International Journal of Application or Innovation in Engineering & Management (IJAIEM)
Web Site: www.ijaiem.org Email: editor@ijaiem.org
Volume 4, Issue 6, June 2015 **ISSN 2319 - 4847**

Network Forensic using DFAI system

Dhwaniket Ramesh Kamble[1], Nilakshi Jain[2], Swati Deshpande[3]

Faculty of Information Technology, Shah and Anchor Kutchhi Engineering College, Mumbai, India.

ABSTRACT

Detection of attacks and prevention of computers from it is a major research topic for researchers throughout the world. In DFAI system, a Genetic Algorithm (GA) based approach is used for generation of rules to detect attacks from the network on the system. A short general idea of the system Digital Forensic tool integrated with Artificial Intelligence (DFAI) is that, genetic algorithm and related detection techniques is provided. The GA will be trained on the Knowledge Discovery Database KDD Cup 99 data set to generate a rule set that can be used to detect attacks on the system. The algorithm takes into consideration different features in network connections of KDD Cup 99 dataset to generate a rule set. Digital Forensics is the science of locating, extracting and analyzing types of data from different devices, which then understand to serve as legal evidence. Digital Forensics focus on finding the digital evidence after a computer security incident has occurred. The system Digital Forensic tool integrated with Artificial Intelligence is used for monitoring and investigating the system by using different Digital Forensic tools.

Keywords: KDD dataset, Genetic Algorithm, Digital Forensic, Artificial Intelligence.

1. INTRODUCTION

Digital Forensics can be defined as the use of scientifically derived and proven methods toward the preservation, validation, identification, analysis, interpretation, documentation and presentation of digital evidence derived from digital sources for the purpose of facilitating or furthering the reconstruction of events found to be criminal, or helping to anticipate unauthorized actions shown to be disruptive to planned operations[1].

The intelligence which is inherited by machines or software is called as Artificial Intelligence. It is an educational grassland which studies the goal to create intelligence. The vital problems or goals of AI follow a line of investigation which include logic, awareness, preparation, learning, ordinary language processing communication, opinion and the ability to move and manipulate objects[2]. Currently popular approaches include statistical methods, computational intelligence and traditional symbolic AI.

Intrusion detection is needed in today's computing environment because it is impossible to keep pace with the current and potential threats and vulnerabilities in our computing systems. The surroundings is continuously sprouting and varying fueled by new technology and the Internet. To make matters worse, threats and vulnerabilities in this environment are also constantly evolving. Attack Detection can help out in managing threats and vulnerabilities in this changing environment. Vulnerabilities are weaknesses in the system. Vulnerabilities can be browbeaten and can be used to negotiate our system. New vulnerabilities are revealed all of the time. Every new technology, product, or system brings with it a new generation of bugs and unintended conflicts or flaws. Also the possible contact from exploiting these vulnerabilities is persistently evolving. In a worst case circumstances, an invasion may cause production downtime, damage of critical information, stealing of confidential information, or other assets, or even negative public relations that may affect an organization.

The Digital Forensic tool integrated with Artificial Intelligence(DFAI) is the tool that can support in shielding a company from invasion by growing the options available to manage the risk from threats and vulnerabilities. The tool could be used to identify an trespasser, and stop the exploit from use by future intruders. DFAI system can become a very powerful tool in an organization's security infrastructure. The system Digital Forensic Tool integrated with Artificial Intelligence is used for network application to detect the threat over the network. The system is used to train and test the packets which specifies the packets identified is a normal packet or it is an attack and then generates the report. The system traces the network activity and detects the attack.

2. REVIEW OF LITERATURE

In 1995, when Crosbie and Spafford [3] applied the several agent technology to detect network anomalies [4]. For these agents they used GA to find, out of the ordinary network behaviours and each agent can keep an eye on one parameter of the network inspection data. The planned style has the benefit when numerous small self-directed agents are used but it has problem when communicating among the agents and also if the agents are not properly initialized the training process can be time consuming.

The dataset has been developed by MIT Lincoln Labs. It contains a broad selection of intrusions computer-generated in a military network atmosphere set up to acquire nine weeks of raw TCP/IP dump data for a local-area network (LAN) simulating a typical U.S. Air Force LAN. The LAN was operated as if it were a true Air Force background, with

various attacks. Hence, this is a high assurance and high superiority data set. [5] They set up an situation to collect TCP/IP dump data from a host located on a simulated military network. Each TCP/IP connection is described by 41 discrete and continuous features (e.g. duration, protocol type, flag, etc.) and labeled as either normal, or as an attack. The KDD cup 99 corrected dataset is 97.6M large and test data unlabeled dataset is 461M large. 65535 records are selected from the each dataset. For this idea, it is decided to use 10% of the training set which contains 494,021 connections. The testing set is the entire set of labeled connections consisting of around 4.9 million connections. Thus, entire data set could be used to test the system on unknown attacks. A connection is a sequence of TCP packets starting and ending at some well defined times, between which data flows to and from a source IP address to a target IP address under some well defined protocol. Each connection is labeled as either normal, or as an attack, with exactly one specific attack type. Every attack on a network can comfortably be placed into one of these groupings.

- **Denial of Service (DoS):** A DoS attack is a type of attack in which the hacker makes a computing or memory resources too busy or too full to serve genuine networking requests and hence denying users access to a machine e.g. apache, smurf, neptune, ping of death, back, mail bomb, UDP storm etc. are all DoS attacks[6][7].
- **Remote to User Attacks (R2L):** A remote to user attack is an attack in which a user sends packets to a machine over the internet, where the user does not have access in order to picture the machines vulnerabilities and utilize privileges which a local user would have on the computer e.g. xlock, guest, ftp_write, xnsnoop, phf, sendmail dictionary etc[6].
- **User to Root Attacks (U2R):** These attacks are exploitations in which the hacker starts off on the system with a normal user account and attempts to misuse vulnerabilities in the system in order to achieve super user privileges e.g. perl, xterm, buffer_overflow [6].
- **Probing:** Probing is an attack in which the hacker scans a machine or a networking device in order to verify weaknesses or vulnerabilities that may later be browbeaten so as to negotiate the system. This technique is commonly used in data mining e.g. satan, ipsweep, portsweep, mscan, nmap etc[6].

Intrusions Detection can be classified into two main categories. They are as follows:

- **Host Based Intrusion Detection:** HIDSs evaluate information found on a single or multiple host systems, including contents of operating systems, system and application files. [8][9]
- **Network Based Intrusion Detection:** NIDSs evaluate information captured from network communications, analyzing the stream of packets which travel across the network. [8][9]

3. EXISTING SYSTEM

There are many existing system tools all of them are less effective than the proposed system. Some of the existing system tools are as follows:

- **Snort:** A free and open source network intrusion detection and prevention system was created by Martin Roesch in 1998 and now developed by Source fire. Through protocol analysis, content searching, and various pre-processors, Snort detects thousands of worms, vulnerability exploit attempts, port scans, and other doubtful behavior.[10][11]
- **OSSEC:** An open source host-based intrusion detection system, performs log analysis, integrity checking, rootkit detection, time-based alerting and active response. In addition to its IDS functionality, it is commonly used as a SEM/SIM solution. Because of its powerful log analysis engine, ISPs, universities and data centre's are running OSSEC HIDS to monitor and analyze their firewalls, IDSs, web servers and authentication logs.[11]
- **Bro**: An open-source, Unix-based network intrusion detection system. Bro detects intrusions by first parsing network traffic to extract its application-level semantics and then executing event-oriented analyzers that compare the activity with patterns deemed troublesome.[11]

3.1 Disadvantages of Existing System

- The intrusion detection system continuously uses additional resources in the system it is monitoring even when there are no intrusions occurring, because the components of the intrusion detection system have to be running all the time. This is the resource usage problem.
- The components of the intrusion detection system are implemented as separate programs, they are susceptible to tampering. An intruder can potentially disable or modify the programs running on a system, rendering the intrusion detection system useless or unreliable. This is the reliability problem.

4. REPORT ON PRESENT INVESTIGATION

To make the DFAI system, GA is chosen. This section gives an overview of Genetic Algorithm (GA), KDD dataset and the system.

4.1 Methodology

The figure below shows the structure of a basic genetic algorithm. First, the GA creates a random population which is then evaluated concerning its level of fitness in a Fitness Function.

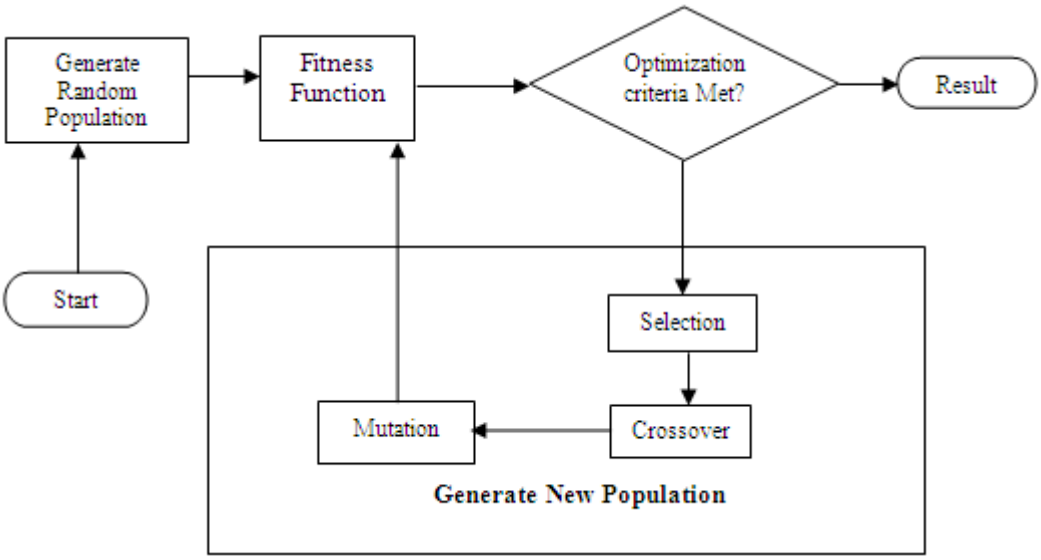

Figure 1 The structure of a basic genetic algorithm

- **Fitness Function**
  A GA Fitness Function typically has the following or similar steps. First, the general outcome is determined based on whether a gene "matches" an existing data set of suspect log record that was obtained from a network device such as a firewall. Then, the function multiplies the "weight" of that field to the degree that the field value "matched" the suspect record field. Typically, the "match" value is either 1 or 0. [12]
- **Selection**
  Once the initial population (of chromosomes) is evaluated, the GA experiments with new generations and iteratively refines the initial outcomes so that those that are most fit are more probable to be ranked higher as results.[12]
- **Crossover**
  The crossover operation creates new chromosomes that share optimistic characteristics of the parent chromosomes while at the same time lowering the negative attributes in a child chromosome.[12]

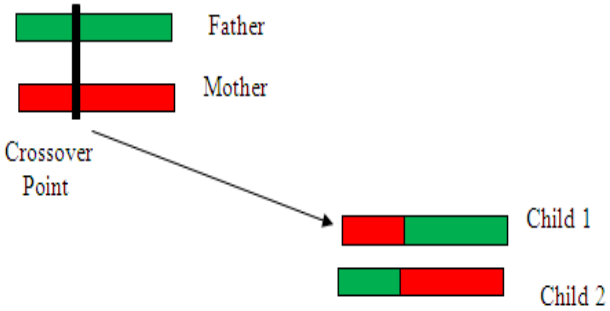

Figure 2 Crossover of chromosomes from the parents to their offspring

- **Mutation**

The final step in the process of generating a new population is mutation. This phase randomly alters a gene's value to create a different one. [12]

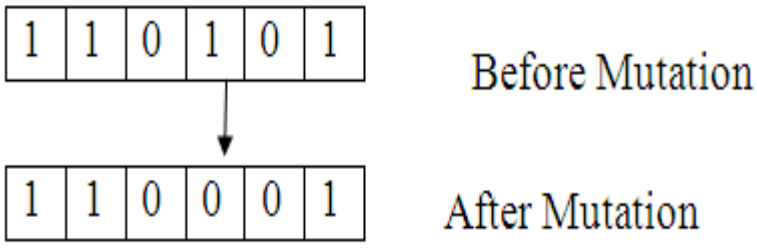

Figure 3 Mutation process for generation a new population.

- **The Rule Set**
  Essentially, the rule set is produced from the output of the GA. For example, the input of Source IP = 1829975662 (which is an IPv4 address of 109.19.54.110)|Destination IP = 1828782356 (which is an IPv4 address of 109.1.1.20) | Destination Port= 8184 | Protocol = 5 | Originator Bytes = 10500 | Responder Bytes = 250000 could produce the following rule:

if

{the connection has following information: source IP 125.19.54.155; destination IP address: 119.1.1.17 ~ 119.1.1.21; destination port number: 8184; the protocol used is FTP; the originator sent more than 10,000 bytes of data; and the responder sent more than 250,000 bytes of data }

then

{log the intrusion and stop the connection} [12]

4.1.1 Advantages

- Traditional methods of optimization use calculus based techniques. These depend on the existence of derivatives, which need to be continuous. Besides many real life problems cannot be readily expressed as mathematical equations.
- Random walks or enumeration type techniques like dynamic programming are not very suitable because they both are extremely inefficient, and have no direction.
- The main feature of GA is its applicability to a wide range of problems. The GA, in its pure form, makes absolutely no use of domain specific knowledge. It works only on the final results; hence the results are always achieved.
- The fact that the GA depends only on payoff data and imposes no preconditions makes it extremely robust and versatile. GA uses random choice as a tool to guide a highly exploitative search through the solution space.

4.2 Flowchart

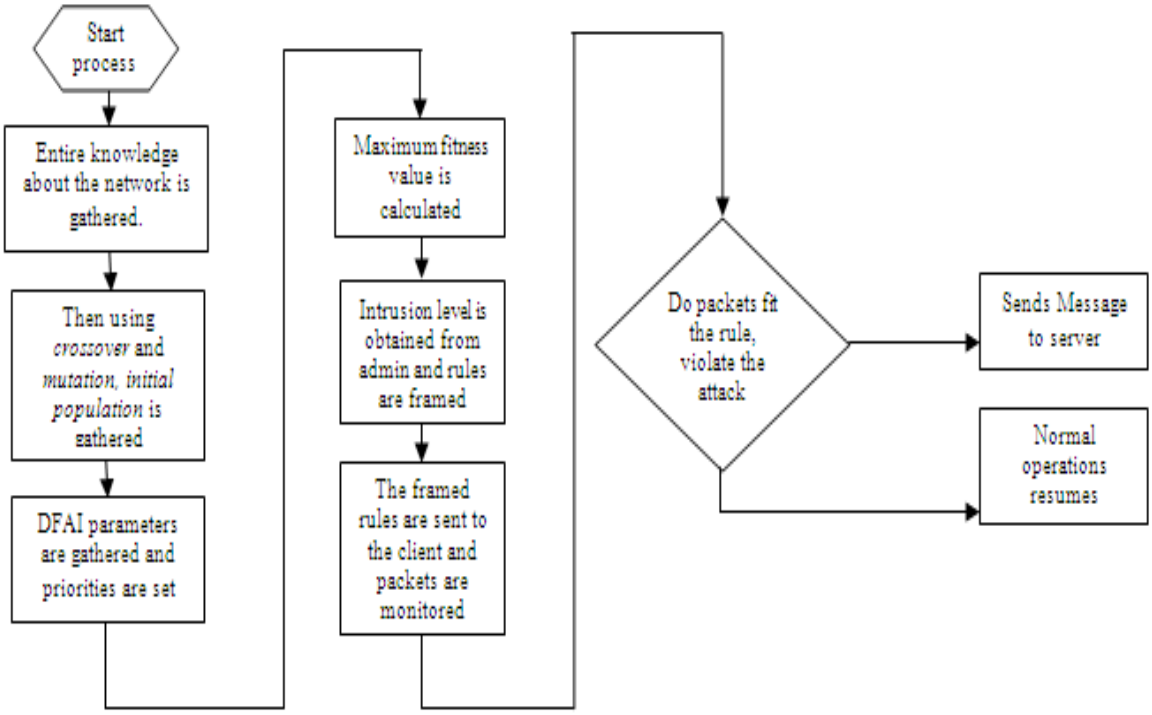

Figure 4 Flowchart of DFAI system

4.3 Implementation Details

To implement the Genetic algorithm and to evaluate the performance of the system, the standard dataset in KDD Cup 1999 “Computer network intrusion detection” competition is used.

4.3.1 KDD Sample Dataset

For the implementation of Genetic algorithm the KDD 99 intrusion detection datasets which are based on the 1998 DARPA initiative is used, which provides designers of intrusion detection systems with a benchmark on which to evaluate different methodologies. A connection is a sequence of TCP packets starting and ending at some well defined times, between which data flows from a source IP address to a target IP address under some well defined protocol. It

results in 41 features for each connection. Normal connections are created to profile that expected in a military network and attacks fall into six categories : ftp_write, ipsweep, smurf, neptune, satan and back.

4.3.2 Implementation Procedure

There are 23 (22+1) groups for each of attack and normal types presented in training data. Number of chromosomes in each group is variable and depends on the number of data and relationship among data in that group. Total number of chromosomes in all groups are tried to keep in reasonable level to optimize time consumption in testing phase. In the testing detection phase, for each test data, an initial population is made using the data and occurring mutation in different features. This population is compared with each chromosomes prepared in training phase. Portion of population, which are more loosely related with all training data than others, are removed. Crossover and mutation occurs in rest of the population which becomes the population of new generation. The process runs until the generation size comes down to 1 (one). The group of the chromosome which is closest relative of only surviving chromosome of test data is returned as the predicted type. Among the extracted features of the datasets, the data taken is only the numerical features, both continuous and discrete.

5.RESULTS AND DISCUSSIONS

From the system Digital Forensic Tool integrated with Artificial Intelligence we get that offline network analysis is done by KDD dataset using Genetic Algorithm. The DFAI system also traces online traffic data and detects intrusion attacks while browsing the network. The system Digital Forensic Tool integrated with Artificial Intelligence trains the KDD dataset, encodes the data by proper feature selection and displays Genetic Rules as shown in Figure 4.3.

Decision Table:

Number of training instances: 248
Number of Rules : 8
Non matches covered by Majority class.
Genetic search.
Start set: 1
Population size: 20
Number of generations: 20
Probability of crossover: 0.6
Probability of mutation: 0.033
Report frequency: 20
Random number seed: 1

Initial population

merit	scaled	subset
4.83871	0	1
67.33871	66.98967	1 2 4
95.16129	96.81088	2 5 6
67.33871	66.98967	1 2 4
92.74194	94.21773	2 3 4 5 6
95.56452	97.24307	1 4 5
87.09677	88.16705	3 6
93.54839	95.08212	1 2 3 4 6
95.56452	97.24307	1 2 4 5
77.41935	77.79446	1 4 6
30.24194	27.22806	2
93.54839	95.08212	3 4 6
4.83871	0	1
95.16129	96.81088	1 5 6
4.83871	0	1
93.54839	95.08212	2 3 4 6
87.09677	88.16705	3 6
95.56452	97.24307	1 2 4 5
74.19355	74.33693	3 4
88.30645	89.46363	1 2 6

```
Generation: 20
merit          scaled         subset
96.77419       96.77419       5
96.77419       96.77419       5
93.54839       93.54839       3 5 6
96.77419       96.77419       5
92.74194       92.74194       3 4 5
92.74194       92.74194       3 4 5
93.95161       93.95161       3 5
96.77419       96.77419       5
 0              0
95.56452       95.56452       4 5
96.77419       96.77419       1 5
92.74194       92.74194       3 4 5 6
96.77419       96.77419       5
96.77419       96.77419       5
96.77419       96.77419       5
96.77419       96.77419       5
93.95161       93.95161       2 3 5
93.95161       93.95161       3 5
96.77419       96.77419       1 5
96.77419       96.77419       5
Evaluation (for feature selection): CV (leave one out)
Feature set: 5,7
 Rules:
 ================================
 src_bytes     class
 ================================
 '(0.5-4.5]'    satan
 '(4.5-54]'     ipsweep
 '(54-130.5]'   ftp_write
 '(1269-28231]' buffer_overflow
 '(-inf-0.5]'   neptune
 '(28231-inf)'  back
 '(822.5-1269]' smurf
 '(130.5-822.5]' normal
 ================================
```

Figure 5 Genetic rules produced on training KDD Dataset.

From the system Digital Forensic Tool integrated with Artificial Intelligence, we get that the KDD dataset is firstly trained and then tested on the rule sets generated and the intruder's attempts are detected using Genetic Algorithm. The Test result of the system Digital Forensic Tool integrated with Artificial Intelligence shows the number of normal packets and the number of attack packets detected as shown in Figure 4.1.

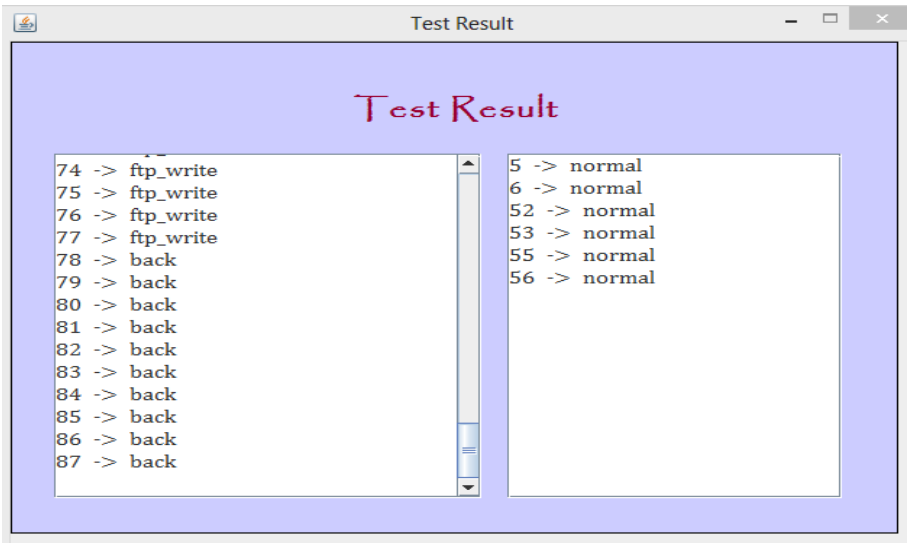

Figure 6 Test Result of KDD Dataset.

Based on the KDD Test Result the efficiency is counted for seven classes i.e ftp_write, ipsweep, normal, smurf, neptune, satan and back and automated graph is generated as shown in Figure 4.2.

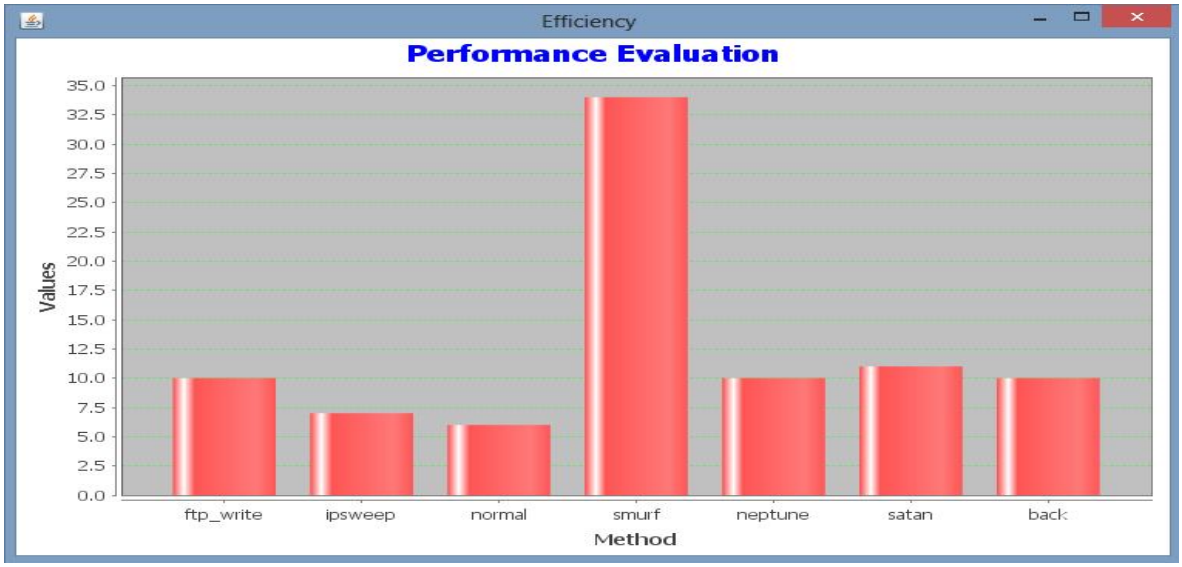

Figure 7 Performance Evaluation of KDD Test Result.

For detecting the intrusion on network, packets transferred in the entire network are evaluated based on the rule sets generated. The network traffic is traced and the traced data is saved with .cap file extension. The traffic data which is saved as .cap file is read and specifies if intrusion has happened or not for SYN Flood attack and TCP ACK storm attack which is displayed in module Attack-1 as shown in Figure 4.4.

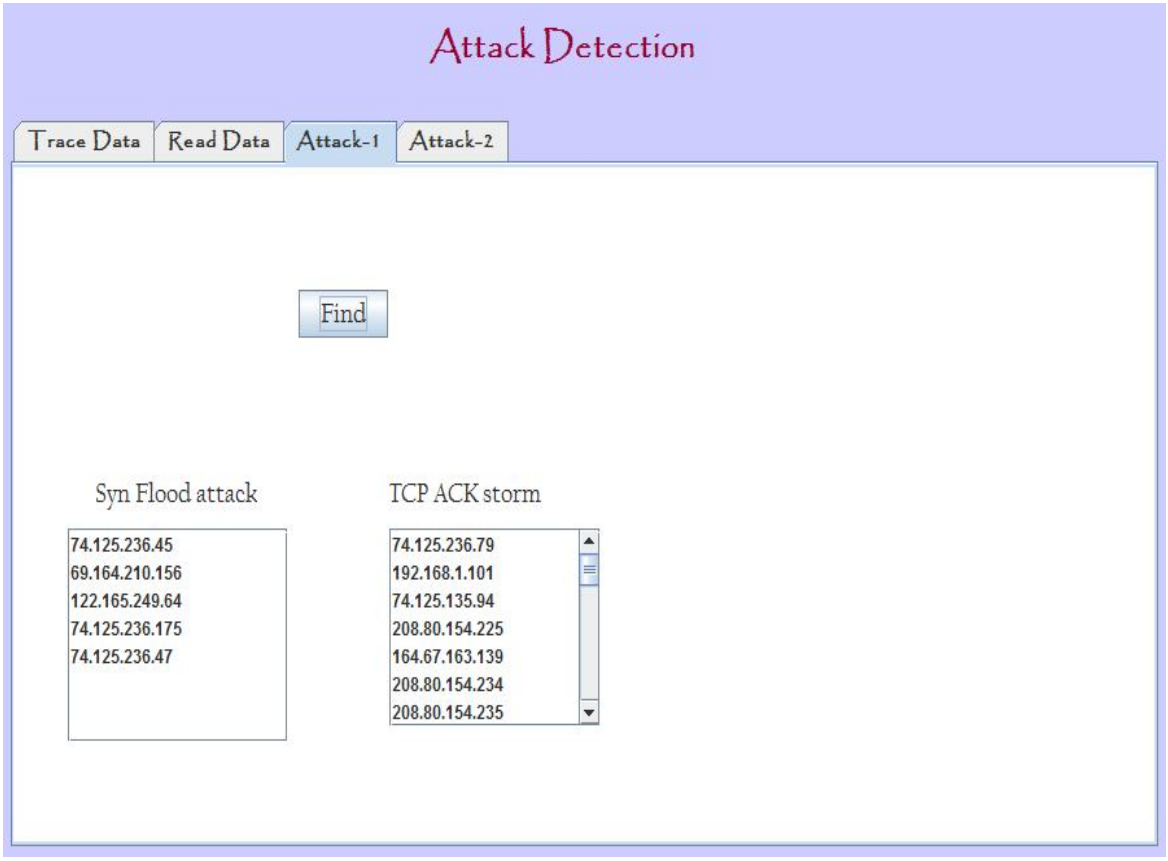

Figure 8 Attack Detection for SYN Flood attack and TCP ACK storm attack.

The saved .cap file is read and specifies if intrusion has happened or not for Fraggle attack and Smurf attack which is displayed in module Attack-2 as shown in Figure 4.5.

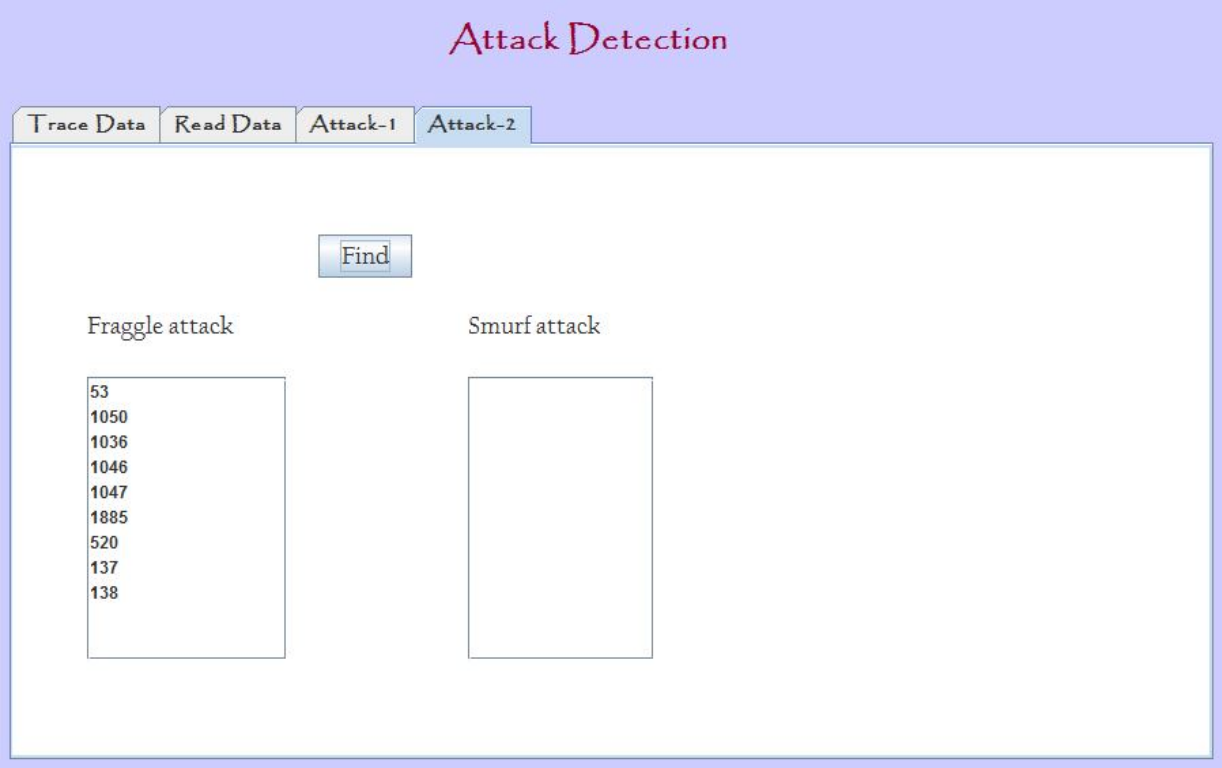

Figure 9 Attack Detection for Fraggle attack and Smurf attack.

Different Digital Forensic tools is used to monitor the system and network. The analysis such as IP locating and finding the IP by the domain name is done for network analysis. The USB Detector digital forensic tool detects and lists all the USB devices that are connected to the systems and reports are generated on the basis of device type, serial number, date created, last plug/unplug date etc. The History Viewer digital forensic tool analyzes the data of Internet Explorer, Google Chrome, Firefox and Windows system. The digital forensic tool monitors the details of URL visited, keyword searched, cookies, download, top visited sites and Input history of the browsers. The tool History Viewer analyzes Windows system which gives the detailed information of systems USB storage, Files and folder accessed and the history of recent documents visited. The reports are generated for both the browser history and Windows system history. The system monitoring digital forensic tool records keystroke, captures the screenshots of the system, shows which application is running on the system, records mouse click events and gives us the details of the websites visited.

6.CONCLUSIONS AND FUTURE WORK

The system Digital Forensic Tool integrated with Artificial Intelligence, aim is to detect an attempt of intrusion to a computer through a network based on looking for particular signatures i.e. looking through a packet's parameters and comparing it to an intrusion signature. To implement and measure the performance of the system, the standard KDD99 benchmark dataset are used and obtained reasonable detection rate. This protects the user's computer and allows user to access the network without the fear of an intrusion attempt. With growing data it is just waste of time and processing to test each packet. So the use of Genetic Algorithm is made to analyze the packet and identify whether the packet is from a previously identified intruder thereby saving valuable time of processing each packet and increasing the speed of the system.

The system Digital Forensic Tool integrated with Artificial Intelligence also throws lights on Digital Forensic Investigation and compiling it to presentable output. The system stores and mines automated results for a security analyst to identify the vulnerabilities and hence improves the system security. Based on different Digital Forensic tools, the digital evidence can be collected from different locations. The system gives a contribution to the Digital Forensic world and speed's up Forensic investigation and brings result faster.

In Future extra tools can be added with extra packet investigation scope in the system. The system will detect an attack and also be able to advise the user and prevent the attack from affecting the computer. The Future work will also include more secure client server architecture can be achieved in the system. Evidences will be purely generated by the system and this application will be made to run on mobile platforms too.

References

[1] Ravneet Kaur, Amandeep Kaur, "Digital Forensics". International Journal of Computer Applications, Volume 50 – No.5, India, 2012.
[2] Dr Faye Mitchell, "Use of Artificial Intelligence in Digital Forensics: An Introduction". Digital Evidence and Electronic Signature Law Review, Volume 7, 2010.
[3] M. Crosbie, E. Spafford, "Applying Genetic Programming to Intrusion Detection", Proceedings of the AAAI Fall Symposium, 1995.
[4] R. H. Gong, M. Zulkernine, P. Abolmaesumi, "A Software Implementation of a Genetic Algorithm Based Approach to Network Intrusion Detection", 2005.
[5] KDD-CUP 1999 Data. [Online]. Available: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html. [Accessed: May. 24, 2015].
[6] Srinivas Mukkamala & Andrew H. Sung. "Identifying Significant Features for Network Forensic Analysis Using Artificial Intelligent Techniques". International Journal of Digital Evidence, Volume 1, Issue 4, New Mexico Tech 2003.
[7] Amrita Anand, Brajesh Patel, "An Overview on Intrusion Detection System and Types of Attacks It Can Detect Considering Different Protocols". International Journal of Advenced Research in Computer Science and Software Engineering 2 (8), pp. 94-98, India, 2012.
[8] Sapna S. Kaushik, Dr. Prof.P.R.Deshmukh, "Detection of Attacks in an Intrusion Detection System". International Journal of Computer Science and Information Technologies, Vol. 2 (3), 982-986, India, 2011.
[9] R. G. Bace, "Intrusion Detection", Macmillan Technical Publishing. 2000.
[10] Kurundkar G.D, Naik N.A, Dr.Khamitkar S.D."Network Intrusion Detection using SNORT". International Journal of Engineering Research and Applications (IJERA), Vol. 2, Issue 2, pp.1288-1296, Nanded, India Mar-Apr 2012.
[11] Surya Bhagavan Ambati, Deepti Vidyarthi. "A brief study and comparison of, open source intrusion detection system tools". International Journal of Advanced Computational Engineering and Networking, Volume-1, Issue-10, Pune, India Dec-2013.
[12] Vivek K. Kshirsagar, Sonali M. Tidke & Swati Vishnu. "Intrusion Detection System using Genetic Algorithm and Data Mining: An Overview". International Journal of Computer Science and Informatics, Vol-1, Iss-4, Aurangabad, India 2012.

AUTHOR

Dhwaniket Kamble is currently working as Assistant Professor in Information Technology Department at Shah and Anchor Kutchhi Engineering College, Mumbai. He has done B.E in Information Technology and currently pursuing M.E in Information Technology.

Nilakshi Jain is currently working as Assistant Professor in Information Technology Department at Shah and Anchor Kutchhi Engineering College, Mumbai. She is having 6 years of teaching experience. She has done M Tech in computer Engineering and currently pursuing Ph.D in Digital forensic field under computer engineering.

Swati Deshpande is currently working as Assistant Professor in Information Technology Department at Shah and Anchor Kutchhi Engineering College, Mumbai. She has completed her M.E. (Electronics).Her teaching experience is almost 17 years. She is also acting as In Charge Head of Department for more than 4 years.