Fuzzy Logic And Network Intrusion Detection System

[1]Bhawna Sinha, [2]S.S. Sahay, [3]Braj Kishor Prasad
[1,3]Assistant Professor, [2]Head of Department
[1,3]Patna Women's College
[2]Bakhtiyarpur College of Engineering

***Abstract*—Intrusion Detection System which are the key part of system defense are 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. Currently available intrusion detection systems focus mainly on determining uncharacteristic system events in distributed networks using signature based approach. Due to its limitation of finding novel attacks, we propose a hybrid model based on improved fuzzy and data mining techniques, which can detect both misuse and anomaly attacks. In the proposed system, we have designed fuzzy logic-based system for effectively identifying the intrusion activities within a network. The proposed fuzzy logic-based system can be able to detect an intrusion behavior of the networks since the rule base contains a better set of rules. In this paper we resent the study of network intrusion detection using fuzzy logic with suitable model. The proposed model has been tested against the live networking environment inside the campus and the results have been discussed.**

***Index Terms*—Intrusion Detection System, Fuzzy logic, hybrid model, network intrusion detection**

I. Introduction

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.

Computer security is one of the top priorities of our modern society. The Internet growth, information sharing, and technology improvement are some of the factors that humans become dependent on. Illegal access to private and confidential data has become a new way of crime. Tools for malicious purposes are freely available for download from the Internet. Hackers all over the world no longer need to have a strong IT background in order to perform attacks.

There are several avenues that help regular users or professional administrators to better protect their data. The first line of defense against intruders is to adopt preventive measures such as physical access control, logical access control (i.e., passwords and encryption) or network access control (i.e., firewalls, VPNs) that dramatically decrease the chances of an intruder to compromise the data. Another preventive measure is to keep the software updated with the latest security patches that are released to avoid software-bug exploits. Another common way not only to prevent but also to combat unwanted activity is the use of Intrusion Detection Systems (IDS) that can detect and also actively or passively respond to intrusions.

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.

The difference between an active and passive response is that in the first case the system takes control over the course of actions that follow the detection of an intrusion (such as, deleting software, blocking a connection), whereas in the case of passive response, the intrusion is reported but the countermeasure decision is left at the human's discretion. Depending on the detection scope, intrusion detection can be categorized into two main classes, host-based and network-based.

The host-based intrusion detection concentrates on the security of a single machine; whereas the network-based intrusion detection concentrates on the security of a network in general. The host-based intrusion detection has the advantage of being able to oversee and monitor all the processes that are initiated on the local host. This type of protection can be very instrumental especially in detecting possible viruses or Trojans that might be intentionally or unintentionally deployed on the protected host.

II. DATA COLLECTION IN INTRUSION DETECTION

Data acquisition is one of the biggest challenges that a network security system must undertake. The decision on the amount of data, and the type and place of the data capturing process dramatically influences not only the performance of the system, but also its trustworthiness and detection scope. Based on the type of data that is collected, the IDSs can be classified into five main categories as follows:

1. *Application-integrated IDSs*: are those IDSs that collect the data out of a single application. They have an embedded sensor inside the application itself that collects and sends the extracted features to the IDS for processing.[1,5, 8]

2. *Application-based IDSs*: are those IDSs that monitor only one application by transparently collecting the necessary data. This is done by the use of external sensors that detect and capture the data exchanged between the monitored application and those third party entities (e.g., applications, hosts) that it interacts with.
3. *Host-based IDSs*: evaluate and keep track of the wellness of a host as an entity by monitoring its applications.
4. *Network-based IDSs*: are those IDSs that are meant to protect the network itself. The network is pictured as a collection of hosts that interact by exchanging messages that are transferred through wires (i.e., in the case of a wired network) or radio waves (i.e., in the case of a wireless network). The sensor of the IDS collects data by sniffing it directly from the network traffic in a transparent way.[8]
5. *Hybrid IDSs*: use two or more of the previously described techniques in order to collect data. The following sections further describe each of the previously mentioned categories, along with their advantages and disadvantages.[11, 4]

III. EXISTING CLASSIFICATIONS OF THE NETWORK DATA FEATURES

The general confusion that exists around the best features to be used from the network data has multiple root causes, and one of them is the lack of universally accepted network feature classifications. Even though there is no unanimous classification consensus regarding the features that can be extracted from raw packet data, most of the papers do make a distinction (even if not directly) between features that are computed with respect to a single TCP connection, versus those that are computed considering multiple TCP connections as follows:

a) **Basic TCP Features:** are those features that characterize a single TCP/IP connection. The names for this category differs from author to author, but the semantic tends to be consistent. [1, 4, 8]

b) **Derived Features:** are those features that can characterize multiple TCP/IP connections at the same time Also known as *Traffic Features* . By their use the system finds similarities that exist between different TCP connections in the network. In order to compute the derived features, two types of sliding window intervals are used. The first approach uses a time window interval of a few seconds (e.g., 5 sec), while the second approach uses a connection window interval of several connections (e.g., last 100 connections).

Thus, the *Derived Features* category is further divided into:

i) **Time-based features:** Includes all the derived features computed with respect to the past x seconds (where x is the size of the time window interval.

ii) **Connection-based features:** Includes all the derived features computed with respect to the latest k encountered TCP connections in the network.

While the first category of features (i.e., Basic TCP Features) are used to characterize and detect attacks that use only one connection, the second category is mostly used to detect attacks that employ multiple connections at the same time (e.g., scanning, DDoS attacks, and worm spreads). Furthermore, the Time-based feature category is mostly used for detecting bursty attacks (i.e., attacks that happen within a short period of interval) such as *worm* and *DDoS*. Finally, the Connection-based features are predominantly used for the detection of stealthy attacks, attacks that happen within a long period of time usually several minutes or even hours. Comparing the pool of available feature classifications with the tremendous amounts of network data that can be monitored and extracted, it is clear that this area greatly suffers from an comprehensive classification schema that takes into account the nature and diversity of feature types that can be seen in the network.

Despite the advantages that this method has, its maintenance proves to be quite a challenge when the security of a whole enterprise is targeted. Moreover, the method remains oblivious to network attacks such as probing, worm, and Denial of Service (DoS).

The network-based approach solves some of the problems that the host-based approach suffers from by its intrinsic superior position.

A network-based IDS (NIDS) has access to network data and also application level data, which allows it to detect network-based intrusions and also some of the application-based intrusions. The disadvantage of this approach is its limitation of detecting those application level attacks that look normal from the network point of view.

For instance, a Trojan that opens a back channel to another host will appear as a perfectly legitimate connection as seen by the NIDS. The huge amount of data that a NIDS can analyze also inevitably leads to computationally intensive tasks that are very hard to cope with. That is why some of the NIDSs compromise their in-depth analysis for a better performance. More and more enterprises adopt a hybrid approach by enforcing their machines to have certain antivirus products while also having a network-based appliance.

This approach takes advantage of both methods providing a comprehensive coverage for the enterprise. Regardless of its type, there are three main approaches that an IDS can implement for the detection of attacks. The first approach is the signature-based one. This type of detection engine uses signatures to identify intrusions. An intrusion is identified if it matches any of the signatures that the IDS has. The method is very precise and does not produce ambiguous results due to the signatures that it has; however, its main disadvantage is the lack of versatility against new attacks or unseen variants of the known attacks.

The second type of detection is the specification-based one. This type of detection uses the protocol or application specification to create a normal profile of allowed actions. If during the running time the monitored application or protocol does not comply with their specifications, an intrusion is signaled. The advantage of this approach is that by defining the normal behavior everything that deviates from it will be signaled and reported. Due to the increasing complexity of applications the main challenge of this technique remains to determine and create the normal allowed profiles. Finally, the last type of detection technique is the anomaly-based one. This type of technique is not as precise as the signature-based one, but can be very instrumental in detecting variations of attacks as well as new ones. During a training period, by using machine learning techniques, a normal profile is created, which will be constantly compared with a running profile extracted at runtime. Once the difference between the normal and running profiles exceeds a certain threshold, an anomaly is signaled. The main disadvantage of such method is the relative high number of false positives that it produces.

IV Network Intrusion Detection System Using Fuzzy logic

Recently, several researchers focused on fuzzy rule learning for effective intrusion detection using data mining techniques. By taking into consideration these motivational thoughts, we have developed a fuzzy rule based system in detecting the attacks. This system, anomaly-based intrusion detection 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 can be able to provide better classification rate in detecting the intrusion behavior. Even though signature-based systems provide good detection results for specified and familiar attacks, the foremost advantage of anomaly-based detection techniques is their ability to detect formerly unseen and unfamiliar intrusion occurrences. On the other hand and in spite of the expected erroneousness in recognized signature specifications, the rate of false positives in anomaly-based systems is generally higher than in signature based ones. 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

Feature Evaluation Module

The *Feature Evaluation Module* implements the feature performance evaluation as previously described in Chapter 3.3. Figure 4.5 depicts the underlying block diagram of this module. This module is implemented as a combination of MATLAB and Java procedures. This whole process is executed once, after the *Statisical Profiler Module* has exhausted all its input data. The whole evaluation process is designed as a sequential process that consists of four tiers. Each individual tier can be executed only after the previous tiers have completely exhausted the data that they work with. For this reason, there are three temporary databases that act like buffers between adjacent tiers. The only functionality that the databases have is to store data until is needed at the next tier. As depicted in Figure-i, the first processing tier is done for each feature tuning combination. This module implements the previously presented algorithm for Fuzzy evaluation of f_i against ξ_j attack while using τ_k tuning.

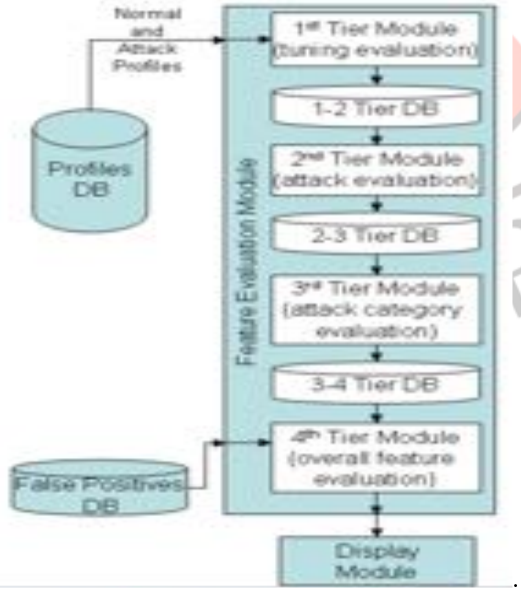
Normal and Attack Profiles
Profiles DB
Feature Evaluation Module
1st Tier Module (tuning evaluation)
1-2 Tier DB
2nd Tier Module (attack evaluation)
2-3 Tier DB
3rd Tier Module (attack category evaluation)
3-4 Tier DB
4th Tier Module (overall feature evaluation)
False Positives DB
Display Module

Figure-i : The overall view of the Feature Evaluation Module block diagram

We use MATLAB Fuzzy Logic Toolbox to implement the fuzzy inference. Next, after all the possible combinations are exhausted, the second tier starts its processing stage for each individual attack-feature combination. When all the possible combinations are exhausted, the third tier starts evaluating each feature against all the defined attack classes. The fourth and final tier uses both, the information provided by the antecedent tier, as well as the information stored in the *False Positives DB*.

V. FUZZY DECISION MODULE

This section describes the designing of fuzzy logic system for finding the suitable class label of the test dataset. Zadeh in the late 1960s introduced Fuzzy logic and is known as the rediscovery of multivalued logic designed by Lukasiewicz.[14] The designed fuzzy system contains 34 inputs and one output, where inputs are related to the 34 attributes and output is related to the class label (attack data or normal data). Here, thirty four-input, single-output of Mamdani fuzzy inference system with centroid of area defuzzification strategy was used for this purpose. Here, each input fuzzy set defined in the fuzzy system includes four membership functions and an output fuzzy set contains two membership functions. Each membership function used triangular function for fuzzification strategy.

In the testing phase, the testing dataset is given to the proposed system, which classifies the input as a normal or attack. The obtained result is then used to compute overall accuracy of the proposed system. The overall accuracy of the proposed system is computed based on the definitions, namely precision, recall and F-measure which are normally used to estimate the rare class prediction. It is advantageous to accomplish a high recall devoid of loss of precision.

VI. CONCLUSIONS

Internet and data fraud has become one of the most challenging cybernetic acts that security officers around the world try to combat. The more critical and confidential the data is the more appealing it is for attackers. The impact of a successful attack on an institution can have disastrous consequences such as privacy breach, data loss, or service interruption. Researchers around the world constantly develop and improve NIDS that are meant to combat such threats. For a NIDS to function properly all of its building blocks and processing components need to be properly designed. The feature selection stage is one of the first steps that needs to be addressed. This step can be considered among the top most important ones, since the overall performance and detection scope of the NIDS directly depends on it. Despite its importance we believe that the feature selection phase for intrusion detection has not been sufficiently studied and explored by the research community.

The main focus of this paper is on mining the most useful network features for attack detection. In order to do this, we proposed a network feature classification schema as well as a deterministic feature evaluation procedure that helps to identify the most useful features that can be extracted from network packets. We have developed an anomaly based intrusion detection system in detecting the intrusion behavior within a network. A fuzzy decision-making module was designed to build the system more accurate for attack detection, using the fuzzy inference approach. An effective set of fuzzy rules for inference approach were identified automatically by making use of the fuzzy rule learning strategy, which are more effective for detecting intrusion in a computer network. At first, the definite rules were generated by mining the single length frequent items from attack data as well as normal data. Then, fuzzy rules were identified by fuzzifying the definite rules and these rules were given to fuzzy system, which classify the test data.

REFERENCES

[1] KDD 99, *The fifth international conference on knowledge discovery and data mining*, http://kdd.ics.uci.edu, Website, Last accessed Octomber 2005.

[2] J. L. Verdegay A. Sancho-Royo, *Methods for the construction of membership functions*, vol. 14, 1999, pp. 12-30.

[3] Magnus Almgren and Ulf Lindqvist, *Application-integrated data collection for security monitoring*, Proceedings of Recent Advances in Intrusion Detection, 4th International Symposium, (RAID 2001) (Davis, CA, USA) (W, L. M Lee, and A. Wespi, eds.), Lecture Notes in Computer Science, Springer-Verlag Heidelberg, October 2001, pp. 22-36.

[4] R. Basu, R.K. Cunningham, S. E. Webster, and R. P. Lippmann, *Detecting low-profile probes and novel denial-of-service attacks*, Proceedings of the workshop on Information Assurance and Security, United States Military Academy (IEEE, ed.), June 2001, pp. 5-10.

[5] M. Ben-Bassat, *Handbook of statistics 2: Classification, pattern recognition and reduction of dimensionality*, ch. Use of Distance Measures, Information Measures and Error Bounds in Feature Evaluation, pp. 773-791, North Holland, 1982.

[6] Berk, G. Bakos, and R. Morris, *Designing a framework for active worm detection on global networks*, Proceedings of the IEEE InternationalWorkshop on Information Assurance (Darmstadt, Germany), March 2003, pp. 13-23.

[7] Joachim Biskup and Ulrich Flegel, *Transaction-based pseudonyms in audit data for privacy respecting intrusion detection*, Proceedings of Recent Advances in Intrusion Detection, 3rd International Symposium, (RAID 2000)(Toulouse, France) (H. Debar, L. M, and S.F. Wu, eds.), Lecture Notes in Computer Science, Springer-Verlag Heidelberg, October 2000, pp. 28-48.

[8] R.G.Bace, "Intrusion Detection", Macmillan Technical Publishing, Indianapolis, USA, 2000.

[9] Marcos M. Campos, Boriana L. Milenova, "Creation and Deployment of Data Mining-Based Intrusion Detection Systems in Oracle Database 10g", in Proceedings of the Fourth International Conference on Machine Learning and Applications, 2005.

[10] R. Agrawal, T. Imielinski, A., Swami, "Mining association rules between sets of items in large databases", in Proceedings of 1993 ACM SIGMOD Intl. Conf. on Management of Data, Washington, DC, pp. 207–216, 1993.

[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

[13] Mahbod Tavallaee, Ebrahim Bagheri, Wei Lu and Ali A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set", in Proceedings of the Second IEEE international conference on Computational intelligence for security and defense applications, pp. 53-58, Ottawa, Ontario, Canada, 2009.

[14] Zadeh, L.A., "Fuzzy sets", Information and control, vol.8, pp. 338-353, 1965.