HIDS with minimize property

By

M.SC. Hadeel Amjed Saeed

Computer Science\ Al-Turath University College

Abstract

Security system is the immune system for computers which is similar to the immune system in the human body. This includes all operations required to protect computer and systems from intruders. The aim of this paper is to develop an anomaly-based intrusion detection system (IDS) that can promptly detect and classify various attacks. Anomaly-based IDSs need to be able to learn the dynamically changing behavior of users or systems. In this paper are experimenting with packet behavior as parameters in anomaly intrusion detection. There are several methods to assist IDSs to learn system's behavior. The proposed IDS use a back propagation artificial neural network (ANN) to learn system's behavior. A new operation has been added to this work by minimize the property of packet from 22 properties to 4 main properties. The KDD'99 data set had been used in the experiments and the obtained results satisfy the work objective.

Keywords- minimize property, anomaly detection, IDS, intrusion, KDD’99, neural network

نظام حاسوبي لكشف التطفل مع تقليلل الخصائص

هديل امجد سعيد

كلية التراث الجامعة

الخلاصة

نظام الأمن هو الجهاز المناعي لأجهزة الكمبيوتر التي هي مماثلة لجهاز المناعة في جسم الإنسان .وهذا يشمل جميع العمليات اللازمة لحماية أنظمة الكمبيوتر من الدخلاء و .الهدف من هذا العمل هو القائم على (IDS) نظام كشف التسلل امكانية الكشف فورا وتصنيف الهجمات المختلفة . أساس الشذوذ في حاجة لتكون قادرة على تعلم السلوك تغيير حيوي من المستخدمين أو الأنظمة .في هذا البحث تم استخدام تدريب سلوك الحزمة الداخله للحاسوب كمدخلات في كشف التسلل ،واستخدمت الشبكات العصبية لتعلم سلوك الصحيح للنظام .كما تم إضافة عملية جديدة للبحث عن طريق تقليل خصائص الحزمة من 22خاصية الى 4خواص اساسية وظهرة زيادة في نسبة الكشف حوالي 10%. كما تم استخدام KDD'99عليها تلبية لهدف العمل .

I. Introduction

Computers are considered one of the most important technological advances of the twentieth century. In the last few years, the use of computers in the home and in business was increased considerably. However, security and privacy issues have been in long existence, even before the computer became a vital component of organizations' operations. Now, security is a big and increasingly-important issue for all networks and computer in today's enterprise environment. Internet (as many other things) is double-edged. It is the entrance to many beneficial things. Unfortunately, it opens the way for a lot of harmful things to login into your device. Hackers and intruders have made many successful attempts to bring down high-profile companies networks and systems. Many methods have been developed to secure the system infrastructure and communication over the internet such as the use of firewalls, intrusion detection, and encryption [1] [2].

Intrusion detection is the process of monitoring the events occurring in a computer system or network and analyzing them for signs of intrusion. It aims to protect the confidentiality, integrity, and availability of critical networked information systems [3] [4]. Intrusion detection system (IDS) is a system that gathers and analyzes information from various areas within a computer or a network to identify attacks made against these components. The IDS uses a number of generic methods for monitoring the exploitations of vulnerabilities. IDSs can be characterized by depending on three main aspects [5] [6]:

- **The data source: In this case, we have host-based, network-based, or hybrid IDSs. Host based IDS monitor's computer components (such as operating system, packet, system log, etc.).**

Network based IDS monitors the network (such as traffic). Hybrid IDS combines host with network for monitoring computer and network together.

- **The model of intrusion detection: Here we have anomaly detection, misuse detection, or hybrid detection. Anomaly based IDS monitoring depends on the behavior of system. Misuse based IDS monitoring depends on signature to data. Hybrid techniques combine anomaly with misuse.**
- **The audit collection and analysis: Here IDSs are divided into either centralized or decentralized (distributed) IDSs. In centralized IDSs, monitoring, detection, and reporting are controlled directly from a central location. In decentralized IDSs, monitoring and detection are controlled from a local control node with hierarchical reporting to one or more central location(s).**

A more detailed classification diagram of IDSs is shown in Fig (1). In this work, we have developed anomaly-based IDS that use ANN for attack detection and classification. Also is used the KDD'99 data set for training and testing of our system. The developed system can be operated in three main modes: detection mode (for differentiating between normal and abnormal actions), detection and classification mode (for further classifying abnormal events into four main attack types: DOS, PROB, U2R, or R2L), and detailed classification mode (for detailed classification of abnormal events into 29 sub-attack types). The remaining of this paper is organized as follows: Section 2 represents a theoretical background of IDSs. The design stages of the proposed IDS are introduced in Section 3. Next, Section 4 discusses some implementation issues and experimental results. Finally, the paper is concluded in Section 5.

II. Intrusion Detection Systems (IDSs)

IDS is an important component for security system. It complements security of other technologies through the provision of information for management. It does not only detect attacks that are discovered by other security elements, but also attempts to provide notification of new attacks that cannot be expected by the other ingredients. This is done by continuously monitoring and analyzing the events that occur in the computer system or network from inside or outside. There are three main steps in the process of intrusion detection, which are: Monitoring and analyzing traffic and system; identifying abnormal activities; and assessing severity and raising alarms [7]. Thus, IDS can be considered as a defense system that monitors traffic on the network or analyze the activity of the system. In this way, it can detect various activities that might endanger the security of the system. Furthermore, as the IDS usually stores basic cases of the data in records, this provides valuable information and could be used as evidence in a lawsuit against the aggressor. Hence, IDSs are important elements for completing the infrastructure for information security and lead the logical complement to the firewall [8].

The IDS and other security components work together to offer an integrated high level of security for the whole system in a way that is much similar to the work of a house security system. In house security systems, different tools are used for protection such as supplying the surrounding wall with barbed wire, putting guards on the door, using cameras for monitoring, etc. [6]. Thus, both types of security systems have three basic security functions (monitoring, detection, and response) and include a number of elements for implementing these functions [8] [9].

There may be other components to be implemented also. Some effective IDSs have the ability to send alerts to the administrator who will receive notice of possible security incident. The activities of the system may not only identify the particular incident and issuing appropriate warning but also respond automatically for this event (such as disabling of the user account) [10]. There are continuous efforts by developers to develop new and more efficient tools for intrusion detection. However, many users are still in a need for more education about the IDS, its tools, and how to use it. Tools used in IDSs should be able to distinguish between attacks from inside sources within the organization and external (internet) sources [11].

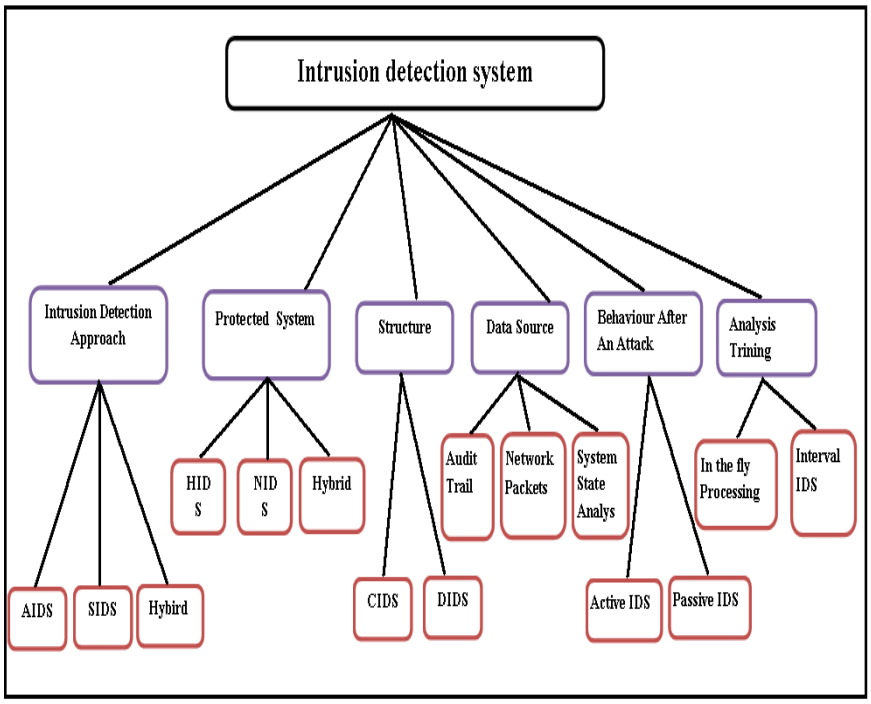


Figure 1. Classification of IDSs [11].

A. Artificial Neural Networks (ANNs)

The computational changes in the last several decades have brought growth to new technologies. One of these technologies is artificial neural networks (ANNs). Over the years, ANNs have given various solutions to the industry. Designing and implementing intelligent

systems have become an important activity for the innovation and development of better products for human life. Examples might include the case of the implementation of artificial life and giving solution to interrogatives that linear systems are not able to resolve [12]. An ANN is a mathematical model to deal with the information that is inspired by the way biological nervous system, such as the brain, performs in information processing. A key element in this model is the structure of this novel information processing system. The system is adaptive and is consisting of a large number of interrelated processing elements (neurons) working together to solve specific problems. ANN such as people learns by example [13], [14], [15].

Thus, ANN can be considered as a network with many simple processors (which are the neurons). These processors receive data from abroad or from other neurons. Data will be received through the channels of communication, known as weights. The ANN needs to be trained (or learned) in order to reach the best output. Basically, learning is a process by which the free parameters (i.e., synaptic weights and bias levels) of the ANN are adapted through a continuing process of stimulation by the environment in which the network is embedded. The type of learning is determined by the manner in which the parameter changes take place. In a general, the learning process may be classified as supervised or unsupervised [14], [16]. Using ANNin implementing IDSs is an important application of the various applications of ANNs.

B. <u>Knowledge Discovery and Database (KDD'99)</u>

The DARPA 1998 intrusion detection evaluation program was prepared and managed by MIT Lincoln labs. In 1999, KDD'99 became the wildly used data set for the evaluation of anomaly detection

methods. The work of DARPA included about 4 gigabytes of compressed raw (binary) and TCP dump data in 7 weeks of network traffic, which could be processed into about 5 million connection records, each with about 100 bytes. The two weeks of testing data had around 2 million connection records. KDD'99 training data set consists of approximately 4,900,000 single connection vectors each of which contains 41 features and was classified as either normal or attack include (DOS, PROB,U2R, R2L). KDD'99 features could be classified into three groups; basic features, traffic features, and content features [17], [18].

III. The Proposed System

The proposed system for anomaly-based intrusion detection is composed of four main stages, as depicted in Fig (2). These stages are monitoring, detection, classification, and alerting. More details of these stages are presented in the following subsections.

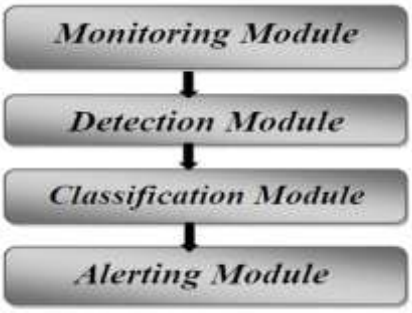


Figure (2). Main Stages of The Proposed IDS.

A. Monitoring Module

The monitoring module is used to give an interface to a system administrator. It provides actual tools to view and manage processes inside of the system such as packet capturing. It consists of a number of management tools that allow the user to view profiles and the data transfer. This module includes reading the system activity to host such

as traffic, process, open, delete, create file, and other operations. It collects the useful data and puts them in profile to determine if data is normal or abnormal. The monitoring algorithm can be described as follows:

Algorithm 1- Monitoring Algorithm

1. *Start when the computer is turned-on.*
2. *Specify the target system to be monitored.*
3. *Monitor the specified target system.*
4. *Check if any suspected operations have occurred on a file, folder or traffic, then add them to database.*
5. *Go to step 3.*
6. *End*

The first important operation of this module for traffic monitoring is packet capturing. The application programming interface (API) for packet capturing (Pcap) enables the capturing of every packet coming to host from any workstation through the connecting network,as shown in Fig (3).

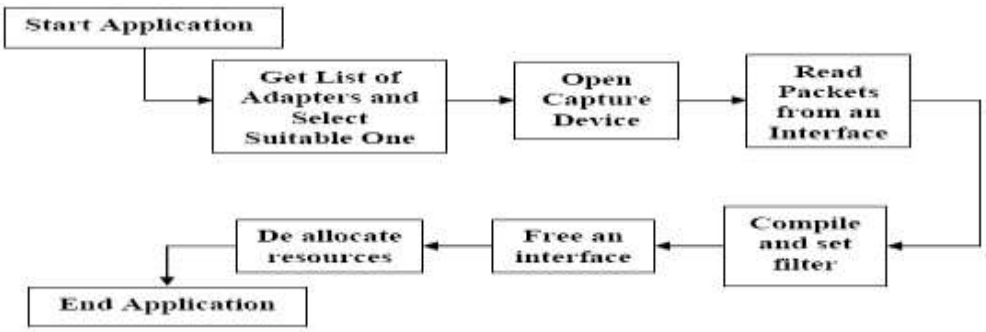


Figure (3). Block Diagram of Packet Observation Process.

B. Detection Module

To enable the anomaly detection in this system, the detection module works on five main steps. These steps are: Reading From Source (Feature Selection), Ranking, Encoding, Normalization and De-Normalization, and finally Anomaly Detection Phase.

1. Feature Selection: This is the first step for detection. It starts after the monitor completes capturing the packets that passed through the network. The packets source and the destination addresses can be specified. The source ports and the destination ports can also be specified. The packets can be captured at the network layer (IP) and/or the transport layer (TCP, UDP, ICMP). The values for each packet header field will be considered according to the features of KDD'99.

2. Ranking: Selecting the important properties and categorizing them are very important for IDSs. There can be a large number of properties and features that can be used in the monitoring process and not all of them are necessary or useful. Dealing with some of them may contain some disadvantages for IDS (such as the difficulty of getting them). Hence, it is very important the use of the principle of ranking to determine and categorize the features into three divisions: preliminary, secondary, and less important. Preliminary features represent the most useful features. Secondary features are of less impact on the monitoring process. Finally, the less important features are of the slightest impact on the detection process. This categorization can accelerate the whole operation of the system. In our system, for example, when considering only 4 preliminary out of the 41 KDD'99 features and good results have been obtained as will be shown later.

3. Encoding: A code can be defined as the rule for converting a piece of information another form or representation. It includes two types of

operations: encoding and its complement, decoding. The encoding is done without using any keys and the non numerical data (records in the database) are converted into serial numbers. This process analyzes the data and if they are normal, it assigns (1) to them. If they are abnormal it assigns (zero). The used encoding algorithm is shown below:

Algorithm 2- Encoding Algorithm

1. *Start*
2. *Read from DB number of records and set n = number of records*
3. *no = 0*
4. *If no <= n, then go to step 6*
5. *Else go to step 9.*
6. *Read a record from DB*
7. *If the data item is not numeric then go to step8, else go to step 9*
8. *Execute convert data*
9. *no = no+1*
10. *End*

4. Normalization and de-normalization: Normalization is the process of organizing data efficiently in a database. This includes establishing relationships between columns of tables in database and makes them more flexible. There are three goals in the normalization process: eliminating redundant data (redundancy), partial dependencies, and transitive dependencies. De-normalization is the process which comes after division operation by multiplying the resulting values of the data by weights. This gives a property of importance in the process of detection so as to split features into the above mentioned three levels (initial properties, secondary, and not important).

5. Anomaly detection: The purpose of this phase is to detect intrusions by using ANN. Anomaly detection is a general category of intrusion detection. It works by identifying activities which vary from patterns of

users, or groups of users. Anomaly detection typically involves the creation of knowledge which contains the profiles about monitored activities. Algorithm 3 below is a general description of detection process.

Algorithm 3- Detection Process

1. *Begin*
2. *Turn on the monitoring system*
3. *Select specific fields from monitoring system database and arrange them in vectors*
4. *While (New vectors are available) DO*
   a. *Begin*
      i. *Read a vector from monitoring system database.*
      ii. *Apply encoding algorithm.*
      iii. *Input the vector to the designed ANN*
      iv. *If the output of ANN is (1) then mark the specified Vector as normal*
      v. *Else mark the specified vector as abnormal*
   b. *If no more vectors are found then Exit*
5. *End (while)*
6. *End*

The number of input nodes of the ANN (4 or 22 or 41) represents the number of the selected features of the network traffic for host. The ANN used also includes two hidden layers (with 10, 20 , 25 and 30 nodes respectively) and an output layer of 29 nodes (29 classifications of attacks). The numbers of hidden layers and nodes in them have been determined based on the back propagation (BP) computation process. The training process for finding the best weights for detection has consumed more than three weeks. The testing process time has been less than 20 hours using our testbed computers.

C. Classification Module

There are several techniques that can be used in the process of classifying data such as ANNs, statistical methods, genetic algorithms, and others. In this project, ANNs have been used in classification of data. The results can only be obtained after completing both of

training and testing phases. The intrusion data have been classified into five categories. The first category represents normal data and the other four are attack types. These attack types are probing, denial-of-service (DoS), remote-to-local (R2L), and user-to-root (U2R) attacks. Each of these categories indeed contains sub-types of attacks, as shown in Table 1 [19].

Table (1) Classified attack types

Probing	DOS	U2R	R2L
ipsweep, mscan, nmap, portsweep, Sendsaint , satan.	apache2, back, land, mailbomb, neptune, pod, processtable, smurf, teardrop, udpstorm.	buffer_overflow httptunnel, loadmodule, perl, ps, rootkit, sqlattack, xterm.	ftp write, guess_passwd, imap, multihop, named, phf, send-mail, snmpgetattack, snmpguess, spy, warezclient, warezmaster, worm, xlock, xsnoop.

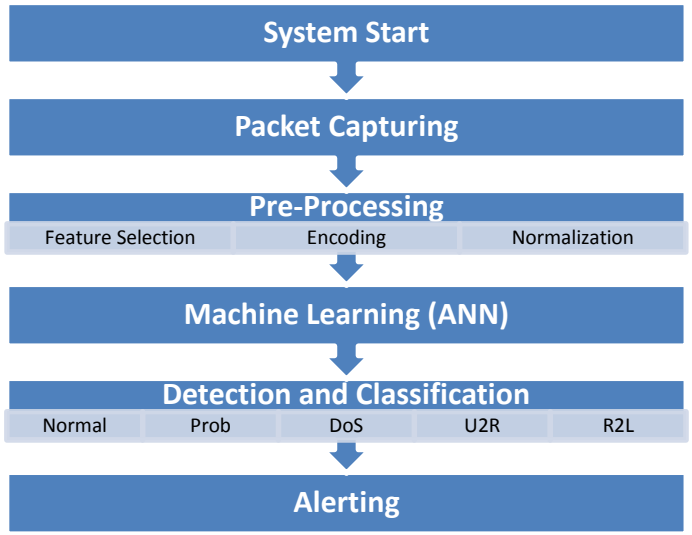
D. <u>Alert Module</u>

This is the final stage of the proposed system. This stage involves identifying the events that occurred whether abnormal or not, then sending the required signals to alert administrator (or user) accordingly.

IV. System Implementation and Results

The proposed IDS has been implemented using VB.NET 2008 language on Pentium 4 machines running under Windows XP and Vista. Before the system running, two additional programs need to be downloaded and installed. These are the Winpcap and Sharppcap programs. WinPcap is an open source library for packet capturing and network analysis for the Win32 platforms. It can be used directly or indirectly.

In our implementation, we are following the indirect approach. So, the system uses Sharppcap with Winpcap. Sharppcap is an open source program that is written in C#.NET and can be called in VB.NET as library (dll). Fig. 4 represents the block diagram of the implemented system.



Figure(4). Block Diagram Of The Implemented System.

Packet capturing (Pcap) process, which is the first step in the system operation, enables to capture the incoming and outgoing packets in the network. The Filter part of Pcap gives the user the required flexibility to filter every aspect of raw packets (such as protocol type, source address, destination address, and others). The ANN was trained on data taken from KDD CUP'99 to differentiate between normal and abnormal packets. Attacks (abnormal actions) are further classified into four main types, as mentioned previously. After the training of the ANN and finding the best detection rate, the best weights have been saved in a file to be used in program execution during the testing phase. The main IDS interface window appears as in Fig(5).

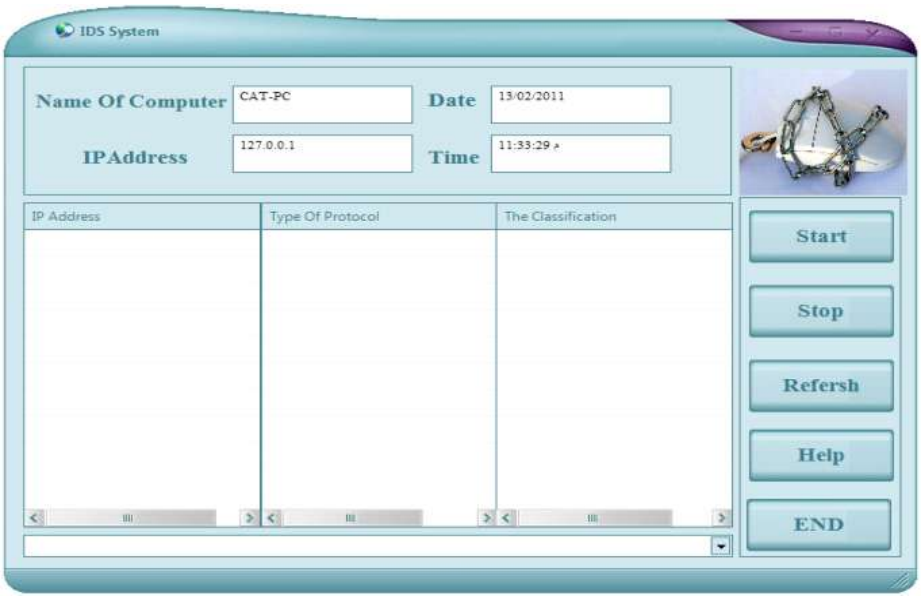


Figure (5). Main IDS Interface Window.

The training and testing have been done using back propagation ANN method for classifying various actions. The Detection Rate (DR) and False Positive rate (FP) have been calculated for different scenarios. The considered scenarios in our experiments are as follows:

- **Detection Only Scenario: This case includes merely detection of attack by deciding whether packets are normal or abnormal. The training and testing have been performed with the 44 and 22 features, respectively, as shown in Fig. 6.**
- **Detection and Classification: This scenario includes detection of packets and classification of them into normal or one of the four main attack types (DOS, Prob, U2R, or R2L). The training and testing here have been done with 22 features and two styles have been considered: training without normalization and training with normalization. The results are shown in Fig. 7.**
- **Detection and Detailed Classification: This case includes detecting the packets and classifying them into normal or main attack**

type (DOS, Prob, U2R, or R2L) and also classifying them into sub-attack categories (according to that in Table 1). This gives a total of 29 types. Indeed, the training and testing here have been performed with 22 features and two styles have been considered: training without normalization and training with normalization, as shown in the results in Fig. (8).

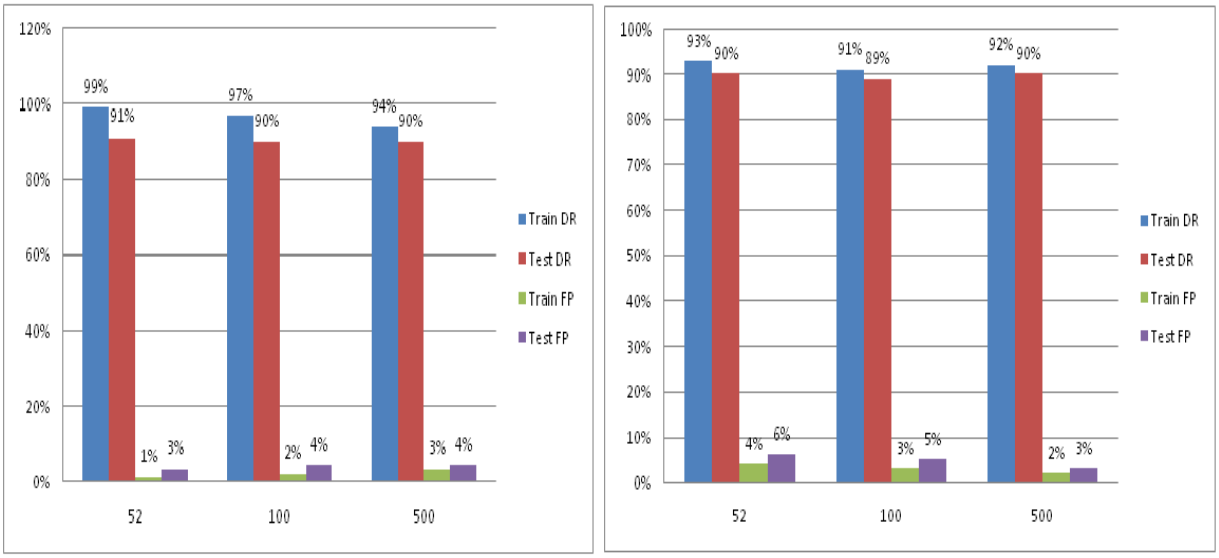


Figure 6. Detection results of training and testing with 41 and 4 features, respectively.

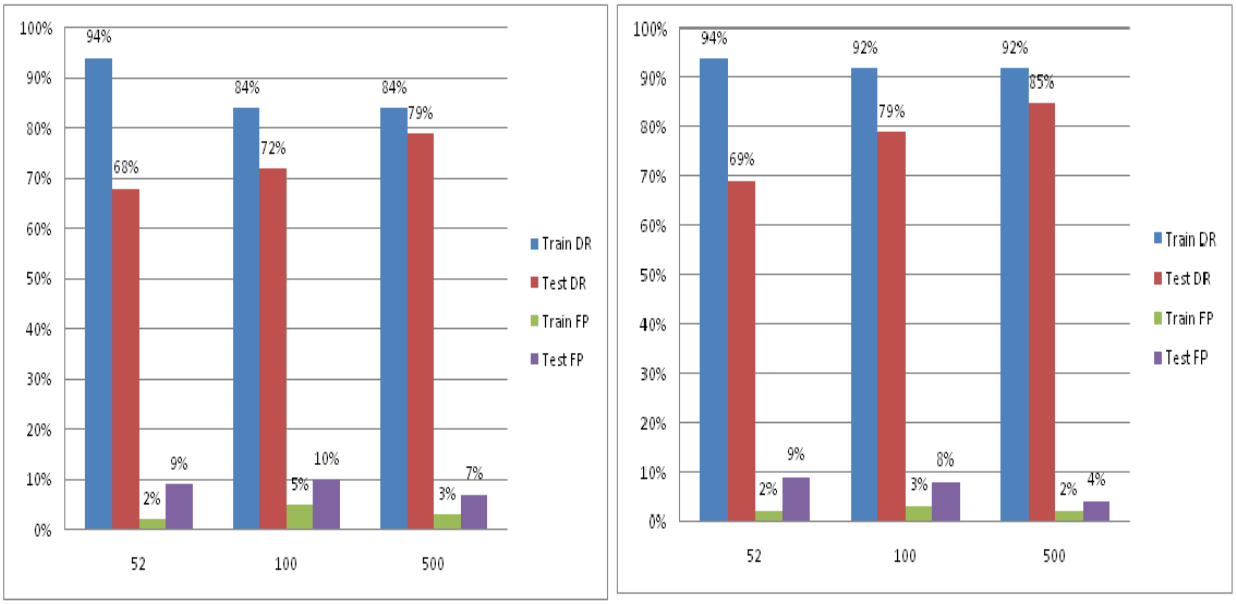


Figure (7). Results of detection and classification without and with normalization, respectively.

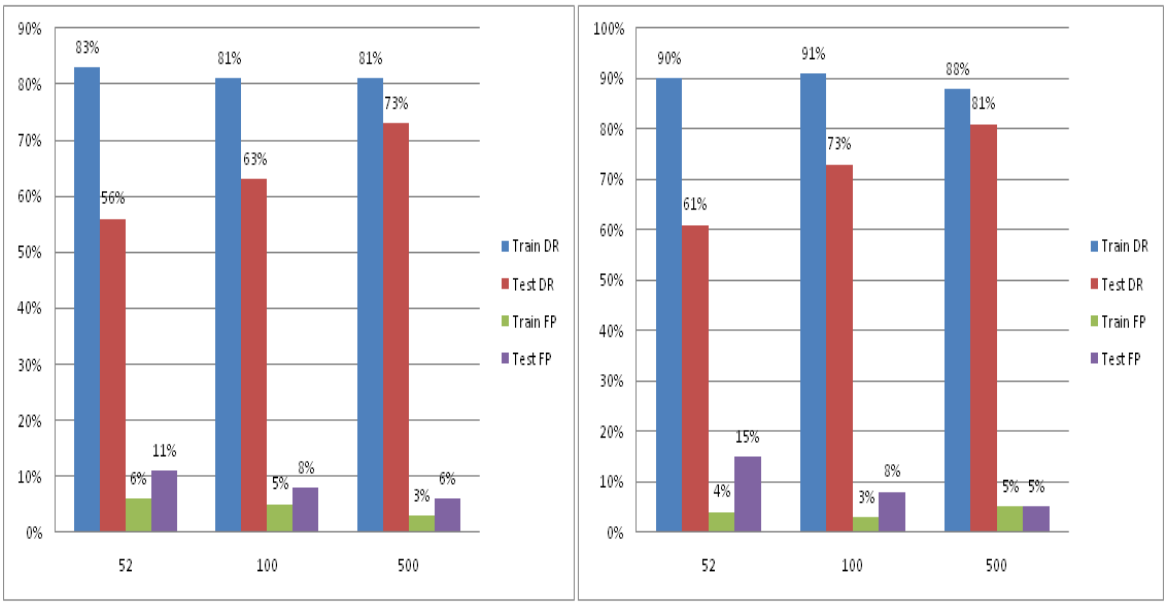


Figure 8. Results of detection and detailed classification without and with normalization, respectively.

V. Conclusion And Future Recommendation

The machine learning procedure using for IDS design and implementation enables the system to adapt to new environments. This facilitates the detection of unknown attack anomalies by the developed IDS. Indeed, the modular architecture of the system enables it to be easily extended, configured, and/or modified. This is can be done by adding new features or by replacing features when they need to be updated. However, the training of the ANN requires a very large amount of data and considerable time to ensure that the results are accurate. Another issue is that there is some kind of compromise between increasing the classification levels and the percentage of detection. Thus, a trade off is required. Furthermore, the obtained results of training with 41 features have been better than those with training with 22 features. Also, results of testing with normalization

have been generally better than the results of testing without normalizing. As a future direction of development of this work, other techniques such as genetic algorithms, fuzzy algorithms, or hybrid methods can be used for detection and classification.

References

1. D. Herrmann, "A practical guide to security engineering and information assurance", 2002, www.auerbach-publications.com.

2. S. Kiran, "Exploring a novel approach for providing software security using soft computing systems", International Journal of Security and Its Applications, Vol. 2, No. 2, pp. 51- 58, 2008.

3. S. Alexander, "An anomaly intrusion detection system based on intelligent user recognition", Ph.D. Thesis, Faculty of Information Technology, University of Jyväskylä, Finland, 2002.

4. S. Mansour and A. Sha'bani, "Fast neural intrusion detection System Based on Hidden Weight Optimization Algorithm and Feature Selection", World Applied Sciences Journal, No. 7 (Special Issue of Computer & IT), pp. 45-53, 2009.

5. J. Daejoon , H. Taeho, and H. Ingoo "The neural network models for IDS based on the asymmetric costs of false negative errors", Pergamon, Journal of Expert Systems with Applications, No. 25, pp. 69–75, 2003.

6. R. Baceand P. Mell, "NIST Special Publication on Intrusion Detection Systems", 2002.

7. W. Jeffery, "Information Security Policy", California State University, Sacramento Information Security Office, July 2009.

8. P. Kazienko and P. Dorosz, "Intrusion detection Systems (IDS): Part I," Windows Security, 11.43 WIB, September 2004, www.WindowSecurity.com.

9. V. Theuns and H. Ray, "Intrusion detection techniques and approaches", Journal of Computer Communications, Vol. 25, No. 15, pp. 1356 - 1365, 2002.

10.W. Mahoney and W. Sousan," IDEA: A new intrusion detection data source", The 2nd International Conference on Information Security and Assurance, Korea, April 2008.

11.L. Theodoros and P. Konstantinos, "Data mining techniques for (network) intrusion detection systems" Department of Computer Science and Engineering, UC Riverside, CA, USA, 2005.

12.R. Ghosh, "A novel hybrid learning algorithm for artificial neural networks", Ph.D. Thesis, School of Information Technology, Griffith University, 2002.

13.S. Jonas, "Neural network computations using Mathematica", A Tutorial by Wolfram Research, http://www.mvs.chalmers.se/~sjoberg/ September 2005.

14.M. Hagan, Neural Network Design. Cengage-Nelson, Canada, 2008.

15.V. Konstantinos, "Machine learning approaches to medical decision making ", PhD Thesis, Department of Computer Science, University of Bristol. March 2001

16.K. Girish, "Artificial neural networks ", Indian Agricultural Research Institute PUSA, New Delhi-110 012, http://www.iasri.res.in/ebook/ , 2005.

17.S. Peddabachigaria, A. Abrahamb, C. Grosanc, and J. Thomasa, "Modeling intrusion detection system using hybrid intelligent systems," Journal of Network and Computer Applications, Vol. 30, pp. 114-132, 2007.

18.V. Matthew and K. Philip, "PHAD: Packet header anomaly detection for identifying hostile network traffic" Department of Computer Sciences, Florida Institute of Technology Technical Report CS-2001-04, 2001.

19.R. Lippmann et al, "Evaluating intrusion detection systems: The 1998 DARPA off-line intrusion Detection Evaluation," DARPA Information Survivability Conference and Exposition (DISCEX'00), Vol. 2, pp. 12-26, 2000.