# Network intrusion detection system by using genetic algorithm

**Hamizan Suhaimi[1], Saiful Izwan Suliman[2], Ismail Musirin[3], Afdallyna Fathiyah Harun[4], Roslina Mohamad[5]**
[1,2,3,5]Faculty of Electrical Engineering, Universiti Teknologi MARA, Malaysia
[4]Faculty of Computer and Mathematical Sciences, Universiti Teknologi MARA, Malaysia

**Article Info**

***Article history:***

Received Apr 1, 2019
Revised Jul 1, 2019
Accepted Jul 12, 2019

***Keywords:***

Genetic algorithm
Intrusion detection
Pattern recognition

## ABSTRACT

Developing a better intrusion detection systems (IDS) has attracted many researchers in the area of computer network for the past decades. In this paper, Genetic Algorithm (GA) is proposed as a tool that capable to identify harmful type of connections in a computer network. Different features of connection data such as duration and types of connection in network were analyzed to generate a set of classification rule. For this project, standard benchmark dataset known as KDD Cup 99 was investigated and utilized to study the effectiveness of the proposed method on this problem domain. The rules comprise of eight variables that were simulated during the training process to detect any malicious connection that can lead to a network intrusion. With good performance in detecting bad connections, this method can be applied in intrusion detection system to identify attack thus improving the security features of a computer network.

***Corresponding Author:***

Saiful Izwan Suliman,
Faculty of Electrical Engineering,
Universiti Teknologi MARA, 40450 Shah Alam, Selangor, Malaysia.
Email: saifulizwan@uitm.edu.my

## 1. INTRODUCTION

In the era of Internet and unlimited access of information, network security becomes one of the most important aspect to look into in order to keep confidential data and information from unauthorized third party access [1, 2]. Network Intrusion Detection System (NIDS) is an important field of research since it deals with many possibilities and aspects in the real-time application especially in terms of network security. It autonomously requires detection of any intrusion and send the gathered information to the authority [3-5]. A network intrusion is known as any action of breaking into the system illegally without the owner's consent.

Many of the applications in computer system nowadays are executed without full intervention or monitoring by in-charged personnel. This with the restricted computational and communication resources of the computer network increase the possibility of intrusions and unauthorized access into the network [6-9]. Computer networks should not rely solely on human action to avoid or overcome this illegal access of its system. Therefore, essential security system is needed to protect the confidential data and information in the networks [10, 11].

In this paper, intrusion detection technique based on metaheuristic approach known as Genetic Algorithm was developed in order to be applied in a computer network. The method will identify and calculate the differences between the behaviour of the unauthorized connection and normal connection using a proposed fitness (objective) function [12, 13]. The proposed technique was executed in two phases; training and testing. The dataset utilized in this study consists of a wide variety of intrusions connections simulated in military network environment and one of the most investigated dataset in this area [14-16].

D. U. S. Rajkumar and R. Vayanaperumal came out with the idea of deploying the Leader Based Intrusion Detection System (LBIDS) into access network in order to detect and prevent DOS such as Sybil and Sinkhole [16]. They used three core security challenges such as authentication, preventing DOS attack and positive incentive provision by implementing the simulation in NS2 software. From the simulation results, the proposed approach proved its ability to fulfil the quality of service in the network. Meanwhile, a group from American University of Kuwait, has designed an efficient intrusion detection framework in WSNs, and recommended a new method that help in detecting and confining intrusive actions in the network. Based on research method proposed by M. Khanafer, et al, they proposed a new beacon-enabled 802.15.4 MAC that is aimed to improve the performance in terms of power conservation, without undermining other important performance parameters [17]. As a result, Markov mathematical model was demonstrated and proved that the approach achieves its goals without affecting the other important performance metrics. The study about two common attacks that often occur in the Wireless Local Area Networks (WLAN) was conducted by J. Afzal, et al [18]. It can detect the attacks by applying the concept of Wireless Intrusion Detection System ((WIDS). In their study, they manage to obtain the detection accuracy of 89% and 93% for the two afore-mentioned attacks. It shows that the efficiency of the proposed attack signatures in WSN.

The same method with a few additional improvements were proposed by M.S. Hoque, M. A. Mukit, and A. N. Bikas [19]. In their experiments, they included another type of connection that can be detected in IDS which are Remote to User Attacks (R2L) and User to Root Attacks (U2R). In 2013, a group of study from University of Mumbai has proposed the same approaches but with a little improvement. P. U. Kadam and P. P Jadhav has proposed an accuracy and effectiveness rule generation for different categories of abnormal connection detection [20]. At least 7 rules were created to identify each data and detect the attacks connection. As enhanced, they used Weka tool to remove the redundant data from KDD'99 Cup in order to improve the detection rate and system performances. Different from previous paper, this group also take another type of attack as main resources which are R21 and R2r. From the results, the detection rate for DoS attacks dominating the highest rate followed by Probe and normal connection with 97.80%, 81.25% and 76.12% respectively. The detection rate for R21 and R2r still low which are 23% and 30.70% respectively. As whole results, the detection rate for some attack connection like DoS attacks remain higher than 90 % detection rate. However, there are some depreciation rate for Probe attacks and normal connection if compared to the previous paper. Non-dominated sorting Genetic Algorithm or NSGA-II is one type of GA that have multiples objective. The idea was proposed by A. Tamimi, D. S. Naidu and S. Kavianpour in 2015 [21]. In this method, they consider features connection and generates the rules by using two different fitness function. The results were optimized by define the different objectives using NSGA-II. As the outcome, they able to fulfil their objective which are to use the effect of one feature on next generations without ignore it and calculate the sum of them to prevent the ignoring of features.

The comparison between GA and Decision Tree (C4.5) Algorithm were proposed by S. Akbar et al in 2012 [22]. C4.5 algorithm was used to create a set of rule that can recognize and classify dissimilar pattern of assault links. In their research, they have create six rules to classify six type of attacks connections. These attacks fall into 4 categories known as DoS, root to local, U2R and probing attacks. The performance of two algorithm was studied by running the test separately to identify the performance between two methods. From the test experiment, it shows the results where the enhanced GA shows detection rate higher than enhanced C4.5. The FPR also biased to enhanced GA where it indicates the smallest value compared to enhanced C4.5 algorithm. From the results, it can be concluded that the performance of GA is better than C4.5 algorithm. An optimized IDS using GA was proposed by S. Kumar and S. Dalal [23]. In their research, they have extend the rule generation set by integrating it with network sniffer to detect Denial of Service (DoS) attacks. With the use of KD'99 cup dataset, they separates the data into two parts; training and testing parts where GA was applied in the first parts. The testing data was also combined with the network sniffer and generated rule set. As outcome, it was capable to, stop the attacks by terminate its connection. In the final end, they were able to reach 97 % detection rate of intrusions estimated by this method.

S. E. Benaicha et al from Algeria has proposed an IDS using GA with an improved selection operator and initial population [24]. The experiment was tested on using Network Security Laboratory Knowledge Discovery and Data Mining (NSL-KDD99) benchmark dataset.The system was implemented using Java language in NetBeans environment and data were stored using MySQL DBMS as database. The results from their experiment indicates that they reach 99.74 % detection rate and 3.74 % False Positive Rate (FPR). It can be concluded that the performance of the detection system is quite high and the FPR is still low.

The study and analysis about improvise the multiclass classification accuracy for IDS was made by S. M. Gaffer, M. E. Yahia and K. Ragad [25]. They introduce Genetic Fuzzy System (GFS) method for IDS where it is the hybrid of fuzzy logic classifier and GA. Fuzzy association rule based classification method was used to gain a compact and accurate classifier with a low cost computational. From the results, they were successfully get detection rate and accuracy more than 90% for DoS and Probe attacks including normal connection while the rest about 73% and above. It is shows that the proposed approaches in their paper are very effective.

D. Narsingyani and O. Kale has proposed a GA in order to optimize false positive in IDS [26]. In their research, they have used the same rule generation of fitness function that has been proposed by S. E. Benaicha et al [9]. Different from them, this approach was used KDD'99 cup dataset to experiment the detection system. The types of attacks that were taken from as main categories attacks in the experimentation are Duration, Protocol, Service, flag, Source byte, Destination byte and Attack-Name. This proposed system was implemented using Java language which is built on third party software package JGAP or GA/GP java toolkit. From the results, they have successfully reduced FPR by increasing the number of rules in training data.

## 2. RESEARCH METHOD

As described earlier, intrusion can be considered as process of attack that can harm the computer network. The intruder can access the system to steal the stored information or gain the knowledge from someone else through their network. Therefore, GA is proposed to overcome this issue. There are several steps involved in GA implementation to detect network intrusion.

Figure 1 shows the step-by-step of GA algorithm implementation for Network Intrusion Detection System in training process. The details method explained below:

```
Step 1: Generate 100 chromosomes randomly.
Step2: Attack recognition between generated chromosomes and
        training data.
Step 3: Fitness function applied to measure fitness value.
Step 4: Data sorted from highest to lowest fitness value.
Step 5: Select top 10 fitness value.
Step 6: Clones 5 times of 10 chromosomes.
Step 7: Crossover between 2 parents of chromosomes.
Step 8: Mutate one of the features in the chromosomes.
Step 9: Calculate fitness value.
Step 10: Data sorted from highest to lowest fitness value.
Step 11: Select top 30 fitness value.
Step 12: Take top 20 from fitness value, top 30 fitness value of
          crossover and 50 chromosomes by randomly generated
Step 13: Repeat 30 times of attack recognition between 100
          population and training data.
Step 14: Final population is obtained for testing process.
```

Figure 1. Training process using GA in IDS

### 2.1. Separate data set into training and testing data set

In KDD Cup 99 data set, there are 41 features that represent the variables used in a computer network [12]. The process of analyzing these all variables is time consuming and requires a large-scale computational steps. Due to this, this research focuses on eight most important features with 6 types of attacks. The dataset contains 284,948 connection data in which 10% of the data was selected as testing data by using pre-set probability value. In the beginning of this study, three different values were investigated for the selection process. The values are 0.2 (20%) 0.3 (30%) and 0.5 (50%). These probability values were used to indicate whether a connection data will be inserted into the pool of testing data. Once the pool is full (which is set to accommodate 10% of the total overall data), this selection process stops. The remaining 90% data were used during the training process with 256,454 of total connections.

Two classes of attacks are the main focus: Denial of service (DOS) and probing attacks. For each field, maximum and minimum number were found out using specific code. The string type of value represented using number and sorted in ascending order. For example, for the 'protocol' features UDP, TCP and ICMP were changed to 10, 11, and 12 respectively. The proposed NIDS begins with training process. As for GA utilized in this study, 100 chromosomes were generated randomly based on the ranges of each fields for the initial population.

### 2.2. Fitness function

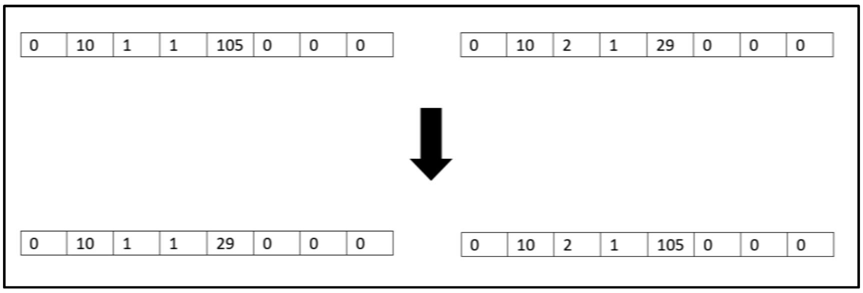
The generated new chromosomes which represent potential solutions were exposed and compared with the training data. This was done to detect any pattern of attack stored in the database. All features were evaluated with the training data set to find the fitness value. The purpose of using fitness function is to select the best fit individuals that would undergo the next stage and create the next generation of chromosomes. In this paper, the proposed fitness function is given by the following formula [1]:

$$F = \frac{a}{A} - \frac{b}{B} \tag{1}$$

Where 'a' represents the number of attacks that detected from comparison of set population and data set while 'A' represents the total number of attacks in dataset, 'b' is the number of normal connection that were detected out of total normal connections, B in the dataset. From that, the fitness value for each chromosome lies in region between -1 and 1. A positive value represents number of attacks correctly classified more than the normal connections. The chromosome is considered of good quality if the fitness value is close to 1. Before the selection, those chromosomes were sorted from the highest to the lowest fitness value. It is essential in order to select the good quality chromosomes. In this study, 10 fittest individuals were selected to undergo next stage which is crossover.

### 2.3. Crossover and Mutation

After 10 fit individuals were selected based on the fitness score, each chromosome was cloned 5 times and this will produce 50 parents of chromosomes in total. An example of this process is in Figure 2.



| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0 | 10 | 1 | 1 | 105 | 0 | 0 | 0 |
| 0 | 10 | 2 | 1 | 29 | 0 | 0 | 0 |
| 0 | 10 | 1 | 1 | 29 | 0 | 0 | 0 |
| 0 | 10 | 2 | 1 | 105 | 0 | 0 | 0 |

Figure 2. Process of crossover

Crossover occurs based on the predetermined crossover rate. This will determine how many features of parents will be inherited by both of the offspring as shown in Figure 2. It is necessary to make sure that the resulting offspring have maintain the range of allowable values in every fields. From the 50 cloned chromosomes, 25 were selected as the 'parent 1' and the remaining 25 as 'parent 2'. After crossover takes place, 50 new chromosomes (offspring) were produced. These chromosomes were then mutated based on probability of mutation to slightly change the gene(s) of the new chromosomes. The mutation process was executed by using single mutation strategy.

### 2.4. Generation of new chromosomes population

After the 50 new chromosomes were mutated, the fitness value for each chromosome was calculated using the same fitness function used before. Then, the mutated chromosomes were sorted based on the highest to lowest fitness value. Thirty chromosomes with the best fitness value were selected as a part of the new population of the next iteration. Besides that, the new population will also comprise of 20 good-quality chromosomes taken from the initial population while the remaining 50 chromosomes were generated randomly. This new population will undergo the same steps and processes iteratively until the pre-set stopping criterion is exhausted. The final population of chromosomes produced will be used during the next step; testing process.

### 2.5. Testing process

Figure 3 above describes the testing process of intrusion detection system using GA. The final population of chromosomes obtained from the training process will be utilized during the testing phase. The process of recognition takes places between the final set of chromosomes and the pre-selected 10%

connection data from the original dataset. The success rate was calculated based on the number of attack connections that can be recognized.

*Step 1: Read testing data one-by-one.*
*Step 2: Present the testing data to each of the chromosomes.*
*Step 3: Compare all features of the testing data with the chromosomes.*
*Step 4: If any of the chromosomes resembles the testing data, then*
*attack is detected. Otherwise, the data is not an attack.*
*Step 5: Calculate the true positive rate of the prediction.*

Figure 3. A testing process of IDS

## 3. RESULTS AND ANALYSIS

Based on the proposed method, the diversity of the raw data and how it is processed influence the final results. In the testing process, interactions were made between the testing dataset and set of chromosomes that were obtained during the training process. In the initial experiments which will divide the dataset into 2 parts (training and testing), investigations were performed to determine the best probability value. Three probability values were tested, which are 0.2, 0.3 and 0.5.

Table 1 shows the results collected by using three different probability values. These values were used to select 10% of data from the raw dataset for testing phase. The fitness value which determines the success rate of detecting intrusions increases from iteration-to-iteration until it stops at a certain value. As shown in the table, the fitness value for the population decreases when the value of selection probability increases.

Table 1. Fitness value for each 0.2, 0.3 and 0.5 probability of random selection

| Iteration | Fitness value | | |
|---|---|---|---|
| | Probability : 0.2 | Probability: 0.3 | Probability: 0.5 |
| 1 | 0.000153139 | 0.000188872 | 0.000204186 |
| 2 | 0.000245023 | 0.000214395 | 0.195597 |
| 3 | 0.000245023 | 0.00392547 | 0.195597 |
| 4 | 0.000245023 | 0.166419 | 0.195597 |
| 5 | 0.000245023 | 0.166419 | 0.195597 |
| 6 | 0.00415008 | 0.166419 | 0.195597 |
| 7 | 0.00415008 | 0.166419 | 0.195597 |
| 8 | 0.00415008 | 0.166419 | 0.409459 |
| 9 | 0.00415008 | 0.166419 | 0.409459 |
| 10 | 0.00415008 | 0.166419 | 0.409459 |
| 11 | 0.00415008 | 0.166419 | 0.409459 |
| 12 | 0.00415008 | 0.166419 | 0.409459 |
| 13 | 0.00415008 | 0.166419 | 0.409459 |
| 14 | 0.00415008 | 0.248025 | 0.409459 |
| 15 | 0.00415008 | 0.248025 | 0.409459 |
| 16 | 0.460567 | 0.248025 | 0.409459 |
| 17 | 0.460567 | 0.453441 | 0.409459 |
| 18 | 0.460567 | 0.453441 | 0.409459 |
| 19 | 0.460567 | 0.453441 | 0.409459 |
| 20 | 0.460567 | 0.453441 | 0.409459 |
| 21 | 0.460567 | 0.453441 | 0.409459 |
| 22 | 0.460567 | 0.453441 | 0.409459 |
| 23 | 0.460567 | 0.453441 | 0.409459 |
| 24 | 0.460567 | 0.453441 | 0.409459 |
| 25 | 0.460567 | 0.453441 | 0.409459 |
| 26 | 0.460567 | 0.453441 | 0.409459 |
| 27 | 0.460567 | 0.453441 | 0.409459 |
| 28 | 0.460567 | 0.453441 | 0.409459 |
| 29 | 0.460567 | 0.453441 | 0.409459 |
| 30 | 0.460567 | 0.453441 | 0.409459 |
| 31 | 0.460567 | 0.453441 | 0.409459 |
| 32 | 0.460567 | 0.453441 | 0.409459 |
| 33 | 0.460567 | 0.453441 | 0.409459 |
| 34 | 0.460567 | 0.453441 | 0.409459 |
| 35 | 0.460567 | 0.453441 | 0.409459 |

Table 2 shows the success detection rate and the average in 30 runs for the three different probabilities selection investigated in this study. The experiment was repeated for thirty times to find the average value. From the results obtained, it indicates that the average of the success rate and the probability value are directly proportionate as illustrated in Figure 4.

Table 2. Success rate of intrusion detection based on each probability value and its average

| Probability of random data selection | Success rate (%) | Average Success rate (30 runs) |
|---|---|---|
| 0.2 | 92.6476 | 93.6754 |
| 0.3 | 98.3119 | 93.764 |
| 0.5 | 99.9825 | 99.8631 |

Success rate is calculated based on the number of attacks that were recognised during the testing process. The probability value used affects the selection of connection data from the raw dataset. As a result, chromosomes that were produced during the training process might be similar with the most data that have been selected for that phase. Therefore, the higher the probability is, the more positive detection it can produce.

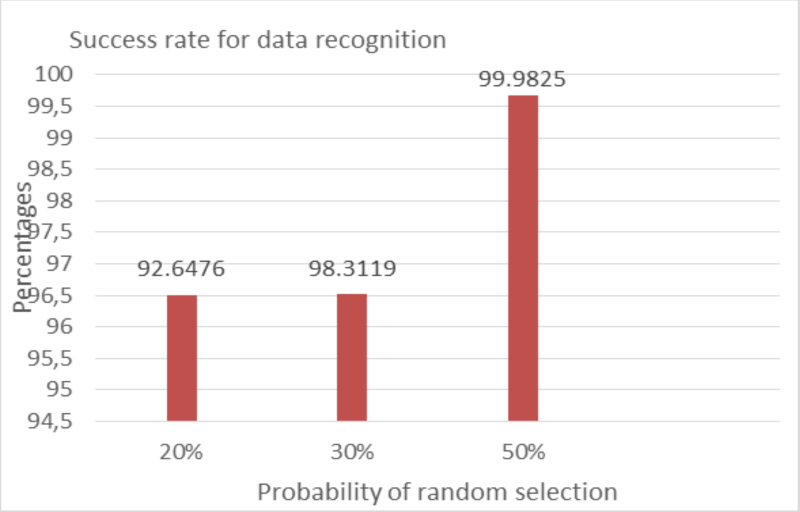


Figure 4. Success rate for three different selection probabilities

## 4. CONCLUSION

In this study, the ideology of evolution in Genetic Algorithm was discussed and utilized to generate the desired solutions for network intrusion detection. Fitness value indicates the quality of a chromosome (candidate solution) that can detect a set of predetermined attack connection data during the training process. The proposed method uses the combination of genetic operators which are cloning, crossover and mutation processes to generate new chromosomes. The genetic processes were conducted in order to produce good quality chromosomes that have high fitness value towards the objective function. These good-quality chromosomes have high possibility/chance to recognize data connection in the network thus lead to intrusion detection. Based on the presented results, the proposed method has the capability to detect any intrusions connection in a network and proven to be a good mechanism to make computer networks more secure.

## ACKNOWLEDGEMENTS

The authors would like to express the gratitude to the Ministry of Education, Malaysia and Universiti Teknologi MARA, Selangor, Malaysia for the financial support given for this project (Geran Bestari) [600-IRMI/MYRA 5/3/BESTARI (026/2017)].

## REFERENCES

[1] I. S. Thaseen and C. A. Kumar, "Intrusion detection model using fusion of chi-square feature selection and multi class SVM," *Journal of King Saud University-Computer and Information Sciences*, vol. 29, pp. 462-472, 2017.

[2] A. Chellam, et al., "Intrusion Detection in Computer Networks using Lazy learning Algorithm," *Procedia computer science*, vol. 132, pp. 928-936, 2018.

[3] J. Jabez and B. Muthukumar, "Intrusion detection system (IDS): anomaly detection using outlier detection approach," *Procedia Computer Science*, vol. 48, pp. 338-346, 2015.

[4] B. Selvakumar and K. Muneeswaran, "Firefly algorithm based feature selection for network intrusion detection," *Computers & Security*, vol. 81, pp. 148-155, 2019.

[5] S. Aljawarneh, et al., "Anomaly-based intrusion detection system through feature selection analysis and building hybrid efficient model," *Journal of Computational Science*, vol. 25, pp. 152-160, 2018.

[6] D. Jianjian, et al., "A Novel Intrusion Detection System based on IABRBFSVM for Wireless Sensor Networks," *Procedia computer science*, vol. 131, pp. 1113-1121, 2018.

[7] A. Shenfield, et al., "Intelligent intrusion detection systems using artificial neural networks," *ICT Express*, vol. 4, pp. 95-99, 2018.

[8] S. Roshan, et al., "Adaptive and online network intrusion detection system using clustering and Extreme Learning Machines," *Journal of the Franklin Institute*, vol. 355, pp. 1752-1779, 2018.

[9] A. Javaid, et al., "A deep learning approach for network intrusion detection system," *Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies (formerly BIONETICS)*, pp. 21-26, 2016.

[10] A. H. Hamamoto, et al., "Network anomaly detection system using genetic algorithm and fuzzy logic," *Expert Systems with Applications*, vol. 92, pp. 390-402, 2018.

[11] S. Mohammadi, et al., "Cyber intrusion detection by combined feature selection algorithm," *Journal of information security and applications*, vol. 44, pp. 80-88, 2019.

[12] N. T. Hanh, et al., "An Efficient Genetic Algorithm for Maximizing Area Coverage in Wireless Sensor Networks," *Information Sciences*, 2019.

[13] D. T. H. Ly, et al., "An improved genetic algorithm for maximizing area coverage in wireless sensor networks," *Proceedings of the Sixth International Symposium on Information and Communication Technology,* ACM, pp. 61-66, 2015.

[14] A. Özgür and H. Erdem, "A review of KDD99 dataset usage in intrusion detection and machine learning between 2010 and 2015," *PeerJ PrePrints*, *4*, p.e1954v1, 2016.

[15] "KDD Cup 1999 Intrusion detection dataset," Available: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html.

[16] U. S. Rajkumar and R. Vayanaperumal, "A leader based intrusion detection system for preventing intruder in heterogeneous wireless sensor network," *2015 IEEE Bombay Section Symposium (IBSS),* pp. 1-6, 2015.

[17] M. Khanafer, et al., "A Review of Intrusion Detection in 802.15.4-Based Wireless Sensor Networks," *2016 IEEE 3rd International Conference on Cyber Security and Cloud Computing (CSCloud)*, Beijing, pp. 95-101, 2016.

[18] J. Afzal, et al., "A Wireless Intrusion Detection System for 802.11 networks," *2016 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*, Chennai, pp. 828-834, 2016.

[19] M. S. Hoque, et al., "An implementation of intrusion detection system using genetic algorithm," arXiv preprint arXiv: 1204.1336, 2012.

[20] P. U. Kadam and P. P. Jadhav, "An effective rule generation for Intrusion Detection System using Genetics Algorithm," vol. 2, 2014.

[21] A. Tamimi, et al., "An Intrusion Detection System Based on NSGA-II Algorithm," *Proceedings - 4th International Conference on Cyber Security, Cyber Warfare, and Digital Forensics, CyberSec*, pp. 58-61, 2015.

[22] S. Akbar, et al., "Improving network security using machine learning techniques," *2012 IEEE International Conference on Computational Intelligence and Computing Research*, pp. 1-5, 2012.

[23] S. Kumar and S. Dalal, "Optimizing Intrusion Detection System using Genetic Algorithm," *International Journal of Research Aspects of Engineering and Management*, vol. 1, pp. 42-45, 2011.

[24] S. E. Benaicha, et al., "Intrusion detection system using genetic algorithm," *2014 Science and Information Conference*, pp. 564-568, 2014.

[25] S. M. Gaffer, et al., "Genetic fuzzy system for intrusion detection: Analysis of improving of multiclass classification accuracy using KDDCup-99 imbalance dataset," *Hybrid Intelligent Systems (HIS), 2012 12th International Conference,* pp. 318-323, 2012.

[26] D. Narsingyani and O. Kale, "Optimizing False Positive In Anomaly based Intrusion Detection using Genetic Algorithm," pp. 72-77, 2007.