VOL. 3, NO.7, July 2012 ISSN 2079-8407
Journal of Emerging Trends in Computing and Information Sciences

http://www.cisjournal.org

Comparison of Selection Methods and Crossover Operations using Steady State Genetic Based Intrusion Detection System

[1] Firas Alabsi,[2] Reyadh Naoum
[1, 2] Middle East University

ABSTRACT

Intrusion Detection Systems are systems built to detect the unwanted attacks. Genetic Algorithm is a method that mimics the process of natural evolution; it was used to support the Intrusion Detection Systems. Genetic Algorithm contains several elements such as population size, evaluation, encoding, crossover, mutation, replacement and stopping criterion. Elements specifications must be determined before using Genetic Algorithm. The performance of Genetic Algorithm depends mainly on these specifications. The aim of this paper is to compare different types of genetic operators and monitor their performance in Intrusion Detection System, to determine the Selection type and Crossover type to be worked together and perform better.

Keywords: *Steady State Genetic Algorithm, Intrusion Detection System, Selection, Crossover.*

1. INTRODUCTION

"Intrusion detection is the act of detecting unwanted traffic on a network or a device. An Intrusion Detection System can be a piece of installed software or a physical appliance that monitors network traffic in order to detect unwanted activity and events such as illegal and malicious traffic" [1]. The strength of the detection system depends mainly on the rules stored within the system. Each rule consists of an action part and a condition part. The condition contains the features values whereas the action determines the attack type as (U2R, R2L, Dos or Probe). In order to determine if the traffic is wanted or unwanted, the system contains a pool of rules that represents the unwanted traffic. The received traffic will be compared with the condition part of each rule stored in the rules pool. If the traffic matches one of the rules which stored in the rules pool, then the traffic is detected as unwanted.

In many cases the rules pool does not have a sufficient number of rules. To solve this problem, Steady State Genetic Algorithm can be used to create new rules and discover the hidden rules which make the rules pool sufficient to detect the unwanted traffic.

"Steady State Genetic Algorithm (SSGA) is used to give a chance for previous rules from previous generation to participate in detecting intrusions in the next generations" [3].

2. STEADY STATE GENETIC ALGORITHM (SSGA) ELEMENTS

2.1 Population

Cleary [4] described the population as a result of a single iteration of genetic algorithm. Iteration can create a new population. Population contains a set of chromosomes; each chromosome is one complete possible solution to the problem to be solved using genetic algorithm.

2.2 Evaluation

For each chromosome there is a fitness function used to evaluate the fitness of each chromosome. Fitness's value reflects the quality of each chromosome.

2.3 Encoding

The gene is a problem parameter; it can be encoded as a binary, integer, or float number.

2.4 Selection

It is the process of selecting the chromosomes to apply Steady State Genetic Algorithm.

TYPES OF SELECTION ARE:

a. Roulette Wheel Selection (RWS):

The chance of a chromosome being selected is proportional to its fitness value. This can be worked as in the following steps:

Step 1: Find the fitness value (fv) for each chromosome in the population using Fitness Function.

Step 2: Calculate sum fitness (Sf) for all the chromosomes in the population:

$$\mathrm{Sf} = \sum_{i=1}^{n} \mathrm{fv}\,i \qquad (1)$$

Step 3: Calculate average fitness (Af) in the Population:

$$\mathbf{Af} = \frac{\mathbf{Sf}}{\mathbf{n}} \qquad (2)$$

Step 4: Find the expected fitness (Ef) for each chromosome in the population:

$$\mathbf{Ef}i = \frac{\mathbf{fv}i}{\mathbf{Af}} \qquad (3)$$

Step 5: Calculate sum expected fitness (Sum Ef) for all the chromosomes in the population:

$$\mathbf{SumEf} = \sum_{i=1}^{n} \mathbf{Ef}i \qquad (4)$$

Step 6: Generate random number (G) in the range [0,SumEf]

$$\mathrm{G} = \mathrm{Rnd}()mod\mathrm{SumEf} \qquad (5)$$

Step 7: Select the chromosome that added its fitness value to the previous chromosomes fitness value's to make (SumEf > = G).

Step 8: Go to step 6, repeat n times, where n is a population size.

b. Elitism Selection:

The idea here is to arrange the chromosomes in the decreasing order according to their fitness values. Then apply the selection with each two chromosomes in the arranged set. In this way, Genetic Algorithm will be applied between strong chromosomes or between weak chromosomes. This means there is no chance to apply Genetic Algorithm between weak and strong chromosomes.

c. Rank Selection:

The rank values can be distributed through the set of chromosomes according to their fitness values, after that, the new fitness values can be calculated using another fitness function. Finally, the roulette wheel can be used to choose the selected chromosomes. This can be worked as in the following steps:

Step 1: Arrange the chromosomes in decreasing order according to its fitness values.

Step 2: Assign a rank value to each chromosome according to its arrangement in the set.

Step 3: Calculate the new fitness value for each chromosome using the following equation [5]:

$$F = max - (max - min) * \frac{rank - 1}{Npop - 1} \qquad (6)$$

Where 1<max<=2 & min = 2-max

d. Stochastic Universal Sampling (SUS):

Instead of spinning the roulette wheel n times as described in Roulette Wheel Selection, in this technique one can spin the Roulette Wheel just once, but after determining n points in the Wheel, where n is a population size. Then choose n chromosomes that situated in front of the determined points.

e. Binary Tournament Selection:

For n times do the following:

- Choose two chromosomes randomly.
- Select the chromosome with the highest fitness value.

2.5 GA operator: Crossover

This process is used to interchange genes between chromosomes to create offsprings. Types of crossover are:

a. Single Point

Step 1: Select the crossover point within a chromosome randomly.

Step 2: Interchange the two parent chromosomes at this point to produce two new offspring's.

b. Two Points

Step 1: Select two points randomly.

Step 2: Interchange the two parent genes between these points.

c. Uniform

According to some probability, crossover will decide the parent contribution in the offspring chromosome. If the mixing ratio is equal to 0.5 this means 50% of genes in the offspring will come from parent 1 and the other will come from parent 2.

2.6 GA operator: Mutation

This process will change the value of randomly selected gene. Types of mutation are:

a. Flip Bit (Used for binary represented genes)

Step 1: Choose one gene randomly.

Step 2: Flip the value of the chosen gene.

b. Boundary (Used for integer and float represented genes)

Step 1: Choose one gene randomly

Step 2: Replace the value of the gene with the upper or the lower value.

c. Uniform (Used for integer and float representation)

Step 1: Choose one gene randomly.

Step 2: Replace the value of a chosen gene with a uniform random value selected between the user specified upper and lower bounds for that gene.

2.7 Replacement

This process will compare between several chromosomes to choose the best. Types of replacement are:

a. Binary Tournament:

It will take two chromosomes and according to their fitness function it will choose the best of them, and ignore the second one.

b. Triple Tournament:

It will replace the worst two chromosomes between three chromosomes by the chromosome with the highest fitness value.

2.8 Stopping Criterion

Starting with an initial population, the evolution process is repeated until the satisfaction of the end condition.

Kumar et al. [2] mentioned common terminating conditions such as:

- The found solution satisfies the minimum criterion.
- A fixed number of generations reached.
- Allocating budget (ex: time, money) reached.
- Successive iterations no longer produce better results.

3. RELATED WORKS

Blickle and Thiele [6] developed a unified and systematic approach to analyze the selection methods based on some basic analysis, and some empirical observations. They regarded proportional selection (such as Roulette Wheel Selection) to be a very unsuited selection scheme.

Picek and Golub [7] presented a large set of Crossover operators used in Genetic Algorithms with binary representation. Their results confirmed the high efficiency of Uniform Crossover and Two Points Crossover.

Sivaraj and Ravichandran [8] described a review of Selection methods and concluded that the strong Selection mechanism reaches equilibrium faster than the weak method. Hence, an appropriate method has to be chosen for the specific problem to increase the optimality of the solution.

Ugtakhbaya and Sodbileg [9] presented a classification Artificial Intelligence (AI) based IDS techniques to help in designing Intrusion Detection System, they proposed Support Vector Machine for IDS to detect Smurf Attack with Accuracy = 99.6%.

4. STATEMENT OF THE PROBLEM

Using Steady State Genetic Algorithm (SSGA) is helpful in discovering the hidden rules which increase the detection rate in Intrusion Detection System (IDS). But (SSGA) has many elements such as Selection and Crossover, each of which has many types. When solving the problem, what are the best types of selection and crossover to be used together, to perform better without consuming time in trying with each type of each element?

5. THE OBJECTIVE OF THE WORK

The problem is to know the suitable types of selection and crossover parameters to be worked together to enhance (SSGA) and perform better within less time when applied in Intrusion Detection System.

6. METHODOLOGY

Using Vb.Net 2008 and SQL server 2008 Intrusion Detection System has been built over a table of U2R which contains 16 records. The contents of the table are presented in the following table which shows the Chromosome ID and the Fitness Value of that Chromosome.

VOL. 3, NO.7, July 2012 ISSN 2079-8407
Journal of Emerging Trends in Computing and Information Sciences

http://www.cisjournal.org

Table 1: ID and fitness value for U2R Chromosomes

Chromosome ID	Fitness Value
1	1.311
2	1.163
3	1
4	1.348
5	3.167
6	1.667
7	3.167
8	3.167
9	3.167
10	3.267
11	3.333
12	3.333
13	3.167
14	3.167
15	3.167
16	2.333

Several Selection processes were applied on the data of Table (1).

The following table presents the arrangement of the chromosomes according to several types of the selection process.

Table 2: The arrangement of ID records due to the Selection type in the first generation

#	RWS	Elitism	Ranking	SUS	Tournament
1	7	5	3	1	7
2	7	7	6	2	5
3	2	8	6	4	7
4	6	6	4	5	5
5	7	4	6	6	7
6	7	1	2	7	5
7	2	2	6	7	8
8	6	3	3	8	6
9	9	11	14	9	14
10	14	12	11	9	13
11	12	10	10	10	12
12	9	9	9	11	14
13	14	13	16	12	9
14	15	14	11	13	12
15	13	15	12	14	12
16	15	16	12	14	11

The first column represents the arrangement of ID's before doing the selection process. The second column represents the ID's arrangement after using RWS method. The third column represents the ID's arrangement after using the Elitism Selection. The fourth column represents the ID's arrangement after using the Ranking Selection. The fifth column represents the ID's arrangement after using the SUS Selection. Finally, the sixth column represents the ID's arrangement after using the Tournament Selection.

(SSGA) applied in Intrusion Detection System to discover the hidden rules. The parameters used in the system are presented in the following table:

Table 3: Genetic Algorithm elements specification (* [10])

Population size	8
Encoding	Real
Evaluation	Reward Penalty Fitness Function *
Selection	RWS, Ranking, Stochastic, Elitism and Tournament
Crossover	Single Point, Two Points and Uniform (mixing ratio = 40%)
Mutation	Flip Bit
Replacement	Binary Tournament Replacement
Stopping Criterion	When Genetic Algorithm Cannot discover additional Rules

The (SSGA) was applied with Roulette Wheel Selection and single point crossover in the first trial. Then it was applied with Roulette Wheel Selection and Two Points Crossover in the second trial. And so on so fourth until applying the Tournament Selection with Uniform Crossover in the 15^{th} trial.

The results are observed for each trial in the generation #10 and generation # 15 to ensure the results.

7. RESULTS AND DISCUSSION

After applying the (SSGA), the following results have been gotten:

Table 4: The choices of different Selection and Crossover types

Selection	Crossover	After 10 Generations		After 15 Generations	
		# of records	Detection Rate	# of records	Detection Rate
RWS	One Point	94	0.46	229	0.53
RWS	Two Point	117	0.53	292	0.53
RWS	Uniform	112	0.4	344	0.4
Elitism	One Point	98	0.4	230	0.4
Elitism	Two Point	125	0.4	349	0.53
Elitism	Uniform	137	0.53	414	0.53
Ranking	One Point	87	0.4	198	0.46
Ranking	Two Point	91	0.46	242	0.53
Ranking	Uniform	110	0.46	346	0.53
SUS	One Point	98	0.4	230	0.40
SUS	Two Point	125	0.4	348	0.46
SUS	Uniform	137	0.53	414	0.53
Tournament	One Point	79	0.46	186	0.46
Tournament	Two Point	110	0.4	283	0.46
Tournament	Uniform	117	0.4	347	0.46

Table (4) shows the choices of Selection and Crossover types and the number of generated records after 10 and 15 generations. Detection Rate columns show the performance of Intrusion Detection System using the discovered rules.

The idea of using (SSGA) is to discover the hidden rules. So (SSGA) will discover all the hidden rules but the process will vary from a choice to another choice in the term of number of generations to discover the rules and the time consuming in discovering rules, Because the time consuming will be as high as the number of generations.

From the results, you can notice the following:

a. For each Selection process, the Two Points Crossover got better results than One Point Crossover.
b. For each Selection process, the Uniform Crossover got better results than Two Points Crossover.
c. Roulette Wheel Selection with One Point Crossover (Choice 1) got better results than Tournament Selection with One Point Crossover (Choice 13). But Roulette Wheel Selection with Uniform Crossover (Choice 3) got worse results than Tournament Selection with Uniform Crossover (Choice 15).
d. Elitism Selection (Choices 4,5 & 6) got the same results as Stochastic Universal Sampling (Choices 10,11 & 12)
e. Elitism Selection with Uniform Crossover (Choice 6) and Stochastic Universal Sampling with Uniform Crossover (Choice 12) both got the best results through different fifteen choices.

8. CONCLUSION

This paper presented the comparative results of Selection with Crossover as (SSGA) elements to be used for Intrusion Detection System. The comparative results show that Uniform Crossover gets the best results wherever it used. And also show that Elitism Selection gets the same results as Stochastic Universal Sampling Selection. To conclude that the best results can be achieved when using Elitism Selection with Uniform Crossover, or using Stochastic Universal Sampling Selection with Uniform Crossover.

9. Future Work

This paper determined the specification of Selection method and Crossover operation to make (SSGA) perform better when used for Intrusion Detection System. But still there are additional works related to this domain:

- Additional work needed to experiment several types of Mutation.
- Additional work needed to experiment several types of Replacement.
- It is better to determine specifications related to all the Steady State Genetic Algorithm elements, which may be used with Intrusion Detection System to perform better.

REFERENCES

[1] Information Assurance Technology Analysis Center (IATAC). (2009). Intrusion Detection System. (Sixth Edition). USA.

[2] Kumar, M., Husian, M., Upreti, N., Gupta, D. (2010). Genetic algorithm: review and application. International Journal of Information Technology and Knowledge Management. Vol (2). No (2). Page 451. Available at: http://www.csjournals.com/IJITKM/PDF%203-1/55.pdf

[3] Al-Sharafat,W. Naoum,R. (2009). Development of genetic-based machine learning for Network Intrusion Detection. World Academy of Science, Engineering and Technology,(55-2009), PP 20-24. Available at: http://www.waset.org/journals/waset/v55/v55-5.pdf

[4] Cleary, B.(2011). “Problems with crossover bias for binary string representations in genetic algorithms”. Master thesis. Californai state university. Long beach. Californa. United States.

[5] Schmidt, M., Stidsen, T. (1997).Hybird systems:genetic algorithms,neural networks and fuzzy logic, DAIMI IR

[6] Blickle, T., Thiele, L. (1995). A Comparison of Selection Schemes used in Genetic Algorithm. TIK-Report, Swiss Federal Institute of Technology, Switzerland. (On-Line), Available: http://www4.hcmut.edu.vn/~hthoang/dktm/Selection.pdf

[7] Picek,S., Golub, M. (2010). Comparison of a Crossover Operator in Binary-Coded Genetic Algorithms. WSEAS Transactions on Computers. Issue (9), Vol.(9), PP (1064-1073).(On-Line), Available:http://www.wseas.us/e-library/transactions/computers/2010/88-211.pdf

[8] Sivaraj,R., Ravichandran, T. (2011). A Review of Selection Methods in Genetic Algorithm. International Journal of Engineering Science and Technology (IJEST). Vol.(3), No. (5), PP (3792-3797). (On-Line), Available:http://www.ijest.info/docs/IJEST11-03-05-190.pdf

[9] Ugtakhbayar, N.Sodbileg, Sh.(2011). "Classification of Artificial Intelligence IDS". Journal of Emerging Trends in Computing and Information Sciences. Vol(2). No (10).PP (529-532).(On-Line), Available at: http://www.cisjournal.org/journalofcomputing/archive/vol2no10/vol2no10_9.pdf

[10] Alabsi,F., Naoum,R.(2012). "Fitness Function for Genetic Algorithm used in Intrusion Detection System". International Journal of Applied Science and Technology. Vol(2). No (4). PP (129-134). (On-line), available at: http://www.ijastnet.com/journals/Vol_2_No_4_April_2012/17.pdf