INTERNATIONAL JOURNAL OF PURE AND APPLIED RESEARCH IN ENGINEERING AND TECHNOLOGY

A PATH FOR HORIZING YOUR INNOVATIVE WORK

NETWORK INTRUSION DETECTION SYSTEM USING FUZZY GENETIC ALGORITHM

MISS PRIYA U KADAM
University of Mumbai MH India

Accepted Date: 16/08/2014; Published Date: 01/09/2014

Abstract: The intrusion detection systems (IDS) are becoming indispensable for effective protection against attacks that are constantly changing in magnitude and complexity. This paper proposes 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. The method is tested on the benchmark KDD'99 intrusion dataset and compared with other existing techniques available in the literature. The results are encouraging and demonstrate the benefits of the proposed approach. The KDD99 dataset is a benchmark dataset that is used in various researches while network dataset is an online network data captured in actual network environment. Genetic algorithm uses an development and collection that uses a chromosome-like data structure and develop the chromosomes using selection, crossover and mutation operators. Fuzzy rule is a machine learning algorithm that can sort network attack data.

Keywords: Fuzzy genetic algorithm, intrusion detection, intrusion detection system, KDD Cup 1999 dataset

Corresponding Author: MS. PRIYA KADAM

Access Online On:

www.ijpret.com

How to Cite This Article:

Priya Kadam, IJPRET, 2014; Volume 3 (1): 51-60

PAPER-QR CODE

INTRODUCTION

The speedy increase in the use of computer and computer network in today's civilization seeks extremely secured and trusted communication. There are a range of approaches being utilized in intrusion detections, but any of the systems is not completely perfect. Intrusion is referred as any set of events that try to negotiate the integrity, confidentiality, or availability of a computer resource. The process of finding out the irregular activities is known as intrusion detection. There are two methods of detection signature based and anomaly based. The signature-based method tries to match computer action to stored signatures of known anomalies or attacks. Anomaly Detection as the another approach detects anomalies by first learning the characteristics of normal activity.

P. Jongsubsuk et al [1] [4] Proposed Intrusion detection system using fuzzy genetic algorithm. The input data is well known KDD99 dataset and network dataset. The fuzzy logic is programmed using four attributes. This algorithm has detection rate of 97.5 %. M. Hoque et al [2] Proposed Intrusion detection using genetic Algorithm to efficiently detect various types of network intrusions. To put the system into practice and measure the performance the KDD99 benchmark dataset is used and obtained detection rate. To measure the fitness of a chromosome the standard deviation equation with distance is used.

Terrence P. Fries [3] introduced a fuzzy-genetic intrusion detection system and was tested using the KDD Cup 1999 Dataset. Feature subset selection and optimization of the fuzzy rule set, both using genetic algorithms, were performed using the 10% training subset of the KDD dataset.

J. Gómez and E. León [5] proposed fuzzy and genetic algorithm to categorize activities of intrusion. The input data is KDDCup99 dataset which consists of 42 features. The fuzzy rule is modified using evolutionary technique and genetic algorithm. The algorithm can categorize the data into five classes including DoS, Probe, R2L, U2R and Normal. This algorithm has detection rate of 98.28 % .

N. Ngamwitthayanon et al [6] proposed a Fuzzy-Adaptive Resonance Theory (ART) for network irregularity detection with feature-reduction dataset. They reduced amount of attribute of KDD99 dataset to 14 attributes. This theory has 98.96% of detection rate. W. Li [7] described a method using GA to identify irregular network intrusion. The method includes both quantitative and definite features of network data for deriving classification rules. though, the addition of quantitative feature can amplify detection rate but no tentative results are offered.

In this paper a fuzzy-genetic approach to detect network intrusion is proposed. To implement and calculate the performance of the system the KDD99 benchmark dataset and network dataset are used. The fuzzy logic is encoded using six attributes. The rest of this paper is presented as follows. In section II, Network datasets are discussed. Section III presents Fuzzy Genetic algorithm. Then Section IV presents result and analysis. Finally in section V conclusion of the experiment is given.

Block diagram and algorithms

GA Rule Generation

Input: Encoded binary string of length n (where n is the number of features being passed), number of generations, population size(, crossover probability (Pc), mutation probability (Pm).

Output: A rule set generation for IDS.

1. Initialize the population randomly with the size of each chromosome to be 41.

2. Initialize N (total number of records in the training set), Pc=0.84 and Pm=0.85.

3. for each chromosome in the new population

4. Calculate fitness= Fx/Sum (Fx)

5. End for

6. Select 50% best fit chromosome and remove worse fit chromosome.

7. Apply Crossover to best selected chromosome.

8. Apply Mutation for each chromosome to generate new population .go to step no3.

9. Stop

GA Parameters

GA has some general elements and parameters which can be defined:

• **GA Operators** The different GA parameter selection mutation and crossover are the most successful parts in the algorithm as they are contribute in the generation of each population.

- **Selection** phase where population individuals with superior fitness are selected, otherwise it gets damaged.

- **Crossover** is a method in each pair of each individuals selects arbitrarily participates in exchanging their parent's genes with each other, until an entire new population has been generated.

- **Mutation** flips some of the bits in an individual, and since all bits could be filled, there is low probability of predicting the change.

• **Fitness Function** The fitness function is defined as a function which scales the value individual relative to the rest of population. It generates the best possible solutions from the amount of candidates located in the population.

In preprocessing phase the KDDCUP99 Dataset is processed by using Weka tool which is used to remove the redundant data from existing Dataset which result in tested Dataset. The removal of redundant data or records from Dataset it improves the detection rate of desired result and improves the performance of our system.

In detection phase the Genetic Algorithm is applied on chosen features data set and locate fitness for every rule with the following fitness function.

Fitness = Fx / sum (Fx)

Where Fx is the fitness of individual x and sum (Fx) is the entire fitness of all individuals.

III. Fuzzy Genetic Algorithm

In this paper we use a fuzzy genetic algorithm described in P. Jongsubsuk [1].

There are two phases in training phase we will improve the rules by using evolutionary concept from genetic algorithm then in testing phase we will use these rules to classify data.

Initial rules ();

while{

for each record {

for each rule{

```
    for each attribute{
        prob = fuzzy();
         totalprob = totalprob + prob;
}
     If (totalprob > threshold)
                  class is attack;
else
                 class is normal;
 }
compare the predicted result with actual result
find A, B, α, and β,
 }
 calculate fitness
create next generation
          preserve_best()
          crossover()
          mutation()
          alien()
 }
```

To compute a fitness value we used following formula

$$fitness = \frac{\alpha}{A} - \frac{\beta}{B}$$

A is total number of attack records.

B is total number of normal records.

α is total number of attack records correctly identified as attack

β is total number of normal records incorrectly classified asattack (false positive).

A. Fuzzy Algorithm

B. In our proposed research work we finally classify the data base on different conditions.

C. Suppose we have two results *X1* and *X2*

D. IF x1 is medium AND x2 is small THEN class is 1

E. IF x1 is medium AND x2 is large THEN class is 2

F. IF x1 is large AND x2 is small THEN class is 2

G. IF x1 is small AND x2 is large THEN class is 3

II. Datasets

In this experiment two input datasets are used as KDD99 dataset and network dataset.

A. KDD99 Dataset

KDD dataset was obtained from the 1998 DARPA Intrusion Detection assessment plan held by MIT Lincoln Labs. The dataset was formed in a replicated in a military network background in which U.S. Air Force LAN which was subjected to replicated attacks [3]. The KDD99 dataset used is the 20% file which contains Normal, Denial of Service (DoS), Probe connection instances. In this paper only six attributes are considered from the dataset shown in Table II.

B. Online Dataset

We obtained online dataset by capturing online network data. We used a packet sniffer to filter the packets on the network. After that, the same procedure used in KDD99 dataset to detect anomalies is followed for online dataset to detect intrusion.

EXPERIMENTAL RESULTS

The implementation was done on java. First, we have created five matrices: the matrix containing the U2R-events, the matrix containing R2L-events, the matrix containing the Probing events, the matrix containing the DOS events and the matrix containing the normal connections. Then, the normalization phase is launched where the various attributes of connections of all matrices are normalized.

We have obtained five normalized matrices U2R, R2L, Probing, Normal and DOS. The next step is the generation of fuzzy rules. To do this, we used the “rand” functio n (random number to generate random numbers that must be among the five values (1, 2, 3, 4. 5) which correspond to (Small, Medium Small, Medium, Medium Large and Large).

We have applied the FGA on the five matrices Rand representing fuzzy rules. All experiments were performed on a laptop CPU Core 2 Duo 2.0 Ghz (x 2) with 3 Go of Ram.

To evaluate the performance of the approach, we used the following measures:

True Positives (TP): is the number of normal connections classified by the genetic approach as normal.

False Positives (FP): is the number of normal connections classified as attacks by the genetic approach.

True Negatives (TN): is the number of attack connections classified as attacks by the genetic approach.

False Negatives (FN): is the number of attack connections classified as normal by the genetic approach.

Specificity: It describes the ability to identify negative results (test the reliability of the given method).

The success rates are as follows:

- **99.99%** for DOS class,
- **98.50%** for Normal class,
- **77.05%** for R2L class,
- **74.20%** for U2R class,
- **89.45%** for Probing class.

Comparison of results with different algorithms

Algorithm	Normal	U2R	R2L	DOS	Probing
	%	%	%	%	%
FGA	**98.50**	**74.20**	**77.05**	**99,99**	**89.45**
FLS	10	95	85	80	80
Hybrid EFS	98.5	76.3	89	98.5	82.5
C4.5	95.9	21.1	30.2	97.1	76.3
5-NN	96.3	25.4	3.8	96.7	87.5
EFRID	92.78	88.13	7.41	98.91	50.35
NB	94.2	25	5.4	79.4	90.4
Naïve Bayesian	97.68	11.84	8.66	96.65	88.33

From above TABLE it can be seen that interesting result are obtained. For all the five classes U2R, R2L, DOS, Probing and Normal, the FGA finds good results compared to the other methods.

CONCLUSION

In this paper, we proposed and implemented a fuzzy genetic algorithm for solving the intrusion detection problem. The results showed the effectiveness of this classification in the field of intrusion detection. We plan to study in future work the parallelization of the proposed approach in the hope to minimize the computation time consuming by the FGA algorithm.

REFERENCES

1. M. Saniee Abadeha, , J. Habibia, C. Lucasb, “Intrusi on detection using a fuzzy genetics-based learning algorithm”, Journal of Network and Computer Applications 30 (2007) 414–428.

2. Saniee Abadeha, J. Habibia, and E. Soroush. “Induct ion of Fuzzy Classification systems via evolutionary ACO-based algorithms”, IJSSST, Vol. 9, No. 3, September 2008.

3. D. Aha, and D. Kibler, “Instance-based learning algorithms”, Machine Learning, vol. 6, pp. 37-66, 1991.

4. N. Ben-Amor, S. Benferhat, and Z. Elouedi. Naive Bayes vs Decision Trees in Intrusion Detection Systems. In Proceedings of the ACM symposium on Applied computing, pages 420–424. ACM Press, 2004.

5. Dalila Boughaci, Samia Bouhali and selma ordeche,” A Fuzzy Local Search for intrusion detection”. ACIT 2011.

6. Dalila Boughaci, Habiba Drias, Ahmed Bendib, Youcef Bouznit, and Belaid Benhamou. “Distributed Intrusion Detection Framework Based on Mobile Agents”. In Proceedings of the International Conference on Dependability of Computer Systems, pages 248–255. IEEE Press, 2006.

7. H. Debar, M. Becker, and D. Siboni. “A neural netwo rk component for an intrusion detection system”. In Proceedings of the IEEE Symposium of Research in Computer Security and Privacy, pages 240-250, May 1992.

8. Kapil Kumar Gupta, Baikunth Nath, Kotagiri Ramamohanarao, “Layered Approach Using Conditional Random Fields for Intrusion Detection”. IEEE Trans. Dependable Sec. Comput. 7(1): 35-49 (2010).

9. M. Gao, M. C. Zhou, “Fuzzy intrusion detection base d on fuzzy reasoning Petri nets,” in Proceeding of the 2003 IEEE International Conference on Systems, Man and Cybernetics, 5-8, pp. 1272-1277, Washington D. C., Oct. 2003.

10. J. Gomez, and D. Dasgupta, "Evolving Fuzzy Classifies for Intrusion Detection," in Proceedings of the 2002 IEEE Information Assurance Workshop. 2002.

11. Ishibuchi H, Murata T. “Techniques and applications of genetic algorithms-based methods for designing compact fuzzy classification systems”. Fuzzy theory systems techniques & applications, V.3, section 40. New York: Academic Press; 1999. p. 1081–109.

12. H.S. Javitz, A. Valdes, T.F. Lunt, A. Tamaru, M. Tyson, and J. Lowrance. “Next generation intrusion detecti on expert system (NIDES)”. Technical Report A016-Rationales, SRI, 1993.

13. G.H. John, and P. Langley, “Estimating Continuous Distributions in Bayesian Classifiers”, in Proceedings of the Eleventh Conference on Uncertainty in Artificial Intelligence. pp. 338-345. Morgan Kaufmann, San Mateo, 1995.

14. S. Kumar and E.H. Spafford. “A pattern-matching model for misuse instrusion detection”. In Proceedings of the national computer security conference, pages 11-21, 1994.

15. W. Lee, S. Stolfo, and K. Mok. “Mining Audit Data t o build Intrusion Detection Models”. In Proceedings of the 4th International Conference on Knowledge Discovery and Data Mining, pages 66–72. AAAI Press, 1998.