BIOINFO Security Informatics
ISSN: 2249-9423 & E-ISSN: 2249-9431, Volume 2, Issue 1, 2012, pp.-23-27.
Available online at http://www.bioinfo.in/contents.php?id=344

NETWORK INTRUSION DETECTION SYSTEM & FUZZY LOGIC

THAKARE S.P.[1] AND ALI M.S.[2]

[1]Department of Information Technology, P.R.M.I.T. & R, Badnera, MS, India.
[2]PRM College of Engineering & Management, Badnera, MS, India.
*Corresponding Author: Email- rohit7277@yahoo.com, softalis@hotmail.com

Received: February 21, 2012; Accepted: March 15, 2012

Abstract- Network intrusion detection systems have become a standard component in security infrastructures. The number of hacking and intrusion incidents are increasing alarmingly each year as new technology rolls out. Unfortunately in today's digitally connected world, there is no place to hide. Unfortunately, current systems are poor at detecting novel attacks without an unacceptable level of false alarms. We propose that the solution to this problem is the application of an ensemble of anomaly detection technique which can be applied to network connection data in an offline environment. This paper gives an overview of our research in development of Intrusion detection system (IDS) by applying fuzzy logic .

Keywords- Intrusion detection system, Anomaly Detection, Misuse detection, Fuzzy Logic.

Citation: Thakare S.P. and Ali M.S. (2012) Network Intrusion Detection System & Fuzzy Logic. BIOINFO Security Informatics, ISSN: 2249-9423 & E-ISSN: 2249-9431, Volume 2, Issue 1, pp.-23-27.

Introduction

This paper discusses the fuzzy based network intrusion detection system. Intrusion detection system is increasingly a key part of system defence is used to identify abnormal activities in a computer system. In general, the traditional intrusion detection relies on the extensive knowledge of security experts, in particular, on their familiarity with the computer system to be protected. To reduce this dependence, various data-mining and machine learning techniques have been used in the literature. 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 [1]. An IDS is an automated system that can detect a computer system intrusion either by using the audit trail provided by an operating system or by using the network monitoring tools. The main goal of intrusion detection is to detect unauthorized use, misuse and abuse of computers by both system insiders and external intruders. A good intrusion detection system should be able to distinguish between normal and abnormal user activities. To classify user behaviour whether it is security intrusive or not is not simple because behaviour pattern is unpredictable and unclear. IDS can be categorized based on its monitoring scope and detection techniques. Host-based IDS can also be referred to as stand-alone intrusion detection systems because their monitoring scope is restricted to only a single host in the form of a single process or a single system. With this limitation, it fails to detect intrusions attempted across the network. Meanwhile, network-based IDS's monitor any number of hosts on a network by scrutinizing the audit trails of multiple hosts. Since attempted intrusions can happen via the network, network-based IDS needs to monitor multiple events generated on several hosts to integrate sufficient evidence. Thus, the use of the network traffic information for security auditing is more effective. Host-based and Network-based IDSs mainly employ two detection techniques; anomaly detection and misuse detection.

a. Misuse detection

Misuse detection attempts to model abnormal behaviour based on signatures of the known attacks and known system vulnerabilities.

b. Anomaly detection

Normal behaviour patterns are useful in predicting both user and system behaviour. Here, anomaly detectors construct profiles that represent normal usage and then use current behavioural pattern to detect a possible mismatch between profiles and recognize possible attack attempts [2].Current intrusion detection techniques mainly focus on discovering abnormal system events in computer networks and distributed communication systems. Due to the uncertainty nature of intrusions, fuzzy sets play an important role in recognizing dangerous events and reducing false alarms level [3].

This document is a template. An electronic copy can be downloaded from the conference website. For questions on paper guidelines, please contact the conference publications committee as indicated on the conference website. Information about final paper submission is available from the conference website.

Motivation

Advantages of anomaly detection method are

Possibility of detection of novel attacks as intrusions. Anomalies are recognized without getting inside their causes and characteristics. ability to detect abuse of user privileges. The biggest disadvantage of this method is a substantial false alarm rate.

Misuse (Signature) detection method has the following advantages

Very low false alarm rate, simple algorithms, easy creation of attack signature databases, easy implementation and typically minimal system resource usage. Some disadvantages of this type of detection are; difficulties in updating information on new types and they are inherently unable to detect unknown, novel attacks where a continuous update of the attack signature database for correlation is a must. In general, there is a tradeoffs between the ability to detect new attacks and the ability to generate a low rate of false alarms in the developing an effective IDS.

For specified, well-known intrusion excellent detection results are achieved by signature-based methods. But, they cannot find out unfamiliar intrusions though constructed as a least alteration of previously known attacks. Conversely, the capability of discovering intrusion events which are previously unobserved is the main advantage of anomaly based detection techniques [1] [2].

Objectives

The proposed system aims to fulfil the following objectives:

- To develop system which will have broad attack detection coverage and will not specific in detecting only the previously known attacks.
- To reduce the number of false alarms generated, thereby improving attack detection accuracy.
- To develop anomaly intrusion detection system will be operate efficiently in high speed networks.

Issues such as scalability, availability of training data, robustness are also implicitly addressed.

Literature Review

Two most significant motives to launch attacks are, either to force a network to stop some service(s) that it is providing or to steal some information stored in a network. An intrusion detection system must be able to detect such anomalous activities. However, what is normal and what is anomalous is not defined, i.e., an event may be considered normal with respect to some criteria, but the same may be labeled anomalous when this criterion is changed. Hence, the objective is to find anomalous test patterns which are similar to the anomalous patterns which occurred during training. The underlying assumption is that the evaluating criterion is unchanged and the system is properly trained such that it can reliably separate normal and anomalous events [4].

Depending on the type of analysis carried out, intrusion detection systems are classified as either signature-based or anomaly-based. Signature-based schemes (also denoted as misuse-based) seek defined patterns, or signatures, within the analyzed data. For this purpose, a signature database corresponding to known attacks is specified a priori. On the other hand, anomaly-based detectors attempt to estimate the ‘‘normal’’ behavior of the system to be protected and generate an anomaly alarm whenever the deviation between a given observation at an instant and the normal behavior exceeds a predefined threshold [5].

Anomaly-based intrusion detection triggers an alarm on the IDS when some type of unusual behavior occurs on network. This would include any event, state, content, or behavior that is considered to be abnormal by a pre-defined standard. Anything that deviates from this baseline of “normal” behavior will be flagged and logged as anomalous. “Normal” behavior can be programmed into the system based on offline learning and research or the system can learn the “normal” behavior online while processing the network traffic.

Some examples of anomalous behavior include

- HTTP traffic on a non-standard port, say port 53 (protocol anomaly)
- Backdoor service on well-known standard port, e.g., peer-to-peer file sharing using Gnutella on port 80 (protocol anomaly and statistical anomaly)
- A segment of binary code in a user password (application anomaly)
- Too much UDP compared to TCP traffic (statistical anomaly)
- A greater number of bytes coming from an HTTP browser than are going to it (application and statistical anomaly) [6].

Fuzzy systems have demonstrated their ability to solve different kinds of problems in various applications domains. Fuzzy systems based on fuzzy if-rules have been successfully used in many applications areas. Fuzzy if-then rules were traditionally gained from human experts. Recently, various methods have been suggested for automatically generating and adjusting fuzzy if-then rules without using the aid of human experts. Genetic algorithms have been used as rule generation and optimization tools in the design of fuzzy rule-based systems [7].

There are two main reasons to introduce fuzzy logic for intrusion detection. First, many quantitative features, both ordinal and categorical, are involved in intrusion detection and can potentially be viewed as fuzzy variables. For instance, the CPU usage time and the connection duration are two examples of ordinal measurements. An example of a linear categorical measurement is the number of different TCP/UDP services initiated by the same source host. The second reason to introduce fuzzy logic for intrusion detection is that security itself includes fuzziness. Given a quantitative measurement, a range value or an interval can be used to denote a normal value. Then, any values falling outside the interval will be considered anomalous to the same degree regardless of their different distances to the interval. The same applies to values inside the interval, i.e., all will be viewed as normal to the same degree. Unfortunately, this causes an abrupt separation between normality and anomaly [8].

With the fuzzy input sets defined, the next step is to write the rules to identify each type of attack. A collection of fuzzy rules with the same input and output variables is called a fuzzy system. We believe the security administrators can use their expert knowledge to help create a set of rules for each attack. The rules are created using the fuzzy system editor contained in the MATLAB Fuzzy Toolbox. This tool contains a graphical user interface that allows the rule designer to create the member functions for each input or output variable, create the inference relationships between the various member functions and to examine the control surface for the resulting fuzzy system. It is not expected, however, that the rule designer utterly relies on intuition to create the rules. Visual data mining can assist the rule designer in knowing which data features are most appropriate and relevant in detecting different kinds of attacks [9].

Proposed System Design and Planning

Proposed work will be anomaly based intrusion detection system using fuzzy logic. The primary task is to characterize the target network in terms of suitable network parameters. The parameters are chosen such that their values will change perceivably in normal and intrusive conditions. The features considered will be the commonly seen protocols in the network traffic, the traffic data rate and the flow direction. The proposed anomaly based IDS has two operational modes.

Learning (or training) mode

In this mode, the IDS will learn the normal traffic behavior in terms of representative feature set characterizing the target network. It will collects the statistics of the selected network parameters for different types of days (Week days from Monday to Friday, Saturdays and Sundays) and then stores them into a specified file for subsequent processing. The frequency of statistics collection is set as per requirement; it is set by default to 10 minutes. An IDS is put in this mode for sufficient period to learn the normal network behavior. When IDS is learning the normal behavior, the target network is assumed to be free from attacks and intrusions.

Following attributes will be considered for characterizing the network:

TCP Packet count (incoming, outgoing and within LAN)
UDP Packet count (-----------------’ ’--------------------)
ICMP Traffic (-----------------’ ’--------------------)
The number of TCP connections
Web Traffic (incoming, outgoing)
DNS Traffic (---------’ ’------------)
Data rates TCP traffic in kb/s (---------’ ’------------)
Data rates UDP traffic in kb/s (---------’ ’------------)
Data rates HTTP traffic in kb/s (---------’ ’------------)
Data rates DNS traffic in kb/s (---------’ ’------------)

Once the learning is over, profile for the target network will be generated with the gathered data using a profiler. If statistics collections is done at every 10 minutes and the learning period is say 1 month, total 24 sample values are available for each network parameter corresponding to each hour of the week day. Hence the profile is generated for each hour of the day over entire week. This profile will used by Anomaly detection module during the detection phase. The IDS is also trained to learn the network behavior in the presence of network intrusions. Intrusions are simulated using the MIT-DARPA training data set.

When the network environment changes for genuine reasons, it may result into a number of false positives. In such situations the Anomaly model will be updated by rerunning the training phase on the changed traffic and rebuilding the profile using profiler program.

Input- The file containing the features values logged during the learning phase
Output- files containing the mean, standard deviations and inverse matrices of feature set

Detection mode

In this mode, IDS will detects in real time, the network based attacks leading to abnormal traffic pattern. The abnormality will be decided on the basis of the network profile constructed earlier. The Anomaly detection module samples the selected network parameters at regular intervals, as in the case of learning mode, checks whether they comply with already established network profile for that particular hour and day of the week. If it detects significant deviations, then it triggers alerts.

Input- The file containing the network profile
Output- Sends alert in case an event is detected as intrusion

The input to the proposed system will be raw dataset, which is divided into two subsets such as, training dataset and testing dataset. At first, the training dataset is classified into five subsets so that, four types of attacks (DoS (Denial of Service), R2L (Remote to Local), U2R (User to Root), Probe) and normal data are separated. After that, it simply mines the 1-length frequent items from attack data as well as normal data. These mined frequent items are used to find the important attributes of the input dataset and the identified effective attributes are used to generate a set of definite and indefinite rules using deviation method. Then, we generate fuzzy rule in accordance with the definite rule by fuzzifying it in such a way, we obtain a set of fuzzy if-then rules with consequent parts that represent whether it is a normal data or an abnormal data. These rules are given to the fuzzy rule base to effectively learn the fuzzy system. In the testing phase, the test data is matched with fuzzy rules to detect whether the test data is an abnormal data or a normal data [1, 2, 4].

This anomaly-based intrusion detection system makes use of effective rules identified in accordance with the designed strategy, which is obtained by mining the data effectively. The fuzzy rules generated from the proposed strategy will be able to provide better classification rate in detecting the intrusion behavior. The foremost advantage of anomaly-based detection techniques is their ability to detect formerly unseen and unfamiliar intrusion occurrences. The different steps involved in the proposed system for anomaly-based intrusion detection are described as follows:

1. Classification of training data
2. Strategy for generation of fuzzy rules
3. Fuzzy decision module
4. Finding an appropriate classification for a test input

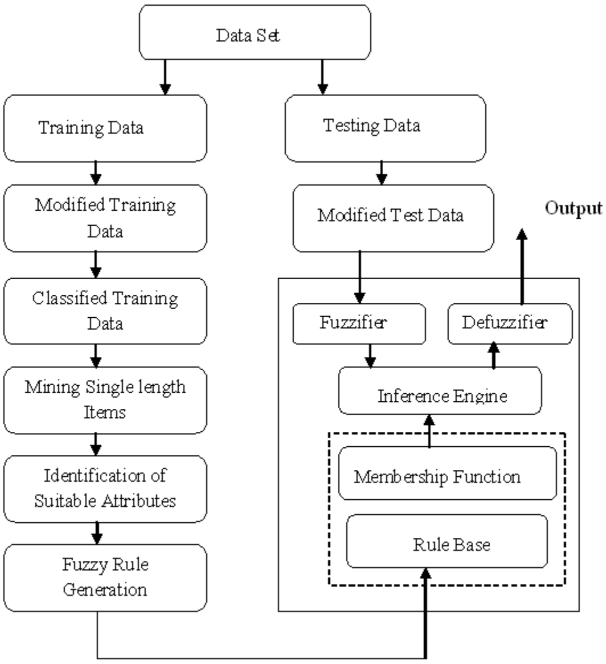

Fig.1- Overview of proposed system

Classification of training data

The first component of the proposed system is of classifying the input data into multiple classes by taking in mind the different attacks involved in the intrusion detection dataset. The dataset we will be take for analyzing the intrusion detection behavior using the proposed system is KDD-Cup 1999 data. Based on the analysis, this data contains four types of attacks and normal behavior data with 41 attributes that have both continuous and symbolic attributes. The proposed system considers only few attributes. Then, the dataset (D) is divided into five subsets of classes based on the class label prescribed in the dataset $D=\{D_i;1\leq\ i \leq 5\}$. The class label describes several attacks, which comes under four major attacks (Denial of Service, Remote to Local, U2R and Probe) along with normal data.

Strategy for generation of fuzzy rules

The five subsets of data are then used for generating a better set of fuzzy rules automatically so that the fuzzy system can learn the rules effectively. It will make use of mining methods to identify a better set of rules. Here, definite rules obtained from the single length frequent items are used to provide the proper learning of fuzzy system.

The process of fuzzy rule generation is given in the following steps.

a. Mining of single length frequent items

At first, frequent items (attributes) are discovered from both classes of input data and by using these frequent items, the significant attributes are identified for the input dataset. It will simply find the 1-length items from each attributes by finding the frequency of the continuous variable present in each attribute and then, the frequent items will be discovered by inputting the minimum support. These frequent items are identified for both class namely, normal and attack (combining four types of attacks).

b. Identification of suitable attributes for rule generation

In this step, it will choose only the most suitable attributes for identifying the classification whether the record is normal or attack. The reason behind this step is that the input data contain 34 attribute, in which all the attributes are not so effective in detecting the intrusion detection. For identifying the suitable attribute, it will make use of deviation method, where mined 1-length frequent items are used.

c. Rule generation

The effective attributes chosen from the previous step will be utilize to generate rules that is derived from the {max, min} deviation. By comparing the deviation range of effective attributes in between the normal and attack data, the intersection points will be identified for the effective attributes. By making use of these two intersection points, the definite and indefinite rules will be generated.

d. Rule filtering

The rules that are generated from the previous step contain definite and indefinite rules. The definite rules are the rules that contain only one classified label in the THEN part and indefinite rule contain two classification label data in the THEN part. The proposed rule filtering technique will filters the indefinite rule and selects only the definite rules for learning the fuzzy system.

e. Generating fuzzy rules

In the proposed system, it will automatically find the fuzzy rules based on the mined 1-length frequent items. The fuzzy rules are generated from the definite rules, where the IF part of the rule is a numerical variable and THEN part is a class label related to attack name or normal. But, the fuzzy rule should contain only the linguistic variable. So, in order to make the fuzzy rules from the definite rules, we should fuzzify the numerical variable of the definite rules and THEN part of the fuzzy rule is same as the consequent part of the definite rules. For example, “IF attribute1 is H, THEN the data is attack and “IF attribute1 is VL, THEN the data is normal”. These fuzzy rules are used to learn the fuzzy system so that the effectiveness of the proposed system will be improved rather than simply using the fuzzy rules without any proper techniques.

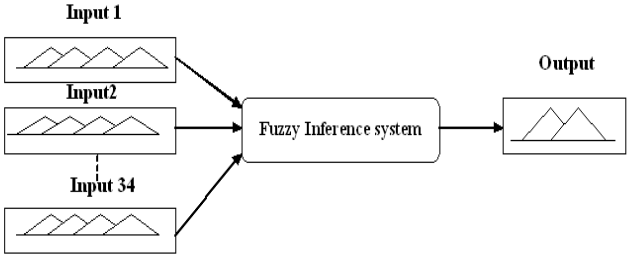

Fig. 2- Proposed fuzzy system

The proposed fuzzy system shown in figure contains 34 inputs and one output, where input are related to the 34 attributes and output is related to the class label (attack data or normal data). Here, each input fuzzy set defined in the fuzzy system includes four membership functions (VL, L, M and H) and an output fuzzy set contains two membership functions (L and H). Each membership function used triangular function for fuzzification strategy. The fuzzy rules obtained from above sub-section are fed to the fuzzy rule base for learning the system.

Finding an Appropriate Classification for a Test Input

For testing phase, a test data from the dataset is given to the designed fuzzy logic system discussed in above sub-section for finding the fuzzy score. At first, the test input data containing 34 attributes is applied to fuzzifier, which will converts 34 attributes (numerical variable) into linguistic variable using the triangular membership function. The output of the fuzzifier is fed to the inference engine which in turn compares that particular input with the rule base. Rule base is a knowledge base which contains a set of rules obtained from the definite rules. The output of inference engine is one of the linguistic values from the following set {Low and High} and then, it is converted by the defuzzifier as crisp values. The crisp value obtained from the fuzzy inference engine is varied in between 0 to 2, where '0' denotes that the data is completely normal and '1' specifies the completely attacked data [1, 11, 12].

System Requirement

- Operating System Requirements: Windows XP Professional and related all Drivers.
- Software Requirement: Matlab-2010.

Hardware Requirements: A Pentium IV or Higher processor with at least 2 GB of RAM.

Conclusion

It is possible to develop an anomaly based intrusion detection system which detects the intrusion behaviour within a network. By introducing fuzzy decision-making module system will be more accurate for attack detection, using the fuzzy inference approach. An effective set of fuzzy rules for inference approach will be identified automatically by making use of the fuzzy rule learning strategy, which is more effective for detecting intrusion in a computer network.

The strategy for implementation is as follows:

The definite rules will be generated by mining the single length frequent items from attack data as well as normal data.

Fuzzy rules will be identified by fuzzifying the definite rules

These rules will be fed to fuzzy system, which will classify the test data.

It is decided to use KDD cup 99 dataset for evaluating the performance of the proposed system and the proposed method is effective in detecting various intrusions in computer networks.

References

[1] Shanmugavadivu R., Nagarajan N. (2011) *Indian Journal of Computer Science and Engineering* (IJCSE), 2(1).

[2] Anazida Zainal, Mohd Aizaini Maarof and Siti Mariyam Shamsuddin (2006) *Postgraduate Annual Research Seminar*.

[3] Yao J.T., Zhao S.L. and Saxton L.V. (2005) *Proceedings of the Data Mining, Intrusion Detection, Information Assurance and Data Networks Security*, SPIE, 5812, 23-30.

[4] Gupta K.K. (2009) *Department of Computer Science and Software Engineering The University of Melbourne*.

[5] Garcia-Teodoroa P., Diaz-Verdejoa J., Macia-Fernandeza G., Vazquez E. (2009) *Computers & security* 28, 18-28.

[6] Fengmin Gong (2003) *White Paper from McAfee Network Security Technologies Group*.

[7] Saniee Abadeh M., Habib J. and Lucas C. (2007) *Journal of Network and Computer Applications*, 30(1), 414-428.

[8] Luo J. and Bridges S.M. (2000) *International Journal of Intelligent Systems*, 15(8), 687-704.

[9] Alanazi H.O., Rafidah Md Noor, Zaidan B.B., Zaidan A.A. (2010) *Journal of Computing*, 2(2).

[10] Mahbod Tavallaee, Ebrahim Bagheri, Wei Lu and Ghorbani A.A. (2009) *Proceedings of the IEEE symposium on computational intelligence in security and defense applications* (CISDA).

[11] *Http://www.ll.mit.edu/mission/communications/ist/corpora/ideval/data/1998data.html.*

[12] *Http://www.sigkdd.org/kddcup/index.php?section=1999&method=data.*