8

Hybrid Intrusion Detection Systems (HIDS) using Fuzzy Logic

Bharanidharan Shanmugam and Norbik Bashah Idris
Advanced Informatics School (AIS),
Universiti Teknologi Malaysia International Campus,
Kuala Lumpur
Malaysia

1. Introduction

The rapid growth of the computers that are interconnected, the crime rate has also increased and the ways to mitigate those crimes has become the important problem now. In the entire globe, organizations, higher learning institutions and governments are completely dependent on the computer networks which plays a major role in their daily operations. Hence the necessity for protecting those networked systems has also increased. Cyber crimes like compromised server, phishing and sabotage of privacy information has increased in the recent past. It need not be a massive intrusion, instead a single intrusion can result in loss of highly privileged and important data. Intusion behaviour can be classified based on different attack types. Smart intruders will not attack using a single attack, instead, they will perform the attack by combining few different attack types to deceive the detection system at the gateway. As a countermeasure, computational intelligence can be applied to the intrusion detection systems to realize the attacks, alert the administrator about the form and severity, and also to take any predetermined or adaptive measures dissuade the intrusion.

2. Need for hybrid IDS systems

This section introduces a classification (Debar *et al*., 1999) of intrusion detection systems that highlights the current research status. This classification defines families of intrusion detection systems according to their properties. There are four different types (Figure 1) of intrusion detection available based on the past (Axelsson, 1998; Richard and Giovanni, 2002) and current researches (Scarfone and Peter, 2007; Sabahi and Movaghar, 2008). The following paragraphs explain the types in detail. Principally, an IDS is concerned with the detection of hostile actions. The intrusion detection approaches can be classified into anomaly based and signature based which any network security tools are mostly using (Ozgur *et al.,* 2005). One more classification can be made by considering the source of data used for intrusion detection. The taxonomy can be given based on the information derived from a single host (named as Host based IDS (HIDS)) and the information derived from complete segment of the network that is being monitored (named as Network based IDS (NIDS).

Any IDS can be categorized upon its operation as standalone or centralized applications that create a distributed system. Standalone systems will be working individually without any agents but centralized applications work with autonomous agents that are capable of taking preemptive and reactive measures.

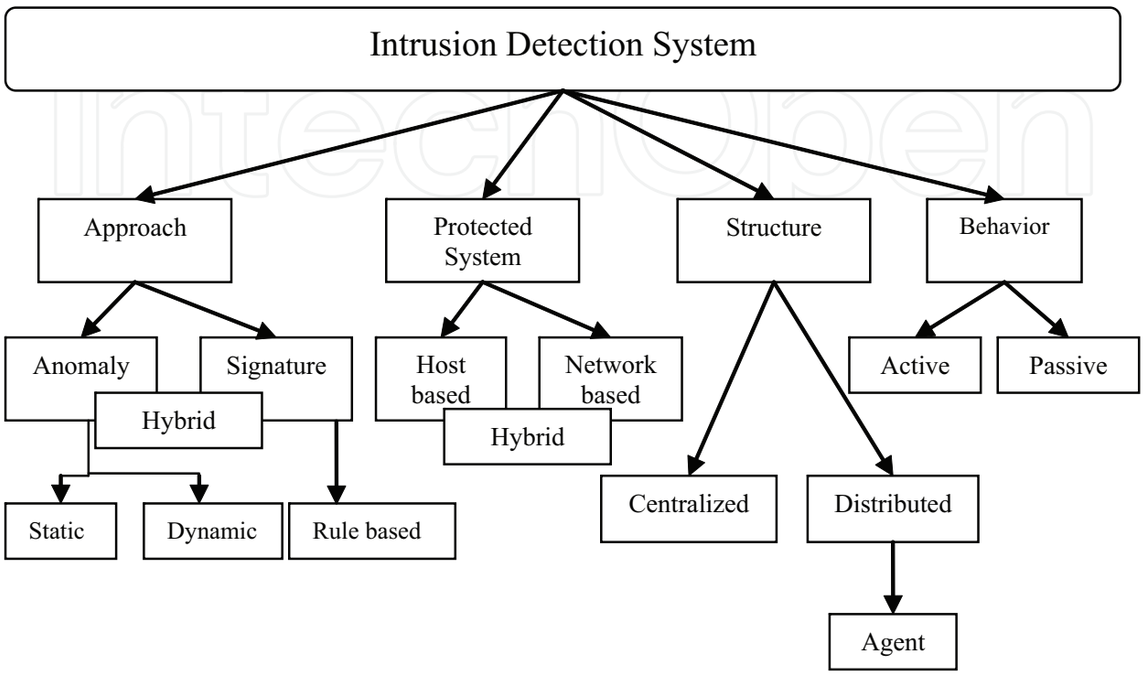


Fig. 1. Classification of intrusion detection systems

An IDS is categorized as behavior-based system, when it uses information about the normal behavior of the system it monitors. Behavior on detection describes the response of the IDS after the detection of attacks. It can be divided into active or passive based on the attack response. These two types of intrusion detection systems differ significantly from each other, but complements one another well. The architecture of host-based is completely dependent on agent-based, which means that a software agent resides on each of the hosts, and will be governed by the main system. In addition, more efficient host-based intrusion detection systems are capable of monitoring and collecting system audit trails in real time as well as on a scheduled basis, thus distributing both CPU utilization and network overhead and providing for a flexible means of security administration. It would be advantageous in IDS implementation to completely integrate the NIDS, su ch that it would filter alerts in a identical manner to HIDS and can be controlled from teh same centralized location. In conclusion, highly secure environment should require both NIDS and HIDS to be implemented for not only providing a complete defence against dynamic attacks but also to effectively and effecintly monitor, respond and detected the computer/network misuse against threats and malicious activities.

3. Different aritificial intelligence approaches in HIDS

Artificial Intelligence (AI) techniques play a vital role by reducing the data used for detection and also classifying the data according to the needs and it is applied in both

techniques (misuse detection and anomaly detection). AI techniques have been used to automate the intrusion detection process; which includes neural networks, fuzzy inference systems, evolutionary computation, machine learning, support vector machines, etc. The following sections will give an overall view (Table 1) about some of the Artificial Intelligence (AI) techniques applied for intrusion detection.

3.1 Artificial neural networks (ANN)

An Artificial Neural Network (ANN) consists of a collection of processing units called as neurons, which are highly interconnected. They have the ability to learn-by-example and also via generalizing from limited, noisy, and complete data too. Neural Networks can be distinguished into two types based on its architecture (Wu and Banzhaf, 2009):

1. **Supervised training algorithms**, where in the learning phase, the network learns the desired output for a given input or pattern. The well known architecture of supervised neural network is the Multi-Level Perception (MLP).
2. **Unsupervised training algorithms**, where in the learning phase, the network learns without specifying any desired output. Self-Organizing Maps (SOM) are popular among unsupervised training algorithms. A SOM tries to find a topological mapping from the input space to clusters.

Lippman and Cunningham (1999) and Ryan *et al.*, (1998) created keyword count based IDS with neural networks. Researchers (Ghosh *et al.,* 1999) created a neural network to analyze program behavior profiles instead of user behavior profiles. Cannady (1998), developed a network-based neural network detection system in which packet-level network data was retrieved from a database and then classified according to nine packet characteristics and presented to a neural network. Self-Organizing Maps (SOMs) have also been used as anomaly intrusion detectors (Girardin and Brodbeck, 1998). SOM was used to cluster and then graphically display the network data for the user to determine which clusters contained attacks. Applications of ANN in intrusion detection can be found in Cansian *et al.*, (1997). However, on contrary to neural networks, self-organizing maps do not provide a descriptive model which explains a particular detection decision.

3.2 Genetic algorithms

Genetic Algorithm for Simplified Security Audit Trials Analysis (GASSATA) proposed by the Me (1998), introduced genetic algorithm for misuse intrusion detection. GASSATA constructed a two dimensional matrix. One axis of the matrix specifies different attacks already known. The other axis represents different kinds of events derived from audit trails. Therefore this matrix actually represents the patterns of intrusions. Given an audit record being monitored which includes information about the number of occurrences of every event; this method will apply genetic algorithms to find the potential attacks appearing in the audit record. However, the assumption that the attacks are dependent only on events in this method will restrict its generality. There are two steps involved in genetic algorithm, one is coding a solution to the problem with a string of bits, and the other is finding a fitness function to test each individual of the population against evaluation criteria. Me (1998) used a standard GA, while Dass used a micro-GA in order reduce the time overhead normally associated with GA. Diaz-Gomez and Hougen (2005) corrected the fitness definition (Me, 1998) used after a detailed analysis and mathematical justification (Diaz-Gomez and Hougen, 2006). The detection rate can be high and false alarm can be low if the fitness

function is well designed. The disadvantage is that it cannot locate the attack in audit trails (Zorana *et al.,* 2009) and it cannot detect novel attacks as it requires more domain specific knowledge (Li, 2006). Genetic based intrusion detection has no ability to detect multiple simultaneous attacks (Suhail *et al.,* 2008) and this detection approach meets computation complexity problems. A novel approach proposed by addresses the problem of detecting masquerading, a security attack in which an intruder assumes the identity of a legitimate user. The approach uses the techniques used in bioinformatics for a pair-wise sequence alignments to compare the monitored session with past user behavior. The algorithm uses a semi-global alignment and a unique scoring system to measure similarity between a sequence of commands produced by a potential intruder and the user signature, which is a sequence of commands collected from a legitimate user. Even though a novel method was proposed, false positive rate is somewhat a lackluster.

3.3 Immune system approach

The Human Immune System (HIS) (Somayaji *et al.,* 1997) protects the body against damage from extremely large number of harmful bacteria and viruses, termed pathogens. It does this largely without prior knowledge of the structure of these pathogens. This property, along with the distributed, self-organized and light weighted nature of the mechanisms by which it achieves this protection, has in recent years made it the focus of increased interest within the computer science and intrusion detection communities. The AIS described by Kephart (1994) is one of the earliest attempts of applying HIS mechanisms to intrusion detection. The paper focuses on automatic detection of computer viruses and worms. They utilized fuzzy matching techniques based on the existing signatures. Aickelin *et al*., (2003) discussed the application of danger theory to intrusion detection and the possibility of combining research from wet and computer labs in a theoretical paper. Sarafijanovic and Boudec (2003) developed an immune based system to detect malfunctioning nodes in a ad-hoc networks. They use Dynamic Source Routing (DSR) protocol to create a series of data sets. A detailed review of the AIS applied to intrusion can be found in (Kim, 2003).

3.4 Fuzzy logic

Fuzzy logic starts and builds on a set of user-supplied human language rules. The fuzzy systems convert these rules to their mathematical equivalents. This simplifies the job of the system designer and the computer, and results in much more accurate representations of the way systems behave in the real world. Additional benefits of fuzzy logic include its simplicity and its flexibility. Fuzzy logic can handle problems with imprecise and incomplete data, and it can model nonlinear functions of arbitrary complexity. Fuzzy logic techniques have been employed in the computer security field since the early 90's (Hosmer, 1993). Its ability to model complex systems made it a valid alternative, in the computer security field, to analyze continuous sources of data and even unknown or imprecise processes (Hosmer, 1993). Fuzzy logic has also demonstrated potential in the intrusion detection field when compared to systems using strict signature matching or classic pattern deviation detection. Bridges (Bridges and Vaughn, 2000), states the concept of security itself is fuzzy. In other words, the concept of fuzziness helps to smooth out the abrupt separation of normal behavior from abnormal behavior. That is, a given data point falling outside/inside a defined "normal interval", will be considered anomalous/normal to the same degree regardless of its distance from/within the interval. Fuzzy logic has a capability

to represent imprecise forms of reasoning in areas where firm decisions have to be made in indefinite environments like intrusion detection.
The model suggested in (Dokas *et al*., 2002) building rare class prediction models for identifying known intrusions and their variations and anomaly/outlier detection schemes for detecting novel attacks whose nature is unknown. The latest in fuzzy is to use the Markov model. As suggested in (Xu *et al.,* 2004) a Window Markov model is proposed, the next state in the window equal evaluation to be the next state of time t, so they create Fuzzy window Markov model. As discussed, researchers propose a technique to generate fuzzy classifiers using genetic algorithms that can detect anomalies and some specific intrusions. The main idea is to evolve two rules, one for the normal class and other for the abnormal class using a profile data set with information related to the computer network during the normal behavior and during intrusive (abnormal) behavior.

3.5 Integrating fuzzy logic with datamining

Data mining techniques have been commonly used to extract patterns from sets of data. Although association rules can be mined from audit data for anomaly detection, the mined rules are at the data level. Many quantitative features are involved in intrusion detection. As per previous researches (Lunt *et al.,* 1992; Varun *et al.,* 2009) IDES classifies the statistical measures into four types: ordinal measures, categorical measures, binary categorical measures and linear categorical measures. Both ordinal measures and linear categorical measures are quantitative. SRIs EMERALD (Lunt *et al.,* 1989) also divides the network traffic statistical measures into four classes: categorical measures, continuous measures, intensity measures and event distribution measures. Example for continuous measures is the connection duration and intensity measure is the number of packets during a unit of time.
The fuzzy sets provide a smooth transition between member and non-member of a set; therefore, there are fewer boundary elements being excluded. An alternative solution using fuzzy sets, introduced by Kuok (Kuok *et al*., 2001) to categorize quantitative variables, offered smooth transitions from one fuzzy set to another. Classification has been repeatedly applied to the problem of intrusion detection either to classify events into separate attack categories (e.g., the 1999 KDD Cup Competition) or to characterize normal use of a network service.
In our research work, the greatest need was to reduce the amount of data needed for processing and the false alarm rate. We are primarily seeking to improve the performance of an existing system rather than trying to replace current intrusion detection methods with a data mining approach. While current signature-based intrusion detection methods have limitations as stated in the previous section, they do still provide important services and this required us to determine how data mining could be used in a complementary way to existing measures and improves it.

4. Different hybrid IDS

To the best of our knowledge there are few research work and papers that have been published in the area of Network Security, particularly in the area of hybrid intrusion detection. But the work of integrating misuse and anomaly detection is very rare. Based on the objective set for our research, Table 1 summarizes the closely related work, with the method used and the findings by the respective researchers for each research work selected.

Researcher and Model	Method	Findings
Lee *et al.*, 2001 IIDS	Classification based anomaly detection	It uses inductive rule generation to generate rules for important, yet infrequent events
Dickerson and Dickerson (2000) FIRE	Classification based anomaly detection	It generates fuzzy sets for every observed feature which are in turn used to define fuzzy rules to detect individual attacks
Barbara *et al.*, 2001 ADAM	Association rules and classification based anomaly detection	It performs anomaly detection to filter out most of the normal traffic, then it uses a classification technique to determine the exact nature of the remaining activity
Zhang and Zulkernine (2006)	-	The outlier detection provided by the random forests algorithm is utilized to detect unknown intrusions
Tajbakhsh *et al.*, 2006; Tajbakhsh, *et al.*, 2009	Association based classification	The proposed method has proved the ability to handle more categorical attributes and the efficiency to classify large data sets especially for IDS.
Kai *et al.*, 2007 HIDS	Association rules	This hybrid system combines the advantages of low false-positive rate of signature-based intrusion detection system (IDS) and the ability of anomaly detection system (ADS) to detect novel unknown attacks.
Jianhui *et al.*, 2008	Prefix tree rule mining	Authors proposed a new rule mining algorithm base prefix tree (PTBA), which compress the fuzzy pattern candidate set and frequent set through constructing a tree structure, thus it can save the memory cost of fuzzy pattern candidate and frequent set.

Table 1. Different hybrid IDS

Lee *et al.* (2001) used a classification algorithm called RIPPER to update the rules used by Network Flight Recorder (NFR), a commercial real-time network-monitoring tool. Manganaris *et al.*, (2000) used association rules from Intelligent Miner to reduce false alarms generated by NetRanger's sensors. MITRE used HOMER (Heuristic for Obvious Mapping Episode Recognition) and BART (Back End Analysis and Review if Tagging) along with clustering analysis for detection. (Lee *et al.*, 1999) proposed an association rule-based data mining approach for anomaly detection where raw data was converted into ASCII network packet information, which in turn was converted into connection-level information. These connection level records contained connection features like service, duration, etc. Association rules were then applied to this data to create models to detect intrusions. They

utilized Common Intrusion Detection Framework (CIDF) to extract the audit data, which is used to build the models, and also to push the signatrues for "novel" attacks ensuring the faster response time for attack detection.

The primary advantage of CIDF is, it is capable of including heterogeneous intrusion detection and response components to share the information and resources in a distributed environments for faster attack detection. The method proposed by Lee *et al.,* (2001) extracts fuzzy classification rules from numerical data, applying a heuristic learning procedure. The learning procedure initially classifies the input space into non-overlapping activation rectangles corresponding to different output intervals. In this sense, our work is similar to that of (Lee *et al*., 2001; Manganaris *et al*., 2000 and MITRE).There are no overlapping and inhibition areas. However, the disadvantage listed is, the high false positive rates which is the primary scaling of all the IDS.

Researchers (Dickerson and Dickerson, 2000) developed the Fuzzy Intrusion Recognition Engine (FIRE) (Figure 2) using fuzzy sets and fuzzy rules. FIRE produces fuzzy sets based on a simple data mining technique by processing the network input data. Then the fuzzy rules are defined by the fuzzy sets to detect attacks. FIRE relies on attack specific as they do not establish any model that represents the current state of the system. On the other hand, FIRE detection is based on the fuzzy logic rules that was created, and applies it to the testing audit data for attack classifications. The authors recorded port scan and probes attacks can be detected highly by using this method. But, the primary disadvantage as noted by authors is the labor intensive rule generation process.

In another work, (Barbará *et al*., 2001) describes Audit Data Analysis and Mining (ADAM) (Figure 3), a real-time anomaly detection system that uses a module to classify the suspicious events into false alarms or real attacks. Customized profiles were built using data mining techniques, and then the classification of observed events are classified into either as attacks or as false alarms. ADAM uses a method that combined association rules along with mining and classification techniques. ADAM builds a "normal" database consists of frequent itemsets by using data that is attack free during the training phase.

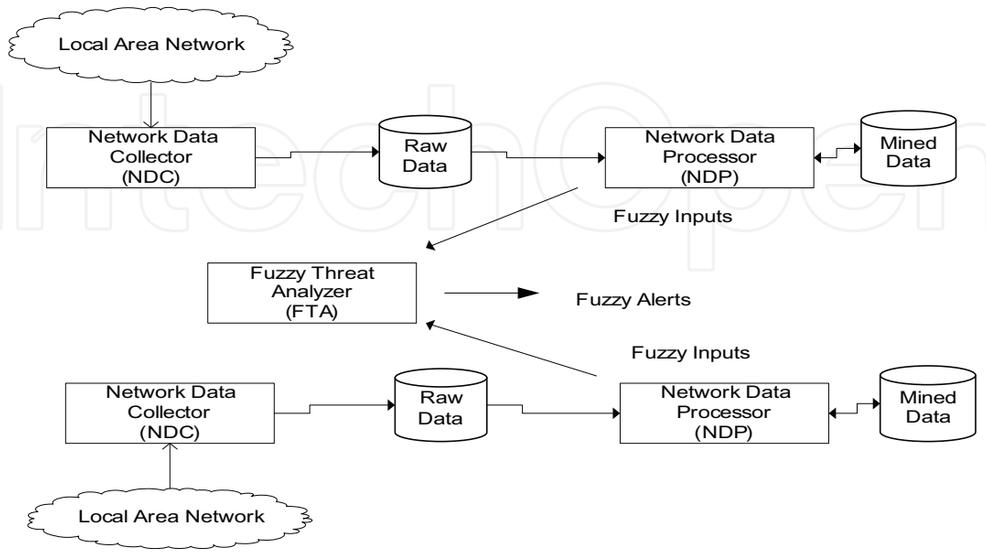


Fig. 2. Fuzzy Intrusion Recognition Engine (FIRE) (Dickerson and Dickerson, 2000)

During the testing (or detection) phase it runs a sliding windoe algorithm to find the frequent itemsets in the last few connections and then compares with the normal item set repository which has already been created. With the remaining item sets which have been flagged as suspicious, ADAM uses a classifier that was trained to classify the known attacks, unknown or a false alarm. Association rules play a major role in gathering necessary data (knowledge) about the nature of audit data. The major drawback is that new type attacks rules need to be given by the external security officer i.e. it does not automate rule generation process and more number of components prevents it from working fast.

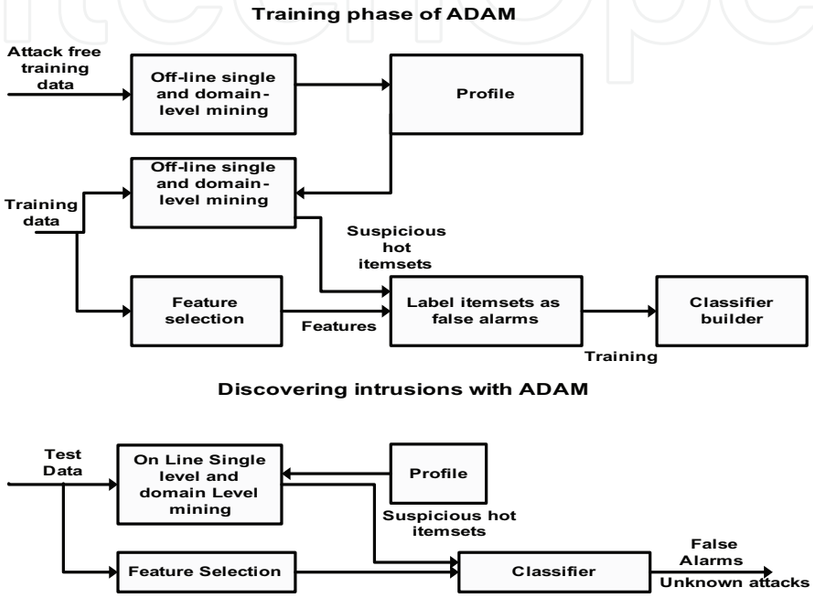


Fig. 3. Training and discovery phases of ADAM (Barbará *et al*., 2001)

Zhang and Zulkernine (2006) detects known intrusions were detected by signature detection module by implementing Random forest algorithm (Breiman, 2001). In the following step, the outlier detection that was produced by random forests algorithm was utilized to detect new or unknown intrusion attempts. Approaches that use both signature detection and anomaly detection produces two set of reports recording the intrusive activities provided they have a correlation component which will analyze and produce perfect results.
Researchers (Tajbakhsh *et al.*, 2006; Tajbakhsh *et al.*, 2009) use association based classification methods (Figure 5) to classify normal and abnormal attacks based on the compatibility threshold. The proposed system consists of training phase and detection phase. In training phase authors use FCM clustering algorithm to define fuzzy membership functions and use hyper edge concept for item / feature reduction. Once rules are defined, then the knowledge base is used in the training phase to match and alert for testing data. FCM an extension of K-means suffer from a basic limitation, i.e. using pair wise similarity between objects and cluster centroids for membership assignment, thereby lacking the ability to capture nonlinear relationships. Since this limitation is not considered by the researchers, there by limiting the system itself in capturing slightly deviated attacks. Next, the system was tested with only 10% of corrected data set from DARPA, which is considered not effective because it is observed (Su-Yun and Ester, 2008) that there is a difference in detection rate and false

positive rates, compared to testing with complete data set and sampling 10% of test data. Also not to forget, that most of the researchers are benchmarked after testing the whole data set and moreover the authors have note mentioned what testing data they used as there are about few weeks of available test data.

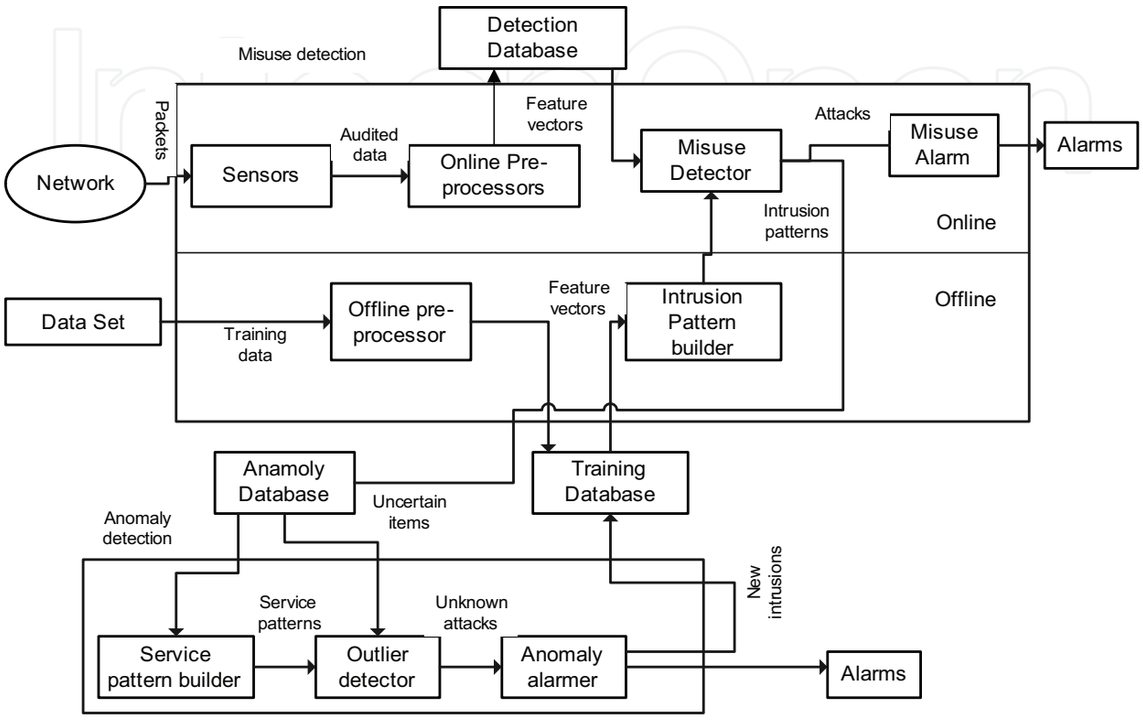


Fig. 4. Misuse and anomaly detection components (Zhang and Zulkernine, 2006)

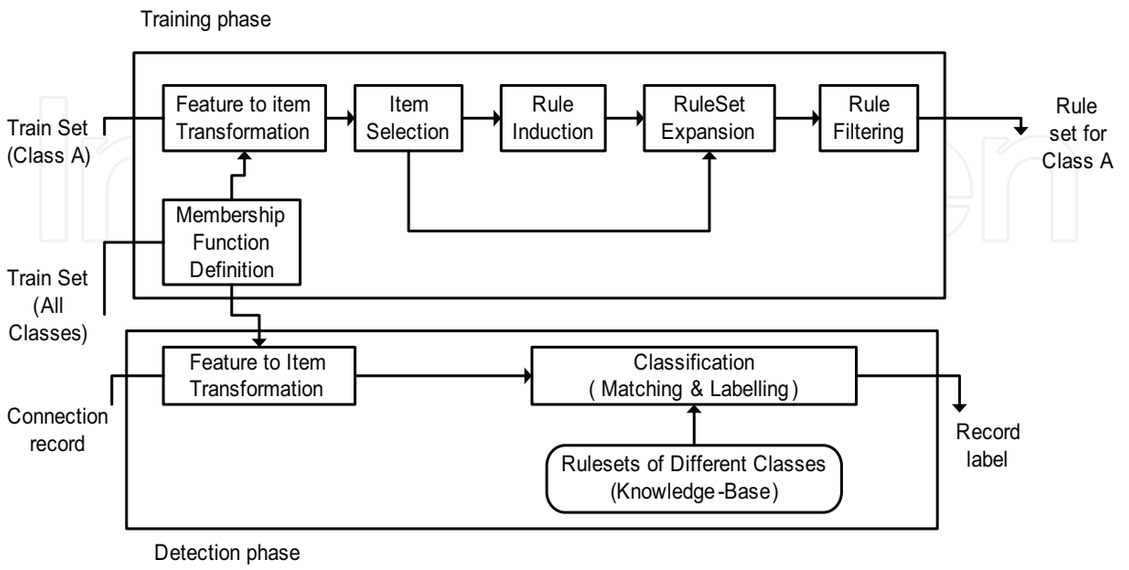


Fig. 5. Block diagram for IDS framework (Tajbakhsh *et al.*, 2009)

The research work (Kai *et al.*, 2007) paper reports the design principles and evaluation results of a new experimental Hybrid Intrusion Detection System (HIDS). This hybrid system combines the advantages of low false-positive rate of signature-based intrusion detection system (IDS) and the ability of Anomaly Detection System (ADS) to detect novel unknown attacks. By mining anomalous traffic episodes from Internet connections, the authors build an ADS that detects anomalies beyond the capabilities of signature-based systems. A weighted signature generation scheme is developed to integrate ADS with SNORT by extracting signatures from anomalies detected. HIDS extracts signatures from the output of ADS and adds them into the SNORT signature database for fast and accurate intrusion detection. The test results of HIDS scheme over real-life Internet trace data mixed with 10 days of Massachusetts Institute of Technology/Lincoln Laboratory (MIT/LL) attack data set (Lippmann *et al.*, 2000), the experimental results showed a 60 percent detection rate of the HIDS, compared with 30 percent and 22 percent in using the SNORT and Bro systems, respectively. This sharp increase in detection rate is obtained with less than 3 percent false alarms. The signatures generated by ADS upgrade the SNORT performance by 33 percent. The HIDS approach proves the vitality of detecting intrusions and anomalies, simultaneously, by automated data mining and signature generation over Internet connection episodes. But, this system is lacking in a major aspect of detection rate, as stated earlier detection rate should be high (some what near to 90%), according to the HIDS, if more than 30% of the attacks are left unnoticed, then the purpose of IDS is getting defeated making it easier for intruders to take control over the protecting networks.
In this research (Jianhui *et al.*, 2008), an intrusion detection model base on fuzzy sets is presented to avoid the sharp boundary problem in rules mining. Considering Apriori algorithm is time-consuming as well as space-consuming; moreover, we propose a new rule mining algorithm base prefix tree (PTBA). PTBA algorithm (Borgelt, 2005) compress the fuzzy pattern candidate set and frequent set through constructing a tree structure, thus it can save the memory cost of fuzzy pattern candidate and frequent set. This characteristic provides a better mining tragedy: if the support degree of a certain node is smaller than the threshold value of support (minsup), the pattern of this node is non-frequent, and then the whole sub-trees whose root node is this node are non-frequent. This characteristic avoids combination explosion and improve mining efficiency prominently. Experiments prove that capability and efficiency of IDS model is obviously improved but, the authors have not addressed the weight supplied on each tree, if the tree goes further down and moreover false positive rate was not recorded and the test data was 10% sampled data of DARPA data set.

5. Proposed hybrid IDS

Our aim is to design and develop an Hybrid Intrusion Detection System (HIDS) that would be more accurate, low in false alarms, Intelligent by using fuzzy mechanisms, not easily deceived by small variations, capable of sniffing and detecting real time packets. The data processor and classifier component summarizes and tabulates the data into carefully selected categories because the amount of data and meta-data associated with network traffic is large. Prior to data analysis, attributes representing relevant features of the input packets must be established. Once the attributes of relevance have been defined, data processor and classifier is employed to compute control variables. Data processor is responsible for accepting raw packet data and produce records for each group as specified by the control variables.

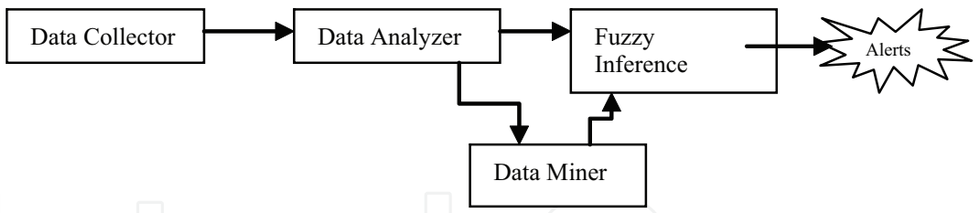


Fig. 6. Overall view of proposed Hybrid IDS

Data mining algorithm by Kuok *et al.*, (1999) is modified and implemented (Shanmugm and Idris 2009)to produce a set of association rules. These rules are expressed as a logic implication with antecedent and consequence. The systems work in two different modes, first being the learning mode and the second being the detection mode. Prior to any data analysis, attributes representing relevant features of the input data (packets) must be established for any IDS. In complex domains like security, attribute selection may play a crucial role. Attributes are represented by names that will be used as linguistic variables by the Data Miner and the Fuzzy Inference Engine and is implemented using the attribute selection algorithm as explained as folows:

Step 1. Initialize the queue S with example set values and the attribute set values.
Step 2. The following steps from 3 until step 7 is performed while CARD(R) is less than the maxsize provided or if the size of the stack is not null.
Step 3. Create a set value which consists of the queue value with maximum support and is a subset of the queue value.
Step 4. Information gain is computed using the formula.
Step 5. Select the attributes with maximum info gain values
Step 6. If the attribute value does not belong to the subset R then
Step 6a. The subset R is the common values of R along with the attributes
Step 7. Following steps are performed for every t_k with their terms of attributes
Step 7a. Example set of terms forms a set based on the examples or the attribute values equals that t_k.

Decision trees are powerful tools in classification and prediction of larger dataset. The attractiveness lies in the formation of rules that is easier for human understanding and the direct usage of those rules with the existing database tools. In majority of the applications especially security, the accuracy of the data classification plays a vital role. In order to define information gain precisely, we need to define a measure commonly used in information theory, called entropy, that characterizes the (im)purity of an arbitrary collection of examples. Given a set S, containing only positive and negative samples with their corresponding labels. The expected information need to classify the DARPA sample is calculated by :

$$I(S_c) = \sum_{i=1}^{c} - p_i \log_2 p_i$$

Where s = total number of samples
c = total classes
$p_i = s_i / s$

For our experiments a feature F with values (f_1,f_2,….f_w) are divided into w training sub sets (s_1,s_2,....s_w) where Sj is the subset which holds the value fj for F. Entropy of the selected feature is

$$E(F) = \sum_{j=1}^{c} \frac{s1j + ... + scj}{s} * I(sij.....scj)$$

More precisely, the information gain, Gain (F) of an attribute A, relative to a collection of examples S, is defined as

$$Gain\ (F) = I(s_1,s_2,....s_c) - E(F)$$

Table 2 shows the information gain calculated for all the attributes based on the equation explained above.

Rank	Information Gain	Feature	Rank	Information Gain	Feature
1	0.988292114	E	13	0.673576388	V
2	0.985914569	C	14	0.671545707	AM
3	0.970739369	J	15	0.662169667	AN
4	0.895566287	AH	16	0.637824522	AO
5	0.844695164	AJ	17	0.591035993	W
6	0.826015494	F	18	0.543491709	AA
7	0.774897786	AI	19	0.516671516	AC
8	0.767343313	AG	20	0.476343726	AF
9	0.767053827	AK	21	0.439147285	L
10	0.724208609	M	22	0.427774836	X
11	0.703067734	A	23	0.391083691	K
12	0.692155232	B	24	0.359009856	D

Table 2. Information gain for DARPA features

6. Implementation results and discussion

To implement and prove the proposed method we used Java 2.0 programming language as our support and implementation tool for IDS. Any research work should be verified with some form of experiment using data. Specifically in the field of Intrusion Detection, testing plays a vital role. To fulfill the above requirements and also to obtain proof of our concept, we tested our system with two sets of data first with DARPA dataset and second, with online data captured inside UTM Citycampus.

Until the year 1998 intrusion detection research has lacked a corpus of data that could serve as the basis for system development, improvement and evaluation. To meet that need, DARPA developed a corpus of data for the DARPA 1998 off-line intrusion detection evaluation, using a network and scripted actors to loosely model the network traffic measured between a US Air Force base and the internet. The latest dataset was added with few more attacks that reflect more real-time data. More details about the 1999 DARPA evaluation data set can be found in Appendix A. For experimental purpose a subset of 1999 DARPA data was used to test the prototype system. A quick glance at the 1999 DARPA

dataset, shows it contains 3 weeks of training data and two weeks of testing data. First and third week data do not contain any attack and this was used as training data for anomaly intrusion detection. Second week contains all types of attacks so this was used for creating attack profiles. We used two weeks of test data for our testing. The prototype was developed using Java programming language. The developed prototype was tested in Pentium 4 2.40 GHz machine with 1 GB RAM. Both testing and real-time data capturing were carried out in the same platform.

The primary aim in our work is to find the relevant attributes with maximum information gain. Table 3 shows the different set of attributes considered for difference classes of attacks ranked accordingly by information gain algorithm. The first attribute found by the attribute selection algorithm, as expected, was ICMP with 0.836 information gain. This is the value with maximum information gain. As a result, the root node of the decision tree is associated with this attribute i.e. root node will carry ICMP.

Types	**Ranking of Features**
NORMAL	AH,E,J,AJ,C,F,A,W,B,M,AC,AK,AF,AG,AI,AN,AA,AM,MO
DOS	AH,E,J,C,AI,AJ
PROBE	E,C,AJ,J,F,AH,B,AG,AI,AK,V,AM,AO,M,AC,AN,AK,W,AF,AA,G,H,L,P,X, AD,AE
U2R	C,J,E,AJ,AM,M,F,AH,AG,AF,AC,AN,AA,V,A,W,L,D
R2L	E,C,J,AH,AJ,AI,AG,AK,F,A,L,M,V,AA,AO,X,AC,AN,W,AM,AF,D,AE
ALL	E,C,J,AH,AJ,F,AI,AG,AK,M,A,B,V,AM,AN,AO,W,AA,AC,AE,AF,L,X,K,D

Table 3. Ranking of attributes for four classes of attacks

The majority of the positive and negative examples will be obviously associated with the ABOVE branch rather than the AVERAGE and BELOW branches. The examples associated with ABOVE were then used for the second iteration, selecting TCP as the most relevant attribute with an information gain of 0.064 and the majority of the training examples in the AVERAGE branch. The third and final iteration selected the FIN attribute with an information gain of 0.00056. This attribute represents tcp packets with the fin flag set. In other words, the attributes ICMP, TCP and FIN are most relevant in this case and are selected to describe the input data in the data mining algorithm.

The aim of this test is to find out the attack type “smurf”. This attack is new to 1999 DARPA data set. The following table describes the attack signature and the description of “smurf”

Attack	**Description**	**Signature**
Smurf	In the *smurf* attack, attackers use ICMP echo request packets directed to IP broadcast addresses from remote locations to create a denial-of-service attack	There is a large number of 'echo replies' being sent to a particular victim machine from many different places, but no 'echo requests' originating from the victim machine.

Table 4. Attack description and signature for smurf

This type of attack detection also required the counter to reach a larger percentage. Hence the threshold was set at 0.6 i.e. the rule must have a firing strength below 0.6 to increase the graph value. Figure 6 shows the firing strength of the rules against the testing data. For the

records from 300 to 500, the firing strength fell to zero and suddenly it shot up to 1 i.e. an increase by 100%. So this sudden rise indicated an attack.

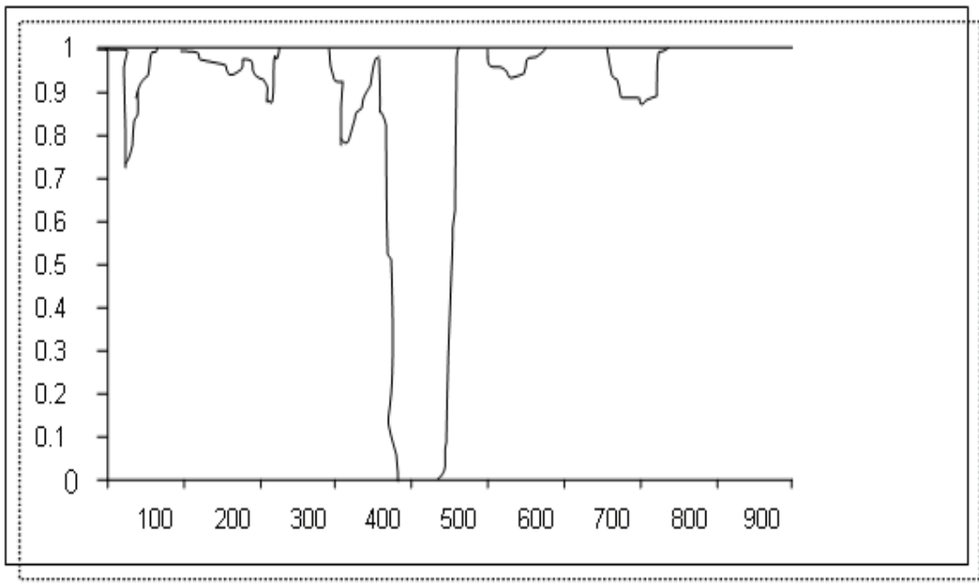


Fig. 7. Consolidated firing strength for rules to detect smurf attack

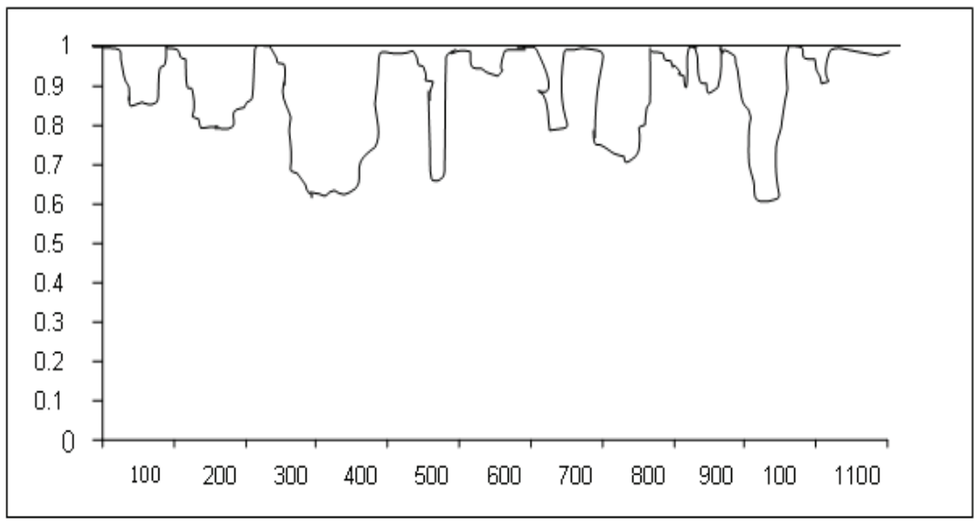


Fig. 8. Consolidated firing strength for real time data

To prove that the system can also work online, the prototype was tested with online data captured in UTM Citycampus. As networking packets constituted huge amount of data, we will however discuss only a small amount of the test data. Here the anomaly detection profiles were based on the DARPA data set. It constituted a clear data i.e. data without any attacks. The data was captured using our own sniffing tool which was written using the Java language. The packages for packet capturing were widely available on the Internet. The data was collected on different days and at different time as follows.

Day 1 : Feb 20th starting from 10.00 a.m. to 6 p.m.
Day 2 : Feb 22nd starting from 10.00 a.m. to 2.00 p.m.

The amount of data collected on the above mentioned days was about 1.76 GB. The collected data was processed and tested with the IIDS. The following graph represents the testing for 20 minutes on day 2 i.e. from 11.30 a.m. to 11.50 a.m.

In the above figure, the firing strengths for most of the records were near to 1. So, there was not much change, which indicated that the prototype did not detect any attack during the testing period. The consolidated results for all four attack types are shown in Table 6.12. The overall analysis and benchmarking of our prototype against others is as shown in Table 6.14. The performance table shows that the detection rate is comparatively higher than all other systems. The false positive rate is also low in comparison to the values obtained from other models.

It was interesting to note that during the experimental stage U2R attacks performance was relatively less, this was because the attacks were distributed across the entire test data. More interesting and important was the fact that the prototype was able to detect the "yaga" attack which occurred during Week 5 day 5. This attack was new and moreover it was not available in the training data set. The proposed prototype was able to detect another new attack, "sechole" which occurred during week 4 day 2. In most cases U2R attack detection performance was always low because of its distribution of attacks and also multiple connections are involved which need more features to be selected.

Attack Types	**Detection %**	**False positive %**
R2L	92.1	10.7
PROBE	98.4	1.8
DOS	94.77	5.5
U2R	69.6	6.7
Average	88.71	6.1

Table 5. Consolidated result for all the four attack types

The R2L attack performance was satisfactory because the prototype was able to detect most of the attacks with high detection rate and with low false positives. The system was able to detect new attacks like net cat, net bus and ncftp, which do not occur in the training data. The total false positive rate seems to be larger but since it has more number of attack instances the value also had increased simultaneously. In the future, we will try to bring down the false positives rate by at least to 0.5%.

During the research, the IIDS were able to detect arppoison, a new type of attack with a high detection rate nearing 95% and with a low false positive < 1%. We were able to achieve 100% detection rate for smurf attacks. These performance shows (Table 6.14) that our system is better in detecting certain types of attacks fully. Some of the researchers mentioned in our earlier chapters use only 10% of the KDD training data. Using such a small amount of training dataset is questionable because the dataset is idealistically simple and moreover 98% of the training data contains "smurf" and "neptune" class. However, for content based attacks which is based on the payload, depending on a feature that is irrelevant with content may lead to false positives.

In a more recent work (Su *et al.*, 2009) have applied fuzzy-association rules using incremental mining. The major advantages cited by the authors are the ability of the system

to create a learning data set based on the real time data and to handle real time network traffic. Even though it has the ability to handle real time systems, the research has not addressed the query frequencies as apriori algorithm will generate huge number of candidates. Using incremental data mining will also lead to increase in the number of queries to the rule databases since the rules are checked every two minutes, which could be a bottle neck especially when the system goes online. However, another limitation (Su *et al* 2009) did not mention is the storage requirements for the huge amount of rules produced by the aprirori algorithm based on the online traffic. This limitation was well handled in (Shanmugam and Idris 2009) and has solved the storage and query frequencies. This was however acknowledged by (Hossain Mahmood *et al*., 2003) who concluded that the incremental mining approach has some advantages, but the application to intrusion detection need to be explored further as it depends on the various aspects like algorithm selected, training and testing data used etc.

We were not able to produce the detection rate, false positives or any other values for real-time data because there is no any universal benchmarking methods available.

Our experimental results are summarized in the below table (Table 6.14) for different attributes. We selected the features by calculating the detection rate for each feature and deleting it one by one. We have not discussed the values due to space restrictions. By the above method we were able to select the appropriate features using the information gain algorithm. Table 8 gives the comparison of using all 41 features against selected 24 features by our proposed method. This clearly reveals the fact that all the features are not important and the selected features had played an important role in improving the overall performance. Initial FIRE (Dickerson and Dickerson, 2000) tests were performed on production local area networks in the College of Engineering at Iowa State University. Using the fuzzy rules, FIRE able to detect nine distinct TCP port scans and separate ICMP (ping) scans of hosts on the network potentially malicious attackers from outside the local network domain. Additionally, it was able to detect non-malicious port scans launched against the system from the local domain. The system also triggered HIGH alerts when seldom seen types of network traffic were observed, in agreement with the Fuzzy Rules used. The system reported a high false positive rate (10.6%) with an average detection rate (79.2%). ADAM (Barbara *et al* 2001) was aimed to detect intrusions of the type Denial of Service (DOS) and Probe attacks (although ADAM can discover other types of intrusions as well).

The overall detection rate recorded was about 84% with a very high false positive rate of 14%.ADAM is a test bed to research which data mining techniques are appropriate for intrusion detection. In spite of the furious activity in intrusion detection, the problem of false alarms and missed attacks (false positives and negatives) is still prevalent.

Features	**Detection %**	**FP %**
All features (41)	77.21	9.23
Selected features (24)	88.71	6.1

Table 6. Different feature selection

The experimental results (Thombini *et al* 2004) shows that the approach drastically reduces the amount of false positives and unqualified events. The proposed model clearly addresses the problem of false positives and the anomaly and signature can be updated as and when needed. The system was tested against HTTP traffic for its effectiveness and the comparison was made against capture traffic on their own and not on DARPA dataset. The proposed

hybrid system (Zhang and Zulkernine (2006)) combines the advantages of these two detection approaches. Besides, the misuse detection can remove known intrusions from datasets, so the performance of the anomaly detection can be improved by applying the misuse detection first. The experimental results (Detection rate of about 86%) show that the proposed hybrid approach can achieve high detection rate with low false positive rate, and can detect novel intrusions. However, some intrusions that are very similar with each other cannot be detected by the anomaly detection. That is a limitation of the outlier detection provided by the random forests algorithm.

By combining the anomaly detection method (Tajbakhsh, A., *et al*. 2006; Tajbakhsh, *et al,*. 2009) with misuse detection method, the false positive error rate in the proposed anomaly detection method is kept as low as in misuse detection scenario. There is a remarkable decrease in the detection rate of the known attacks in the anomaly detection scenario. And in the case of unseen attacks the anomaly scenario performs better than the misuse approach. This is actually the most important advantage of combining both the methods. This method is somewhat near to IIDS detection rate and false positive rate. In the weighted signature generation approach (Kai *et al.,* 2007), only the most frequent patterns of detected anomalies are characterized as signatures. By further eliminating nondiscriminative patterns, the generated signatures are quite specific to anomalies detected. Therefore, the newly generated signatures have quite low false alarm rates. The proposed HIDS results in a detection rate of 60 percent, and False alarm rates are maintained about 3 percent. Alerts from intrusions and anomalies detected need to be correlated to result in an even smaller overhead in the detection process.

Group Name	**Detection rate %**	**False positive %**
Lee *et al.,* 2001 IIDS	78	12.2
Dickerson and Dickerson (2000) FIRE **	79.2	10.6
Barbara *et al.,* 2001 ADAM	84	14
Zhang and Zulkernine (2006)	86	NA
Tajbakhsh *et al.,* 2006; Tajbakhsh *et al.,* 2009	85.5	6.9
Kai *et al.,* 2007 HIDS	60	30
Jianhui *et al.,* 2008	94	NA
IIDS	88.71	6.1

Table 7. Comparison with other selected models

In this work (Jianhui *et al.,* 2008), the model of intrusion detection based on fuzzy sets is suggested and experiment results showed its accuracy and capability. In the process of rules mining, however, we found the select of membership function depended excessively upon the expert knowledge, which necessarily causes the deviation between results and experiment conclusion. The authors need to focus on how to obtain an optimal membership function with minimum overhead. In the experiment (Gongxing and Yimin, 2009), authors

collect the audit data of users' 7 day's normal operations as training data so as to get the user's behaviors. In this system, misuse detection can detect the attack attempt well, but due to no intrusion rule of impersonation attack and legal user attack in the misuse detection rule base, the two attacks can not be detected. Combined with the characteristics of misuse detection and anomaly detection, designs and realizes a new type of intrusion detection system with adaptive ability and applies the Apriori algorithm based on Trie tree to the database intrusion detection system to improve the generation efficiency of rule base.

7. Conclusions and future directions

In the recent years, IDS have slowly changed from host based application to a distributed systems that involves a variety of operating systems. The challenges that lie ahead of us for intrusion detection system, particularly for hybrid systems are huge. First, is the inability to reduce the number of false positives that prevents from intrusion detection systems being deployed widespread. As reported (Varun, 2009), the intrusion detection systems crash because of its in-ability to withstand the heavy load of false alarms. Second, the time take to process the huge amount of data is mounting, a process to reduce the time taken should be considered. Third, there is a lack of standard evaluation dataset that can simulate the real time network environments. The existing evaluation data set DARPA/Lincoln labs are a decade old and they are currently being used to evaluate any intrusion detection systems. There is a need to create a new data set, where it could be used to evaluate the intrusion detection systems for the dynamic topologies. Finally, our system crashed, as it could not withstand the traffic for more than three weeks without restarting, and that issue has to be sorted out using a high-end hardware and systematically re-tuned source code.

8. References

Aickelin, U., P. Bentley, et al. (2003). Danger theory: The link between ais and ids. *Proceedings of Second International Conference on Artificial Immune Systems (ICARIS-03)*,147-155.

Aly El-Semary, Janica Edmonds, et al. (2006). Applying Data Mining of Fuzzy Association rules to Network Intrusion Detection. *Proceedings of IEEE workshop on Information Assurance*,100-107.

Axelsson, S. (1998). Research in intrusion-detection systems: a survey. Goteborg, Sweden,, Department of Computer Engineering, Chalmers University of Technology.

Barbará, D., J. Couto, et al. (2001). ADAM: a testbed for exploring the use of data mining in intrusion detection. *ACM SIGMOD* 30(Special):pp. 15-24

Breiman, L. (2001). Random forests. *Machine Learning* 45(-):pp. 5-32

Bridges, S. M. and R. B. Vaughn (2000). Fuzzy data mining and genetic algorithms applied to intrusion detection. *Proceedings of National Information Systems Security Conference* Baltimore.

Cannady, J. (1999). Artificial Neural Networks for Misuse Detection. *Proceedings of National Information Systems Security Conference*,443-456, Arlington.

Cansian A, M., Moreira E, et al. (1997). Network intrusion detection using neural networks. *Proceedings of International conference on computational intelligence and multimedia applications ICCMA*.

Debar, H., M. Dacier, et al. (1998). A workbech for Intrusion detection systems. *IBM Zurich Research Laboratory*:pp.

Diaz-Gomez. and D. F. Hougen (2005). Analysis and mathematical justification of a fitness function used in an intrusion detection system. *Proceedings of Genetic and Evolutionary Computation Conference*,1591-1592, ACM.

Dickerson, J. E. and J. A. Dickerson (2000). Fuzzy Network Profiling for Intrusion Detection. *Proceedings of 19th International Conference of the North American Fuzzy Information Processing Society*,301-306, Atlanta.

Dokas, P., L. Ertoz, et al. (2002). Data Mining for Network Intrusion Detection. *Proceedings of National Science Foundation Workshop on Next Generation Data Mining*,21-31.

Girardin, L. and D. Brodbeck (1998). A Visual Approach or Monitoring Logs. *Proceedings of 12th System Administration Conference*,299-308.

Gongxing, W. and H. Yimin (2009). Design of a new Intrusion detection system based on database. *Proceedings of International conference on signal processing systems*,814-817.

Hosmer, H. (1993). Security is fuzzy!: applying the fuzzy logic paradigm to the multipolicy paradigm. *Proceedings of 1992-1993 Workshop on New Security Paradigms*,175-184, Little Compton.

Hossain, M., Bridges Susan M, et al. (2003). Adaptive intrusion detection with data mining. *Proceedings of IEEE Conference on Systems, Man and Cybernetics*,3097-3103.

Jianhui, L., T. HUANG, et al. (2008). A Fast Fuzzy Set Intrusion Detection Model. *Proceedings of 2008 International Symposium on Knowledge Acquisition and Modeling*,601-605, ACM.

Kai, H., C. Min, et al. (2007). Hybrid Intrusion Detection with Weighted Signature Generation over Anomalous Internet Episodes. *IEEE Treansactions on dependable and secure computing* 4(1):pp. 41-55

Kephart. J (1994). A biologically inspired immune system for computers. *Proceedings of Fourth International Workshop on Synthesis and Simulatoin of Living Systems,*
Artificial Life,130-139.

Kim, J. (2003). Integrating artificial immune algorithms for intrusion detection. Department of Computer Science. London, University College London: 190.

Kuok, C., A. Fu, et al. (1999). Mining Fuzzy Association Rules in Databases. *The ACM SIGMOID Record* 27(1):pp. 41-46

Lee, W. (2001). A Data Mining framework for construction features and models for intrusion detection systems, Columbia University: 200.

Li, W. (2006). Using Genetic Algorithm for Network Intrusion Detection, Department of Computer Science and Engineering, Mississippi State University,: 10.

Lippmann, R. P., D. J. Fried, et al. (2000). Evaluating Intrusion Detection Systems : The 1998 DARPA Offline Intrusion Detection Evaluation. *DARPA Information Survivability Conference and Exposition* 2:pp. 12-26

Lippmann. R. and Cunningham. R. (1999). Improving Intrusion Detection performance using Keyword selection and Neural Networks. *Proceedings of Proceedings of Recent Advances in Intrusion Detection*,223-230, Indiana.

Lunt, T. F. (1993). Detecting Intruders in Computer Systems. *Proceedings of Conference on Auditing and Computer Technology*.

Manganaris, S., M. Christensen, et al. (2000). A data mining analysis of RTID alarms. *Computer Networks* 34(4):pp. 571-577

Me, L. (1998). GASSATA, a Genetic Algorithm as an alternative Tool for Security Audit Trials Analysis. *Proceedings of 1st International workshop on Recent Advances in Intrusion Detection*,11, Belgium.

Mitrokotsa, A., K. Nikos, et al. (2007). Towards an Effective Intrusion Response Engine Combined with Intrusion Detection in Ad Hoc Networks. *Proceedings of The Sixth Annual Mediterranean Ad Hoc Networking WorkShop*,137-145.

Ozgur, D., T. Murat, et al. (2005). An intelligent intrusion detection system (IDS) for anomaly and misuse detection in computer networks. *Expert Systems and Applications* 29(4):pp. 713-722

Richard, K. and V. Giovanni (2002). Intrusion Detection: A Brief History and Overview. Security and Privacy: 27-30.

Ryan. J., Lin. M., et al. (1998). Intrusion Detection with Neural Networks. *Advances in Neural Information Processing Systems* 10:pp.

Sabahi, F. and A. Movaghar (2008). Intrusion Detection: A Survey. *Proceedings of Third International Conference on Systems and Networks Communications*,23-26.

Sarafijanovic, S. and J. Boudec (2003). An artificial immune system approach with secondary response for misbehavior detection in mobile ad-hoc networks., Ecole Polytechnique Federale de Lausanne.

Scarfone, K. and M. Peter (2007). Guide to Intrusion Detection and Prevention Systems, National Institute of Standards and Technology: 127.

Shanmugam, B. and N. B. Idris (2009). Improved Intrusion Detection System Using Fuzzy Logic for Detecting Anamoly and Misuse Type of Attacks, International Conference of Soft Computing and Pattern Recognition, 2009 pp.212-217.

Somayaji, A., H. Steven, et al. (1997). Principles of a Computer Immune System. *Proceedings of New Security Paradigms Workshop*.

Su., M.-Y., G.-J. Yu., et al. (2009). A real time network intrusion detection system for large-scale attacks based on an incremental mining approach. *Computers and Security* 28(5):pp. 301-309

Suhail, O., S. Vaclav, et al. (2008). Using Genetic Algorithm Approach in Intrusion Detection Systems Techniques. *Proceedings of 7th Computer Information Systems and Industrial Management Applications*,300-307.

Su-Yun, W. and Y. Ester (2008). Data mining-based intrusion detection. *Expert Systems with Applications* 36(3):pp. 5605-5612

Tajbakhsh, A., M. Rahmati, et al. (2006). A new classification approach using fuzzy association rules. *Proceedings of 11th International CSI computer conference*,263-270.

Tajbakhsh, A., M. Rahmati, et al. (2009). Intrusion detection using fuzzy association rules. *Applied Soft Computing* 9(2):pp. 462-469

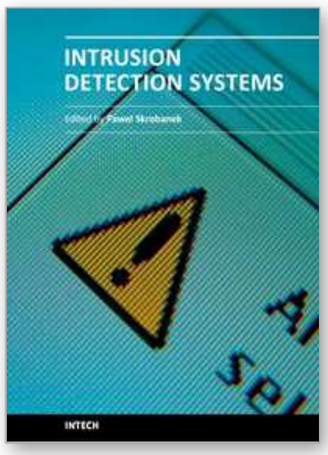
Tombini, E., H. Debar, et al. (2004). A serial combination of anomaly and misuse IDSes applied to HTTP traffic. *Proceedings of 20th Annual Computer Security Applications Conference*,428-437, Arizona.

Varun, C., B. Arindam, et al. (2009). Anomaly Detection - A Survey. *ACM Computing Surveys* 41(3):pp.

Wu.X and W. Banzhaf (2009). The use of computational intelligence in intrusion detection systems: A review. *Applied Soft Computing Journal*:pp. 35

Zhang, J. and M. Zulkernine (2006). A hybrid network intrusion detection technique using random forests. *Proceedings of First International Conference on Availability, Reliability and Security*,262-269., Vienna.

Zorana, B., J. M. Moya., et al. (2009). A Genetic Algorithm-based Solution for Intrusion Detection. *Journal of Information Assurance and Security* 4:pp. 192-199



Intrusion Detection Systems
Edited by Dr. Pawel Skrobanek

ISBN 978-953-307-167-1
Hard cover, 324 pages
Publisher InTech
Published online 22, March, 2011
Published in print edition March, 2011

The current structure of the chapters reflects the key aspects discussed in the papers but the papers themselves contain more additional interesting information: examples of a practical application and results obtained for existing networks as well as results of experiments confirming efficacy of a synergistic analysis of anomaly detection and signature detection, and application of interesting solutions, such as an analysis of the anomalies of user behaviors and many others.

How to reference
In order to correctly reference this scholarly work, feel free to copy and paste the following:

Bharanidharan Shanmugam and Norbik Bashah Idris (2011). Hybrid Intrusion Detection Systems (HIDS) using Fuzzy Logic, Intrusion Detection Systems, Dr. Pawel Skrobanek (Ed.), ISBN: 978-953-307-167-1, InTech, Available from: http://www.intechopen.com/books/intrusion-detection-systems/hybrid-intrusion-detection-systems-hids-using-fuzzy-logic

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

IntechOpen