

Genetic Algorithm Approach to Intrusion Detection System

¹B. Uppalaiah, ²K. Anand, ³B. Narsimha, ⁴S. Swaraj, ⁵T. Bharat

^{1,3}Holy Mary Institute of Technology & Science (HITS COE), Bogaram, Hyderabad, AP, India

²Anurag Engineering College, Anantagiri, Kodad, India

⁴Padmasri Dr. B. V. Raju Institute of Technology, India

⁵Holy Mary Institute of Technology, Hyderabad, AP, India

Abstract

This rapid growth of computer networks for the past decade, security has become a very important issue for computer systems. The detection of attacks by using IDS against computer networks is becoming a major problem to solve in the area of network security. In this paper we are going to present Genetic Algorithm to identify various harmful/attack type of connections. This algorithm takes into consideration of different features in network connections such as a type of protocol type, duration, service, dst_host_srv_count to generate a classification on rule set. Each rule set identifies a specific type of attacks. For this experiment, we implemented Genetic Algorithm and trained it on the KDDCUP99 dataset to generate a set of rules that can be applicable to the IDS to identify and classify different types of attack connections. In this experiment the characters of an attack such as Smurf, Warezmaster, Saint, Mail bomb, multihop, IP sweep, snmpguess, and buffer-overflow were summarized through the KDD99 data set and the effectiveness and robustness of the approach has been proved. These rules will work with high-quality accuracy for detecting the Denial of Service and Probe type of attacks connections and with appreciable accuracy for identifying the U2R and R2L connections. These findings from this experiment have given promising results towards applying GA for Network Intrusion Detection.

Keywords

IDS, Genetic Algorithm, KDDCUP dataset, rule set.

I. Introduction

The recent growth of local area networks and internet gives a convenient and better technology for the users. Even though the emerging technology is more beneficial for the users of the computer systems the security threads are also increasing at a high rate. Organizations are utilizing different technologies to protect the system from the attacks by using firewall, antivirus software, password protection etc., to overcome the threads. It is highly difficult to provide complete security to the system though we have several protection techniques. In the network accessing and exchanging the information may be easy but providing the security for the information is difficult. Intrusion detection recognizes the unauthorized access to the network, mischievous attacks on the computer systems [1-2]. To recognize the attacks and detect the intrusions the intrusion detection technology is more useful. Intruders can be classified into two types 1. External Intruder 2. Internal Intruder. The unauthorized users who enter the system and make changes to the system and access the resource in the network without authorization, is an external intruder. The intruder in the network without user accounts trying to attack the system is an internal intruder.

Intrusion detection systems are classified into two types 1. Misuse detection 2. Anomaly detection. Intrusion detection with known patterns is called misuse detection. Identifying the abnormalities from the normal network behaviors is called anomaly detection. Hybrid detection systems combine both the misuse and anomaly

detection systems. The categorization IDS's and also be done with respect to the location of intrusion. The activities with a particular host can be monitored by a host based IDS, monitoring the network traffic is done by a network-based IDS. The host activities like system calls, application logs, password files, capability/acl databases can be tested for intrusion detection by a host based IDS. The network traffic and individual packets for mischievous traffic is tested by a network based IDS.

II. KDDCUP99 Dataset

To test and to work with the system classifier KDDCup99 [5] dataset is useful. The LAN representing U.S. Air Force LAN was worked with the dataset provided by MIT Lincoln Labs which contains different types of intrusions present in military networking environment to possess nine weeks of raw TCP/IP data merged with multiple attacks of different types. Every TCP/IP connection with features like duration, protocol type, flag etc., is named as either normal with a specific type of attack such as Smurf, Perl etc., each TCP/IP connection was specifically described by different and 41 contiguous. The list of samples for normal class and attack class concluded in 10% of the data set of DARPA with categorization was present in

Table 1: Attack Types and sample size in 10% KDD Data Set

Category	AttackType(Number of Samples)
Normal	Normal(97277)
DOS	Smurf(280790), Neptune(107201), Back(2203), Teardrop(979), Pod(264), Land(21)
U2R	Buffer_overflow(30), Rootkit(10), Loadmodule(9), perl(3)
R2L	Warezcilent(1020), Guess_passwd(53), Warezmaster(20), Imap(12), ftp_write(8), Multihop(7), Phf(4), Spy(2)
Probe	Satan(1589), Ipsweep(1247), Portsweep(1040), Nmap(231)

Table 2: Number of Attacks in Training KDDCUP99 Dataset

Data Set	Normal	Dos	U2R	R2L	Probe
10%KDD	97277	391458	52	1126	4107
Corrected KDD	60593	229853	70	11347	4106
Whole	972780	3883370	50	1126	41102

Table 1, and number of attacks in training KDDCUP99 dataset in Table 2, Attack Types and The four major types of attacks are:

A. Denial of Service Attacks (DoS)

Where a number of requests were sent by the attacker to the host which he wants to attack.

B. User to Root Attacks (U2R)

Getting the right to access from a host by the attacker to obtained the system root access.

C. Remote to User Attacks (R2L)

In this the attacker tries to access the remote machine through the network and also tries to control the system operations like a local user.

D. Probe

The hacker tries to collect the information and services provided by the machines present in the network to develop the ordinary information.

III. Genetic Algorithm Overview

John Holland [3] in 1970s introduced familiar problem solving algorithms called Genetic Algorithms (GAs) which works on the principles of biological development, natural selection and genetic recombination. Genetic Algorithm works on an individual called chromosome [4] and evolves the group of chromosomes to a population of quality individuals. Each chromosome represents a technique to solve the problem. A fitness function will be there for each rule which is a measure of each rules implementation. The evolution of population starts from an initial population of selected chromosomes which gradually improve the fitness value.

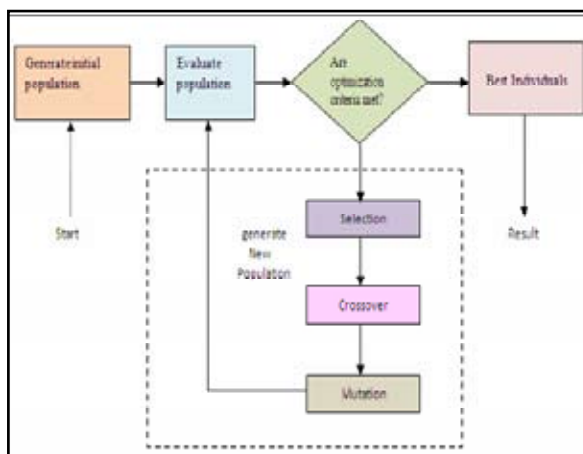


Fig.1: Genetic Algorithm Flow

The three genetic operators selection, crossover, mutation are applied to each individual during the generation process. A group of suitable chromosomes are selected using a fitness function initially eliminating the other individuals. The process continuous by selecting a number of individuals and making pairs each other. The chromosome pair generates one off-string which exchanges their genes around selected cross points. Finally, some individuals are identified and the mutation operations are applied on it. The sub attack labels (smurf, mailbomb, saint, warezmaster etc.,) are recognized with respect to the fitness criteria by selecting the best-fit chromosomes capable of detecting the attacks from every population.

A. Algorithm

1. Rule set Creation using Genetic Algorithm

(i). Input

Set Binary String, Number of generations, Population Size, Crossover Probability, Mutation Probability.

(ii). Output

A set of chosen Features.

Step 1. Initialize the Population arbitrarily

Step 2. Sum of Records in the Training Set

Step 3. For each Chromosome in the New Population

Step 4. Apply uniform Crossover operator to the Chromosome

Step 5. Apply Mutation operator to the chromosome

Step 6. Evaluate Fitness $f(x) = f(x) / f(\text{sum})$

Where, $f(x)$ is the fitness of individual x

and f is the sum of fitness of all individuals in a pop

Step 7. Rank Selection $P_s(i) = r(i) / r_{\text{sum}}$

Where, $P_s(i)$ is probability of selection Individual $r(i)$ is rank of individuals r_{sum} is sum of all fitness values.

Step 8. Choose the top best 60% of Chromosomes Into new population

Step 9. if the number of generations is not reached, go to Step 3.

IV. Related Work

A. Lu [10]

Develop a method to derive a set of classification rules by using Genetic Programming (GP) with help of past data of network. In this method using GP the practical implementation is more difficult due to the system required more data or time.

B. Bridges [8]

Implemented a method to detect both anomalies and network misuses by combining Genetic Algorithm's and Fuzzy data mining technologies. In this method select the most significant network features and locate the best possible parameters of the fuzzy function by using Genetic Algorithm.

C. Xiao [13]

Proposed a methodology to detect abnormal behaviors of network using Genetic Algorithm and information theory. The Genetic Algorithm complexity reduced by use of mutual information. However this methodology applicable only for discrete features.

D. Li [12]

Present to detect network anomalous using Genetic Algorithm. The detection rates may be increased due to quantitative features inclusion. However, no implementation results are available.

E. Crosbie [9]

Proposed a methodology to detect network anomalies using Genetic Programming (GP) and multiple agent technology. When the agents are not properly initialized, the training process takes long time. The communication among small autonomous agents is still a problem.

F. Selvakani [14]

Applied Genetic Algorithm to generate rules for training the IDS. Rules are generated for only Smurf (DoS) attack and Warzmaster (R2L) attack. This performance of this methodology detection rate is low. This survey shows that the proposed Intrusion Detection models for R2L, U2R, Probe attacks get low detection rates using KDDCup dataset. This paper studies two types of attacks for each category i.e., DoS, R2L, U2R and Probe. Observed all the features in the KDDCUP Dataset to detect the attacks.

V. Architecture of the Proposed Model

In the proposed architecture contains two phases. In the first one, the Learning stage, rule set is generated for detecting intruders using network audit data. The second phase, the best rule set with highest fitness value is used for detecting intruders in the Internet world. In the proposed architecture deployed KDDCUP99 Dataset. The Dataset contains 41 features out of those we have selected 3 features to specify each entry of KDDCUP99 Dataset.

The selected features and description are listed in Table 3: Each feature as one gene representing in 1 byte and three collected genes form as a chromosome.

If <condition>

Then <action>

Every rule construction is based on simple “IF-THEN” format as follows

The Condition generally decides the action by comparing current network connection and rules in IDS.

The action field refers what action on to be taken. That action decided by security policies of an organization.

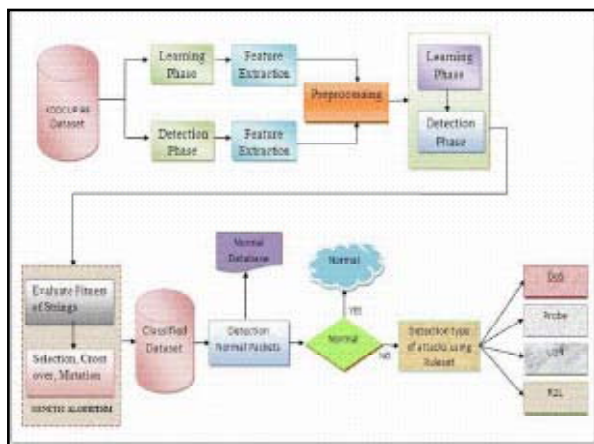


Fig. 2: The simple Architecture of the proposed model

Table 3:

DOS	R2L	U2R	Probe
Rule 1: if duration=0 and protocol_type=icmp and dst_host_srv_count=255 and then smurf	Rule 3: if duration=0 V duration <=289 and protocol_type=tcp and dst_host_srv_count >=1 ^ <=128 and then warezmaster	Rule 5: if duration=0 V duration <=289 and protocol_type=udp and src_bytes=0 and then snmpguess	Rule 7: if duration=0 and protocol_type=icmp and dst_host_srv_count >=1 ^ <=255 and then psweep
Rule 2: if duration=1 V 5 V 11 and protocol_type=tcp and dst_host_srv_count >=2 ^ <=247 and then mailbomb	Rule 4: if duration=0 and protocol_type=icmp V top V udp and dst_host_srv_count >=1 ^ <=20 and then multihop	Rule 6: if land=0 and protocol_type=tcp and dst_host_srv_count <=100 and then buffer-overflow	Rule 8: if duration=0 and duration <=11 and protocol_type=icmp V top V udp and dst_host_srv_count >=1 ^ <=255 and then saint

The following Table 3, shows the sample rules to classify DoS attack (Smurf, Mailbomb), R2L attack (Warezmater, multihop), U2R attack (Snmpguess, Buffer-overflow), Probing attack (ip-sweep, saint)

Table 4: Selected Features

Feature	Description	Number of Genes
Duration	Duration of the connection	1
Protocol_type	Connection protocol	1
dst_host_srv_count	Count of connections having the same destination host and using the same service	1

In preprocessing, the symbolic features are converted into binary form and normalize the data.

In the detection phase we have applied Genetic Algorithm on selected features dataset and find fitness for each rule using the following fitness function.

$$\text{Fitness} = f(x) / f(\text{sum}) \quad (1)$$

Where $f(x)$ is the fitness of entity x and f is the total fitness of all entities

Rank Selection is parallel to relative selection. Individuals in a inhabitants are sorted and ranked based on their fitness value.

$$P_s(i) = r(i) / \text{rsum} \quad (2)$$

Where, $P_s(i)$ is probability of selection individual

$r(i)$ is rank of individuals

rsum is sum of all fitness values.

We collect the classified dataset from the Genetic Algorithm and rules applied to detect the errors.

VI. Experimental Setup

The main aim of our experiment was focused to generate rule set for eight types attacks belongs to four categories. We would like to extend our implementation to all the attack types and features, huge training time complexity of the algorithm, very large data sets.

For our implementation we have used C# in .net suite to develop Genetic Algorithm. We used Windows based computer with Pentium core duo processor 2.0 GHz, 250 GB Hard disk, 1GB of RAM to execute the program.

For our experiment we limited our search space to fields three only. These fields provide information about duration, protocol type, dst_host_srv_count. We want to extend our implementation to more fields for further experiments. The population size for each generation is user selectable.

VII. Results

From the above implementation we have successfully generate some rules those classify the mentioned attack connections. Applying Genetic Algorithm on selected feature set and finds the fitness value for each generation.

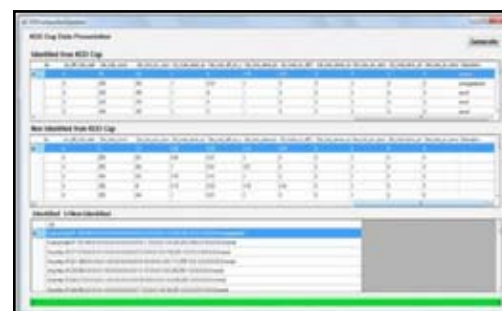


Fig. 3: showing identified and non-identified KDDCUP Data Presentation

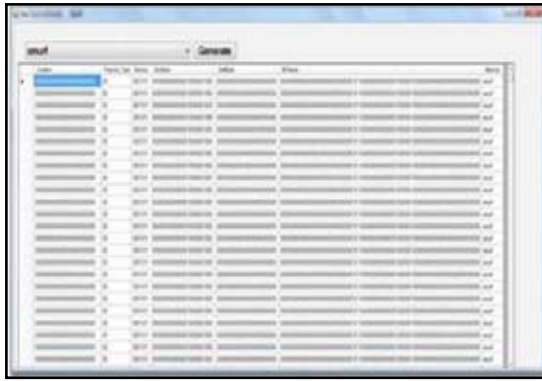


Fig. 4: Showing converted binary string of selected features
KDDCUP Data Presentation

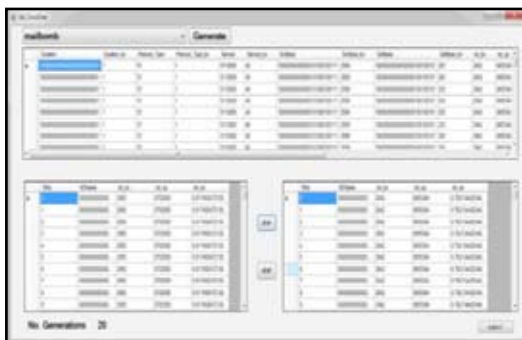


Fig. 5: Showing Selection, crossover, mutation operations

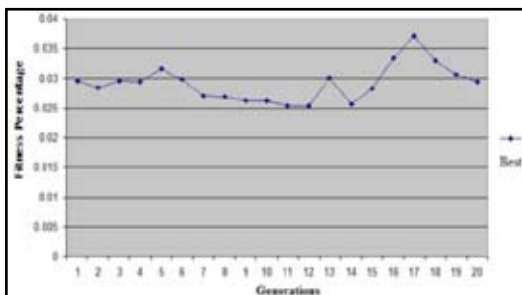


Fig.6 showing fitness of the population graph for smurf attack

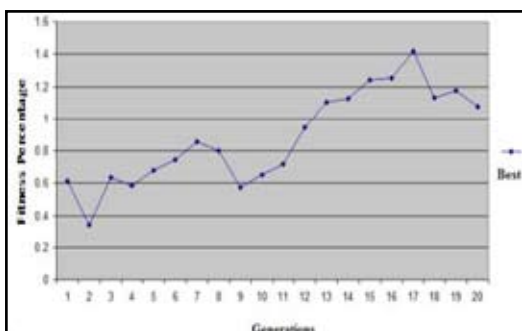


Fig.7 showing fitness of the population graph for bufferoverflow attack

[illegible]

Fig: 8 Attacks Captured By Software

VIII. Conclusion

The paper presents the Genetic Algorithm for the Intrusion detection system for detecting DoS, R2L, U2R, Probe from KDD99CUP data set. The architecture of the system along with implementation of the software for the proposed technique is also discussed. The time to get thorough with three features to describe the data will be reduced with a combination of Genetic Algorithm based IDSs. This provides a high rate of the rule set for detecting different types of attacks. The results of the experiments are good with an 83.65% of average success rate and got satisfied.

Our system is more flexible for usage in different application areas with proper attack taxonomy. As the intrusions are becoming complex and alter rapidly an IDS should be capable to compete with the threat space. Genetic Algorithm detects the intrusion while correlation techniques identify the features of the network connections.

The results shows that we have specified set of rules and high Dos, R2L, U2R, Probe attack detect rate. In may try to improve the results of the whole system and reduce the complexity of the model. Optimizing the parameters present in the algorithm reduces the training time. More reduction techniques may be referred to get valuable features in future.

IX. Acknowledgments

We thank International Journal of Computer Science and Technology (IJCT). Also we thanks to Holy Mary Institute of Technology and Science (HITS COE), Hyderabad & Anurag Engineering College (AEC), Kodad, A.P, India. for finding and supporting this research.

References

- [1] Balasubramanyam JS, Garcia-Fernandez JO, Isaco D, Spatford E, Zamboni D, "An architecture for intrusion detection using autonomous agents", Proceedings of 14th annual computer security applications conference, 1998.
- [2] Heberlein LT, Mukherjee B, Levitt K N, Mansur DL, "Towards Detecting Intrusions in a Networked Environment", Proceedings of 14th department of energy computer security group conference, 1991.
- [3] Holland J, "Adaptation in natural and artificial system", Ann Arbor. The University of Michigan Press, 1975.
- [4] Polhlheim H, "Genetic and Evolutionary Algorithms: Principles, Methods and Algorithms", [Online] Available: <<http://www.geatbx.com/docu/index.html>>, accessed in 2006.
- [5] "KDD cup 1999 data", [Online] Available: <http://www.kdd.ics.uci.edu/databases/kddcup99/kddcup99.html>
- [6] Gong R.H, Zulkemine.M, Anolmaesumi.P, "A Software Implementation of a Genetic Algorithm Based approach to Network Intrusion Detection", Proceedings of the SNPD/SAWN'05, pp.19-27, Aug 2005.
- [7] W. Li, "Using Genetic Algorithm for Network Intrusion Detection", Proceedings of the United States Department of Energy Cyber Security Group, 2004.
- [8] Bridges, Susan, Rayford B. Vaughn, "Intrusion Detection via Fuzzy Data Mining", In Proceedings of 12th Annual Canadian Information Technology Security Symposium, pp. 109-122, Ottawa, Canada, 2000.
- [9] Crosbie, Mark, Gene Spafford, "Applying Genetic Programming to Intrusion Detection", In Proceeding of 1995 AAAI Fall Symposium on Genetic Programming, pp. 1-8, Cambridge, Massachusetts, 1995.

- [10] W. Lu, I. Traore, "Detecting new forms of network intrusion using genetic programming", Computational Intelligence Vol.20, Issue 3, 2004, pp. 475-494.
- [11] Bridges S.M, Vaughn R.B, "Fuzzy Data Mining and Genetic Algorithms Applied to Intrusion Detection", Proceedings of 12th Annual Canadian Information Technology Security Symposium, pp. 109-122, 2000.
- [12] Li W, "A Genetic Algorithm Approach to Network Intrusion Detection", SANS Institute, USA, 2004.
- [13] Xiao. T, QU. G, Hariri. S, Yousif. M, "An efficient Network Detection Method Based on Information Theory and Genetic Algorithm", Proceedings of the 24th IEEE International Performance Computing and Communications Conference, Phoenix, AZ, USA, 2005.
- [14] Selvakani S, R.S. Rajesh, "Genetic Algorithm for framing rules for Intrusion Detection", IJCSNS, Vol.7, No.11, 2007.
- [15] Z Bankovic, D Stepanovic, S Bojanic, O Nieto-Taladriz, "Improving network security using algorithm approach", Computers and Electrical Engineering 33, 2007 pp. 438-451.
- [16] B. Addullah, I. Abd-alghafar, Gouda I. Salama, A. Adbalhafez, "Performance Evaluation of a Genetic Algorithm Based Approach to Network Intrusion Detection System", ASAT-13-CE-14, May, pp. 26-28, 2009.
- [17] Chittur.A, "Model Generation for an Intrusion Detection System using Genetic Algorithms", High school Honors Thesis, [Online] Available: <http://www1.cs.columbia.edu/ids/publications/gaidsthesi01.pdf>, accessed in 2006.



B. Uppalaiah received Bachelor of Technology (Information Technology) from Kakatiya University. Master of Technology (Software Engineering) from Jawaharlal Nehru Technological University (JNTUH). My research interests include Data Mining, Network Security, Information Security and Web Services. Presently working as an Assistant Professor in the department of Computer Science

and Science (HITS COE), India.



K. Anand received Master of Technology (Computer Science Engineering) from Jawaharlal Nehru Technological University (JNTUH). His research interests include Data Mining, Network Security, and Mobile Computing. Presently working as an Assistant Professor in the department of Information Technology at Anurag Engineering College (AEC), Anantagiri (V), Kodad (M), Nalgonda Dist., India.



Banothu Narsimha received Master of Technology (Computer Science Engineering) from Jawaharlal Nehru Technological University (JNTUH). His research interests include Data Mining and Network Security. Presently working as an Assistant Professor in the department of Computer Science at Holy Mary Institute of Technology and Science (HITS COE), India.



Sadhu Swaraj received Master of Technology (Software Engineering) from Jawaharlal Nehru Technological University (JNTUH). His research interests include Data Mining, Network Security and Software Engineering. Presently working as an Assistant Professor in the department of Computer Science at Padmasri Dr. B. V. Raju Institute of Technology, India.

Talari Bharat pursuing Master of Technology (Computer Science Engineering) from Jawaharlal Nehru Technological University (JNTUH). His research interests include Data Mining and Network Security. Presently working as an Assistant Professor in the department of Computer Science at Holy Mary Institute of Technology (HIT COE), India.