'FUTURIZM-2014, (14TH & 15TH March 2014), 'P.D.E.As College of Engineering, Manjari[Bk]', 'IIP Cell,'Manjari Road, Hadapsar, Pune-412307. India.', 'Chief Editor:Dr.K.R.Harne', Editors: Prof R V Patil, Prof Niraja Jain

International Journal of Innovative Technology & Adaptive Management (IJITAM)
ISSN: 2347-3622, Volume-1, Issue-10 , July, 2014

Intrusion Detection System Using Hybrid Approach

Pallavi Shimpi, Rupali Mangrule
MIT College Aurangabad, Dr.B.A.M.University, Aurangabad, India
Department of Computer Science & Technology, MIT College, Aurangabad, India
(pallavi_shimpi27@rediffmail.com, ramangrule@yahoo.com)

Abstract **- It is our nature to guard our belongings and valuable items, we work too hard to earn what we have and we guard it to protect it from breakage, damage, and from anyone taking it from us. All of our households have locks, some of them might have a security alarm systems, a safe box, and maybe even video surveillance, and this is what we can use to protect our valuables. These types of security will protect us from anyone trying to steal, break-in, or enter unauthorized. As we care for our physical belongings, we also aim a strong concern for our computer, on-line transmissions, private documents, networks and overall privacy. To assist us with monitoring any out of the norm action in our network we will require the Intrusion Detection System.**

Keywords **– Association Rule, Fuzzy Logic, Genetic Algorithm, Intrusion Detection System**

I. INTRODUCTION

With the rapid development of network technology, the network computer system has become the intrusion target of hackers, network system security faces a huge threat, and intrusion detection technology becomes the hot topic in the field of network security [1]. As a result of the various advantages offered by the Internet, businesses have become more open to supporting Internet-powered initiatives such as customer care, e-commerce, and extranet collaboration. However this presents a new challenge. Many enterprise networks have been broken into by hackers. Intrusion Detection is an important component of infrastructure protection mechanisms. Given the increasing complexities of today's network environments, more and more hosts are becoming vulnerable to attacks and hence it is important to look at systematic, efficient and automated approaches for Intrusion Detection.

Intrusion incidents to computer systems are increasing because of the commercialization of the Internet and local networks. Computer systems are turning out to be more and more susceptible to attack, due to its extended network connectivity. The usual objective of the aforesaid attacks is to undermine the conventional security processes on the systems and perform actions in excess of the attacker's permissions. These actions could encompass reading secure or confidential data or just doing vicious destruction to the system or user files. A system security operator can detect possibly malicious behaviors as they take place by setting up intricate tools, which incessantly monitors and informs activities. Intrusion detection systems are turning out to be progressively significant in maintaining adequate network protection. An intrusion detection system (IDS) watches networked devices and searches for anomalous or malicious behaviors in the patterns of activity in the audit stream. Capability of discriminating between standard and anomalous user behaviors should be present in a good intrusion detection system. This would comprise of any event, state, content, or behavior that is regarded as abnormal by a pre-defined criterion.

As an important supplement of the traditional prevention intrusion technology, intrusion detection is another fence to protect network computer systems. So to establish an effective and real-time intrusion detection system is a huge engineering task. After the emergence of a new class of invasion, intrusion detection system needs to real-timely update the invasion match model, because in today world with increasingly high degree of information technology, even in a very short time delay, the new invasion would result in very large hazards. However, if only depends on the knowledge and experience of the system builders to analyze, classify the attack scene and system weak points, extract the characteristics of invasive means to store in the feature database, and then manually write the rules and

International Journal of Innovative Technology & Adaptive Management (IJITAM)
ISSN: 2347-3622, Volume-1, Issue-6 , March, 2014

models matched with the new invasion, if until when the feature the invasion monitoring packets extracted is same to the characteristic of the database it will be judged as invasion, it is very likely that during this period of manual analysis and the preparation of the rules, the new invasion way has resulted in a significantly enough network disaster [2]. In such a system design process, human factors play a decisive role, because the system cannot adapt to the complex network environment, has not enough prevention for the endless new attack means of hackers, and the system self-adaptability and effectiveness of detection is extremely limited. Intrusion detection has emerged as a significant field of research, because it is not theoretically possible to set up a system with no vulnerabilities. One main confrontation in intrusion detection is that we have to find out the concealed attacks from a large quantity of routine communication activities. Several Machine Learning algorithms, for instance Neural Network, Support Vector Machine, Genetic Algorithm, Fuzzy Logic , and Data Mining and more have been extensively employed to detect intrusion activities both known and unknown from large quantity of complex and dynamic datasets. Generating rules is vital for IDSs to differentiate standard behaviors from strange behavior by examining the dataset which is a list of tasks created by the operating system that are registered into a file in historical sorted order. Various researches with data mining as the chief constituent has been carried to find out newly encountered intrusions. The analysis of data to determine relationships and discover concealed patterns of data which otherwise would go unobserved is known as data mining. Many researchers have used data mining to focus into the subject of database intrusion detection in databases.

II. LITERATURE SURVEY

The survey conducted explores the history of research in intrusion detection as performed in software in the context of operating systems for a single computer, a distributed system, or a network of computers. Various international conference papers, text books and Internet are the major source information considered under literature survey.

2.1 Introduction

Intrusion Detection System is one of the most important aspects of Computer/Network Security nowadays, though it has been introduced much earlier. Intrusion is not the only factor associated with breaching the network or computer security but plays a major role to show its existence and dominance in rupturing the overall system security.

Computer security is a branch of computer technology known as information security as applied to computers and networks. The computer security area itself is too vast because of the various different types of network existing in real life with their great contribution. The objective of computer security includes protection of information and property from theft, corruption, or natural disaster, while allowing the information and property to remain accessible and productive to its intended users. The term computer system security means the collective processes and mechanisms by which sensitive and valuable information and services are protected from publication, tampering or collapse by unauthorized activities or untrustworthy individuals and unplanned events respectively.

2.2 Computer Security

The strategies and methodologies of computer security often differ from most other computer technologies because of its somewhat elusive objective of preventing unwanted computer behavior instead of enabling wanted computer behavior [1].

2.2.1 Security by design

The technologies of computer security are based on logic. As security is not necessarily the primary goal of most computer applications, designing a program with security in mind often imposes restrictions on that program's behavior.

There are four approaches to security in computing; sometimes a combination of approaches is valid:

1. Trust all the software to abide by a security policy but the software is not trustworthy (this is computer insecurity).
2. Trust all the software to abide by a security policy and the software is validated as trustworthy (by tedious branch and path analysis for example).
3. Trust no software but enforce a security policy with mechanisms that are not trustworthy (again this is computer insecurity).
4. Trust no software but enforce a security policy with trustworthy hardware mechanisms.

Many systems have unintentionally resulted in the first possibility. Since approach two is expensive and non-deterministic, its use is very limited. Approaches one and three lead to failure. Because approach number four is often based on hardware mechanisms and avoids abstractions and a multiplicity of degrees of freedom, it is more practical. Combinations of approaches two and four are often used in

International Journal of Innovative Technology & Adaptive Management (IJITAM)
ISSN: 2347-3622, Volume-1, Issue-6 , March, 2014

a layered architecture with thin layers of two and thick layers of four.

There are various strategies and techniques used to design security systems. However, there are few, if any, effective strategies to enhance security after design. One technique enforces the principle of least privilege to great extent, where an entity has only the privileges that are needed for its function. That way even if an attacker gains access to one part of the system, fine-grained security ensures that it is just as difficult for them to access the rest.

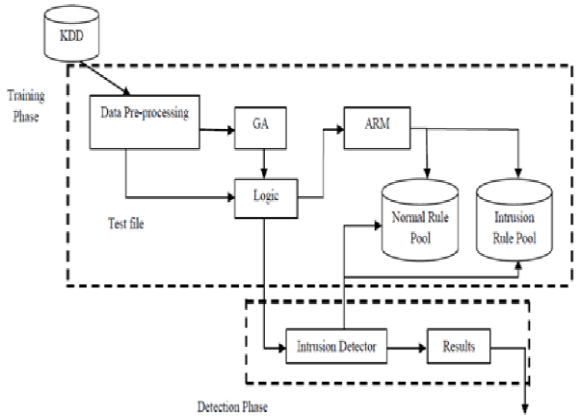
Furthermore, by breaking the system up into smaller components, the complexity of individual components is reduced, opening up the possibility of using techniques such as automated theorem proving to prove the correctness of crucial software subsystems. This enables a closed form solution to security that works well when only a single well-characterized property can be isolated as critical, and that property is also assessable to math. Not surprisingly, it is impractical for generalized correctness, which probably cannot even be defined, much less proven. Where formal correctness proofs are not possible, rigorous use of code review and unit testing represent a best-effort approach to make modules secure.

The design should use "defense in depth", where more than one subsystem needs to be violated to compromise the integrity of the system and the information it holds. Defense in depth works when the breaching of one security measure does not provide a platform to facilitate subverting another. Also, the cascading principle acknowledges that several low hurdles do not make a high hurdle. So cascading several weak mechanisms does not provide the safety of a single stronger mechanism.

III. METHODOLOGY

Hybrid Approach:

This approach is a hybrid approach which is genetic algorithm, fuzzy logic and class-association rule mining algorithm. Due to a hybrid approach, this proposed system works for both misuse and anomaly intrusion detection system.



4.1 PROPOSED SYSTEM OVERVIEW

Association rules:

As one of the most popular data mining methods for wide range of applications, association-rule mining is used to discover association rules or correlations among a set of attributes in a dataset. The relationship between datasets can be represented as association rules. An association rule is expressed by $X \Rightarrow Y$, where X and Y contain a set of attributes. This means that if a tuple satisfies X, it is also likely to satisfy Y. The most popular model for mining association rules from databases is the a priori algorithm [8]. This algorithm measures the importance of association rules with two factors: support and confidence. However, this algorithm may suffer from large computational complexity for rule extraction from a dense database.

Association rules were originally developed as a tool for analysis of retail sales. A piece of sales data usually includes information about a transaction, such as transaction date and items purchased. Association rules can be used to find the correlation among different items in a transaction. For example, when a customer buys item A, item B will also be purchased by the customer with the probability of 90%. Agrawral and Srikant have presented some fast algorithms to mine association rules, including algorithm Apriori. Using the notation of Agrawal and Srikant, let D = {T1, T2…Tn} be the transaction database with n transactions in total and I = {i1, i2… im} be the set of all the items where each if $(1 \leq j \leq m)$ represents one kind of item. Then each transaction Tl $(1 \leq l \leq n)$ in D records the items purchased, i.e., Tl I. Define an itemset as a nonempty subset of I.

International Journal of Innovative Technology & Adaptive Management (IJITAM)
ISSN: 2347-3622, Volume-1, Issue-6 , March, 2014

An association rule will have the form: X ⇒ Y, c, s, where X I, Y I, X and Y are disjoint itemsets. Here s represents the support of this association rule and c represents the confidence of this association rule. Assume the number of transactions that contains both the itemset X and the itemset Y is n'; support(X∪Y)=n'/n) and c=support(X∪Y)/support(X).Intuitively, support(X) can be viewed as the occurrence frequency of the itemset X in the whole transaction database D, while c indicates that when X is satisfied, there will be the certainty of c that Y is also true .Two thresholds, minconfidence (representing minimum confidence) and minsupport (representing minimum support), are used by the mining algorithm to find all association rules X ⇒ Y, c, s, such that c ≥ minconfidence and s ≥ minsupport.

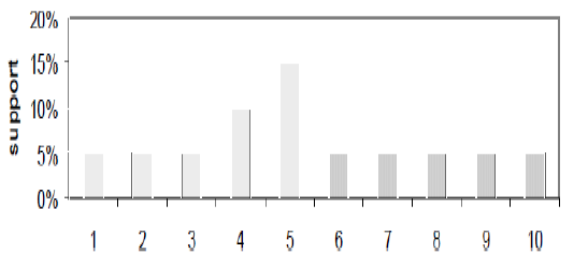
Integration of Fuzzy Logic with data mining

Although association rules and frequency episodes can be mined from audit data for intrusion detection, the mined rules or episodes are at the data level. Integrating fuzzy logic with association rules allows one to extract more abstract patterns at a higher level.

Mining Fuzzy Association Rules

Srikant and Agrawal have described a very popular algorithm for mining quantitative association rules that partitions quantitative attributes into different intervals.

Unfortunately, a sharp boundary problem‖ results from using interval partitions. For example, suppose [1, 5] and [6, 10] are two intervals created on a quantitative attribute as shown in Figure 2.5. If the minimum support threshold is set at 30%, the interval [6, 10] will not gain enough support regardless of the large support near its left boundary, as shown in Figure 2.5. That is to say, although the value 5 has a large support and lies near the interval [6, 10], it will not make any contribution when counting the support of [6, 10].



In intrusion detection, the sharp separation of intervals may raise additional problems. For example, suppose the interval [1, 5] is mined as a normal pattern for the quantitative attribute. The values 6 and 10 will both be considered abnormal regardless of the difference in their deviations from the normal pattern. Likewise, a normal behavior that varies slightly from normal may fall outside the interval representing a normal pattern and be considered an anomaly. Similarly, an intrusion with a small variance may fall inside the interval and be undetected.

To address the sharp boundary problem‖ , Kuok, Fu, and Wong have proposed to mine fuzzy association rules by using fuzzy sets to categorize a quantitative attribute. In the above example, the two intervals will be replaced by two fuzzy sets. Suppose the value 5 has membership degree of 0.9 in the first set and 0.3 in the second set. Then it will contribute 0.9 to the support of the first fuzzy set and 0.3 to the second one. However, this means that the value 5 will be more important than other values since the sum of its contributions to different fuzzy sets has become greater than 1.

Overview of the rule mining based on GNP

A class-association-rule mining algorithm based on GNP has been proposed [19][20]. In this section, the outline of GNP and its class association- rule mining is briefly reviewed.

GNP is one of the evolutionary optimization techniques, which uses directed graph structures instead of strings and trees. The phenotype and genotype expressions of GNP are shown in Fig 2.6. GNP is composed of three types of nodes: start node, judgment node, and processing node. Judgment nodes, *J*1, *J*2 , . . . , *Jm* (*m* is the total number of judgment functions), serve as decision functions that return judgment results so as to determine the next node. Processing nodes, *P*1 *P*2….*Pn* (*n* is the total number of processing functions), serve as action/processing functions. The practical roles of these nodes are predefined and stored in the function library by supervisors. Once GNP is booted up, the execution starts from the start node, then the next node to be executed is determined according to the connection between nodes and a judgment result of the current activated node. Fig. 2.6 also describes the gene of a node in a GNP individual. NT*i* represents the node type such as 0 for start node, 1 for judgment node and 2 for processing node. ID*i* serves as an identification number of a judgment or processing node, for example, NT*i* = 1 and ID*i*

International Journal of Innovative Technology & Adaptive Management (IJITAM)
ISSN: 2347-3622, Volume-1, Issue-6 , March, 2014

= 2 represents node function *J*2. *Ci*1, *Ci2*…. denote the node numbers connected from node *i*. The total number of nodes in an individual remains the same during every generation. Three kinds of genetic operators, i.e., selection, mutation, and crossover, are implemented in GNP.

1) *Selection*: Individuals are selected according to their fitness.

2) *Crossover*: Two new offspring are generated from two parents by exchanging the genetic information. The selected nodes and their connections are swapped each other by crossover rate *Pc* .

3) *Mutation*: One new individual is generated from one original individual by the following operators. Each node branch is selected with the probability *Pm*1 and reconnected to another node. Each node function is selected with the probability *Pm*2 and changed to another one.

IV. CONCLUSION

Data mining methods are capable of extracting patterns automatically and adaptively from a large amount of data. Various methods related to intrusion detection system are studied and compared. Crisp data mining methods such as ADAM method, Random Forest algorithm are used for intrusion detection but suffer from sharp boundary problem which gives less accurate results. In proposed method, use of fuzzy logic overcomes the sharp boundary problem. Class-Association rules have been used to mine training data to established normal patterns for anomaly detection. An actual intrusion with a small deviation may match the normal patterns and thus not be detected. Therefore, integration of fuzzy logic with class-association rules and GA generates more abstract and flexible patterns for anomaly detection.

In this paper, we have proposed a GA-based fuzzy Class Association Rule Mining with Sub-Attribute Utilization and its application to classification, which can deal with discrete and continuous attributes at the same time. In addition, this method was applied to both misuse detection and anomaly detection.

ACKNOWLEDGMENT

I have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. I would like to extend my sincere thanks to all of them.

I am highly indebted to Ms.Rupali Mangrule for her guidance and constant supervision as well as for providing necessary information regarding the project & also for their support in completing the project.

I would like to express my gratitude towards my parents & member of MIT College for their kind co-operation and encouragement which help me in completion of this project.

REFERENCES

[1]Mabu S., Chen C., Shimada K., "An Intrusion-Detection Model Based on Fuzzy Class-Association-Rule Mining Using Genetic Network Programming," IEEE Transactions Systems, Man, Cybernetics C, Application and Reviews, volume 41, number 1, pp. 130–139, January 2011.

[2]Hoque M., Mukit M. and Bikas M., "An Implementation of Intrusion Detection System using Genetic Algorithm," International Journal of Network Security & Its Applications (IJNSA), Vol.4, No.2, March 2012.

[3]Lu W. and Traore I., "Detecting new forms of network intrusion using genetic programming," Computer Intelligence, volume 20, no. 3, pp. 474–494, 2004.

[4] Kaliyamurthie K., Parameswari D., Suresh R., "Intrusion Detection System using Memtic Algorithm Supporting with Genetic and Decision Tree Algorithms," IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 2, No 3, March 2012.

[5] Scarfone, Karen; Mell, Peter (February 2007). "Guide to Intrusion Detection and Prevention Systems (IDPS)". Computer Security Resource Center (National Institute of Standards and Technology)(800–94).http://csrc.ncsl.nist.gov/publications/nistpubs/800-94/SP800-94.pdf. Retrieved 1 January 2010.

[6] Sathya s., Ramani R., Sivaselvi K., "Discriminant Analysis based Feature Selection in KDD Intrusion Dataset," International Journal of Computer Applications (0975 – 8887), Volume 31– No.11, October 2011.

[7] Kddcup 1999data [Online]. Available: kdd.ics.uci.edu/ databases/kddcup99/kddcup99 .html.

[8] Han J., Kamber M., "Data Mining," Morgan Kaufmann Publishers, 2001.

[9] Shetty M. and Shekokar N., "Data Mining Techniques for Real Time Intrusion Detection Systems," International Journal of Scientific & Engineering Research Volume 3, Issue 4, April 2012.

[10] Gong R., Zulkernine M., Abolmaesumi P., "A Software Implementation of a Genetic Algorithm Based Approach to Network Intrusion Detection," Proceedings of the Sixth International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Dis tributed Computing and First ACIS International Workshop on Self-Assembling Wireless Networks, IEEE, 2005