root attacks take advantage of programs that are not careful about the way they manage temporary files. Finally, some User to Root vulnerabilities exists because of an exploitable race condition in the actions of a single program, or two or more programs running simultaneously.

A Remote to User attack occurs when an attacker who has the ability to send packets to a machine over a network but who does not have an account on that machine exploits some vulnerability to gain local access as a user of that machine. There are many possible ways an attacker can gain unauthorized access to a local account on a machine. Some of the attacks discussed within this section exploit buffer overflows in network server software (imap, named, sendmail). The Dictionary, FtpWrite, Guest and Xsnoop attacks all attempt to exploit weak or miss configured system security policies. The Xlock attack involves social engineering in order for the attack to be successful the attacker must successfully spoof a human operator into supplying their password to a screensaver that is actually a Trojan horse.

Probes Attacks

In recent years, a growing number of programs have been distributed that can automatically scan a network of computers to gather information or find known vulnerabilities. These network probes are quite useful to an attacker who is staging a future attack. An attacker with a map of available machines and services on a network can look for weak points of the network. Some of these scanning tools (satan, saint, mscan) enable even a very unskilled attacker to very quickly check hundreds or thousands of machines on a network for known vulnerabilities (Kendall, 1999).

The proposed fuzzy genetic system

The main focus of this section of the paper is on the design of good analysis for fuzzy and genetic methods. An intelligent analysis detecting the new and unknown attacks. The more the diversity of the attacks, the more difficult the study and creation of good algorithm for them for security expert and it is required to extract more and suitable criteria to detect their behavior. In addition, the relation and combination of various criteria are determined to describe each specific behavior accurately. Fuzzy systems can cover the complexities of such systems and are effective in reduction of error by natural fuzzy view to the criteria.

The system analyzer is genetic-based fuzzy system and fuzzy classifiers are responsible for their detection and decision-making. Thus, most of the criteria and the features are fuzzy linguistic variables. The final detection of the analyzer is fuzzy. It means that analyzer output besides attack event alarm determines the validity percent and its occurrence. As the data space and describing rules of the network behavior is a complex space, genetic algorithms are applied for learning the rules base in the analyzer. The architecture of the proposed intrusion detection system is shown in Fig 9. In this architecture, the main components of intrusion detection system including sensor, analyzer and user medium are observed. The audit data source is sensor input.

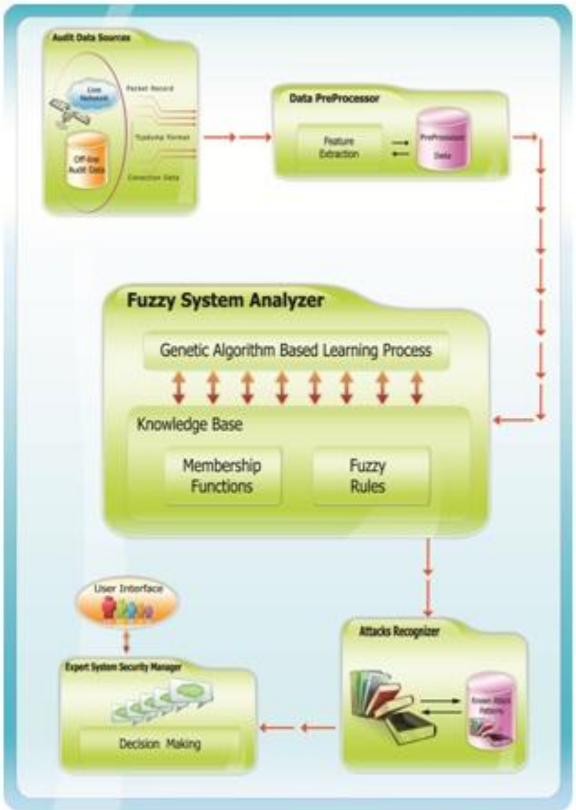


Fig. 9. The architecture of the proposed fuzzy genetic system.

In this Figure, the data base rules applied by analyzer fuzzy system are similar to Sugno controller rules (Zimmermann, 1996). The general form of the rules of this controller is as following:

Rule r: if X1 is A1 and X2 is A2 and … and Xn is An) then u is fr(X1 ,X2 ,…Xn)

In the above equation, f function is linear. In the required application, f is a function taking only two various values. One value corresponding with the set of network normal behaviors and another one is correspondent with the attacks set. Each simple rule in rules base is as the above. A set of simple rules forms the normal behavior classifier and another set is the corresponding classifier with the attacks. In other words, the analyzer applies two combinational rules as DNF form for intrusion detection. In the mentioned fuzzy system, the fuzzy operator minimum is correspondent with and maximum is correspondent with or. Fuzzy inference in this system is as in each classifier, at first the activity of each simple rule as the conjunction normal form is calculated. Then, the simple rule results are combined with maximizing. The result shows the event occurrence degree defining by function f.

The outputs of both classifier, forms the final output of the fuzzy system. The required membership function for description of the fuzzy space of each criterion is achieved by C-Mean classification algorithm (Zimmermann, 1996). This algorithm is implemented before applying the genetic algorithm to design the classifiers. Thus, the details of membership functions are not in the evolution trend. Some of the advantages of this method are good fuzzy description of the criteria and reduction of complexity and genetic algorithm cost. Membership functions are applied in fuzzification section of the system. This section fuzzificate the data receiving from the input media pre-processor for the analysis. The source of receiving the audit data, provides the supervised data as the raw input of intrusion detection system. Normally, the volume of raw data is very much and decision-making is complex based on it. It is required that the criterion extraction unit receives the data receive output and extract the important criteria for the analysis. The pre-processor section prepares the extracted data for delivery to the fuzzy system. Pre-processing operation is defined based on the type of system and the applied criteria including the data normalization and numericalization of discrete values series, etc. In the proposed architecture, analyzer output is fuzzy and a unit is considered for the final decision-making. This unit is shown as intrusion detector in the figure. The reason of separating this section from non-fazzification unit of the analyzer is separating its performance. Although it is possible that this unit is only responsible for simple thresholding but its role in customization of the intrusion detection system is optimum. This unit based on two-state absolute result besides generating the buzzer, gives some information about the attack for system security expert. In various organizations and networks and for various security authorities, the event sensitivity and the type of attack are different. By regulating this section, we can hear the alarm only when the event validity is more than the defined limit. This possibility is useful for anomaly detection systems generating the alarm.

Implementation

The learning method in the proposed system is based on the method introduced by Pittsburg and disjunction normal term method is applied for classifying chromosome display. This method is capable in presenting the various rules.

Its suitable performance is explained with genetic operators and fitness function in the following.

The selection of the fitness function correspondent with non-educational examples detection is very difficult. Thus, our aim is defining fitness function that is correspondent with classifier detection power in new and non-educational examples. In the optimization of the classifiers, the aim is simultaneous satisfaction of the conditions and completeness aims. Thus, using T-norm operator is necessary in the combination of the criteria. In other words, as the suitable rule is the rule covering all the set examples well and not including the examples of other sets. Thus, we considered the conjunction combination of these two criteria. Thus, the risk of evolution trend to local optimized solutions, empty rule and general rule is eliminated. Thus, fitness function of classifier accuracy is defined as following:

$$Fitness = (TP/(TP+FN)) \times (TN/(TN+FP))$$

The calculation of confusion matrix elements for fuzzy classifiers in the proposed method is defined as follows:

$$TP = \sum_{i=1}^{P} predicted\,(class_1 data_i)^{1/4}$$

$$FP = \sum_{i=1}^{q} prediction\,(class_2 data_i)^{1/4}$$

$$TN = \sum_{i=1}^{q} (1 - prediction\,(class_2 data_i\,))^{1/4}$$

$$FN = \sum_{i=1}^{p} (1 - prediction\,(class_1 data_i))^{1/4}$$

The above equations are dedicated to set 1 classifier. $prediction\,(class_a data_i)$ is the data belonging of i th of set n to the related set. The number of the first set elements is p and the number of the second set elements is q. To calculate the confusion matrix elements instead of the sum of detection, the sum of the square squares is applied. The application of the sum of the square squares causes that the rules have general detection of their set elements. In other words, in this method the general score method detecting all the examples, as average 0.25 is more than the specific rule that is only including a set of the examples of the set. In addition, the fuzzy aggregation operator is applied for correct estimation of fitness function. This operator is considered suitable with the special application of the problem. To do this, W_{agg} function is defined as following in Fig 10.:

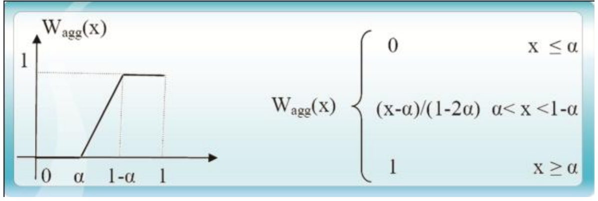


Fig. 10. The applied aggregation function in fitness function.

By W_{agg} function, the equations of the calculation of confusion matrix elements are changed as following:

$$TP=\sum_{i=1}^{P}[W_{agg}(predicted\,(class_1data_i))\times predicted\,(class_1\,data_i)]^{1/4}$$

$$FP=\sum_{i=1}^{q}[W_{agg}(prediction\,(class_2data_i))\times predicted\,(class_2data_i)]^{1/4}$$

$$TN=\sum_{i=1}^{q}[W_{agg}(1-prediction\,(class_2data_i))\times(1-predicted\,(class_2data_i))]^{1/4}$$

$$FN=\sum_{i=1}^{p}[W_{agg}(1-prediction\,(class_1data_i))\times(1-predicted\,(class_1data))]^{1/4}$$

To consider the rule perception criterion, the rule simplicity is considered. The rules review method is selected as $Simp$ simplicity criterion is defined as well. This criterion is a function of total number of simple rules Rc and the total number of conditions in combined rule Cc:

$$Simp=1/Cc+Rc/Cc$$

By these considerations, the final fitness function for the evaluation of a classifier is as following:

$$Fitness=W_1\times Fitness_{accuracy}+W_2\times Simp$$

The fuzzy sets are defined before evolution exactly and only their numbers are mentioned in the rule. Thus, the un-coded fuzzy set parameters are not in evolution process.

The cross over operator is single-point cross and cross over point is selected only between the simple rules (conjunction normal terms). The reason of applying this limitation in cross over operator is the consistency of rules operation with the capability of a person in rules population. Based on a simple rule (conjunctive normal form), it will have better genetic feature in inheritance compared to one condition or the related components (variable and fuzzy set).

Mutation operator is used as including addition and omission of the gene. In these stages, gene is a variable or fuzzy set and some considerations are considered for the validity of the chromosomes. The specific case considered in applying mutation operator is the variable probability in selection of a valid media and fuzzy set. A fuzzy rule is as following:

Fuzzy rule: IF condition THEN consequent [Weight].

Condition :(x1 is [not] A1) and (x2 is [not] A2)… and (xk is [not] Ak)

In representation of the rules, only the condition is coded and in high-level representations, any component of the condition, fuzzy variable, fuzzy set and (is not and is) are separately coded. Generally, in random generation of the initial population and mutation operator application, is and is not have similar probability for presenting in the rule. It seems that this issue is not correct practically and in linguistic description of the phenomenon by an expert, (is) is applied more than (is not). In intrusion detection process, the presence probability of thee two components are varied and this change is considered both in creating the initial population and mutation operator application. The rules are in the form of DNF-3.

$$DNF-3:(A_1\wedge A_2\wedge A_3)\vee(A_4\wedge A_5\wedge A_6)\vee...\vee(A_{3k-2}\wedge A_{3k-1}A_{3k!})$$

To generate each of the evolution initial population, it is required to define the initial form of the classifier fuzzy rules. To do this, in fuzzification stage, C-mean clustering was used to determine the membership functions. Three numbers among the functions are selected. For discrete attributes, absolute sets that are non-overlapping fuzzy sets are applied as in Fig 11. In other words, triangular membership function with zero bases is used.

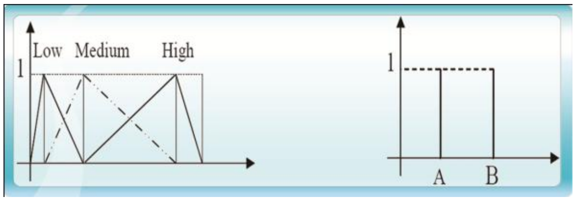


Fig. 11. An example of fuzzy space with three sets and a space example with two absolute sets.

In some of the applications, a fixed fuzzy space is considered for all the features (Gomez2, 2002) and in some of them, the space of fuzzy sets are obtained in co-evolution with the classifiers (Mendes, 2001). The selection of the parents or the chromosomes generating the next generation is done by tournament method.

Co-evolution is one of the best techniques of this paper in implementation of the proposed system. In this evolutionary trend, the aim is the evolution of the classifiers expecting good collaboration of them in detection. For the evolution of the classifier, the existing view is the representation of all the classifiers in a chromosome based on Pittsburg view. In this method, the search space is getting bigger and more complex. In the proposed co-evolution, the search space is the same as search space in the separate evolution. The execution cost is not increased on condition of not complication of the fitness function. Another view is the one-directional co-evolution. This method means that in co-evolution, only one of the population is evolved based on the collaboration method with another population and another population is evolved independently. The proposed fitness function is defined to determine the value in the collaboration of two chromosomes of two various populations as:

$$Fitness_{accuracy} = ((Comp_1 a + Comp_2 b)/2) \times ((Comp_2 a + Comp_1 b)/2)$$

$Comp$ is completeness criteria and a, b show that the required parameter is attributed to set a chromosome or set b.

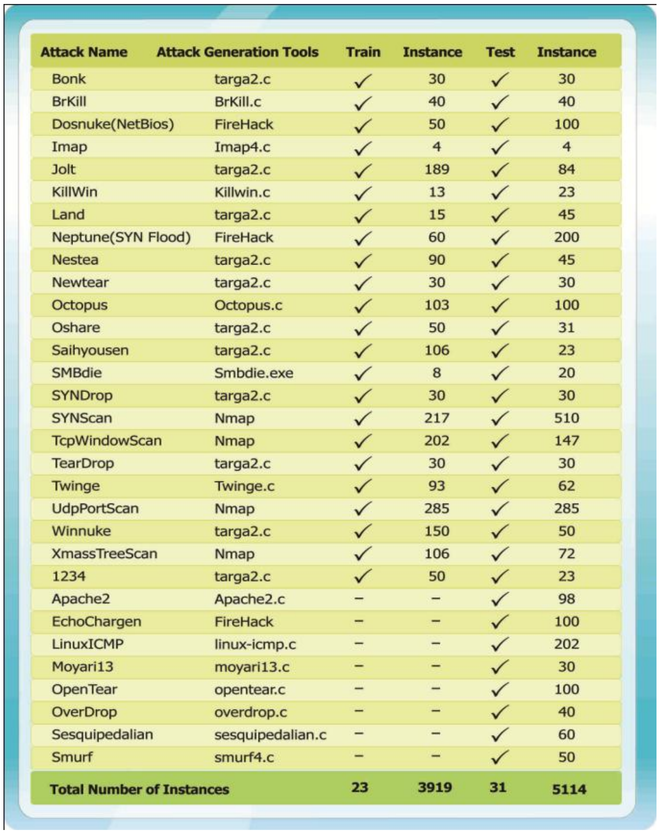
When the fuzzy rules are applied in classification, the rule activation degree and its validity degree show the belonging of the example to the required rule set in the detection of the examples by the classifiers. Thus, the rule activation in each example is multiplied by the rule validity degree, and then the corresponding set with the example is determined.

$CF = TP/(TP + FP)$

During the detection, the examples based on only classifier have detection threshold 0.25. In co-detection of the classifiers, the example belongs to the set that the corresponding classifier had the highest detection. In binary classification issues, we can use both above methods and choose the best method based on the improvement of the problem progress.

Test, evaluation, comparison and analysis of the results

Table 4. Train and test data attack types with their tools and instances.



Attack Name	Attack Generation Tools	Train	Instance	Test	Instance
Bonk	targa2.c	✓	30	✓	30
BrKill	BrKill.c	✓	40	✓	40
Dosnuke(NetBios)	FireHack	✓	50	✓	100
Imap	Imap4.c	✓	4	✓	4
Jolt	targa2.c	✓	189	✓	84
KillWin	Killwin.c	✓	13	✓	23
Land	targa2.c	✓	15	✓	45
Neptune(SYN Flood)	FireHack	✓	60	✓	200
Nestea	targa2.c	✓	90	✓	45
Newtear	targa2.c	✓	30	✓	30
Octopus	Octopus.c	✓	103	✓	100
Oshare	targa2.c	✓	50	✓	31
Saihyousen	targa2.c	✓	106	✓	23
SMBdie	Smbdie.exe	✓	8	✓	20
SYNDrop	targa2.c	✓	30	✓	30
SYNScan	Nmap	✓	217	✓	510
TcpWindowScan	Nmap	✓	202	✓	147
TearDrop	targa2.c	✓	30	✓	30
Twinge	Twinge.c	✓	93	✓	62
UdpPortScan	Nmap	✓	285	✓	285
Winnuke	targa2.c	✓	150	✓	50
XmassTreeScan	Nmap	✓	106	✓	72
1234	targa2.c	✓	50	✓	23
Apache2	Apache2.c	–	–	✓	98
EchoChargen	FireHack	–	–	✓	100
LinuxICMP	linux-icmp.c	–	–	✓	202
Moyari13	moyari13.c	–	–	✓	30
OpenTear	opentear.c	–	–	✓	100
OverDrop	overdrop.c	–	–	✓	40
Sesquipedalian	sesquipedalian.c	–	–	✓	60
Smurf	smurf4.c	–	–	✓	50
Total Number of Instances		**23**	**3919**	**31**	**5114**

For system test and training, similar to (Azarkasb, 2009; Azarkasb, 2013), this paper attempted that by collecting the existing tools in producing the attacks produce some of the existing attacks against the network agent and collected the traffic as known attacks traffic. To collect normal traffic in a working day, after 10 minutes of network traffic and in three other days at the peak of traffic, 5 to 10 minutes of network traffic as normal traffic consisting of 5114 examples were collected. Thus, a set of test including 23 types of attacks in training data with 8 types of new attacks not existing in training data, consisting of 3919 instances were provided. In the existing attacks in the mentioned data with applied instruments to produce them and the number of applied instances are shown in Table 4.

To test the system, the genetic algorithm was applied on the data set and fuzzy system rules. Any initial population has 150 individuals created randomly. The evolution trend is replicated at most 150 generations. The special parameters and explanations of the test are expressed briefly as:

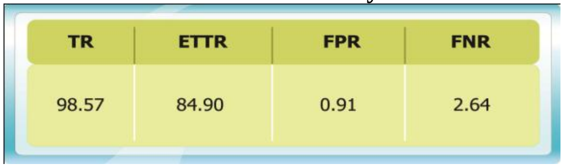
- **Membership function:** It is considered for continual values of five Trapezoid fuzzy sets low, medium low, medium, high, medium high are considered and was achieved for each separate criterion by applying C-Mean algorithm. When the sets example members have low distance in algorithm output, the number of fuzzy sets for the criterion is less than 5. For discrete values, for the number of the values taken by the criterion, an absolute set is considered. The continual data before applying algorithm are normalized in 0, 1 interval.
- **Chromosome:** Fuzzy combinational rules as DNF-3 form describe a full classifier and form a chromosome. The fuzzy variables, fuzzy set number of one of the give sets and relation is or is not are coded separately.
- **Initial population:** For each set, a random population with 150 people is created.
- **Selection:** The selection of the generating units of the second generation is done by tournament method.
- **The number of the new generated chromosome in each generation:** Four parents are selected and four new people enter the next generation. For replacing the children due to the fixed number of the number of population, the people with low fitness are excluded.
- **The probability of (is not) in generating the new generation, the probability of applying mutation operator and parameter α in aggregation function:** this parameter is selected to test as empirically.
- **The number of generation:** The maximum evolution trend during 150 generations is tested.

- **Fitness function parameters:** Fitness function with parameters $W_1 = 0.995$(for accuracy criterion of the rule) and $W_2 = 0.005$(for the simplicity of the rule) is evaluated.

During the evolution, after 25 iterations, pruning of the rules is used. In this trend, the existing iterative simple rules are omitted in a classifier. To reduce the implementation costs of this trend, it can be applied only based on the best people of the population.

The evaluation results are shown in Table 5 based on the True detection Rate (TR), Exact True Type detection Rate (ETTR), False Positive detection Rate (FPR), and False Negative detection Rate (FNR).

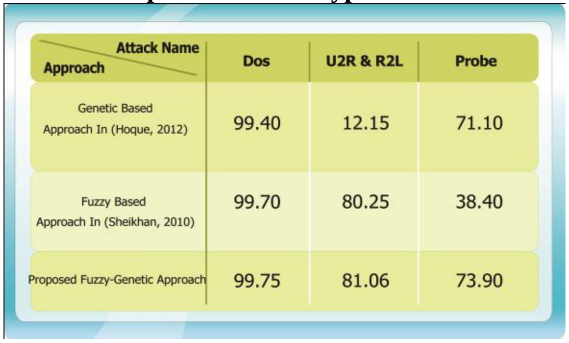
Table 5. Detection performance of fuzzy-genetic intrusion detection system.



TR	ETTR	FPR	FNR
98.57	84.90	0.91	2.64

In the test, the false positive error is lower than the false negative error. It means that the classifiers learnt normal behavior well and low mistake alarm was generated. The high error and high generation of false alarms in the real environment in which we expect the observation of normal behavior, is unsuitable and it is possible to make the security expert hopeless of using such a system. The observation of the false negative error in the system is expected as the system is based on learning and the number of new attacks in the test data is high. To compare the performance of proposed system we evaluated it with two other approaches as shown in table 6 based on three above attacks categories.

Table 6. The comparison of the efficiency of the presented method with other related methods by the separation of the type of attacks.



Attack Name Approach	Dos	U2R & R2L	Probe
Genetic Based Approach In (Hoque, 2012)	99.40	12.15	71.10
Fuzzy Based Approach In (Sheikhan, 2010)	99.70	80.25	38.40
Proposed Fuzzy-Genetic Approach	99.75	81.06	73.90

Conclusion

In this paper, the intrusion detection system is designed and evaluated and its analysis engine is the fuzzy genetic system. The genetic algorithm is applied as it can extract both normal behavior models and intrusion models in the form of fuzzy rules of educational data. The evolutionary learning methods of the classifiers are evaluated as general methods in the paper. In this evaluation, better results of the existing similar systems results were obtained.

In the evaluation in this paper, a set of data was provided by implementation of attacks generation tools including suitable criteria to do detection operation. In the tests with the data set, the analyzer by fuzzy rules base obtained in learning process identified the service denial attacks and scanning attacks. The required system detects the new and unknown attacks as relatively well. The positive false error is low both in detection by the aid of normal behavior classifier and in detection by attacks classifier and this advantage detects using the proposed method in real systems and the systems based on anomaly detection.

As the proposed system of a system is based on rule, its performance is understood for the security expert. This advantage is manifested during the fuzzification of the fuzzy rules. As the similar fuzzy rules with human linguistic description, explain various states. In the introduced system of genetic design output is presented and modified by the security expert. The results of detection are activated with the reasons of rules trend, event detection degree as the attack can be presented to the security person, and he applied the required security measurements with the probable changes in the system.

The proposed system is based on machine learning. It is better that machine learning is beside the expert human knowledge and help it not replacing it. By this attitude, we can increase the system capability. Thus, in this work, we can present some simple recommendations being useful about the construction of an operational and commercial system as explained in the next section. Some of the recommendations lead into the limitation of genetic algorithm search space and the rest of the path is easier to achieve the optimum answer.

Development grounds and future works

In this section, improving system's performance we try to recommend some flowing future objectives:

- In determining the membership functions and fuzzy space of the criteria, we can use expert knowledge.
- According to the previous knowledge, for various criteria, different presence probability is controlled in the rules. This view can be similar to the method presented in this paper for (is not) both in the initial population and evolution trend.
- It is better to omit the contradictory combination of the criteria of search space. Indeed, in implementation of this idea, the exceptional

considerations are used to decrease the algorithm implementation costs. This can be considered as a part of pruning of the rules.

- Instead of using the random initial population in genetic algorithm, we can apply expert knowledge in determining the initial space of the search and put the simple or combined pre-defined rules in the population.
- The data normalization trend in pre-processing can be dependent upon the criterion and with better methods. For example, the logarithm normalization is suitable for some of the criteria than linear normalization.
- Presentation of the rules to the expert after education stage and obtaining the consent before the application can increase the system efficiency.

The items were the recommendations to increase the system efficiency by the aid of expert knowledge. To improve the machine-learning trend of the system, the following recommendations are raised.

- As the decrease of which type of error is of great importance in the required system can design the fitness function for genetic algorithm to prefer the classifiers with the required priority. The proposed fitness function leads into the reduction of both types of errors in evolutionary trend with similar priority.
- In the tests performed in this paper, it is shown that using only one classifier had better result of simultaneous use of the classifiers. It is possible to combine the suitable mechanisms of the detection results of both classifiers not contradictory with each other and obtain better accuracy on detection. Using theory of evidence in (Shafer, 1976) are some of the recommendations.
- The separate education of different kinds of attacks and creation of a combinational detection system can have better results.
- The detection of new attacks is not adequate based on the recognized attacks and the system needs the updating section of the rules. This section can be based on general learning method. Except this view, the control attitude and applying the gradual and on line evolutionary learning methods can be useful.
-

Correspondence author:
Sayed Omid Azarkasb
Artificial Intelligence MSc,
Faculty of Computer Engineering, Qazvin Branch
Azad University, Tehran, Iran.
Cellular Phone: +989123515900

REFERENCES

[1] Abdullah B, Abd-alghafar I, Gouda I. Salama, Abd-alhafez A. (2009). Performance Evaluation of a Genetic Algorithm Based Approach to Network Intrusion Detection System. 13th International Conference on Aerospace Sciences and Aviation Technology (ASAT).

[2] Azarkasb S.O. (2013). An Efficient Expert System for Intrusion Detection. American Journal of Scientific Research, Issue 86, pp. 53-80.

[3] Azarkasb S.O, Shiry Ghidary S. (2012). Logs Correlation: Current Approaches, Promising Directions, and Future Policies. Journal of Basic and Applied Scientific Research. Volume 2, Issue 5, pp. 4413-4422.

[4] Azarkasb S.O, Shiry Ghidary S. (2009). New approaches for intrusion detection based on logs correlation. IEEE International Conference on Intelligence and Security Informatics (ISI), Dallas, TX, US.

[5] Boughaci D, Herkat M.L, Lazzazi M.A. (2012). A specific fuzzy genetic Algorithm for intrusion detection. In Proceedings of the Second International Conference on Communication and Information Technology (ICCIT).

[6] Boughaci D, Bouhali S, Ordeche S. (2011). A Fuzzy Local Search for intrusion detection. ACIT.

[7] Bridges S.M. Vaughn R.B. (2000). Fuzzy data mining and genetic algorithms applied to intrusion detection. Proceedings of twenty third national information systems security conference, Baltimore, MD.

[8] Chittur A. (2001). Model Generation for an Intrusion Detection System Using Genetic Algorithms. Ossining High School, NY.

[9] Crosbie M, Spafford E, (1995). Applying Genetic Programming to Intrusion Detection. Proceedings of the AAAI Fall Symposium.

[10] Dickerson J, Juslin E, Koukousoula J. (2001). Fuzzy intrusion detection. In the proceedings of North American fuzzy information processing society (NAFIPS), Vancouver, Canada.

[11] Fodor J, Marichal J.L, Roubens M. (1995). Characterization of some aggregation functions arising from MCDM problems, fuzzy logic and soft computing. Edited by Bouchon-Meunier B., Yager R.R and Zadeh L.A., World Scientific, Singapore, pp. 194-201.

[12] Freitas A.A. (2002). A survey of evolutionary algorithms for data mining and Knowledge discovery. In A. Ghosh , and S. Tsutsui, editors, Advances in evolutionary computation , Springer-verlag, pp. 819-845.

[13] Gao M, Zhou M.C. (2003). Fuzzy intrusion detection based on fuzzy reasoning Petri nets. In Proceeding of the IEEE International

Conference on Systems, Man and Cybernetics, Washington D.C., pp. 1272-1277.

[14] Gomez1 J, Gonzalez F, Dasgupta D, Nasraoui O. (2002). Complete expression trees for evolving fuzzy classifiers systems with genetic algorithms and application to network intrusion detection. In proceedings of NAFIPS-FLINT joint conference, New Orleans, La, pp. 469-474.

[15] Gomez2 J, Dasgupta D. (2002). Evolving fuzzy classifiers systems for intrusion detection. In proceedings of IEEE Workshop on Information Assurance.

[16] Gong R.H, Zulkernine M. Abolmaesumi P. (2005). A Software Implementation of a Genetic Algorithm Based Approach to Network Intrusion Detection. In Sixth IEEE ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing (SNPD), Maryland.

[17] Goyal A, Kumar C. GA-NIDS: A Genetic Algorithm based Network Intrusion Detection System. Evanston, Illinois, 2008.

[18] Gupta K.K. (2009). Robust and Efficient Intrusion Detection Systems. Submitted in total fulfillment of the requirements of the degree of Doctor of Philosophy, Department of Computer Science and Software Engineering, The University of Melbourne.

[19] Gupta P, Shinde S.K. (2011). Genetic Algorithm Technique Used to Detect Intrusion Detection. Advance in Computing and Information Science, Volume 198, pp. 122-131.

[20] Herrera F, Magdalena L. (1997). Genetic fuzzy systems. Tatra Mountains mathematical publication, Volume 13, pp. 93-121.

[21] Hoque M.S, Mukit M.A, Bikas, M.A.N. (2012). An Implementation of Intrusion Detection System Using Genetic Algorithm. International Journal of Network Security & Its Applications (IJNSA), Volume 4, No. 2.

[22] Kendall K. A (1999). Database of Computer Attacks for the Evaluation of Intrusion Detection Systems. Master's thesis, Massachusetts Institute of Technology.

[23] Li W. Using Genetic Algorithm for Network Intrusion Detection. SANS Institute, USA, 2004.

[24] Lu W, Traore I. (2004). Detecting New Forms of Network Intrusion Using Genetic Programming. Computational Intelligence, vol. 20, pp. 3, Blackwell Publishing, Malden, pp. 475-494.

[25] Mendes R.R.F, Voznika F.D.B, Freitas A.A, Nievola J.C. (2001). Discovering fuzzy classification rules with genetic programming and Co-evolution. In Principles of Data mining and knowledge discovery, Proc 5th European conference PKDD, Lecture notes in artificial intelligence, Springer-Verlag.

[26] Middlemiss M, Dick G. (2003). Feature selection of intrusion detection data using a hybrid genetic algorithm/KNN approach. Design and application of hybrid intelligent systems, IOS Press Amsterdam, pp. 519-527.

[27] Mukkamala M, Sung A.H, Abraham A. (2005). Intrusion detection using an ensemble of intelligent paradigms. Journal of Network and Computer Applications, Volume 28, Issue 2, pp. 167-182.

[28] Peddabachigari S, Abraham A, Grosan C, Thomas J. (2007). Modeling intrusion detection system using hybrid intelligent systems. Journal of Network and Computer Applications, Volume 30, Issue 1, pp. 114–132.

[29] Peng T, Leckie C, Ramamohanarao K. (2007) Information sharing for distributed intrusion detection systems. Journal of Network and Computer Applications, Volume 30, Issue 3, pp. 877–899.

[30] Pillai M.M, Eloff J. H. P, Venter H. S. (2004). An Approach to Implement a Network Intrusion Detection System using Genetic Algorithms. In Proceedings of the 2004 annual research conference of the South African institute of computer scientists and information technologists on IT research (SAICSIT), pp. 221-228.

[31] Sadiq Ali Khan M. (2011). Rule based Network Intrusion Detection using Genetic Algorithm. International Journal of Computer Applications, 18(8): 26-29.

[32] Saniee Abadeh M, Habibi J, Lucas C. (2007). Intrusion detection using a fuzzy genetics-based learning algorithm. Journal of Network and Computer Applications, Volume 30, Issue 1, pp. 414-428.

[33] Saniee Abadeh M, Habibia J, Soroush, E. (2008). Induction of Fuzzy Classification systems via evolutionary ACO based algorithms. International journal of simulation, systems, science, technology (IJSSST), Vol. 9, No. 3.

[34] Shafer G. A (1976). Mathematical theory of evidence. Princeton university press.

[35] Shah H, Undercoffer J, Joshi A. (2003). Fuzzy Clustering for intrusion Detection. In Proceedings of the 12th IEEE International Conference on Fuzzy Systems, IEEE Press, Volume 2, pp. 1274-1278.

[36] Sheikhan M, Sharifi Rad M. (2010). Misuse Detection Based on Feature Selection by Fuzzy Association Rule Mining. World Applied Sciences Journal, Vol. 10, pp. 32-40.

[37] Xia T, Qu G, Hariri S, Yousif M. (2005). An Efficient Network Intrusion Detection Method Based on Information Theory and Genetic Algorithm. Proceedings of the 24th IEEE International Performance Computing and Communications Conference (IPCCC '05), Phoenix, AZ, USA.

[38] Zimmermann H.J. (1996). Fuzzy set theory and its application. Kluwer Academic Publishers.

4/2/2013